



République Algérienne Démocratique Et Populaire
Ministère De L'enseignement Supérieur Et De La Recherche Scientifique
Université Abou Bakr Belkaid-Tlemcen
Faculté De Technologie
Département De Génie Électrique Et Électronique

Filière : Génie Industriel
Projet de Fin d'Etudes de Master
Spécialité : Ingénierie de la production

Intitulé :
Implémentation d'un Digital Twin des stations de
La chaîne MPS 500 sous Flexsim

Présenté par :
ABDOUNE Khadidja Farah

Devant le jury :

<u>Président :</u>	BETAOUAF Talib Hichem	MCB	Université de Tlemcen
<u>Encadrants :</u>	GHOMRI Latéfa	MCA	Université de Tlemcen
	CASTAGNA Pierre	Professeur	IUT de Nantes
	MEKKADER Mohamed El Amine	Ingénieur en R&D	Université de Tlemcen
<u>Examineur :</u>	SARI Mohammed	Enseignant	Université de Tlemcen
<u>Invité :</u>	MALIKI Fouad	MCB	ESSA de Tlemcen

Année Universitaire : 2019/2020

REMERCIEMENTS

Ce projet n'aurait pas été possible sans l'intervention, consciente, d'un grand nombre de personnes. Je reconnais que chacune a apporté, à des degrés divers, mais avec une égale bienveillance, une contribution positive à la finalisation de ce projet.

Mes remerciements vont à mon enseignante et mon encadrante GHOMRI Latéfa de l'université de Tlemcen qui m'a initiée à la recherche et m'a prodigué de nombreux conseils avisés. La confiance et la patience qu'elle m'a accordées ont été déterminantes dans l'accomplissement de ce projet.

Je voudrais remercier CASTAGNA Pierre, professeur des universités à l'IUT de Nantes, pour m'avoir accueillie. Je lui suis également reconnaissante pour le temps conséquent qu'il m'a accordé, pour toute son aide, ses conseils précieux et son écoute qui ont été prépondérants pour la bonne réussite de ce projet. Je suis ravie d'avoir travaillé en sa compagnie car outre son appui scientifique, il n'a pas hésité à m'accueillir chaleureusement et à faire part d'une gentillesse et d'une bienveillance exceptionnelle à mon égard durant la conjoncture sanitaire due au COVID 19.

Mes sincères remerciements à CARDIN Olivier et NOUIRI Maroua pour m'avoir bien accueillie chaleureusement au sein de l'IUT de Nantes et avoir fait preuve de générosité. Je les remercie particulièrement pour leurs remarques constructives, leurs questions, et leurs idées qui m'ont permis d'améliorer la qualité de mon travail.

Je témoigne ma reconnaissance et ma gratitude à tous les enseignants qui ont apporté leur aide, ont guidé mes réflexions et ont contribué à la réussite de ce parcours universitaire ainsi qu'à ce projet, par leurs paroles, leurs écrits, leurs conseils et leurs critiques.

Ces cinq ans passés en étant étudiante en génie industriel furent riches tant sur le plan scientifique qu'humain. Un remerciement à mes ami(e)s qui ont embelli mes expériences, m'ont soutenu et épaulé pendant mes périodes de doutes. Je suis particulièrement reconnaissante aux membres d'*Archiclub* pour tous les moments de bonheur et de joie qui m'ont permis de grandir et de m'ouvrir au monde.

Remerciements

Je remercie tout particulièrement HAMZAOUI Adel pour sa relecture méticuleuse bien que pénible de ce manuscrit ainsi que YELLES CHAUCHE Abdelkrim d'avoir accepté d'orienter mes réflexions et de répondre à mes questions.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à mes parents et à ma famille qui ont toujours été là pour moi. Leur soutien inconditionnel et leurs encouragements sont pour moi les piliers fondateurs de ce que je suis et de ce que je fais.

Un remerciement aussi particulier et chaleureux à la famille qui m'a accueillie durant la pandémie, qui m'a soutenue et m'a encouragée : à Annie, Marinela, Gomette et Naïa. Malgré la distance qui me séparait de mes proches, leur confiance, leur encouragement, leur tendresse, leur amour, leur générosité d'âme et d'esprit, leur bienveillance m'ont accompagnée durant cette période.

Une pensée pour terminer ces remerciements à un être qui nous a quittés beaucoup trop tôt, une personne que j'affectionne particulièrement et qui m'a énormément marquée durant mon parcours universitaire par sa joie de vivre, sa gaité, son humour, sa gentillesse et son dévouement. Mr BENSMAIN Abderrahmane m'a toujours encouragée et poussée de l'avant, il a bercé mon âme par ses conseils, me transmettant sa passion de vivre. Il vivra à tout jamais dans mon cœur. Que dieu l'accueille dans son vaste paradis.

ABDOUNE Khadidja Farah

TABLE DES MATIÈRES

REMERCIEMENTS	II
TABLE DES MATIÈRES	IV
LISTE DES TABLEAUX.....	VIII
LISTE DES FIGURES.....	IX
LISTE DES SIGLES.....	XIII
LISTE DES ANNEXES.....	XV
INTRODUCTION GÉNÉRALE.....	1
CHAPITRE 1 INDUSTRIE 4.0 ET AU-DELÀ	4
1.1 Introduction :.....	4
1.2 Industrie 4.0 :.....	5
1.3 Les systèmes de production cyber-physiques (CPPS) :.....	6
1.4 Les leviers technologiques de l'industrie 4.0 :.....	6
1.4.1 Big Data :	7
1.4.2 Robots autonomes :.....	7
1.4.3 Les systèmes d'intégration verticale et horizontale :	7
1.4.4 Internet industriel des objets (IIOT) :.....	8
1.4.5 Cyber sécurité :.....	8
1.4.6 Infonuagique (Cloud) :.....	8
1.4.7 Fabrication additive :.....	9
1.4.8 Communication Inter Machines (M2M) :.....	9

Table des matières

1.4.9	La réalité augmentée :	9
1.4.10	La simulation :	10
1.5	Digital twin:	11
1.5.1	Les composants d'un Digital Twin :	13
1.5.2	Niveau d'intégration :	14
1.6	Les stratégies de la maintenance :	16
1.7	Revue de littérature de la maintenance prédictive :	17
1.8	Défis :	19
1.9	Conclusion :	20
CHAPITRE 2 OUTILS DE MODÉLISATION, DE SIMULATION ET DE PROGRAMMATION		21
2.1	Introduction :	21
2.2	Flexsim :	21
2.3	CIROS :	23
2.4	STEP 7:	24
2.5	NetToPlcsim :	24
2.6	OPC :	24
2.6.1	Les spécifications d'OPC :	25
2.6.2	L'architecture d'OPC :	26
2.7	KepserverEX :	26
2.8	Python et ses packages d'analyse de données :	26
2.9	Le système de production modulaire MPS500 :	27
2.9.1	Le système de transport :	28

Table des matières

2.9.2	Les différentes stations.....	28
2.10	La ligne d'assemblage :.....	30
2.11	Conclusion :.....	30
CHAPITRE 3 ARCHITECTURE DE LA MAINTENANCE PRÉDICTIVE BASÉE SUR LE DIGITAL TWIN POUR LES CPPS		31
3.1	Introduction :.....	31
3.2	Digital twin utilisé dans la maintenance prédictive :	31
3.3	Architecture d'implémentation :	32
3.3.1	Le système de production cyber-physique (CPPS).....	33
3.3.2	Acquisition des données.....	33
3.3.3	L'ensemble des modules détecteurs	34
3.3.4	Centre intelligent d'aide à la décision en maintenance	34
3.4	Machine Learning:	35
3.4.1	Régression logistique (LR) :	36
3.4.2	Réseau de neurones artificiels (ANN) :.....	37
3.4.3	Machine à vecteurs de support (SVM) :.....	38
3.5	Conclusion :.....	39
CHAPITRE 4 IMPLÉMENTATION ET ETUDE DE CAS		40
4.1	Introduction :.....	40
4.2	Etude de cas sur la ligne d'assemblage :.....	40
4.2.1	Système de production :	41
4.2.2	Communication des données :.....	43
4.2.3	Modélisation numérique sous Flexsim :.....	43

Table des matières

4.2.4	Enregistrement des données sur MySQL :	46
4.2.5	Analyse des données sur Python	47
4.2.6	Apprentissage automatique sur Python	48
4.2.7	Conclusion.....	53
4.3	Etude de cas sur les stations de la MPS 500	53
4.3.1	Fonctionnement de la station de tri :	54
4.3.2	Observation de la station sur Ciros	55
4.3.3	Programmation API.....	55
4.3.4	Test du programme API	56
4.3.5	Configuration du réseau	58
4.3.6	Modélisation 3D sur Flexsim :	62
4.3.7	Configuration de l'émulation sur Flexsim	63
4.3.8	Résultats	67
4.4	Conclusion.....	68
CONCLUSION ET PERSPECTIVES		69
BIBLIOGRAPHIE		71
TP 0 : initiation à Flexsim		74
TP 1 : Introduction aux concepts généraux de modélisation		77
TP 02 : Introduction au Process Flow		87
RÉSUMÉ.....		94
ABSTRACT		94
تلخيص		XCIII

LISTE DES TABLEAUX

Tableau 4-1 Distribution des classes de la perturbation ε	47
Tableau 4-2 Résultats de l'analyse de données.....	48

LISTE DES FIGURES

Figure 1.1 les différentes type de modélisation d'un système	10
Figure 1.1.2 les éléments d'un modèle de Digital Twin ¹⁹	14
Figure 1.3 Flux de données dans un Digital Model	15
Figure 1.4 Flux de données dans un Digital Shadow	15
Figure 1.5 Flux de données dans un Digital Twin	16
Figure 2.1 Exemple d'un modèle 3D sur Flexsim	22
Figure 2.2 Exemple d'un Process Flow sur Flexsim	22
Figure 2.3 MPS 507 FMS	28
Figure 2.4 La ligne d'assemblage de l'IUT de Nantes	30
Figure 3.1 Le schéma de l'architecture proposée	33
Figure 3.2 La courbe de la fonction Sigmoidale	37
Figure 3.3 Réseau de neurone artificiel.....	38
Figure 3.4 Machine à vecteurs support	39
Figure 4.1 Architecture d'implémentation.....	41
Figure 4.2 Système de transport de la MPS 500	41
Figure 4.3 la ligne d'assemblage de l'IUT	42
Figure 4.4 Process Flow de l'arrivée d'une palette	43
Figure 4.5 Arrivée de la palette sur le convoyeur	44
Figure 4.6 Process flow du traitement d'un lecteur	44
Figure 4.7 Arrivée de la palette au lecteur	45
Figure 4.8 Synchronisation du détecteur.....	46

Liste des figures

Figure 4.9 Connexion sur le module Database Connector	47
Figure 4.10 Organigramme de l'algorithme de classification	50
Figure 4.11 Importation des bibliothèques.....	51
Figure 4.12 Création du modèle	51
Figure 4.13 Entraînement des données	52
Figure 4.14 Matrice de confusion	52
Figure 4.15 Architecture d'implémentation.....	54
Figure 4.16 La station de Tri de la MPS 500 sur CIROS.....	55
Figure 4.17 Table des mnémoniques de la station de tri.....	56
Figure 4.18 Chargement du programme STEP 7 sur CIROS	56
Figure 4.19 Interface de EzOPC et PLCSIM lors de la simulation.....	57
Figure 4.20 Configuration d'un réseau Ethernet sur STEP 7	58
Figure 4.21 choix du protocole TCP/IP sur PLCSIM.....	59
Figure 4.22 Configuration de l'adressage sur Nettoplcsim	59
Figure 4.23 Démarrage du serveur sur Nettoplcsim	60
Figure 4.24 Configuration matérielle de Kepserver.....	60
Figure 4.25 Importation des entrées et des sorties sur Kepserver.....	61
Figure 4.26 Extraction des entrées et des sorties sur TIA Portal Exporter	61
Figure 4.27 Modèle 3D de la station de Tri sur Flexsim.....	62
Figure 4.28 Process Flow de la station de tri	63
Figure 4.29 Connexion OPC DA sur la boîte d'émulation.....	64
Figure 4.30 Interface des entrées.....	65
Figure 4.31 Interface des sorties	66

Liste des figures

Figure 4.32 Client OPC sur Kepserver.....	67
Figure 4.33 Schéma synthétisant la démarche d'implémentation	68
Figure 0.1 Interface de Flexsim.....	74
Figure 0.2 Flux d'une entité sur le modèle 3D de Flexsim.....	76
Figure 0.3 Le déplacement d'un jeton sur le Process Flow de Flexsim	76
Figure 0.4 disposition du premier modèle.....	78
Figure 0.5 les unités d'un modèle de simulation sur Flexsim	78
Figure 0.6 Connexions aux objets type	79
Figure 0.7 distribution statistique de la loi exponentielle	80
Figure 0.8 les évènements survenant lors de la création d'une pièce	81
Figure 0.9 Propriétés du stock.....	82
Figure 0.10 configuration des propriétés du stock	83
Figure 0.11 Fenêtre du Flow Item Bin	83
Figure 0.12 Propriétés du poste de contrôle	84
Figure 0.13 Configuration des Triggers	84
Figure 0.14 configuration des temps opératoires des machines.....	85
Figure 0.15 Configuration des Triggers à l'entrée du stock	85
Figure 0.16 Taux d'occupation des postes.....	86
Figure 0.17 Diagramme de Gantt.....	86
Figure 0.18 Disposition du deuxième modèle.....	87
Figure 0.19 utilisation d'une ressource mobile.....	88
Figure 0.20 établissement d'une connexion centrale.....	88
Figure 0.21 les différentes ressources mobiles.....	89

Liste des figures

Figure 0.22 Dimensionnement du stock.....	90
Figure 0.23 Process flow de l'opérateur	90
Figure 0.24 création d'un jeton.....	91
Figure 0.25 affectation d'une ressource.....	91
Figure 0.26 Séquence de tâches de l'opérateur.....	92
Figure 0.27 configuration de l'activité de chargement de pièce	92
Figure 0.28 changement de la couleur de la pièce	92

LISTE DES SIGLES

ANN	Artificial Neuronal Network
API	Automate Programmable Industriel
CIROS	Computer Integrated Robot Simulation
CoDeSys	Controller Development System
COM	Component Object Model
CPPS	Cyber Physical Production System
CPS	Cyber Physical System
DCOM	Distributed Component Object Model
DP	Decentralised Peripheral
DT	Digital Twin
E/S	Entrée/Sortie
ERP	Enterprise Resource Planning
IA	Intelligence Artificielle
IHM	Interface Homme-Machine
IIOT	Industrial Internet of Things
IMS	Intelligent Maintenance System

Liste des sigles

LR	Logistic Regression
MES	Manufacturing Execution System
ML	Machine Learning
OLE	Object Linking and Embedding
OPC	Open Platform Communication
PLM	Product Lifecycle Management
PROFINET	Process Field Net
RFID	Radio Frequency identification
SCADA	Supervisory Control And Data Acquisition
SED	Simulation à évènements discrets
SVM	Support Vectors Machines
TCP	Transmission Control Protocol
TIC	Technologies de l'information et de la communication

LISTE DES ANNEXES

Annexe A – Travaux pratiques sous flexsim74

INTRODUCTION GÉNÉRALE

Au fil du temps, l'écriture, l'agriculture ont été inventées, de grandes civilisations en Mésopotamie, dans la vallée de l'Indus et en Égypte ont vu le jour. Les grandes pyramides de Gizeh ont été érigées. De gigantesques cités comme Rome et Athènes ont été les témoins de brillantes civilisations disparues, tout comme d'autres empires qui après un essor glorieux se sont éteints tels la chute de Constantinople. Des courants de pensée ont influencé l'histoire et le monde comme la Renaissance.

Nous voilà maintenant à l'époque moderne, Nous avons construit des objets en métal qui volent dans l'espace, modifié le climat de notre planète, exploré le monde et sommes même allés sur la Lune. Une époque d'essor technologique, de miracles scientifiques capables de réaliser ce qui faisait partie du tableau de la science-fiction il y a tout juste un demi-siècle, mais qui ironiquement reste tout bêtement impuissant devant une menace micro biologique immobilisant les capacités de toute l'humanité.

Le théâtre d'événements de l'histoire qui est celle de l'Homme moderne est sans doute stupéfiant avec l'impact des grandes découvertes, des guerres mondiales et des révolutions politiques, scientifiques et notamment industrielles.

La transition de l'artisanat à l'industrialisation a vu le jour lors de la mise au point de la machine à vapeur donnant vie la production en petites séries. L'apparition de l'électricité a permis de moderniser les moyens de production, à l'aube du taylorisme et du fordisme, donnant naissance à la production de masse et à la standardisation. L'avènement de l'électronique et de l'automatisation a induit à la flexibilité des chaînes de production.

Aujourd'hui, nous vivons la révolution des TIC (Technologie de l'information et de la Communication). Les TIC font que la demande des clients est de plus en plus personnalisée. Mais les TIC ont aussi permis la 4^{ième} révolution industrielle, en favorisant l'apparition des produits intelligents, et le développement de nouveaux processus, produits et services. L'intérêt de l'exploitation de ces nouvelles opportunités offertes la révolution des TIC est démontrée par de nombreux chercheurs et entreprises qui cherchent à mettre en place des usines dites « intelligentes » capable d'une plus grande réactivité et flexibilité et caractérisée par une

Introduction générale

interconnexion des machines et des systèmes. Celles-ci est basée sur les systèmes cyber-physiques (CPS) et l'internet des objets (IoT).

Aux prémices de cette révolution, les systèmes de production ont évolué de monolithiques vers des systèmes plus autonomes, plus complexes et plus imprévisibles. Ces mêmes systèmes doivent souvent faire face à des problèmes de maintenance et des aléas de production (pannes, etc.)

L'un des concepts clés de l'industrie 4.0 cherche à remédier à ces aléas. Il imagine une copie virtuelle du système réel. Dans un premier temps, cette copie virtuelle permet de visualiser le comportement d'une chaîne de production ou d'un groupe de machines, grâce aux données détectées et aux appareils intelligents connectés. Ensuite, en synchronisant cette copie virtuelle avec le système réel, avec l'aide de modèles mathématiques, d'outils d'analyse de données et d'outils d'Intelligence Artificielle (IA), on cherche à prévoir et optimiser le comportement du système de production à chaque phase du cycle de vie du produit, et ceci en temps réel. Ce concept n'est autre que le jumeau numérique ou le « Digital Twin ».

L'objectif primaire de ce projet de fin d'étude de Master est d'implémenter un jumeau numérique sur les stations de la MPS 500 de Festo Didactique du laboratoire de productique de l'université de Tlemcen (MELT) ainsi que sur la ligne d'assemblage du Laboratoire des Sciences du Numériques de Nantes (LS2N) en utilisant le logiciel de simulation de flux Flexsim, en vue de faire de la maintenance prédictive et réactive en utilisant des techniques d'intelligence artificielle. En effet, Ce travail est le résultat d'une collaboration très fructueuse entre le laboratoire de productique de l'université de Tlemcen (MELT) et le LS2N.

Dans le premier chapitre de ce mémoire, nous allons définir l'industrie 4.0 et ses différents piliers technologiques qui nous aideront à suivre le cheminement de ces travaux. Nous nous focaliserons après cela sur le digital twin et ses niveaux d'intégration industriels pour finalement clôturer sur la définition des différentes stratégies de maintenance et une synthèse sur les recherches actuellement menées dans ce domaine.

Le deuxième chapitre est dédié aux outils logiciels de programmation, de modélisation et de communication que nous avons utilisés afin d'implémenter ce travail mais aussi une présentation de la MPS500 et de la ligne d'assemblage, le matériel physique sur lequel nous avons effectué une étude de cas.

Introduction générale

Dans le troisième chapitre, nous exposerons l'architecture proposée basée sur le digital twin et des techniques d'intelligence artificielle visant à améliorer la maintenance prédictive dans un système de production. Le dernier chapitre sera consacré à notre méthode d'implémentation d'un jumeau numérique. Nous montrerons comment on peut synchroniser la réalité et le virtuel et exploiter les données collectées afin d'améliorer la fiabilité d'un système.

CHAPITRE 1 INDUSTRIE 4.0 ET AU-DELÀ

1.1 Introduction :

Aujourd'hui, l'évolution des technologies de l'information et de la communication (TIC) a conduit à la naissance d'une nouvelle révolution industrielle. L'industrie 4.0 a conduit au besoin des entreprises de visualiser un système ou un processus dans le monde numérique permettant de fournir des informations sur la performance de celui-ci.

L'évolution des systèmes de production cyber physiques permet qu'ils soient plus réactifs et capables de s'adapter au mieux aux aléas de production comme les pannes.

Dans ce contexte, le jumeau numérique des systèmes physiques émerge comme une solution efficace, permettant de collecter des données grâce à des capteurs, mesurant ainsi le niveau d'usure des systèmes. Il aide à comprendre les problèmes plus rapidement et procéder de manière pragmatique et rapide à des adaptations pour résoudre ces problèmes. Face à ces enjeux industriels de maintenance, plusieurs travaux de recherche ont été menés dans la littérature.

L'objectif de ce travail est l'implémentation d'un Digital Twin des stations de Festo Didactique de la MPS500 et la ligne d'apprentissage de l'IUT de en utilisant le logiciel de simulation Flexsim en vue de faire de la maintenance prédictive et réactive

Dans ce chapitre, nous faisons connaître le cadre de la recherche qu'est « l'industrie 4.0 ». Nous fournirons les précisions nécessaires en ce qui concerne cette dernière en citant les principaux leviers technologiques. Nous nous focaliserons ensuite sur « le digital twin ». Par la suite, nous consacrerons une partie à la présentation de la maintenance et nous exposerons une revue de littérature au sujet des méthodes utilisées dans ce domaine en vue de connaître les progrès récents.

1.2 Industrie 4.0 :

Aujourd'hui, la nouvelle génération d'usines connectées, robotisées et intelligentes est connue sous le nom de l'industrie 4.0. Il s'agit de la nouvelle révolution numérique interconnectant le monde physique et le monde digital donnant aux machines, aux produits et aux collaborateurs la possibilité d'interagir entre eux. [1]

Apparu en 2011 pour la première fois au forum mondial de l'industrie de Hanovre naissant d'une réflexion allemande, le concept de « Industrie 4.0 » ou « Industrie du futur » a défini une nouvelle façon de voir les moyens de production [2] pour faire face aux nouveaux défis, en utilisant de nouvelles technologies, créant de nouveaux business modèles pour tendre à une transformation dite digitale.[3]

Cette révolution a pour objectif de travailler avec un niveau d'automatisation élevé assurant ainsi une efficacité et une efficience de la production et une réduction des coûts. Les entreprises travaillent ensemble dans un écosystème digital avec les fournisseurs et les clients. [3] L'idée principale est d'exploiter le potentiel des nouvelles technologies et concepts comme [4] l'utilisation de l'internet des objets, l'intégration de nouveaux processus techniques, la migration vers « smart factory » . L'Industrie 4.0 prend appui sur la communication en temps réel pour surveiller et agir sur les systèmes physiques. Son déploiement [5] requiert une intégration de différents savoir-faire propres aux technologies numériques.

Diverses stratégies nationales de développement industriel ont emboité le pas de l'Allemagne, qui dès 2011 mettait en place le projet « Industrie 4.0 ». On voit ainsi apparaître « l'usine du Futur » en France, « usine intelligente » au Japon, « High value manufacturing Catapult » au Royaume Uni, « La fabbrica Del futuro » en Italie, « Advanced manufacturing partnership » aux États Unis, ou encore le projet « Made in China 2025 » en Chine. Ces efforts abordent différentes perspectives portant sur l'émergence de nouveaux processus, produits et services.

1.3 Les systèmes de production cyber-physiques (CPPS) :

Le terme systèmes cyber-physiques (CPS) a été défini comme étant les systèmes naturels conçus par l'humain (l'espace physique) fortement intégrés avec les systèmes de communication et de contrôle (l'espace cyber). Leurs principales caractéristiques sont la décentralisation et l'autonomie.[6]

Les systèmes de production cyber physiques (CPPS), quant à eux, s'appuient fortement sur les développements les plus récents de la technologie de l'information et de la communication (TIC), ce qui a conduit à la 4e révolution industrielle. D'après [7], « *Les systèmes de production cyber-physiques sont des systèmes de systèmes d'éléments autonomes et coopératifs qui se connectent les uns aux autres de manière dépendante de la situation, à tous les niveaux de la production, des processus aux machines, en passant par les réseaux de production et de logistique, améliorant ainsi les processus décisionnels en temps réel, en réponse à des conditions imprévues et évolutions dans le temps* »

1.4 Les leviers technologiques de l'industrie 4.0 :

Les progrès technologiques ont entraîné une augmentation spectaculaire de la productivité depuis l'aube de la révolution industrielle. Les usines à moteur à vapeur au XIXe siècle, l'électrification a conduit à la production de masse au début du XXe siècle, et l'industrie est devenue automatisée dans les années 1970. Au cours des décennies qui ont suivi, les progrès technologiques industriels ne font que progressés dans le domaine de l'informatique, de la communication et de l'e-commerce. [8].

Maintenant, durant cette quatrième vague de progrès technologique, la montée en puissance de la nouvelle technologie industrielle numérique s'appuie sur des avancées technologiques fondamentales. Dans cette transformation, les capteurs, les machines, les pièces et les systèmes informatiques sont interconnectés pour créer un écosystème cyber physique. Les piliers de l'industrie 4.0 permettent de collecter et d'analyser les données rapidement, garantissant ainsi une hausse de la productivité manufacturière et stimulant la croissance industrielle et économique, car les cellules jadis isolées se transforment en un flux de production entièrement intégré, automatisé et optimisé.

1.4.1 Big Data :

Les données massives et leur analyse visent à traiter des volumes de données énormes provenant de source multiples et de formats différents, permettant ainsi une meilleure prise de décision en temps réel[5]. Selon Forrester [8], quatre dimensions définissent le Big Data : le volume de données, la variété de données, la vitesse de génération de nouvelles données et d'analyse et enfin la valeur des données [9].

Exemple : le fabricant de semi-conducteurs « Infineon Technologies » a réduit les défaillances de produits en corrélant les données capturées lors de la phase de test à la fin de la production avec les données collectées dans plus tôt dans le processus. De cette façon, Infineon a pu identifier les modèles qui aident à décharger les puces défectueuses tôt dans le processus de production et à améliorer la qualité de la production.

1.4.2 Robots autonomes :

Dans le paradigme de l'industrie 4.0, les robots deviennent plus autonomes, plus flexible et interagissent de plus en plus avec l'humain grâce aux nouvelles technologies de connectivité, de façon sécurisée, ces robots sont appelés « cobots ». Ces robots peuvent accomplir des tâches avec une précision et une intelligence remarquables tout en se focalisant sur la flexibilité[10].

1.4.3 Les systèmes d'intégration verticale et horizontale :

De nos jours, les systèmes informatiques ne sont pas complètement intégrés. Les entreprises, les fournisseurs et les clients sont faiblement liés. L'avènement de l'industrie 4.0 a permis une cohésion entre les entreprises, les départements, les fonctions. L'industrie 4.0 comporte trois dimensions d'intégration : une intégration horizontale de la chaîne de création de valeur, une intégration verticale pour une reconfiguration des processus de production et une intégration d'ingénierie durant le cycle de vie d'un produit [10].

Exemple : Dassault Systèmes et BoostAeroSpace ont lancé une plateforme de collaboration pour l'industrie européenne de l'aérospatiale et de la défense. La plate-forme, AirDesign, sert d'espace de travail pour la collaboration de conception et de fabrication et est disponible en tant que service

sur un cloud privé. Elle gère la tâche complexe d'échanger des données de produits et de production entre plusieurs partenaires.

1.4.4 Internet industriel des objets (IIOT) :

Selon la norme ISO/IEC 30141, l'internet des objets est défini comme étant une infrastructure d'interconnexion d'entités physiques, de systèmes, de sources d'information et de services intelligents traitant les informations du monde physique et du monde virtuel.

Dans l'industrie 4.0, l'internet des objets industriel permet de connecter une variété de ressources numériques les reliant à des technologies embarquées leur permettant d'interagir, de communiquer, de collecter, d'échanger et d'analyser. Ceci permet de décentraliser la prise de décision, avec une réactivité en temps réel.

Internet des objets(IoT) devrait aussi être connu sous le nom de l'internet de tout (IoE) qui consiste en l'internet des services(IoS), l'internet des services manufacturiers(IoMs), internet des personnes (IoP).[11]

Exemple : Bosch Rexroth, un fournisseur de systèmes d'entraînement et de contrôle, a équipé une installation de production de vannes avec un processus de production décentralisé semi-automatisé. Les produits sont identifiés par des codes d'identification par radiofréquence, et les postes de travail « savent » quelles étapes de fabrication doivent être effectuées pour chaque produit et peuvent s'adapter pour effectuer l'opération spécifique.

1.4.5 Cyber sécurité :

A l'aube de l'industrie 4.0, l'échange de données industrielles et le contrôle à distance des systèmes de production laissent apparaître un risque critiques de la cyber sécurité. Cela a fini par accroître les mesures de sécurité en faisant recours à des standards et des protocoles de communication plus fiables comme le VPN, P2P, PIR, TLS, DNSSEC et les étiquettes RFID.[6]

1.4.6 Infonuagique (Cloud) :

Le cloud donne accès à des ressources informatiques partagées (Réseaux, serveurs, stockage, applications et services) en externalisant la gestion de celles-ci ce qui implique une réduction des

coûts. L'infonuagique facilite le partage de données entre les sites et les systèmes. Dans l'industrie 4.0, de nombreuses ressources sont impliquées dans la virtualisation des infrastructures, nous notons : Software as a service (SaaS), Infrastructure as a service (IaaS), Platform as a service (Paas).[5]

1.4.7 Fabrication additive :

Les besoins et les exigences des clients changent continuellement, les entreprises se doivent de relever le défi et de s'adapter à ces derniers en personnalisant les produits tout en réduisant les délais de fabrication.

Dans le contexte de l'industrie 4.0, elles adoptent de plus en plus la fabrication additive comme l'impression 3D, principalement utilisée pour le prototypage et la production des composants personnalisés de petits lots, dont la conception est complexe. La production devient plus rapide et moins chère en utilisant des technologies tels que la fusion de laser sélective ou la méthode de dépôt par laser.[5]

1.4.8 Communication Inter Machines (M2M) :

Les technologies de communication inter machines connaissent une croissance. Les technologies M2M créent des réseaux locaux en s'appuyant sur des protocoles et des standards tels que : TCP/IP, OPC DA, OPC UA, DLNA...Etc.[5]

1.4.9 La réalité augmentée :

L'industrie peut utiliser la réalité augmentée pour fournir aux travailleurs des informations en temps réel afin d'améliorer la prise de décision et les procédures de travail. Les travailleurs peuvent recevoir des instructions de réparation sur la façon de remplacer une pièce particulière lorsqu'ils examinent le système réel à réparer.

La réalité augmentée vient superposer en temps réel sur l'environnement réel physique des informations à la réalité environnante. Elle permet de redéfinir la manière de surveiller, de contrôler, d'optimiser ou encore de rendre autonome les produits, les processus ou les services de l'Industrie 4.0.

1.4.10 La simulation :

La simulation est déjà présente dans différentes sphères de l'ingénierie. Son application dans l'industrie 4.0 consiste à simuler le processus de fabrication. Elle vise à tester et optimiser le fonctionnement et le paramétrage afin de visualiser les changements avant de les transposer dans le monde réel. Elle consiste à imiter le comportement d'un système réel ou bien à prédire le comportement d'un système sous-jacent qui n'existe pas encore

Il existe différentes méthodes de modéliser et de simuler un système réel comme illustré dans la figure 1.1 ci-dessous ou nous nous intéresserons à la simulation des flux particulièrement.

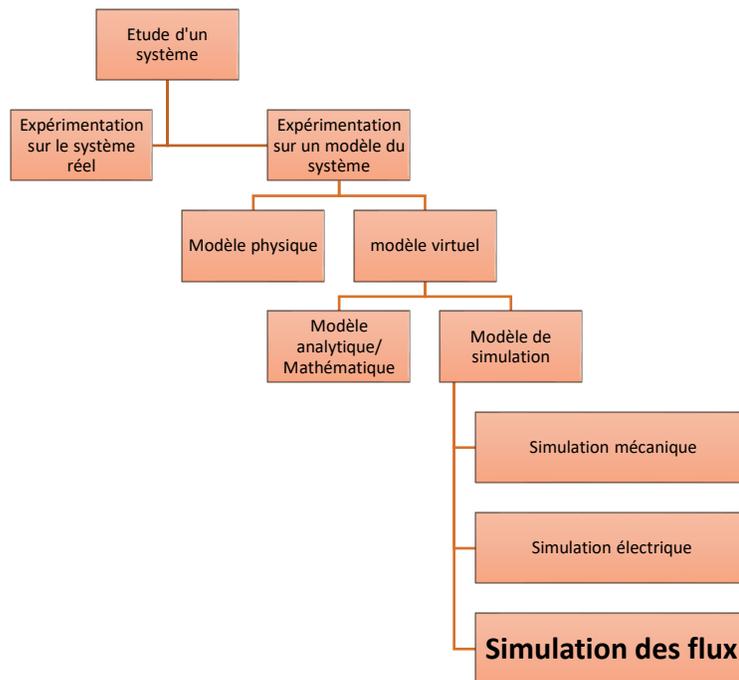


Figure 1.1 les différents type de modélisation d'un système

- **La simulation de flux :**

La construction d'un modèle dynamique et l'utilisation expérimentale de ce dernier sont toutes les deux intégrées dans la simulation des flux de production. Ces modèles sont une représentation

des systèmes réels capable de reproduire le fonctionnement de ce dernier et ont une capacité de décrire la chaîne de production avec un degré de précision important.

Elle est principalement utilisée afin d'étudier les flux physiques qui sont définis comme étant un déplacement d'une entité physique ou immatérielle dans un système donné

- **Simulation à événements discrets (SED) :**

Dans la simulation, on distingue : les modèles statiques, pour lesquels le temps n'est pas un paramètre et les modèles dynamiques, pour lesquels le comportement est une fonction du temps.

A l'intérieur des modèles dynamiques de simulation, on retrouve les modèles à événements discrets dans lesquels les états changent lors de la survenance d'événements tels le début ou la fin d'une opération. Dans une simulation à événements discrets, les flux essentiels que l'on examine sont composés d'éléments isolables que l'on peut appeler "Entités".

1.5 Digital twin:

La première définition du Digital Twin a été donnée en 2002 par Michael Grieves à l'Université du Michigan dans le cadre d'une présentation de l'industrie concernant la gestion du cycle de vie du produit (PLM) [12], où il expliquait que la représentation numérique devrait inclure de manière optimale toutes les informations concernant le système actif qui pourraient être potentiellement obtenues à partir de son inspection approfondie dans le monde réel.

Mais le digital twin existait bien avant, l'ère pionnière de l'exploration spatiale a vu le jour il y a environ 60 ans. À ce moment-là, La NASA avait comme défi de concevoir des objets qui pourraient voyager si loin qu'ils seraient au-delà de la capacité humaine à voir, surveiller ou les modifier directement. La NASA conçut alors un jumeau numérique d'un système physique utilisé pour faire fonctionner, simuler, tester un système sous-jacent. En juillet 1969, la navette Apollo 11 amenant Neil Armstrong sur la lune, était l'un des premiers appareils dotés d'un twin.

Le jumeau numérique (en allemand: digitaliseur Zwilling) [13] fut défini pour par la NASA en 2010 comme une simulation d'un véhicule ou d'un système utilisant le meilleur modèle possible afin de représenter l'image de son jumeau réel.

Une définition plus détaillée et dans le domaine de la recherche largement reconnue est donnée dans [14]: le « *jumeau numérique est une simulation multi-physique, multi-échelles et intégrée d'un produit complexe et utilise les meilleurs modèles physiques disponibles, mises à jour par les capteurs pour refléter la vie de son jumeau correspondant.* »

En termes de fabrication, la définition suivante d'un Digital Twin est donnée dans [15]: « *Le DT consiste en une représentation virtuelle d'un système de production qui est capable de fonctionner sur différentes disciplines de simulation caractérisées par la synchronisation entre le système virtuel et réel, grâce aux données détectées et aux appareils intelligents connectés, aux modèles mathématiques et à l'élaboration de données en temps réel. Le rôle d'actualité dans les systèmes de fabrication de l'industrie 4.0 est d'exploiter ces fonctionnalités pour prévoir et optimiser le comportement du système de production à chaque phase du cycle de vie en temps réel.* »

L'industrie et les recherches académiques définissent un jumeau numérique de plusieurs façons différentes. Selon certains, un jumeau numérique est un modèle intégré d'un produit conforme à la construction destiné à refléter les défauts de fabrication et être continuellement mis à jour pour inclure l'usure subie pendant l'utilisation [16]. D'autres définitions largement diffusées décrivent le jumeau numérique comme un modèle numérique animé par les capteurs d'un objet physique qui simule l'objet dans un cadre réel.

Bien que la terminologie ait changé au fil des ans, le concept de création d'un jumeau numérique et physique en une seule entité est resté le même depuis son émergence.[17] Le jumeau numérique est basé sur le monde réel, mesurant en temps réel des données à travers un tableau de bord. Ces mesures nous permettent de visualiser l'évolution de système ou du processus dans le monde numérique, permettant de fournir des informations sur la performance de celui-ci.

Aujourd'hui, les entreprises utilisent de plus en plus les fonctionnalités des jumeaux digitaux des systèmes physiques, permettant de collecter des données grâce à des capteurs, mesurant ainsi le niveau d'usure des systèmes. Mais aussi avancées, et puissantes que ses fonctions soient-elles, l'objectif d'un digital twin reste le même, comprendre les problèmes plus rapidement et procéder de manière pragmatique et rapide aux commandes nécessaires pour résoudre ces problèmes.

Un jumeau numérique diffère des modèles de conception assistée par ordinateur (CAO) ou de la technologie de l'internet des objets (IoT). La conception assistée par ordinateur est totalement

encapsulée dans un ordinateur à l'intérieur d'un environnement simulé modélisant des objets et des systèmes plutôt complexes. [18] Quant à l'internet des objets, il permet de mesurer la position ou de donner le diagnostic d'un objet donné. Mais aucune des deux technologies ne permet une interaction entre le système réel et le système virtuel. C'est sûrement l'interactivité forte, complète en temps quasi réel entre le monde physique et le monde réel qui constitue la véritable puissance du jumeau numérique. Il peut promettre des modèles riches et réalistes, analysant continuellement le flux de données récoltées. Il peut être utilisé tout au long du cycle de vie du produit en vue de simuler, optimiser et prédire le système de production.

1.5.1 Les composants d'un Digital Twin :

La figure présente ci-dessous représente un processus de fabrication réel et son jumeau numérique virtuel ainsi que les éléments qui les composent et les relie [19], nous citons :

- **Les capteurs** qui permettent au jumeau numérique de récolter les données en temps réel afin de reproduire le même comportement que le processus de fabrication.
- **Les données** récoltées en temps réel ainsi que les données opérationnelles de l'entreprise, on peut retrouver même les dessins techniques, les données externes sont combinées.
- **Intégration des données** communiquées par les capteurs au monde numérique grâce à la périphérie, les interfaces de communication et la sécurité.
- **Analyse des données** au moyen de simulation et de visualisation par le jumeau numérique.
- **Actionneurs** – Le jumeau numérique va produire la même action que le modèle réel au moyen des informations transmises par les actionneurs.

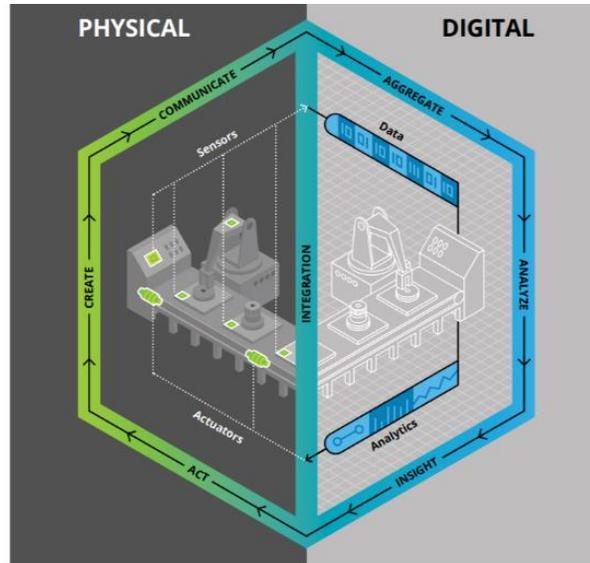


Figure 1.1.2 les éléments d'un modèle de Digital Twin[19]

1.5.2 Niveau d'intégration :

Sur la base des définitions données d'un jumeau numérique dans n'importe quel contexte, on pourrait identifier une compréhension commune des jumeaux numériques, en tant que contreparties numériques d'objets physiques. Dans ces définitions, les termes Digital Model, Digital Shadow et Digital Twin sont souvent utilisés comme synonymes. Cependant, les définitions données diffèrent dans le niveau d'intégration des données entre l'homologue physique et numérique. Certaines représentations numériques sont modélisées manuellement et ne sont liées à aucun objet physique existant, tandis que d'autres sont entièrement intégrées à l'échange de données en temps réel. Par conséquent, les auteurs souhaitent proposer une classification des jumeaux numériques en trois sous-catégories, en fonction de leur niveau d'intégration des données. [12]

- **Digital model :**

Un modèle numérique est une représentation numérique d'un objet physique existant ou planifié qui n'utilise aucune forme d'échange de données automatisé entre l'objet physique et l'objet numérique.

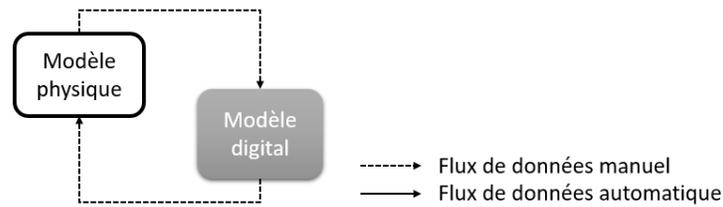


Figure 1.3 Flux de données dans un Digital Model

- **Digital Shadow :**

Sur la base de la définition d'un modèle numérique, s'il existe en outre un flux de données unidirectionnel automatisé entre l'état d'un objet physique existant et un objet numérique, on pourrait se référer à une telle combinaison comme ombre numérique. Un changement d'état de l'objet physique entraîne un changement d'état de l'objet numérique, mais pas l'inverse.

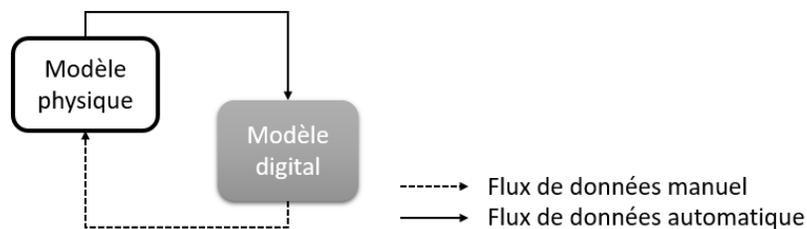


Figure 1.4 Flux de données dans un Digital Shadow

- **Digital Twin :**

Si, en outre, les flux de données entre un objet physique existant et un objet numérique sont entièrement intégrés dans les deux sens, on pourrait l'appeler Digital Twin. Dans une telle combinaison, l'objet numérique pourrait également agir comme instance de contrôle de l'objet physique. Il peut également y avoir d'autres objets, physiques ou numériques, qui induisent des changements d'état dans l'objet numérique. Un changement d'état de l'objet physique entraîne directement un changement d'état de l'objet numérique et inversement.

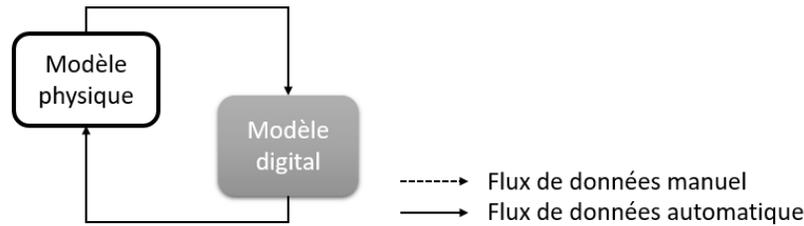


Figure 1.5 Flux de données dans un Digital Twin

1.6 Les stratégies de la maintenance :

La norme AFNOR NF X 60-010 a défini la maintenance comme étant « *un ensemble des actions permettant de maintenir ou de rétablir un bien dans un état spécifié ou en mesure d'assurer un service déterminé. Bien maintenir, c'est assurer ces opérations au coût optimal.* »

Les stratégies de maintenance sont classées en deux groupes principaux: réactif et préventif [20]. La maintenance réactive, représente les interventions sur un équipement une fois que celui-ci est défaillant. Cette stratégie de maintenance corrective permet de ne pas élaborer un calendrier de maintenance à l'avance et s'assure que rien n'est fait tant que tout se passe bien. Cependant, dès que cela devient nécessaire, l'équipement souffre de temps de panne et d'un coût de réparation importants.

Afin d'éviter ces pertes, effectuer des interventions de maintenance préventive sur un équipement avant sa défaillance est devenue assez courant. Ces opérations sont planifiées périodiquement sans interrompre la production. Cependant, celle-ci peuvent être faites plus souvent que nécessaire.

Une stratégie plus efficace qui prend en compte l'état réel de l'équipement ainsi que toutes ses pannes précédentes, serait la maintenance prédictive, qui est basée sur le suivi de l'état de la machine à des intervalles réguliers ou en continu.

1.7 Revue de littérature de la maintenance prédictive :

De l'inspection visuelle et des signes qui peuvent être détectés avec les yeux, les oreilles et le nez aux techniques avancées et élaborées basées sur des modèles d'intelligence artificielle, la maintenance a évolué de manière sauvage.

Les capteurs sont désormais disponibles pour augmenter la fiabilité, réduire les coûts mais aussi analyser et prévoir les dégradations et les pannes en introduisant la maintenance prédictive. Sujet de recherche dans le paradigme de l'industrie 4.0, appelé aussi « surveillance en ligne », « maintenance conditionnelle » ou « maintenance basée sur les risques », où des mesures sont prises en continu pour détecter les tendances et anticiper les problèmes, permettant une réponse immédiate aux pannes potentielles en utilisant des méthodes d'apprentissage automatique (ML) et autres. Les systèmes industriels peuvent continuer à fonctionner sans interruption inutile. Lorsqu'une maintenance est nécessaire, elle est très ciblée - les techniciens sont informés des composants qui doivent être inspectés, réparés et remplacés.

Différents projets de recherche et travaux connexes ont été réalisés pour tenter d'approcher cette problématique. En voici quelques-uns :

Afin d'éviter les temps d'arrêt importants et les coûts de réparation dus à une maintenance corrective classique, les constructeurs des avions s'intéressent de plus en plus à la stratégie de maintenance prédictive [5]. Il s'agit d'un type de stratégie de maintenance préventive basée sur des mesures prises en continu pour détecter les défauts et anticiper les problèmes. [21] Les auteurs ont revu les approches, les méthodes et les outils liés à la maintenance prédictive dans les systèmes de fabrication et proposent une plate-forme intégrée de maintenance prédictive.

Dans [22] Nguyen et al. ont proposé un examen des approches basées sur la simulation, qui ont été largement utilisées dans le contexte de la maintenance. Dans ces approches, le comportement du système est reproduit et simulé à travers les résultats du système respectif.

Récemment, les techniques d'intelligence artificielle et les applications Big Data fournissent des supports techniques pour le développement efficace du système de production en collectant des données avec précision et en temps opportun, en analysant et en traitant les données, en identifiant les causes profondes, en réduisant les coûts et en tirant des informations précieuses pour

l'amélioration de la maintenance [23]. Dans la littérature, il y a eu plusieurs travaux sur le rôle de l'intelligence artificielle en ce qui concerne la maintenance du système de production [24].

L'apprentissage automatique (ML) est largement utilisé dans la surveillance des conditions et la prédiction des défauts et l'analyse prédictive. Les techniques de ML sont une branche des méthodes d'IA basées sur l'utilisation d'énormes quantités de données pour apprendre et identifier des modèles.

Dans Shcherbakov et al. [25] les auteurs ont proposé un modèle conceptuel pour un système d'aide à la décision proactif basé sur des analyses prédictives en temps réel et conçu pour la maintenance de systèmes cyber-physiques afin d'optimiser leurs temps d'arrêt. Chaudhuri A. [26] a proposé une machine à vecteur de support flou modifié hiérarchique (HMFSVM) pour comprendre les tendances des défaillances du véhicule. La méthode proposée est comparée aux approches couramment utilisées comme la régression logistique, les forêts aléatoires et les machines à vecteurs de support. Les auteurs de [27] Lee et al. ont proposé une architecture de référence basée sur le deep learning pour le cyber-système physique. Ils explorent le concept d'une machine à commande numérique (CNC) à réseaux de neurones utilisée dans le contrôle d'un atelier.

Les experts du centre pour le système de maintenance du renseignement (IMS) a créé Watchdog Agent Technology [28], une approche pour l'évaluation et la prévision de la dégradation des performances des produits, pour la modélisation et la prise de décision avec une interaction humaine nécessaire. Cette technologie comprend l'analyse du domaine temporel, l'analyse en composantes principales (ACP), les logiques floues (FLS), la régression logistique (LR), le réseau de neurones artificiels (ANN), les réseaux de croyances bayésiennes et les machines à vecteurs de support (SVM) [29]. Un système intelligent de maintenance prédictive (SIMAP) [30] a été développé pour le diagnostic en temps réel des processus industriels basés sur le réseau neuronal pour détecter les anomalies. Dans [31], la logique floue est utilisée pour fournir une modélisation du comportement d'une expérience de maintenance intégrée au système de maintenance intelligent proposé. Pour estimer la dégradation de la défaillance du roulement dans [32], LR a été utilisée en combinaison avec la machine à vecteur de pertinence (RVM) afin de prédire la probabilité de défaillance.

Les diverses applications des réseaux de neurones artificiel (ANN) [33] dans l'évaluation des risques de panne et l'analyse de détection précoce de panne ont été passées en revue avec des exemples de leur utilisation dans les cas de maintenance prédictive. Alors que dans[34], la machine à vecteurs de support (SVM) a été utilisée pour le diagnostic des pannes du système de freinage hydraulique automobile.

Comme nous pouvons le conclure de la revue de la littérature, plusieurs méthodes ont été proposées pour la maintenance prédictive (modélisation mathématique, techniques basées sur la simulation, etc.). Les méthodes citées précédemment tiennent compte des masses de données accumulées au fil des ans à partir de capteurs intégrés de CPPS (données historiques) pour prendre des décisions de maintenance efficaces. Peu de travaux utilisent les données en temps réel pour détecter la déviation du système et traitent en temps réel le dysfonctionnement du système.

1.8 Défis :

Bien que l'implémentation de la maintenance prédictive par le biais d'un jumeau numérique offre de nombreux avantages, son déploiement peut poser les défis suivants:

Un jumeau numérique doit refléter précisément toutes les propriétés d'un jumeau physique, y compris celles mécaniques (pression d'aspiration, température de conception, etc.) et électriques (capacité, conductivité, etc.).

Des plans détaillés des défaillances d'une machine sont nécessaires. Pour prévoir les défaillances, un jumeau numérique doit être alimenté avec des données sur les modes de défaillance de l'équipement. Ces données doivent être collectées pendant une longue période pour observer une machine tout au long de son processus de dégradation.

Un jumeau numérique nécessite un remodelage avec tout changement dans la configuration ou l'état de l'élément de l'équipement. Toute modification affectant les performances de l'équipement nécessite un changement de son modèle et des algorithmes sous-jacents. De telles modifications au niveau de la machine ou au niveau de l'usine ne sont pas toujours reflétées dans les spécifications de l'usine, donc, ne peuvent pas être simulées avec précision, ce qui intensifie le risque d'erreurs.

Bien que le déploiement d'une maintenance prédictive basée sur le digital twin nécessite beaucoup de temps et de travail, la technologie offre la possibilité de reconnaître en temps opportun les

perturbations des performances des actifs, de prévoir les problèmes potentiels et de simuler divers scénarios de maintenance. Il aide les entreprises à éliminer les temps d'arrêt des machines, à réduire les coûts de maintenance des équipements, à améliorer la fiabilité des équipements et à prolonger leur durée de vie.

La complexité de l'utilisation de l'intelligence artificielle dans l'automatisation industrielle nécessite que les fabricants collaborent avec des spécialistes pour trouver des solutions personnalisées.

Pour cela un certain nombre d'éléments doivent être considérés pour répondre aux besoins de l'implémentation :

- Collecte de données historiques
- Capture de données en direct via des capteurs
- Agrégation de données
- Connectivité via les protocoles de communication Intégration avec les automates
- Applications IA: apprentissage automatique et autres techniques

1.9 Conclusion :

Dans ce chapitre, nous avons commencé par une présentation du paradigme de l'industrie 4.0 et des systèmes de production cyber-physiques ainsi que les piliers technologiques principaux. Nous avons ensuite exposé le digital twin ainsi que les différents concepts qui l'entourent. Puis, nous avons présenté les stratégies de maintenance nous sommes plus particulièrement intéressés à la maintenance prédictive et les différents travaux de référence qui ont étudié ce concept.

Ce chapitre met en évidence l'apport de nos travaux et situe la problématique dans le contexte de recherche global ou nous avons défini les généralités afin de comprendre le cheminement de ce qui suivra.

Dans le chapitre 2, nous présenterons les outils logiciels et matériels que nous avons eu à notre disposition afin d'implémenter le digital twin.

CHAPITRE 2 Outils de modélisation, de simulation et de programmation

2.1 Introduction :

Le défi d'implémenter un digital twin s'avère de taille, en effet plusieurs fonctionnalités doivent être assurées comme la programmation des automates programmable, la collecte des données à partir du système de production, la communication, l'agrégation et l'analyse des données. Pour cela, différents outils de programmation, de communication et de modélisation ont été utilisés. Dans ce chapitre, nous présenterons l'aspect matériel, constitué de la chaîne de production MPS500 et ses stations ainsi que la ligne d'assemblage mais aussi l'aspect logiciel comme Flexsim, CIROS, SIMATIC STEP 7, OPC...etc. dans le but de comprendre l'utilité de chaque outil et son rôle dans l'architecture globale.

2.2 Flexsim :

Flexsim est un logiciel de simulation d'événements discrets puissant et simple à utiliser permettant de construire des modèles en trois dimensions d'un système réel fournissant des animations graphiques réalistes et une analyse de performance importante. Il peut être utilisé dans la production, la santé, l'exploitation minière, la logistique, etc.

Un modèle de simulation sous Flexsim peut être composé de deux principales parties :

- **Un modèle 3D** du système physique qui peut représenter les machines, les convoyeurs, les systèmes de stockage, les moyens de manutentions, etc.

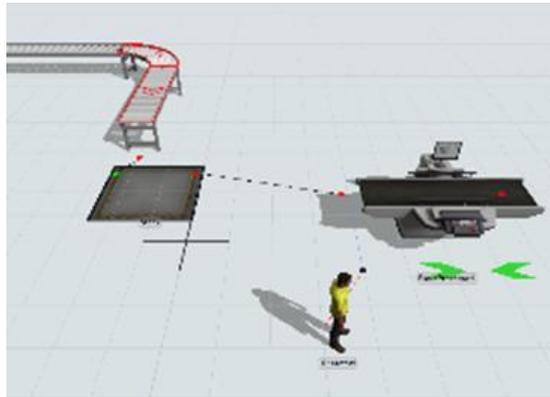


Figure 2.1 Exemple d'un modèle 3D sur Flexsim

Un flux est modélisé par le déplacement des flow items, ces derniers représentent le mouvement des produits dans le système.

- **Un ou plusieurs Process Flow** décrivant l'enchaînement des comportements logiques du système. Il est représenté sous forme d'un ensemble d'étapes ou chacune représente une tâche et peut nécessiter une ou plusieurs ressources. On peut le considérer comme un organigramme alternatif à la programmation.

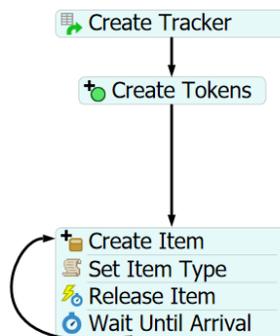


Figure 2.2 Exemple d'un Process Flow sur Flexsim

Le Process flow est parcouru par des jetons qui modélisent l'évolution des activités du Process Flow, un peu comme les jetons d'un réseau de Pétri.

2.3 **CIROS** :

CIROS est un outil pédagogique de simulation 3D dédié à la modélisation des dispositions du matériel et des processus des chaînes pédagogiques FESTO. Il permet la simulation des cellules de travail robotiques et des usines de fabrication automatisées et la visualisation des séquences complexes.

Il met à disposition des exemples basés sur des applications pratiques, les exercices se concentrent sur les procédures opérationnelles avec une approche holistique du processus d'apprentissage. CIROS donne à l'utilisateur des modèles préconfigurés des stations commercialisées par FESTO Didactique, lui permettant ainsi de se familiariser avec le matériel sans risque d'endommagement des machines

Il assure les fonctionnalités suivantes :

- **Familiarisation avec la méthodologie et la fonctionnalité des stations automatisées :**

Il permet de se familiariser avec les composants d'entraînement, les connexions, les schémas de circuit électriques et pneumatiques, il permet d'apprendre également à utiliser les stations automatisées.

- **Programmation d'API :**

Ciros comprend un environnement didactique virtuel qui se concentre sur les systèmes pilotés par API. Les modèles peuvent être immédiatement pilotés au moyen de l'API S7 virtuel intégré, à partir de STEP 7 ou du TIA Portal au moyen du contrôleur SIMATIC simulé S7-PLCSIM ou, via EasyPort, à partir de n'importe quel API matériel externe.

Chaque station d'une installation possède sa propre commande virtuelle dotée d'un programme qui lui est propre et qui peut être modifié ou entièrement recréé à tout moment.

CIROS assiste l'utilisateur dans cette tâche en simulant le comportement du processus, en se familiarisant avec les composants et le mode de fonctionnement des automates mais aussi grâce à la programmation de ces derniers et la création de programmes stables.

- **La programmation de robots :**

CIROS met à disposition de l'utilisateur des cours et des formations permettant d'apprendre la manipulation et la programmation des robots industriels d'une façon simple et efficace et la mise en service des systèmes robotiques.

2.4 STEP 7:

STEP7 est le logiciel Siemens IEC 61131 pour la configuration et programmation des systèmes d'automatisation SIMATIC S7. STEP7 V 5.x est utilisé comme système de programmation pour l'automate central avec CPU 314C-2 PN/DP de la MPS500.

Ce logiciel permet d'écrire des programmes structurés avec des blocs paramétrables, de vérifier la cohérence et la logique du programme, faire une configuration du matériel et du réseau et injecter les programmes aux API.

Les outils List Statement, Ladder Diagram et Function Block Diagram sont fournis en standard par Siemens, tandis que S7-SCL et S7-GRAPH sont des outils d'ingénierie facultatifs.

L'outil S7-PLCSIM, fourni en standard, émule le matériel du contrôleur et permet le test fonctionnel des programmes API. Ce contrôleur virtuel peut être couplé à la simulation.

2.5 NetToPlcsim :

L'outil NetToPlcsim permet d'utiliser le simulateur d'api virtuel PLCSIM de siemens avec d'autres logiciels via un protocole TCP/IP. Il fonctionne comme un serveur de données ou simulateur de port Ethernet afin de connecter les logiciels de SIMATIC Manager avec un logiciel tiers sans disposer de matériel physique.

2.6 OPC :

Lorsque l'automate ne dispose pas d'interface de communication ou plusieurs marques d'automates sont utilisés traitant différents formats de données, l'utilisation d'un serveur OPC s'avère la meilleure solution. Cette technologie client-serveur assure la communication entre par

exemple un API et une troisième application tierce, telles que les systèmes IHM / SCADA ou les outils de simulation.

L'OPC (Open Platform Communications) est un standard de communication qui permet l'interopérabilité entre les différents fabricants de logiciels et de matériels permettant la sécurisation des flux de données lors de leur circulation.

En règle générale, le serveur OPC peut être fourni par un fabricant d'API ou être un outil tiers (par exemple de Softing, Matrikon, Kepware). Il communique avec l'automate et échange les données avec les clients OPC connectés.

De nombreux outils de simulation tels que par ex. Arena, AutoMod, CIROS Studio, Delmia Automation, Demo3D / Emulate3D, Experior, Process Simulate, SIMIT, Tara VRcontrol, Visual Components 3D Create / 3D-Simulate, Virtual Universe et WinMOD fournissent un client OPC.

2.6.1 Les spécifications d'OPC :

L'OPC et ses différentes spécifications classiques, basés sur les technologies Microsoft Windows OLE / COM / DCOM peuvent être séparées en deux catégories. La première inclut :

- OPC DA (Data Access) : la lecture et l'écriture des données en temps réel
- OPC HDA (Historical Data Access) : appliqué sur les données historiques horodatées fournies par les appareils.
- OPC AE (Alarms and Events) : utilisé pour le contrôle des systèmes et l'envoi des alarmes au logiciel.

La deuxième catégorie regroupe une seule spécification OPC Architecture Unifiée (UA). Elle élimine la nécessité d'utiliser un ordinateur Microsoft Windows avec les versions antérieures d'OPC. UA combine la fonctionnalité des interfaces OPC existantes avec de nouvelles techniques comme XML et les services Web. Ce protocole intègre de base des mécanismes de sécurité. Flexible et complètement indépendant, OPC UA est considéré comme le protocole de communication idéal pour la mise en œuvre de l'Industrie 4.0.

2.6.2 L'architecture d'OPC :

L'architecture d'un réseau fonctionnant avec le standard OPC se compose toujours de 3 parties principales :

1) Un dispositif matériel ou logiciel

Ce dispositif matériel ou logiciel contient les données à exploiter.

2) Un serveur OPC

Un serveur OPC est un logiciel qui va collecter les données à exploiter. Il existe des serveurs OPC pour les différentes marques d'automates (Siemens, Allen Bradley, Omron, GE, Schneider, Honeywell, Emerson, Yokogawa, ABB, etc ...) et des serveurs OPC pour des applications comme Excel (DDE), pour les bases de données (ODBC), pour des protocoles de communication connus comme le Modbus ou encore DNP3.

3) Des clients OPC

Un client OPC est un logiciel qui met en œuvre les spécifications du standard OPC et qui peut communiquer avec tout serveur OPC. Un client OPC peut se connecter à un serveur OPC de n'importe quel fabricant pour échanger avec d'autres clients OPC.

2.7 KepserverEX :

KepserverEX est un serveur de communication OPC, certifié par la fondation OPC, parmi les solutions les plus répandues. Cette dernière dispose d'une liste d'interfaces de communication dont OPC DA, HDA, AE, etc.

KepserverEX se place de nos jours au cœur de l'industrie 4.0 et l'internet industriel des objets grâce à l'interopérabilité des systèmes au travers d'un système de transmission des données fiable et sécurisé.

2.8 Python et ses packages d'analyse de données :

Python est un langage dynamique et extensible, qui permet une approche modulaire et orientée objet de la programmation. Il connaît depuis quelques années grâce à l'essor de la data science une

grande utilisation. Il possède des différents packages spécifiquement adaptés aux statistiques, l'analyse de données, ou encore la visualisation.

Les bibliothèques de science des données et d'apprentissage automatique les plus populaires telles que Scikit Learn, Numpy, Panda et Matplotlib permettent d'écrire du code élégant et compact, mais également d'implémenter des modèles et de résoudre des problèmes rapidement. Nous utiliserons celle-ci dans l'implémentation de l'algorithme de la maintenance prédictive.

- **Scikit learn** est l'un des packages Python les plus utilisés pour le Machine Learning. Il comprend de nombreuses méthodes pour faire de la classification, de la régression ou encore du clustering.
- **Numpy** est utilisé pour la manipulation des vecteurs et des matrices. Il permet de gérer des tableaux, de les importer et les exporter et de travailler dessus.
- **Panda** donne la possibilité de manipuler de grands volumes de données et les structurer de manière simple et intuitive.
- **Matplotlib** est considéré comme un package puissant et complet pour la visualisation des données.

2.9 Le système de production modulaire MPS500 :

La chaîne de production MPS-500 située au laboratoire de génie productique MELT de la faculté de technologie de Tlemcen est un système modulaire conçu par FESTO ayant pour but l'apprentissage et la formation didactique facilitant l'appréhension, la familiarisation avec le matériel industriel et l'interaction avec les différents composants mécaniques, électriques, pneumatiques, les interfaces de communication, etc.

Composé de plusieurs stations communiquant avec un système de transport central permettant ainsi l'échange de l'information, celles-ci peuvent être disposées différemment en respectant un certain ordre donnant ainsi une référence propre à chaque configuration.

Nous retrouvons dans le laboratoire de productique MELT, la chaîne de production MPS – 500 configurée en MPS 507 FMS, se compose de six principales stations reliées par un système de convoyage.

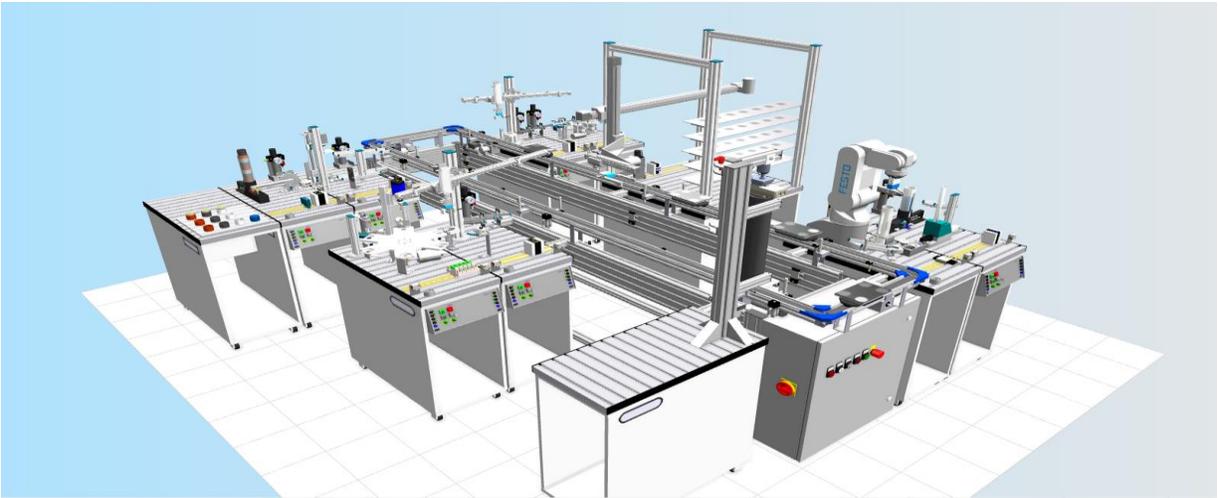


Figure 2.3 MPS 507 FMS

2.9.1 Le système de transport :

Le système de transport, considéré comme l'élément central de la chaîne de production, il sert à transporter les pièces à usiner d'une station à une autre et échanger les données entre celles-ci. Afin de pouvoir positionner les pièces à usiner et les transmettre aux autres stations, le système de transport nécessite plusieurs composants. Porte-pièces, palettes, unité de levage et d'indexation sont quelques-uns d'entre eux.

2.9.2 Les différentes stations

Le système de transport se charge de transporter les pièces d'une station à une autre. On retrouve six positions de travail :

- **Position de travail 1**

Les stations de distribution et de contrôle se trouvent à la position de travail 1. Sur la station de distribution, les pièces à usiner sont séparées et transmises à la station de contrôle. La station de distribution n'est pas directement reliée au système de transport, elle se trouve à côté de la station de contrôle. Celle-ci est accouplée au système de transport et assure outre le contrôle des pièces à usiner, le transfert des pièces à usiner au système de transport.

Le flux des matériaux commence à cette position de travail.

- **Position de travail 2**

La station de manipulation se trouve à la position de travail 2, elle assure le transport des matériaux vers **la station d'usinage**. Celle-ci se trouve directement à côté de la station de manipulation et est responsable du perçage d'un trou puis du contrôle du perçage.

- **Position de travail 3**

Un système vidéo est installé à la position de travail 3, la caméra est montée directement sur le système de transport et se trouve au-dessus de la position de travail. De cette manière, aucun déplacement de la pièce à usiner n'est nécessaire. La caméra est posée sur un chariot MPS ou un plan de travail.

- **Position de travail 4**

La station d'assemblage robotisée se trouve à la position de travail 4. Elle se compose de deux chariots MPS. Le robot RV-2AJ est monté sur le chariot se trouvant directement à côté du système de transport. Il assure la manipulation des pièces à usiner et des composants d'assemblage. Un vérin modèle est ici monté par un robot.

- **Position de travail 5**

La station de magasin central automatisé est à disposition à la position de travail 5 pour le stockage des matériaux. Les pièces à usiner peuvent y être stockées et déstockées.

- **Position de travail 6**

La station de tri se trouve à la position de travail 6. A l'aide de **la station de manipulation**, située directement à côté du système de transport, elle reçoit ses pièces à usiner et les trie d'après la couleur du vérin.

Le flux des matériaux s'achève à cette position de travail.

2.10 La ligne d'assemblage :

La ligne d'assemblage automatisée est située au département Qualité, Logistique Industrielle et Organisation (QLIO) de l'Institut Universitaire de Technologie (IUT) de Nantes.

Cette ligne comprend six postes de travail et un magasin de palettes. Ces palettes transportent les produits et sont équipées d'une étiquette RFID permettant le stockage des prestations à effectuer sur les produits transportés.

Le système est contrôlé par quatre automates programmables industriels SCHNEIDER et un MES (Manufacturing Execution System) INTOUCH très répandu dans l'industrie est fabriqué par WONDERWAR, système d'exécution de fabrication industrielle. Ce MES permet l'affection d'opérations aux six postes de travail et la définition de gammes de fabrication des produits. Il permet ensuite de lancement d'Ordres de Fabrication (OF) permettant de déclencher des productions.

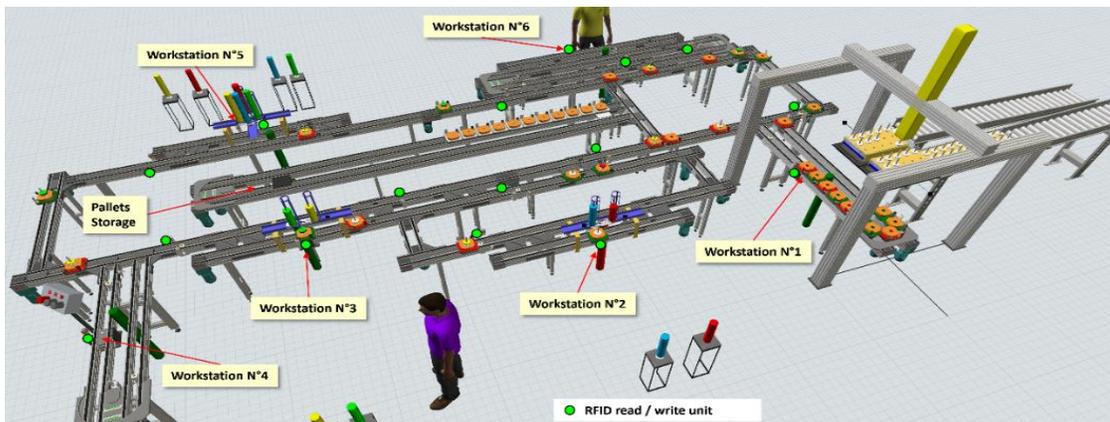


Figure 2.4 La ligne d'assemblage de l'IUT de Nantes

2.11 Conclusion :

Dans ce chapitre, nous avons introduit les différents outils utilisés afin de se familiariser avec leurs principales fonctionnalités et de rendre l'appréhension de l'implémentation plus facile à exposer et comprendre. Le chapitre suivant est l'essence de nos travaux proposant une architecture d'implémentation d'un digital twin en vue de faire de la maintenance prédictive et réactive.

CHAPITRE 3 ARCHITECTURE DE LA MAINTENANCE PRÉDICTIVE BASÉE SUR LE DIGITAL TWIN POUR LES CPPS

3.1 Introduction :

Les événements et les incidents non planifiés qui peuvent survenir à tout moment dans des environnements industriels réels augmentent la complexité des systèmes de production. Dans la quatrième révolution industrielle, le secteur industriel évolue rapidement avec des technologies émergentes comme les systèmes cyber-physiques de production (CPPS), Internet des objets (IoT), l'intelligence artificielle (AI), le jumeau numérique (DT), etc. Cependant, l'efficacité et la fiabilité de ces systèmes restent discutables dans de nombreuses circonstances. Pour relever ce défi, une architecture basée sur des techniques d'IA et le DT visant à élaborer une maintenance prédictive et réactive dans les CPPS est proposée dans ce chapitre, nous introduirons les techniques d'apprentissage automatique éventuellement utilisées. L'outil proposé vise à améliorer la fiabilité du système et aide les superviseurs de maintenance à ajuster les décisions de maintenance.

En effet, une bonne stratégie de maintenance réduit considérablement les coûts d'exploitation des systèmes concernés et augmente la fiabilité et la disponibilité globale des systèmes pour entreprendre les opérations. Le manque de connaissances sur les systèmes de production, leurs équipements et les processus associés compliquent davantage la gestion de ces systèmes.

3.2 Digital twin utilisé dans la maintenance prédictive :

La création d'un duplicata numérique d'une machine ou d'une chaîne de production pour modéliser son état et simuler son comportement, va permettre de détecter tout écart de fonctionnement entre cette duplication numérique et le système réel et de prévenir les dysfonctionnements éventuels du système et la préparation d'intervention de maintenance préventive.

Les données sont au cœur du développement de tout algorithme de maintenance prédictive, qui peut être utilisé pour la détection des dysfonctionnements. Dans notre cas, ces données seront issues des écarts entre le modèle numérique et le système réel. Des caractéristiques significatives sont extraites de ces données dans une étape de prétraitement et utilisées pour former un algorithme d'apprentissage automatique pour prédire les pannes dans un intervalle de temps donné.

La stratégie proposée est donc de détecter un éventuel décalage entre le modèle simulé et le modèle réel. On peut distinguer deux cas, le premier c'est lorsque les systèmes dans la réalité mettent plus de temps à réagir et que l'évènement se produit plutôt dans le digital twin. Ou dans le cas inverse lorsque l'évènement se déclenche dans le système physique plus tôt que dans le jumeau numérique, le retard va s'accumuler et ainsi les deux modèles ne seront plus synchronisés. [35]

Si un décalage au même endroit est récurrent sur le digital twin, cela signifiera que ce système représente un dysfonctionnement ou une anomalie à ce niveau dans la réalité.

Le but est de développer un modèle capable de prédire une panne à partir de ces données précédentes recueillies par le Digital Twin dans une base de données.

Pour comprendre comment les jumeaux numériques permettent la maintenance prédictive, nous proposons dans ce qui suit une architecture liant le système cyber physique de la production avec un modèle observateur alimenté par un algorithme d'apprentissage automatique.

3.3 Architecture d'implémentation :

Les anomalies dans les systèmes de production cyber physiques peuvent être dues à des causes internes ou externes. L'architecture proposée vise à détecter rapidement les défauts du système susceptibles de compromettre la fiabilité du système de production.

L'objectif principal est de localiser les problèmes, d'évaluer leurs criticités puis d'avertir les responsables de maintenance ou les opérateurs via les outils IIot.

La figure 3.1 présente l'organigramme de l'architecture proposée. Celle-ci est structurée en quatre parties principales détaillées ci-après.

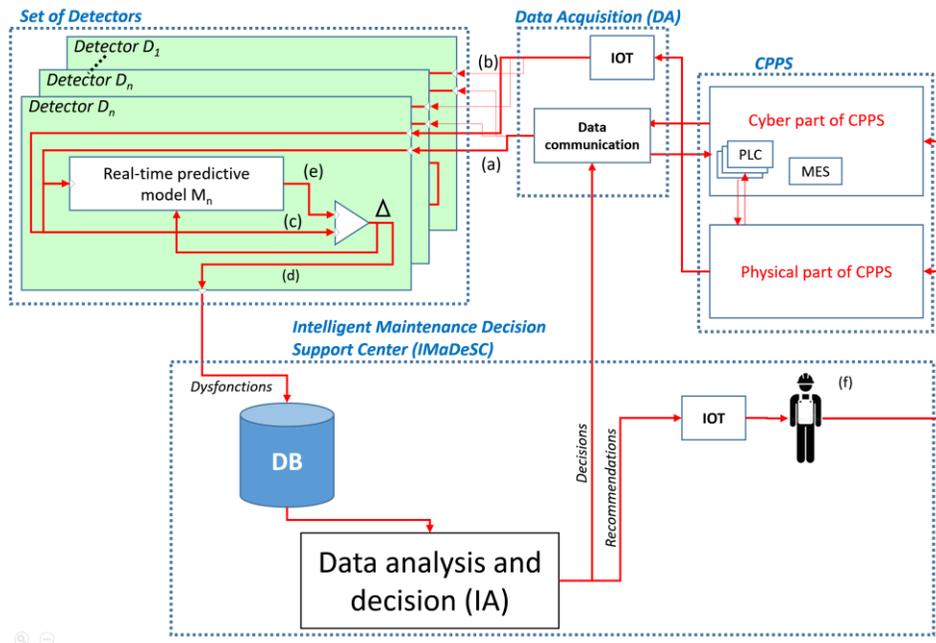


Figure 3.1 Le schéma de l'architecture proposée

3.3.1 Le système de production cyber-physique (CPPS)

La première partie est le **CPPS**. Nous pouvons décomposer ce CPPS en une partie physique, y compris les postes de travail, les moyens de stockage et de transfert. Ces différents moyens sont contrôlés par la partie logique du CPPS, y compris des infrastructures telles que les automates programmables (API), un système d'exécution de fabrication (MES) et d'autres outils logiciels.

3.3.2 Acquisition des données

Une partie fondamentale dans la maintenance prédictive est l'acquisition des données des équipements. Cette fonction est importante car elle permet de connaître l'état et le comportement du CPPS.

Les composants du CPPS comme les MES ou ERP fournissent des données d'équipement spécifiques à la production par exemple les ordres de fabrication, les heures de fonctionnement des machines. D'autres composants comme les API font référence à des données mesurables concernant une pièce en cours de traitement mais aussi les informations provenant des capteurs nécessaires au pilotage. Le module de communication permet cet échange avec le CPPS (a sur la

figure 3.1). Les Iot peuvent nous renseigner sur certaines informations si besoin comme la température, la pression, les vibrations (b sur la figure 3.1).

3.3.3 L'ensemble des modules détecteurs

La troisième partie du CPPS est constituée d'un **ensemble de n modules Détecteurs** (D1 à Dn) qui représentent des modèles comportementaux du système réel utilisés pour détecter les dysfonctionnements du CPPS. Le cœur de chaque détecteur est un modèle prévoyant le comportement en temps réel.

Basé sur les informations des capteurs CPPS et Iot, fournies par le module d'acquisition de données (DA), ce modèle prédit le comportement "idéal" du système.

Le modèle prédictif en temps réel M_i détermine la différence (Δ) entre le comportement idéal (e) et le comportement réel du système physique (c). Le résultat de la soustraction renseigne sur le dysfonctionnement arrivé dans le CPPS. Chaque détecteur est concerné par un aspect particulier du CPPS (aspect événementiel en temps réel, aspect thermique, aspect énergétique, aspect économique, ...)

3.3.4 Centre intelligent d'aide à la décision en maintenance

La quatrième partie, le centre intelligent d'aide à la décision permet de traiter les écarts détectés, entre le comportement virtuel idéal et le comportement réel.

Les dysfonctionnements observés (d) détectés par le modèle prédictif M_i sont stockés dans une base de données créant un historique de l'utilisation de l'équipement dans le temps.

Ces données historiques sont utilisées comme base pour une analyse ultérieure des données à l'aide d'un algorithme d'apprentissage automatique en détectant des modèles anormaux et en identifiant des scénarios de défaillance récurrents. L'utilisation de méthodes d'apprentissage automatique vise à améliorer l'efficacité globale du système.

L'analyse des données historiques : Le résultat de cette analyse est la prédiction du comportement futur du système, indiquant l'occurrence potentielle d'une défaillance dans une période de temps et la notification de l'opérateur du dysfonctionnement ainsi que de sa criticité. Par conséquent, des actions peuvent être établies avant le dysfonctionnement.

L'analyse des données immédiates : elle permet de prendre des décisions concernant la maintenance réactive sur la base d'un seuil de temps de blocage estimé, préalablement fixé par l'opérateur.

Les résultats de cette analyse peuvent être des recommandations transmises à l'opérateur pour l'aider à entreprendre des réponses réactives qui seront appliquées au CPPS. Ces recommandations d'ajustement des décisions de maintenance seront transmises via l'Iot (montres connectées, smartphone, outils de réalité virtuelle, etc.).

3.4 Machine Learning:

L'intelligence artificielle est une constellation de nouvelles technologies donnant aux machines l'aptitude de percevoir, comprendre, agir et apprendre.

L'apprentissage automatique est une subdivision de l'intelligence artificielle. Les algorithmes de ce dernier permettent aux ordinateurs de s'entraîner sur des données afin d'automatiser le processus de décision.

Deux principales méthodes de Machine Learning sont largement adoptées de nos jours, l'apprentissage supervisé basé sur des données historiques disponibles et étiquetées préalablement et l'apprentissage non supervisé utilise des données non étiquetées, de sorte que l'algorithme d'apprentissage trouve tout seul des points communs parmi ses données d'entrée.

Dans le cadre de l'apprentissage supervisé, nous retrouvons deux approches : la classification et la régression.

La classification est le processus de recherche de modèles mathématiques et statistiques pour séparer les données en plusieurs classes où les données peuvent être divisées en binaires ou en plusieurs étiquettes discrètes. Il est utilisé pour classer les modèles normaux et anormaux, défectueux et sains dans une certaine période de temps dans la maintenance prédictive.

Alors que la régression est le processus de recherche d'un modèle pour distinguer les données en valeurs réelles continues. Elle est utilisée pour prédire le temps utile restant de l'équipement industriel en général.

Cependant, le défi est de choisir la technique appropriée et la plus efficace afin de présenter un socle d'aide à la décision solide.

Il existe plusieurs méthodes capables de détecter les défaillances, de la plus simple à la plus complexe. Selon la littérature, les plus pertinentes et les plus utilisées dans les modèles de maintenance prédictive sont la régression logistique (LR), le réseau neuronal artificiel (ANN) et la machine à vecteur de support (SVM).

3.4.1 Régression logistique (LR) :

La régression logistique est une technique d'apprentissage automatique qui est utilisé pour les problèmes de classification, c'est un algorithme d'analyse prédictive basé sur le concept de probabilité.

Elle utilisée pour attribuer des observations à un ensemble discret de classes pour analyser les problèmes lorsqu'il existe une ou plusieurs variables indépendantes qui déterminent un résultat. Le résultat est mesuré avec une variable dichotomique dans laquelle il n'y a que deux résultats possibles.

Certains des exemples de problèmes de classification sont le spam par courrier électronique ou non, la fraude aux transactions en ligne ou non, la tumeur maligne ou bénigne

Il existe deux types de classification dans cette méthode, à savoir binaire et multilinéaire.

La régression logistique utilise la fonction logistique Sigmoides, qui est une fonction S-discriminante pour comprimer la sortie d'une équation linéaire entre 0 et 1 et renvoyer une valeur de probabilité.

L'expression de la fonction sigmoïde est la suivante :

$$f(x) = \frac{1}{1 + e^{-x}}$$

La courbe de la fonction Sigmoides est telle que :

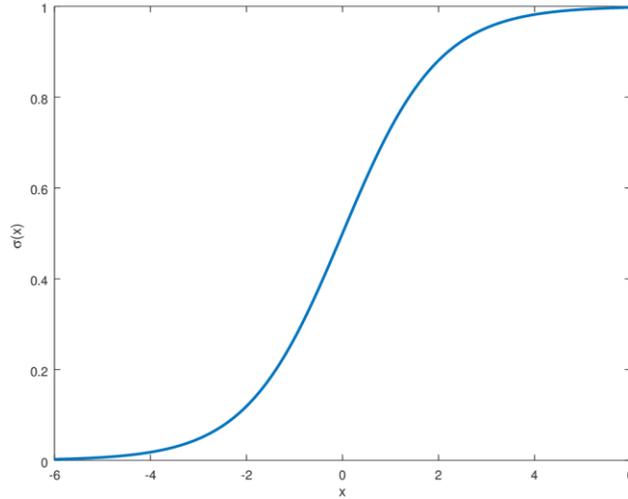


Figure 3.2 La courbe de la fonction Sigmoide

3.4.2 Réseau de neurones artificiels (ANN) :

Les réseaux de neurones, communément appelés réseaux de neurones artificiels (ANN), sont une simulation de la fonctionnalité du cerveau humain dans les problèmes d'apprentissage automatique (ML) conçue pour traiter des informations liées entre elles. L'utilisation la plus courante des ANN est la régression et la classification.

Le bloc de construction de base est similaire à un véritable réseau de neurones.

Les neurones sont connectés en couches afin qu'une couche puisse communiquer avec d'autres couches formant un réseau neuronal. Les couches internes autres que les couches d'entrée et de sortie sont appelées couches cachées. Les sorties d'une couche sont envoyées aux entrées d'une autre couche.

La signification de chacune des entrées est multipliée par un poids. Elles sont ensuite additionnées par une fonction de sommation et transmises à une fonction d'activation telle que la fonction logistique pour produire une sortie.

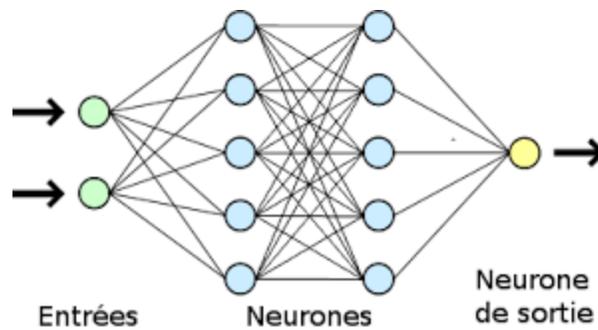


Figure 3.3 Réseau de neurone artificiel

Il est très puissant quand il y a des ensembles de données massifs, ils peuvent apprendre des tâches en considérant les données d'entrée. [33] En raison de leur capacité à apprendre des exemples, ANN a reçu une attention importante, il peut montrer des résultats prometteurs pour évaluer les données afin de soutenir les activités de maintenance prédictive.

3.4.3 Machine à vecteurs de support (SVM) :

La machine à vecteurs de support est un modèle linéaire pour les problèmes de classification et de régression. Il peut résoudre des problèmes linéaires et non linéaires et fonctionne bien pour de nombreux problèmes pratiques.

L'idée de SVM est de créer un hyperplan dans un espace à N dimensions qui sépare les données en classes.[34] Pour séparer les classes de points de données, il existe de nombreux hyperplans possibles qui pourraient être choisis. Le but est de trouver un plan qui a la marge maximale, c'est-à-dire la distance maximale entre les points de données des deux classes. L'optimisation de la distance de marge fournit un certain renforcement afin que les futurs points de données puissent être classés avec plus de confiance.

Les vecteurs de support sont des points de données plus proches de l'hyperplan et influencent la position et l'orientation de l'hyperplan. En utilisant ces vecteurs de support, la marge est maximisée. La suppression des vecteurs de support modifiera la position de l'hyperplan. Ces points aident principalement à construire un modèle SVM.

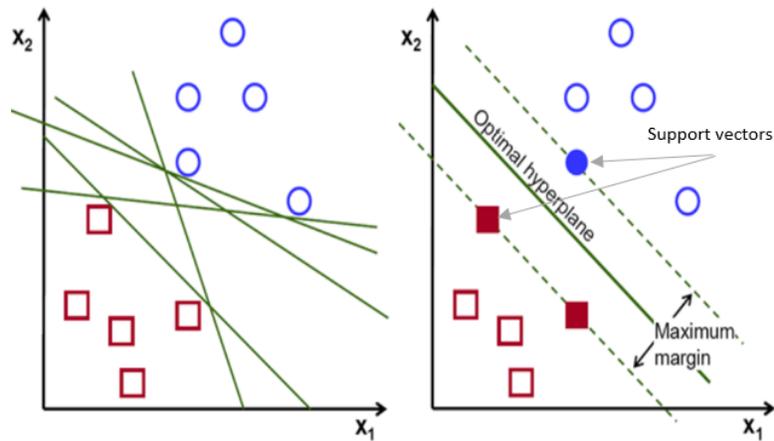


Figure 3.4 Machine à vecteurs support

3.5 Conclusion :

Dans ce chapitre, nous avons proposé une architecture d'implémentation d'un digital dans le contexte de l'industrie 4.0. Notre approche est basée sur la simulation d'un modèle comportemental, sur l'analyse de données et l'apprentissage automatique d'une autre part, afin d'être capable de détecter les comportements anormaux et calculer la probabilité de survenance d'une panne en se basant sur les données historiques de l'évolution d'un système de production.

Dans le prochain chapitre, nous aborderons une étude de cas sur laquelle nous appliquerons notre approche

CHAPITRE 4 IMPLÉMENTATION ET ETUDE DE CAS

4.1 Introduction :

Ce dernier chapitre est réparti en deux parties. Dans la première, nous avons pris pour une étude de cas, la ligne d'assemblage de l'IUT de Nantes et le système de transport de la MPS 500 sur lesquels nous avons appliqué notre approche de maintenance prédictive basée sur le digital twin. Quant à la deuxième partie, elle est consacrée aux démarches de création d'un digital twin des stations de la MPS500 en utilisant Flexsim et CIROS. Nous introduisons dans cette partie les principes de programmation des automates, de test, de simulation d'émulation et de communication

4.2 Etude de cas sur la ligne d'assemblage :

Le détecteur est construit à l'aide d'un modèle de simulation d'événements discrets fonctionnant en temps réel sous le logiciel « **Flexsim** ».

Le modèle numérique simulé communique avec un automate virtuel « **S7-PLCSIM** » à l'aide d'un module d'**émulation** grâce à un serveur OPC DA « **Kepserver** »

Ce modèle simule le comportement idéal du système physique et détermine la différence avec le comportement physique en temps réel.

Les écarts détectés sont enregistrés dans une base de données sur « **MySQL** » via un module « **Database Connector** » et sont utilisées pour une analyse à l'aide d'un algorithme d'apprentissage automatique sur **Python** afin de détecter les potentiels dysfonctionnements.

Cette preuve de concept n'est faite que sur un seul convoyeur, contenant un point d'entrée A et un module de lecture / écriture L1

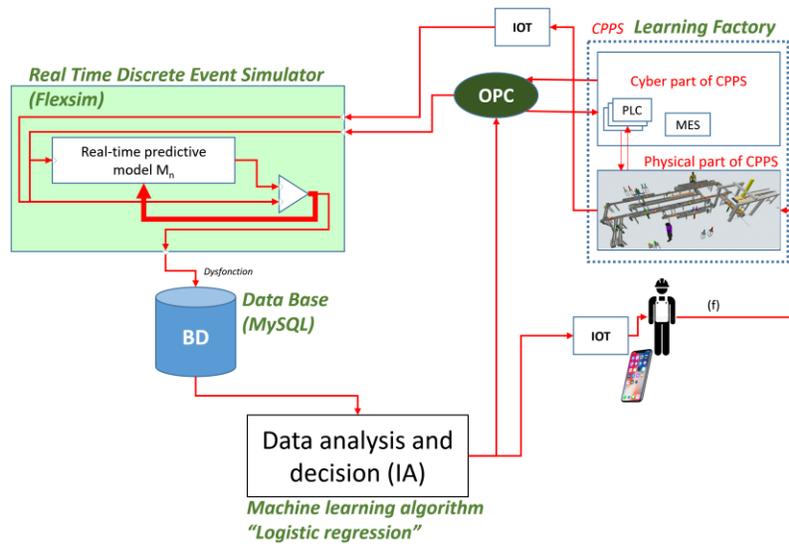


Figure 4.1 Architecture d'implémentation

4.2.1 Système de production :

4.2.1.1 Système de transport de la MPS 500 :

Les porte-pièces sont dotés d'un système d'identification des E/S. Un nombre différent de perçages sur le dessous des porte-pièces lui donne son numéro personnel d'identification. De cette manière, il est possible d'accompagner le porte-pièces de spécifications et lire cet identifiant devant chaque stoppeur. Six capteurs sont mis dans le système de transport devant chaque position de travail

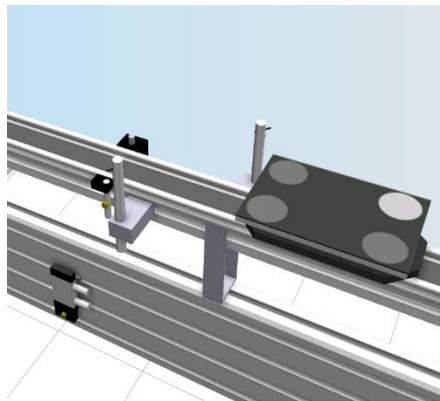


Figure 4.2 Système de transport de la MPS 500

4.2.1.2 Ligne d'assemblage :

Les palettes passent par un réseau de convoyeurs complexes. A chaque point de commutation, une unité de lecture RFID lisant le numéro d'identification de la palette permet de décider de l'orientation des palettes. Dix-sept unités de lecture sont utilisées pour identifier la position des palettes.

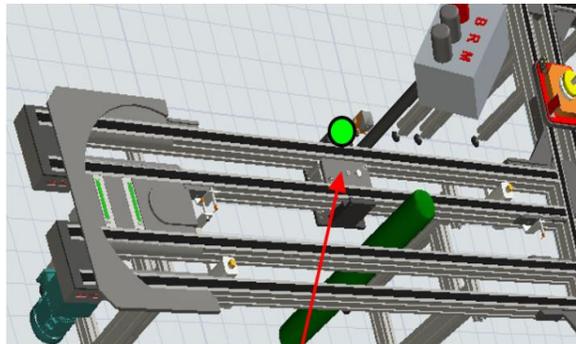


Figure 4.3 la ligne d'assemblage de l'IUT

4.2.1.3 Fonctionnement :

Nous remarquons une similitude dans le comportement de ces deux systèmes, notamment dans le convoyage et la lecture de la palette. On décrit dans ce qui suit le fonctionnement opérationnel typique et général de ces derniers.

Une palette se déplace en direction d'un stoppeur/ butée, celle-ci est arrêté, le numéro d'identification est lu, le capteur s'active et la palette est relâchée.

4.2.1.4 Programmation API :

Un programme d'automate est écrit sur STEP 7 afin de représenter ce fonctionnement décrivant les temporisations, les capteurs et les actionneurs devant un lecteur. Nous utilisons S7-PLCSIM afin de simuler un API virtuel.

4.2.2 Communication des données :

Nous utilisons le protocole Ethernet TCP /IP d'un OPC serveur qui permet de connecter les automates programmables à des applications client OPC, notamment HMI, SCADA, ERP, ou comme dans notre cas le logiciel de simulation Flexsim. Ce dernier est à son tour muni d'un outil d'émulation permettant d'utiliser le protocole OPC, configurer cette connexion et importer les entrées et les sorties souhaitées. De cette manière, les données entre l'automate virtuel et le modèle Flexsim sont échangées.

4.2.3 Modélisation numérique sous Flexsim :

Le module de lecture / écriture L1 est situé à une distance D du point A. V est la vitesse du convoyeur. A l'instant t0, l'automate virtuel envoie les informations conduisant à la création d'une palette dans le simulateur. A la date t1, l'automate envoie les informations indiquant l'arrivée de la palette réelle au point L1. t1 a été programmé de telle sorte que :

$$t1 = t0 + \frac{D}{V} + \varepsilon$$

Avec ε étant une variable aléatoire telle que $\varepsilon = \text{Uniforme}(-D / V, D / V)$ nous permettant d'introduire une perturbation.

4.2.3.1 Arrivée des palettes sur la ligne

Une variable automate (Start) indique le départ d'une palette sur la ligne réelle. Nous lisons le numéro de la palette donnée par l'API puis nous créons la palette virtuelle qui part sur la ligne. Nous mettons ensuite le jeton dans la liste « Liste Palettes » qui a donc autant de jetons qu'il y a de palettes sur la ligne.

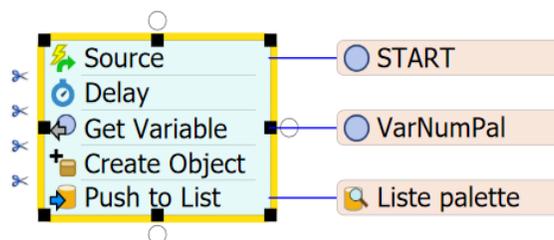


Figure 4.4 Process Flow de l'arrivée d'une palette

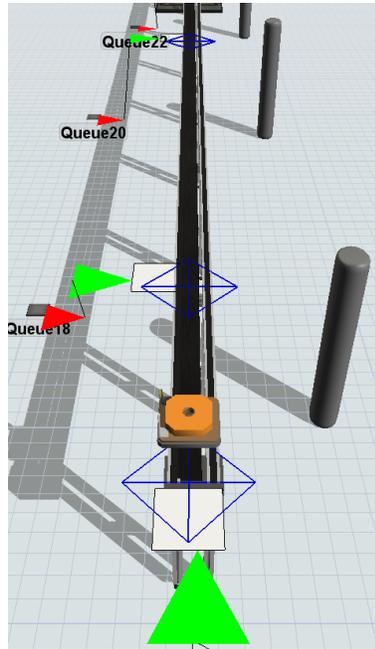


Figure 4.5 Arrivée de la palette sur le convoyeur

4.2.3.2 Traitement d'un lecteur

Lorsqu'une palette virtuelle arrive sur le point décision qui représente le capteur dans le système réel, nous inscrivons dès lors la date d'arrivée de celle-ci. Celle-ci est bloquée sur le convoyeur jusqu'à l'arrivée de la palette réelle.

Lorsque la variable B001 indique l'arrivée de la palette réelle sur le lecteur, nous récupérerons le N° de cette palette. Nous récupérerons le jeton qui se trouve dans la liste Palettes et qui nous indique le flowitem « Palette » correspondant.

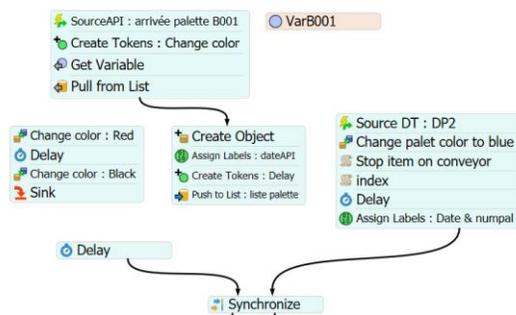


Figure 4.6 Process flow du traitement d'un lecteur

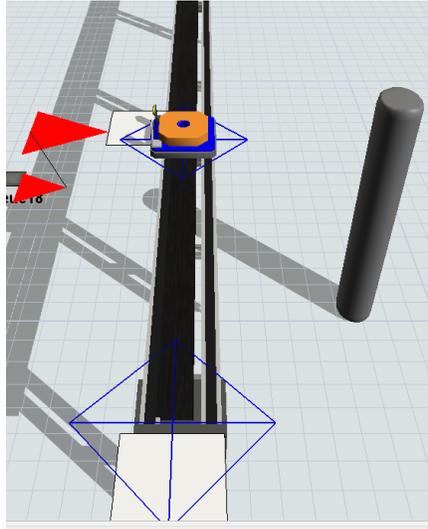


Figure 4.7 Arrivée de la palette au lecteur

4.2.3.3 Synchronisation

Il est très important de synchroniser le modèle virtuel avec la réalité. Deux phénomènes peuvent se produire : La palette réelle peut être en retard par rapport à la palette virtuelle. Cela se produit surtout lorsque la palette réelle est bloquée. L'autre possibilité est que la palette réelle soit en avance sur la palette virtuelle.

Pour chaque lecteur, deux événements doivent donc être détectés : l'arrivée de la palette virtuelle au point du lecteur virtuel et l'arrivée de la palette réelle au point du lecteur réel.

Lorsque la palette virtuelle arrive au point de l'unité virtuelle, elle est bloquée pour attendre l'arrivée de la palette réelle. Mais il est important de générer une alerte avant l'arrivée de la vraie palette, notamment en cas de blocage de cette vraie palette. C'est la fonction de la boucle (d) Figure 4.8. Bien sûr, si la palette réelle arrive pendant la période D_{lim} qui est un temps d'attente fixé préalablement, l'alerte ne sera pas envoyée.

Lorsque la palette réelle arrive au point réel de l'unité RFID, une copie de la palette virtuelle est créée au point de l'unité virtuelle. Ceci est important pour que le comportement du modèle virtuel soit cohérent avec la réalité.

Le détecteur analyse les différences entre les dates d'arrivée des palettes virtuelles dans le modèle d'observation et les dates d'arrivée réelles fournies par l'automate virtuel.

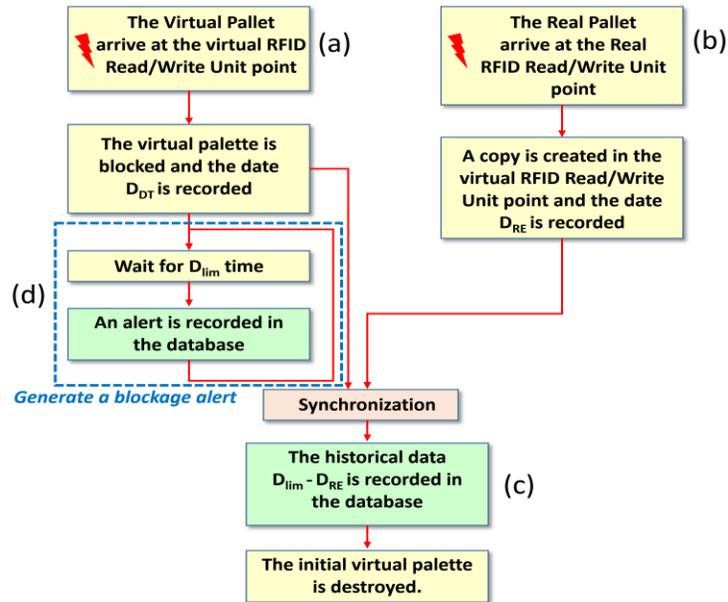


Figure 4.8 Synchronisation du détecteur

4.2.4 Enregistrement des données sur MySQL :

Nous connectons Flexsim à MySQL grâce au module « database connector » et nous enregistrons dans la table d'une base de données MySQL, le temps d'arrivée de la palette virtuelle et le temps d'arrivée de la palette réelle, l'écart Δ entre eux ainsi que le lecteur ou la position concernée, le numéro d'identification de la palette, en temps réel.

Celle-ci nous permettra d'avoir un historique des événements survenant sur notre ligne.

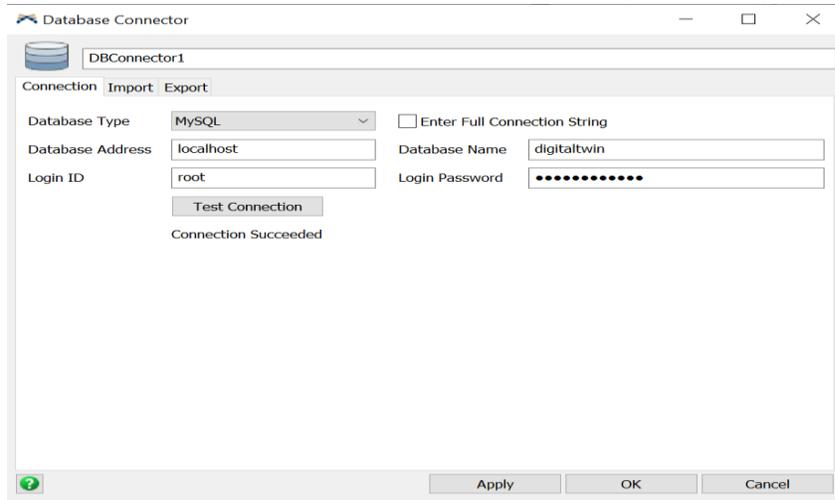


Figure 4.9 Connexion sur le module Database Connector

4.2.5 Analyse des données sur Python

La base de données est à son tour connectée à Python afin de traiter ces enregistrements via le package « Mysql connector Python ».

Chaque fois qu'une palette est créée, une nouvelle valeur de perturbation ε est introduite. Comme le montre le tableau 1, cette population de valeurs de ε peut être divisée en cinq classes. Si ε est négatif, cela correspond à une arrivée précoce de la palette en L1. Sinon, cela correspond à une arrivée tardive.

Tableau 4-1 Distribution des classes de la perturbation ε

Classe -2	$-3700 \text{ ms} \leq \varepsilon < -2000 \text{ ms}$	Arrivée très précoce
Classe -1	$-2000 \text{ ms} \leq \varepsilon < -500 \text{ ms}$	Arrivée précoce
Classe 0	$-500 \text{ ms} \leq \varepsilon < 500 \text{ ms}$	Arrivée à l'heure
Classe 1	$500 \text{ ms} \leq \varepsilon < 2000 \text{ ms}$	Arrivée en retard
Classe 2	$2000 \text{ ms} \leq \varepsilon < 3700 \text{ ms}$	Arrivée très en retard

Un programme sur Python est écrit pour démontrer la possibilité d'analyser ces données. Il est destiné à classer les enregistrements en classes. Le tableau 2 montre le nombre d'enregistrements dans la base de données SQL pour chaque classe de ϵ ainsi que le nombre d'enregistrements que le programme Python a pris en compte dans chaque classe.

Tableau 4-2 Résultats de l'analyse de données

Classes	Nombre de retards générés par l'API	Nombre de retards observés par Python
Classe -2	15	16
Classe -1	11	10
Classe 0	6	6
Classe 1	7	7
Classe 2	11	11

Cela signifie que les perturbations que nous avons générées en entrée sur l'automate sont bien détectées comme étant des écarts en sortie sur notre modèle, ce qui démontre l'efficacité de notre digital Twin. Etablir ces classes, aide également à mesurer des écarts et fixer le seuil de criticité des pannes.

4.2.6 Apprentissage automatique sur Python

L'étape suivante consiste à mettre en œuvre des méthodes d'intelligence artificielle pour interpréter les résultats et prédire les causes probables de dysfonctionnement.

Les bibliothèques de science des données et d'apprentissage automatique les plus populaires telles que Scikit Learn, Numpy, Pandas et Matplotlib permettent d'écrire du code élégant et compact, mais également d'implémenter des modèles et de résoudre des problèmes. Nous avons utilisé ces packages dans un programme sur Python afin de construire d'une façon simple et basique nos

algorithmes à savoir Régression logistique, Machine à vecteurs support et réseau de neurone artificiel.

Afin d'entraîner le modèle d'apprentissage, nous utilisons les observations simulées pour prédire les modèles défectueux.

Pour l'instant, le modèle ne vise qu'à détecter les écarts entre le système réel et son modèle virtuel, faisant une classification binaire en faisant la différence entre un comportement normal et un comportement anormal ayant de l'avance ou du retard. Nous verrons ça plus en détails dans les étapes qui suivent.

4.2.6.1 Implémentation des algorithmes d'apprentissage automatique :

Le principe de chaque algorithme est différent, en général, la régression logistique est un très bon algorithme tout usage, surtout quand il y a peu de fonctionnalités et que nous avons besoin de probabilités. Les ANN sont très flexibles lorsque de nombreuses données et fonctionnalités sont nécessaires. Ces deux méthodes produisent non seulement une étiquette de classe pour un élément de données, mais aussi une probabilité, P .

Alors que les SVM sont bons lorsque nous avons un ensemble de données de haute dimensionnalité et que nous n'avons pas beaucoup de données et ne considérons qu'une distinction dichotomique entre les deux classes et attribue des étiquettes de classe, 0 ou 1, à un élément de données inconnu.

La méthode d'implémentation d'un modèle de classification reste toutefois la même, l'organigramme suivant illustre le principe général de cette dernière.



Figure 4.10 Organigramme de l'algorithme de classification

A. Comprendre la problématique et définir l'objectif

Dans un premier lieu, nous souhaitons construire un algorithme de classification binaire sous apprentissage supervisé afin de prédire les pannes, cela signifie que notre modèle classera les évènements en deux classes à savoir, comportement normal et comportement défectueux, l'objectif étant de détecter les écarts entre le modèle réel et le modèle virtuel.

B. Acquérir et intégrer les données, et si nécessaire les transformer et les adapter

Nous utiliserons afin de construire ce modèle un échantillon de 140 observations simulées sur Flexsim avec différentes perturbations, nous attribuerons à la classe contenant les évènements sains un 0 et à la classe contenant les évènements potentiellement défectueux à case du seuil d'attente dépassé un 1.

Notre analyse se basera sur la date d'arrivée de la palette au système réel, la date d'arrivée de la palette au système virtuel, le numéro de palette, le lecteur concerné, ces derniers paramètres représentent les variables connues mises en entrées.

Nous traiterons nos données sur un classeur Excel que nous importerons sur Python à l'aide du package Pandas.

C. Importer les packages et les fonctions nécessaires

Scikit learn, Numpy, et Matplotlib nous permettent d'écrire un code simple et efficace avec des fonctions préalablement construites. Nous importons Matplotlib pour la visualisation et NumPy pour les opérations de tableau. Nous aurons également besoin des fonctions qui définissent l'algorithme comme LogisticRegression mais aussi de `classification_report ()` et `confusion_matrix ()` de scikit-learn afin d'analyser les résultats obtenus:

```
Python

import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix
```

Figure 4.11 Importation des bibliothèques

D. Construire un modèle de classification

Nous construisons le modèle de l'algorithme souhaité, dans notre cas nous avons construit trois modèles distincts pour chacune des méthodes à savoir, régression logistique, réseau de neurones artificiel et machine à vecteurs supports.

Nous précisons les paramètres du modèle et ajustons ces derniers selon notre problème.

```
Python

model = LogisticRegression(solver='liblinear', random_state=0)
```

Figure 4.12 Création du modèle

L'instruction ci-dessus crée une instance de `LogisticRegression` et lie ses références au modèle de variable. `LogisticRegression` possède plusieurs paramètres facultatifs qui définissent le comportement du modèle et de l'approche comme la régularisation utilisée, la tolérance pour l'arrêt de la procédure, calculer l'ordonnée à l'origine, le nombre maximum d'itérations, le solveur à utiliser pour ajuster le modèle comme « liblinear », « newton-cg », « lbfgs », « sag » et « saga ».

E. Entraîner le modèle

Maintenant que les variables sont définies et le modèle bien construit, nous utilisons la fonction « fit » afin d'entraîner le modèle et l'ajuster.

```
Python  
model.fit(x, y)
```

Figure 4.13 Entraînement des données

F. Evaluer la performance du modèle

Une fois qu'un modèle est défini, nous évaluons le modèle avec la fonction `.predict ()`, en obtenant les prédictions réelles.

Nous mesurons la performance et la précision du modèle sur la base de test. Il existe certaines façons standard de le faire en fonction des types de problèmes. Pour les problèmes de classification, la matrice de confusion s'avère être une bonne méthode d'évaluation.

Dans le cas d'une classification binaire, la matrice de confusion affiche les nombres suivants:

- Vrais négatifs en position supérieure gauche
- Faux négatifs en bas à gauche
- Faux positifs en position supérieure droite
- Vrais positifs en position inférieure droite

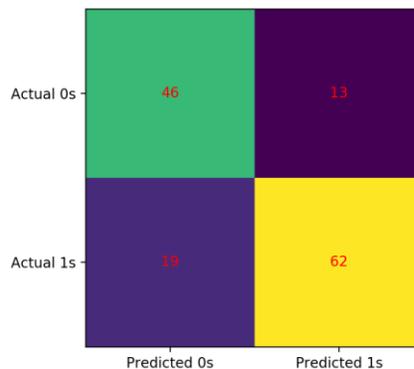


Figure 4.14 Matrice de confusion

4.2.6.2 Résultats :

Sur les 140 observations faites, nous avons obtenu une précision de 75% pour les trois algorithmes, le but est de détecter les écarts et les comportements anormaux redondants. Cette précision peut être améliorée en faisant une analyse de données plus approfondie et plus riche sur le système réel, nous introduirons dans une prochaine version une analyse supplémentaire visant à détecter la défectuosité de la palette ou la latence des unités de lecture / écriture RFID, et proposer des solutions basées sur la criticité du problème.

4.2.7 Conclusion

Des tests d'émulation ont été faits sur la ligne d'assemblage réelle à l'IUT de Nantes, où nous avons réussi à connecter Flexsim aux automates de Schneider par le biais de l'OPC Schneider Unity.

Une validation du principe basée sur une émulation du système actuel réel a été développée sur les postes de travail de la ligne d'assemblage, le bon fonctionnement de cette dernière nous a permis de valider notre architecture proposée basée sur un digital twin et des techniques d'intelligence artificielle pour gérer la maintenance des CPPS.

Nous avons également pu exploiter les modèles de simulation afin d'entraîner nos algorithmes d'apprentissage automatique hors ligne. Ceci nous a permis de générer différentes perturbations et imaginer différents scénarios possibles.

4.3 Etude de cas sur les stations de la MPS 500

Dans cette deuxième partie, nous proposons une architecture à but pédagogique afin d'observer, de se familiariser et d'acquérir les connaissances nécessaires sur les stations de la MPS500 à l'aide de CIROS, mais aussi à créer un modèle d'émulation sur Flexsim et apprendre à faire communiquer ces deux dernières entités par le biais d'un OPC serveur.

CIROS dispose de modèles préconfigurés des stations de Festo Didactique semblables aux stations réelles, celles-ci peuvent être contrôlées par un automate virtuel S7-PLCSIM par le biais d'un programme écrit sur STEP 7 v5.6 en utilisant EzOPC.

Un modèle d'émulation des stations est créé et configuré sur Flexsim et reçoit les ordres de la part de l'automate virtuel via KepserverEX.

Nous aurons alors un système physique représenté par le modèle de Ciros et un modèle miroir simulé sur Flexsim reflétant le même comportement que le système physique.

Nous prenons la station de Tri comme étude de cas dans cette partie-là.

Il est à noter que ce principe d'implémentation a été inspiré par les travaux de Yelles Chaouche Karim et El Ghazi Younes dans leur mémoire de projet de fin d'étude « Implémentation d'un Digital Twin des stations commandées par automates de la MPS500 »[35] ou les étapes de configuration sont plus détaillées.

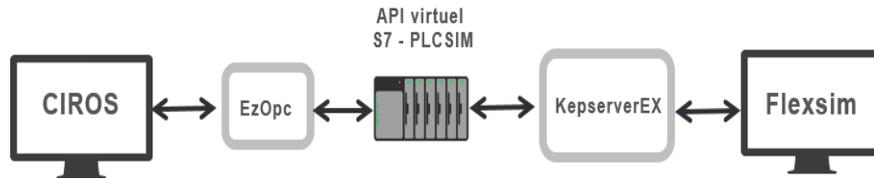


Figure 4.15 Architecture d'implémentation

4.3.1 Fonctionnement de la station de tri :

La station procède au tri de pièces à usiner sur 3 glissières. Les pièces à usiner insérées en début de bande sont détectées à l'aide d'un capteur.

Des capteurs installés en aval du barrage détectent les propriétés de la pièce (noire, rouge, métallique). Le tri des pièces à usiner et leur acheminement aux glissières respectives est assuré par des déviations manœuvrées par des vérins à faible course. Une barrière à réflexion surveille le niveau de charge des glissières.

4.3.2 Observation de la station sur Ciros

Pour mieux observer cette station, nous utilisons le modèle de référence qui se trouve dans la documentation de CIROS de la station désirée dans la section « Help ».

Nous pourrons dès lors lancer la simulation et observer le séquençement du cycle de fonctionnement et le changement des capteurs et des actionneurs.

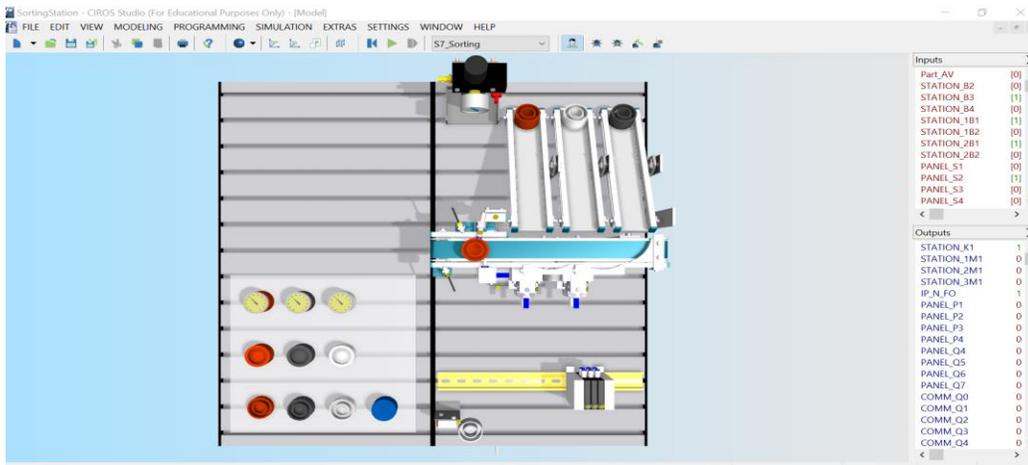


Figure 4.16 La station de Tri de la MPS 500 sur CIROS

4.3.3 Programmation API

Maintenant, nous avons pris connaissance de la description opérationnelle de la station, nous pourrons écrire notre programme sur STEP 7 ou TIA Portal afin de programmer l'automate.

Nous retrouvons l'affectation des entrées et des sorties de l'automate dans la documentation de CIROS avec l'adressage qui leurs sont propre.

Status	Symbol	Adresse	Datentyp	Kommentar
	K1	A 0.0	BOOL	Bandmotor ein / belt motor on
	1M1	A 0.1	BOOL	Weiche 1 ausfahren / Extend switch 1
	2M1	A 0.2	BOOL	Weiche 2 ausfahren / Extend switch 2
	3M1	A 0.3	BOOL	Stopper einfahren / Retract stopper
	IP_N_FO	A 0.7	BOOL	Station belegt / station occupied
	P1	A 1.0	BOOL	Leuchtmelder Start / Start indicator light
	P2	A 1.1	BOOL	Leuchtmelder Grundstellung (Reset) / Reset indicator light
	P3	A 1.2	BOOL	Leuchtmelder Rutsche voll / Slide full indicator light
	Part_AV	E 0.0	BOOL	Werkstück vorhanden / workpiece available
	B2	E 0.1	BOOL	Werkstück aus Metall / Metallic workpiece
	B3	E 0.2	BOOL	Werkstück ist nicht schwarz / Workpiece not black
	B4	E 0.3	BOOL	Rutsche voll / Slide full
	1B1	E 0.4	BOOL	Weiche 1 eingefahren / Switch 1 retracted
	1B2	E 0.5	BOOL	Weiche 1 ausgefahren / Switch 1 extended
	2B1	E 0.6	BOOL	Weiche 2 eingefahren / Switch 2 retracted
	2B2	E 0.7	BOOL	Weiche 2 ausgefahren / Switch 2 extended
	S1	E 1.0	BOOL	Taster Start / Start button
	S2	E 1.1	BOOL	Taster Stop (Öffner) / Stop button (normally closed)
	S3	E 1.2	BOOL	Schalter Automatik-Manuell / Automatic-manual switch
	S4	E 1.3	BOOL	Taster Richten / Reset button
	Em_Stop	E 1.5	BOOL	NOT-AUS entriegelt / Emergency stop unlocked

Figure 4.17 Table des mnémoniques de la station de tri

4.3.4 Test du programme API

Maintenant que nous avons écrit notre programme, CIROS nous donne la possibilité de tester la logique contrôlant les stations propres à nous sous STEP7 ou TIA PORTAL.

Par défaut sur CIROS, les simulations sont contrôlées par l'API virtuel interne. Utiliser cette méthode simple nous permet de se familiariser et manipuler l'interface CIROS.

Pour cela nous utilisons le modèle d'utilisateur de la station de tri, nous chargeons le programme STEP 7, nous pouvons dès lors lancer la simulation et visualiser le fonctionnement de la station.

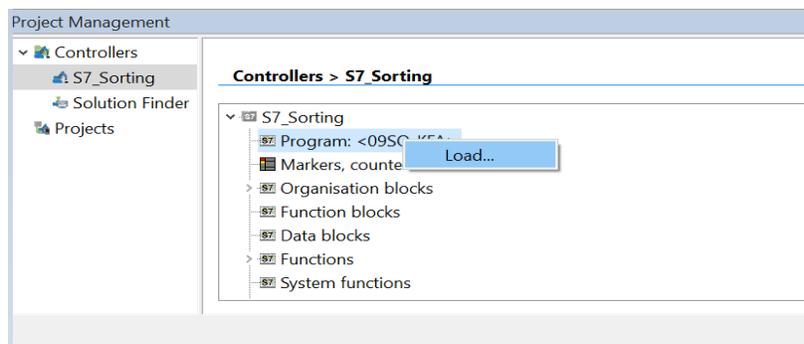


Figure 4.18 Chargement du programme STEP 7 sur CIROS

Chapitre 4 : Implémentation et étude de cas

Cette méthode n'est fonctionnelle qu'avec les programmes sous le format S7P, seule Step7 sauvegarde ses fichiers sous ce format. Elle permet dans un premier temps à tester la logique du programme et l'adapter au fonctionnement correct de la station.

Il existe une autre méthode qui permet plus de flexibilité en codant la logique sur STEP 7 ou sur TIA PORTAL en utilisant un API virtuel comme PLCSIM ou réel. Pour cela nous aurons besoin de EzOPC qui permet de transmettre les données de l'API à CIROS.

Il suffit d'ouvrir le modèle d'utilisateur de la station de tri sur CIROS, nous changeons les paramètres sur CIROS afin que celui-ci soit contrôlé par un automate virtuel externe.

Nous chargerons le programme sur PLCSIM et lancerons la visualisons. EzOPC est ouvert, nous nous assurerons que la connexion est bien établie.

Nous pouvons alors lancer la simulation sur CIROS qui sera détecté comme un OPC client sur EzOPC et nous pouvons à ce stade la contrôler la station de tri à partir de PLCSIM.

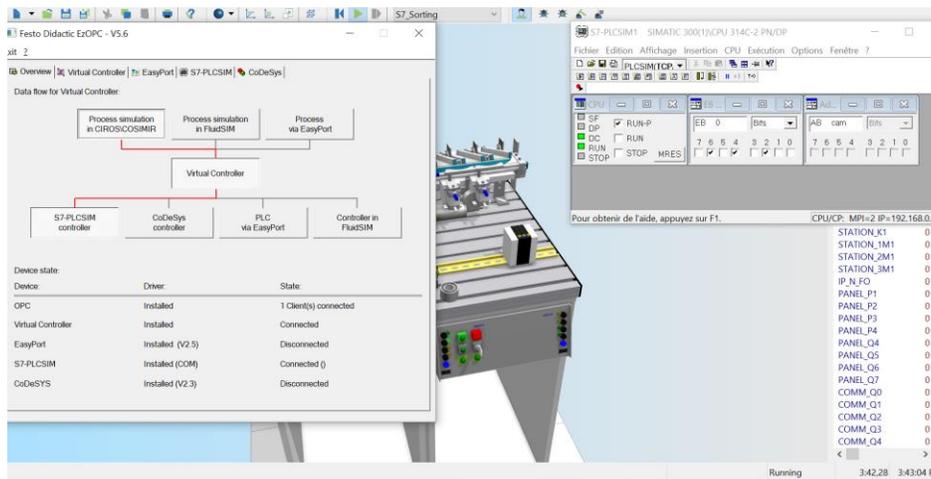


Figure 4.19 Interface de EzOPC et PLCSIM lors de la simulation

Nous aurons ainsi de cette façon connecter notre automate virtuel S7-PLCSIM avec CIROS par le biais de EzOPC, cela nous donne la possibilité de contrôler la station de tri préconfigurée.

4.3.5 Configuration du réseau

Afin de permettre au server KepserverEX d'accéder aux données de l'automate, une configuration du réseau est primordiale. Il est important pour chaque CPU d'avoir sa propre adresse pour éviter toute confusion entre les données.

Le protocole Ethernet TCP /IP nous permet de connecter les automates programmables à notre tierce programme Flexsim. Notons que le choix du protocole dépend de l'application souhaitée.

Nous devons choisir une CPU de type PN afin d'être capable de configurer un réseau Ethernet industriel grâce au protocole PROFINET (PN) comme la CPU 314C-2 PN/DP qui qui contrôle notre station.

Maintenant, nous exécutons STEP 7 (PG) et S7- PLCSIM (API) sur le même PC , cela signifie qu'ils doivent avoir la même adresse IP.

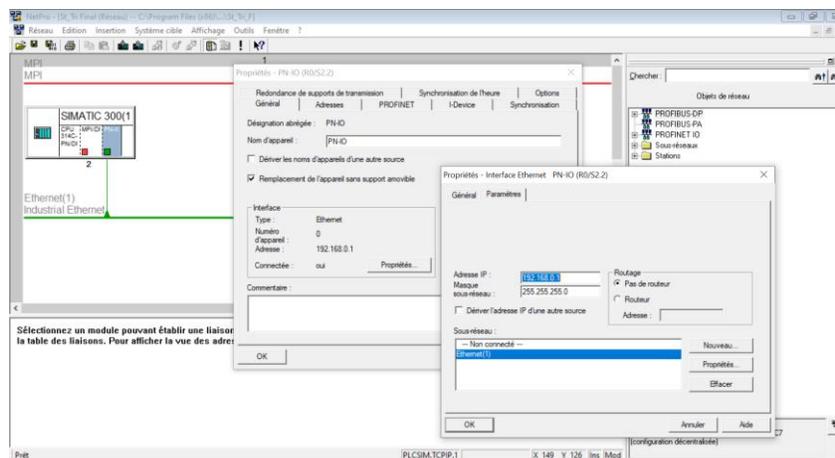


Figure 4.20 Configuration d'un réseau Ethernet sur STEP 7

Sur PLCSIM, nous choisissons PLCSIM TCP/IP dans le simulateur

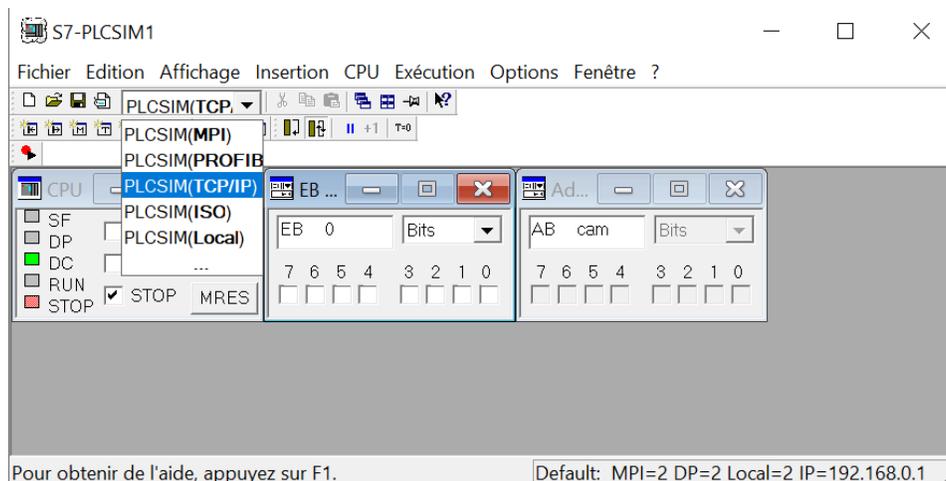


Figure 4.21 choix du protocole TCP/IP sur PLCSIM

Il ne faut pas oublier d'enregistrer, charger et fermer la configuration matérielle.

N'hésitons pas à charger le programme, le connecter en ligne et vérifier que tout fonctionne bien.

Il est impératif d'utiliser NetToPlcsim qui nous permet d'utiliser PLCSIM avec des logiciels tiers via TCP/IP. Pour le configurer, il est important en premier lieu de libérer le port 102 ensuite pour l'adresse network IP, nous ne choisirons pas l'adresse IP de notre carte réseau mais plutôt l'adresse du réseau Wifi sur lequel nous sommes connectés, que nous devons introduire manuellement. L'adresse IP PLCSIM correspond à l'adresse IP de l'automate sur STEP 7. Nous nous assurerons que c'est le bon numéro de rack qui est introduit et nous cocherons enfin la case enable TSAP check.

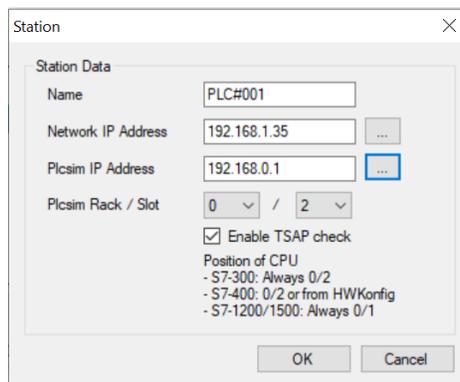


Figure 4.22 Configuration de l'adressage sur Nettoplcsim

En appuyant sur Start server, la connexion est établie.

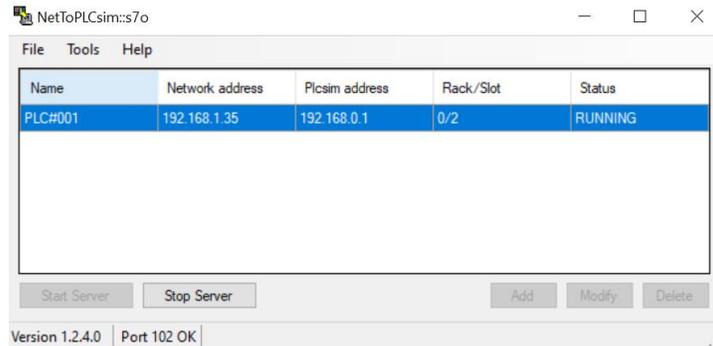


Figure 4.23 Démarrage du serveur sur Nettoplcsim

Nous sommes maintenant prêt à configurer Kepsserver. Il est impératif dans cette étape de définir le canal d'accès, l'automate utilisé, les adressages, les Tags d'entrées et de sorties.

Pour procéder à la configuration, il faut ouvrir un nouveau projet sur Kepsserver, ajouter un nouveau chemin qui correspond à notre paramétrage « Siemens TCP/IP Ethernet », choisir le réseau wifi dans l'adaptateur réseau, d'autres choix sont plausibles mais nous optons pour ce dernier.

Après cela, il suffit d'ajouter le modèle de l'API et son adresse IP. Puisque nous utilisons Nettoplcsim pour se connecter à PLCSIM, l'adresse IP de l'API va correspondre à la même adresse du réseau wifi introduite dans l'étape précédente et non pas à l'adresse IP saisie sur STEP 7.

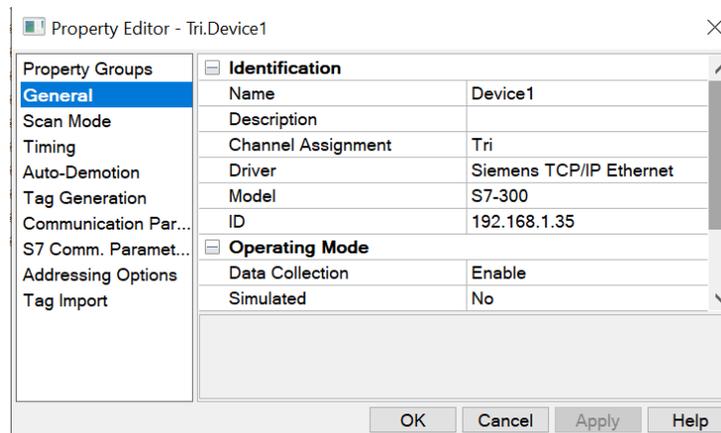


Figure 4.24 Configuration matérielle de Kepsserver

Il est préférable d'importer le programme STEP 7 afin de faciliter l'accès et la visualisation des entrées et des sorties automatiquement au lieu de les saisir manuellement.

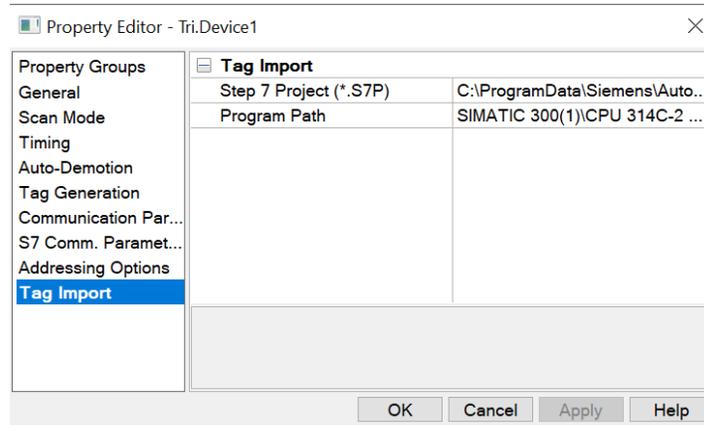


Figure 4.25 Importation des entrées et des sorties sur Kepsserver

Note : Si l'utilisateur souhaite écrire son programme sous TIA PORTAL, il faudra installer TIA Portal Openness V14 SP1 ou une version récente de celle-ci et utiliser TIA Portal Exporter afin d'importer les tags comme illustré dans la figure 4.26.

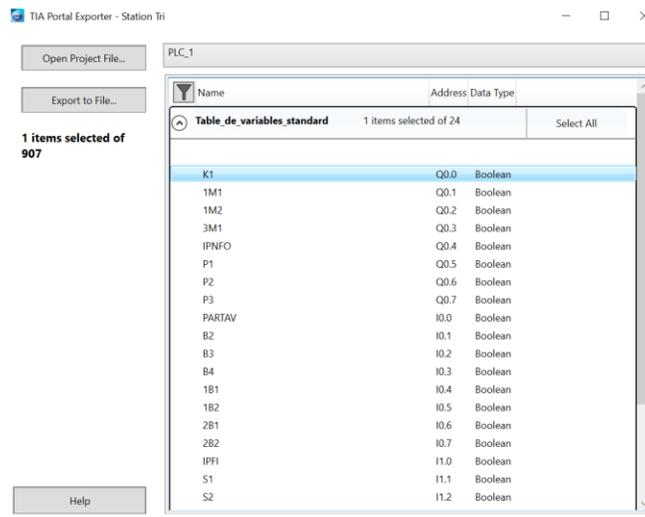


Figure 4.26 Extraction des entrées et des sorties sur TIA Portal Exporter

A la fin de ces étapes la configuration du réseau est terminée.

4.3.6 Modélisation 3D sur Flexsim :

A ce stade là nous commençons à construire notre modèle de simulation en mettant le convoyeur principal ainsi que les trois glissières, nous représenterons les capteurs par des points de décisions et les actionneurs par des moteurs.

Il est possible d'importer des formes 3D fait de Sketch Up sur Flexsim en format .skp en les glissant à partir de la bibliothèque visuelle afin que notre modèle ait plus de ressemblance avec le système réel. D'autres formats sont possibles tel que .3ds .ac .ase .blend .cob .csm .dae .dxf fbx...etc.

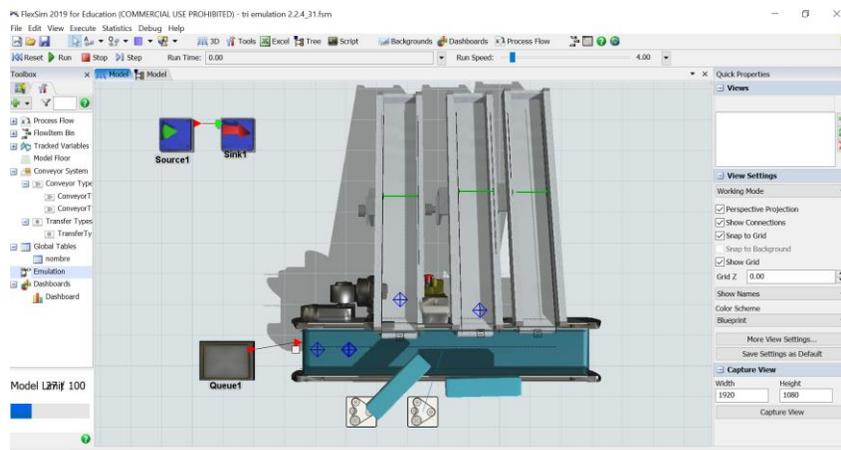


Figure 4.27 Modèle 3D de la station de Tri sur Flexsim

L'étape qui suit consiste à décrire le fonctionnement de la station en utilisant le process flow.

La logique de celui-ci est semblable au programme écrit en Ladder ou en Grafset, il décrit les étapes, les transitions et les temporisations, le changement des variables d'entrées et de sorties, il se chargera de contrôler le fonctionnement du modèle 3D.

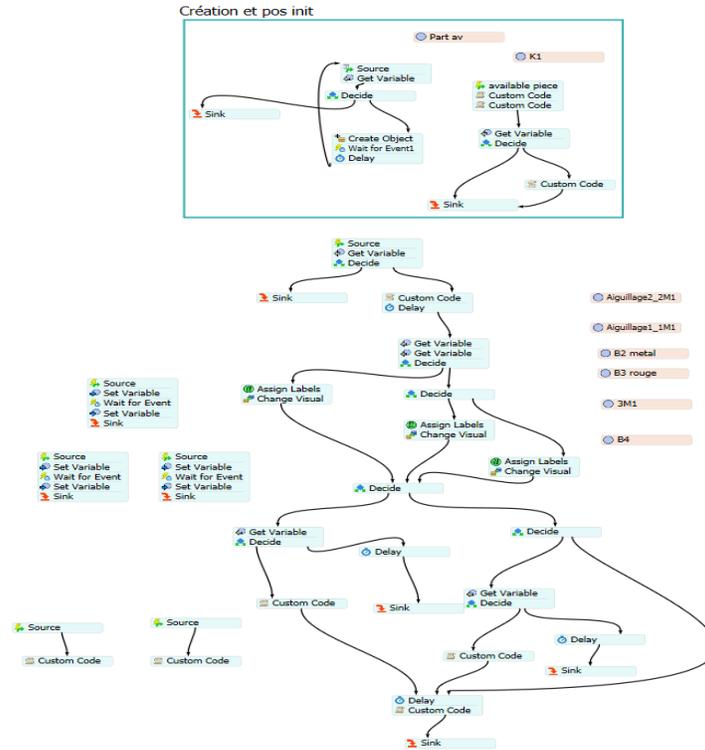


Figure 4.28 Process Flow de la station de tri

Les variables de l'automate sont utilisées comme des variables d'émulation externe qui peuvent représentées soit des capteurs (entrées API) ou contrôleurs (sorties API). Sur Flexsim, nous récupérons le signal de ces dernières afin décider du cheminement du Process Flow.

4.3.7 Configuration de l'émulation sur Flexsim

Lorsque le modèle de simulation est fini et fonctionne correctement, nous pouvons passer à la configuration de la boîte d'émulation de Flexsim qui nous permettra de communiquer et d'échanger les données avec Kepserver.

L'outil d'émulation crée un lien entre Flexsim et les API externes ou les clients / serveurs qui communiquent avec les API. Cet outil peut créer plusieurs connexions et définir des variables pour chacune de ces connexions. Cet outil peut prendre en charge les protocoles suivants : OPC DA, Modbus TCP / IPv4, Modbus TCP / IPv6, Modbus RTU.

L'outil d'émulation est accessible à partir de la boîte à outils.

Dans notre cas, nous allons utiliser Une connexion au serveur OPC local Kepserver utilisant le protocole d'accès aux données (DA).

Lorsque nous configurons une connexion à un serveur dans Flexsim, nous pouvons décider d'activer ou de désactiver cette connexion. Si la connexion est active, Flexsim lit et écrit des données directement à partir d'un serveur ou d'un automate réel. Si la connexion est inactive, elle extraira à la place les données du modèle de simulation théorique que nous avons créé dans Flexsim

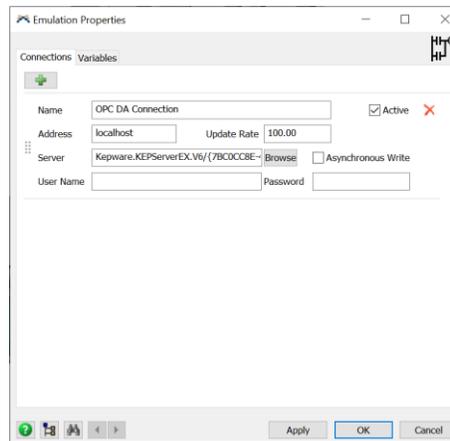


Figure 4.29 Connexion OPC DA sur la boîte d'émulation

Les connexions OPC DA peuvent être réelles, ce qui signifie que Flexsim communiquera via un port Ethernet physique sur l'ordinateur, ou elles peuvent être virtuelles, ce qui signifie qu'un programme logiciel émule une connexion TCP comme c'est le cas pour nous.

Dans l'**adresse IP** de la connexion, nous utilisons localhost pour créer une connexion avec un serveur sur le même ordinateur. Nous pouvons également une adresse IP si le serveur n'est pas sur le même ordinateur.

Update rate est le taux de mise à jour en unités de temps du modèle, pour Flexsim pour interroger le serveur pour les modifications.

Nous préciserons ensuite **le serveur** Kepserver en parcourant une liste de serveurs auxquels nous pouvons nous connecter.

Si la case « **Asynchronous Write** » est cochée, l'écriture de données sur le serveur obligera Flexsim à attendre la fin de l'opération d'écriture avant de poursuivre l'exécution.

Sur l'onglet variables, il existe deux types de variables qui peuvent être ajoutées: les capteurs (entrées API) et les actionneurs (sorties API).

- **Capteurs (entrées API) :**

Les capteurs sont utilisés pour écrire des données sur l'automate ou le serveur. Si la connexion est active, la modification de la valeur du capteur entraînera l'écriture de cette valeur sur l'automate ou le serveur. Si la connexion est inactive, la valeur ne sera modifiée que localement.

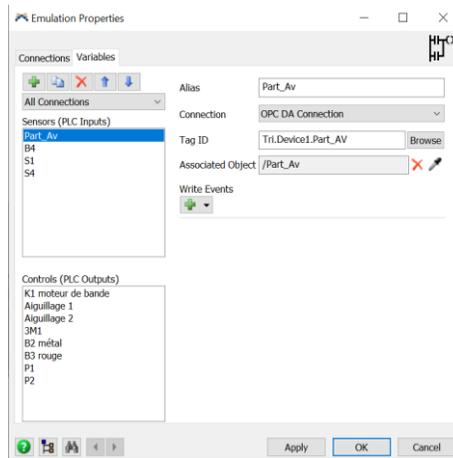


Figure 4.30 Interface des entrées

- **Actionneurs (sorties API) :**

Les sorties sont utilisées pour lire les données de l'automate ou du serveur. Cela permet à la logique du Process flow interne d'agir comme l'API et d'apporter des modifications aux commandes qui peuvent démarrer / arrêter les moteurs, etc.

Si la case « **Subscribe to changes** » qui signifie S'abonner aux modifications est cochée, le serveur informera Flexsim de la valeur modifiée (abonnement) ou Flexsim demandera des modifications à l'aide d'événements.

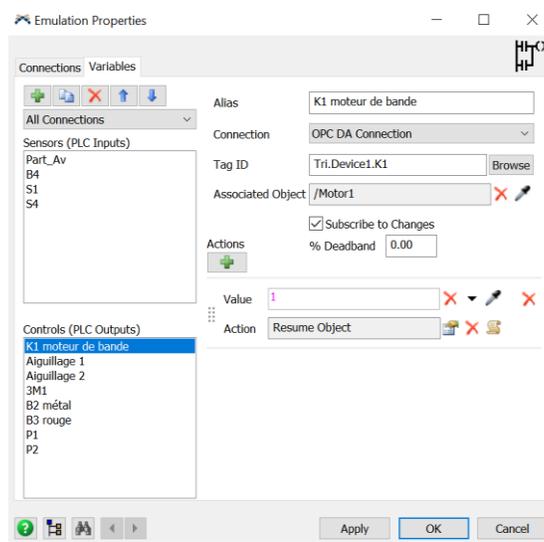


Figure 4.31 Interface des sorties

Les propriétés des actionneurs et des capteurs sont décrites ci-dessous :

Le nom de la variable est le nom utilisé pour identifier la variable dans Flexsim.

Le Tag ID est le chemin de la variable sur le serveur.

L'objet associé lie cette variable à un objet du modèle.

Le **% deadband** peut filtrer les petites modifications de la valeur (bruit) du contrôle, lors de l'abonnement à des changements analogiques.

Les actions nous permettent de définir ce qui se passe avec le changement de la variable en fonction de sa nouvelle valeur. Chaque action est évaluée de haut en bas jusqu'à ce que la nouvelle valeur corresponde au champ Valeur ou qu'elle atteigne une valeur par défaut. L'action sera alors déclenchée et aucune autre action ne sera testée.

Maintenant que tout est configuré, nous pouvons regarder si la connexion est bien établie entre Flexsim et l'automate sur le client rapide de Kepserver.

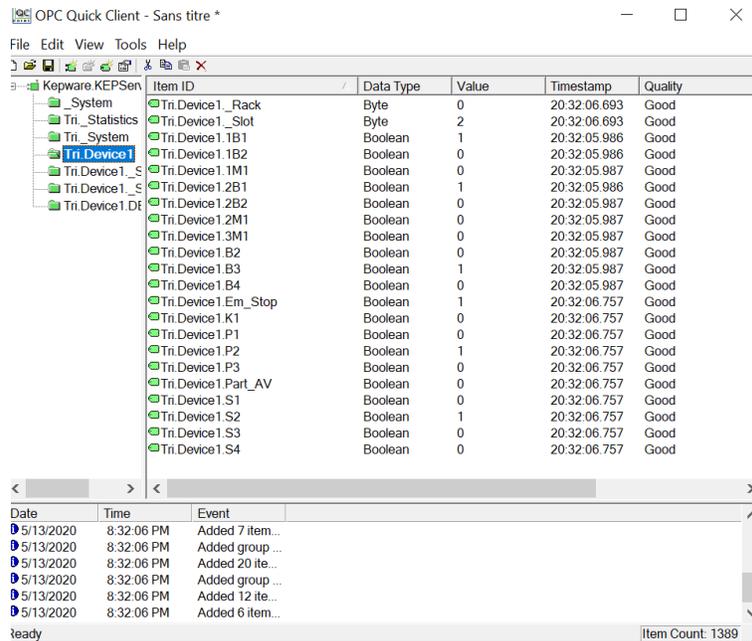


Figure 4.32 Client OPC sur Kepserver

Notons que tous les projets de simulation ne bénéficient pas nécessairement de l'utilisation de l'émulation.

Nous pouvons tester la précision de la logique de l'automate après sa programmation. Une fois la logique du Process Flow est implémentée, nous pouvons connecter Flexsim au système réel et déterminer si la logique de l'automate a été correctement programmée. Au fur et à mesure que le modèle de simulation s'exécute et reçoit les données du système réel en temps réel, nous pouvons comparer le modèle de simulation avec les résultats du système pour valider que la logique du process Flow a été correctement programmée.

4.3.8 Résultats

Maintenant que toute la configuration est mise en œuvre et que notre architecture est implémentée avec succès, Nous obtenons un modèle Flexsim simulé représentant le système virtuel contrôlé par la station de tri préconfigurée sur CIROS représentant le système réel. Nous résumons la démarche et le résultat dans le schéma suivant :

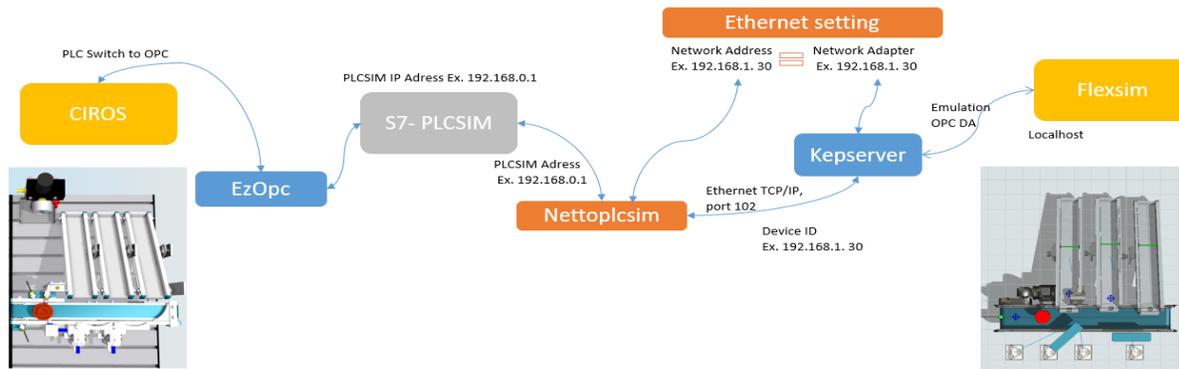


Figure 4.33 Schéma synthétisant la démarche d'implémentation

4.4 Conclusion

Ce chapitre nous a permis de présenter l'architecture du jumeau numérique que nous avons proposée. Il représente donc la somme des travaux faits dans ce projet, Après avoir présenté la ligne d'assemblage et la MPS 500 qui nous ont servi de support pour la validation de concept et ensuite exposé l'intérêt des logiciels utilisés, ce dernier chapitre nous a permis de détailler et appliquer l'architecture proposée et de l'intégrer au CPPS afin d'améliorer sa fiabilité, anticiper les pannes et les temps d'arrêt et de réduire les coûts de maintenance.

D'autre part, ce chapitre est venu élucider de manière pédagogique les démarches de création d'un lien entre Flexsim et CIROS, à savoir maîtriser ces deux entités séparément, manipuler les stations préconfigurées, créer un modèle de simulation adéquat, configurer un OPC afin d'échanger les données. Ceci nous permet de sortir de la sphère de la théorie et de passer à la pratique et l'apprentissage afin d'obtenir une représentation virtuelle de deux systèmes synchronisés en temps réel. La méthode utilisée nous permet de faire l'implémentation sans risquer d'endommager le matériel. Nous pourrons dès lors passer à l'implémentation sur le système réel sans crainte.

CONCLUSION ET PERSPECTIVES

De nos jours, tous les systèmes peuvent être connectés grâce aux technologies de l’IoT et CPS, permettant de fonctionner en autonomie et indépendamment. La dernière révolution industrielle souligne la nécessité de promouvoir la communication et l’échange de données entre les systèmes réels et virtuels afin d’accroître la productivité de l’entreprise. Pour rester compétitif, ils doivent être plus flexibles et réactifs afin de s’adapter aux besoins ainsi qu’aux aléas de fabrication. Au niveau stratégique, ils sont aussi appelés à tirer profit des opportunités prometteuses offertes par l’industrie 4.0.

Pour atteindre notre objectif, nous supposons que les concepts de la communication, la simulation et l’agrégation des données d’un jumeau numérique peuvent améliorer la fiabilité des systèmes de façon efficace. Ce dernier semble être une solution pertinente permettant de modéliser l’état d’un système, simuler ses performances, faciliter l’accès aux informations, donnant ainsi la possibilité de détecter tout écart de fonctionnement et de prévenir les dysfonctionnements éventuels du système et la préparation d’intervention de maintenance. Cela conduira également à favoriser l’anticipation des pannes, ce qui entraînera à son tour de meilleurs coûts et de meilleurs temps.

Pour cela, une analyse de la littérature a été effectuée dans le premier chapitre et nous a permis de dresser des constatations à partir des approches existantes concernant la maintenance prédictive dans les systèmes de production. Nous avons pu conclure que plusieurs auteurs utilisent des méthodes d’intelligence artificielle ou d’optimisation en tenant compte des masses de données accumulées afin de prendre des décisions des opérations de maintenance, mais très peu utilisent les données en temps réel pour détecter les dysfonctionnement d’un système.

Nous avons alors proposé une architecture afin d’implémenter dans les systèmes de production, un jumeau numérique apte à faire de la maintenance prédictive et réactive. Cette architecture repose principalement sur la synchronisation entre le système réel et le système virtuel, l’analyse des données et les techniques d’intelligence artificielle.

Conclusion et perspectives

La simulation joue également un rôle crucial dans notre approche car elle est capable de représenter, de façon assez perspicace, le comportement idéal d'un système, tout en prenant en considération les perturbations aléatoires et les écarts entre les deux systèmes réel et virtuel.

Par la suite, La mise en œuvre de l'architecture proposée a été appliquée sur les postes de la ligne d'assemblage de l'IUT de Nantes. Plusieurs tests réels ont été réalisés sur le terrain et cela semble fonctionner parfaitement.

Une autre approche pédagogique a été proposée afin d'apprendre et de maîtriser les outils Flexsim, OPC, CIROS d'une part mais aussi de manipuler les stations de FESTO Didactique de la chaîne de production de la MPS500.

Les travaux de ce mémoire forment un socle qui ouvre des perspectives intéressantes, à plus ou moins long terme. Différentes techniques d'intelligence artificielle seront mises en œuvre et testées afin de sélectionner la meilleure capable d'optimiser les stratégies de maintenance.

Une autre direction future est de généraliser l'architecture pour obtenir une première conception et mise en œuvre dans le contexte de l'intégration au sein du Digital Twin de CPPS. L'idée serait d'ajouter d'autres détecteurs avec des objectifs différents : minimiser la consommation d'énergie, améliorer la productivité, etc. L'analyse des données historiques dans le centre de maintenance intelligent sera alors basée sur plusieurs objectifs. Ainsi, le Digital Twin sera en mesure d'aider les superviseurs à prendre des décisions fondées sur des compromis en fournissant des données avancées pour l'aide à la décision.

D'autres travaux seront menés sur l'optimisation de la mise en œuvre de capteurs intelligents (localisation, fréquence d'envoi des données, cycle, etc.) pour obtenir des données plus pertinentes via la technologie IoT.

Des TP de simulation sont développés en Annexe afin de permettre à l'étudiant de se familiariser avec l'interface de Flexsim, ainsi que ses bibliothèques et ses concepts de bases.

Les contributions scientifiques de ce mémoire ont fait l'objet d'un papier publié lors d'un atelier européen.

BIBLIOGRAPHIE

1. Industrie 4.0 : définition et mise en œuvre vers l'usine connectée. In: Visiativ Solut. <https://www.visiativ-solutions.fr/industrie-4-0/>. Accessed 8 Mar 2020
2. l'Innovation M de l'Économie et de S'améliorer / Industrie 4.0 - MEI. In: Ministère L'Économie L'Innovation. <https://www.economie.gouv.qc.ca/objectifs/ameliorer/industrie-40/page/guides-et-outils->
3. Alcácer V, Cruz-Machado V (2019) Scanning the Industry 4.0: A Literature Review on Technologies for Manufacturing Systems. *Eng Sci Technol Int J* 22:899–919
4. Rojko A (2017) Industry 4.0 Concept: Background and Overview. *Int J Interact Mob Technol IJIM* 11:77
5. Danjou C, Rivest L, Pellerin R Industrie 4.0 : des pistes pour aborder l'ère du numérique et de la connectivité. 27
6. Lee J, Bagheri B, Kao H-A (2015) A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. <https://doi.org/10.1016/j.mfglet.2014.12.001>
7. Cardin O (2019) Classification of cyber-physical production systems applications: Proposition of an analysis framework. *Comput Ind* 104:11–21
8. Rießmann M, Lorenz M, Gerbert P, Waldner M, Justus J, Harnisch M Industry 4.0: The Future of Productivity and Growth in Manufacturing Industries. 14
9. Witkowski K (2017) Internet of Things, Big Data, Industry 4.0 – Innovative Solutions in Logistics and Supply Chains Management. *Procedia Eng* 182:763–769
10. Vaidya S, Ambad P, Bhosle S (2018) Industry 4.0 – A Glimpse. *Procedia Manuf* 20:233–238
11. Neugebauer R, Hippmann S, Leis M, Landherr M (2016) Industrie 4.0 - From the Perspective of Applied Research. *Procedia CIRP* 57:2–7
12. Kritzinger W, Karner M, Traar G, Henjes J, Sihn W (2018) Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-Pap* 51:1016–1022
13. Beyond the digital twin – making analytics come alive. <https://www.iosb.fraunhofer.de/servlet/is/81714/>. Accessed 12 Mar 2020
14. Glaessgen E, Stargel D (2012) The digital twin paradigm for future NASA and U.S. air force vehicles. <https://doi.org/10.2514/6.2012-1818>
15. Negri E, Fumagalli L, Macchi M (2017) A Review of the Roles of Digital Twin in CPS-based Production Systems. *Procedia Manuf* 11:939–948

Bibliographie

16. Reid JB, Rhodes DH Digital System Models: An investigation of the non-technical challenges and research needs. 10
17. The Mysterious History of Digital Twin Technology and Who Created It. In: Chall. Advis. <https://www.challenge.org/insights/digital-twin-history/>. Accessed 12 Mar 2020
18. West T, Pyster A (2015) Untangling the Digital Thread: The Challenge and Promise of Model-Based Engineering in Defense Acquisition. *INSIGHT* 18:45–55
19. Industry 4.0 and the digital twin technology | Deloitte Insights. <https://www2.deloitte.com/us/en/insights/focus/industry-4-0/digital-twin-technology-smart-factory.html>. Accessed 27 Apr 2020
20. Khazraei K, Deuse J (2011) A strategic standpoint on maintenance taxonomy. *J Facil Manag* 9:96–113
21. Efthymiou K, Papakostas N, Mourtzis D, Chryssolouris G (2012) On a Predictive Maintenance Platform for Production Systems. *Procedia CIRP* 3:221–226
22. Nguyen A-T, Reiter S, Rigo P (2014) A review on simulation-based optimization methods applied to building performance analysis. *Appl Energy* 113:1043–1058
23. Zhu L, Yu FR, Wang Y, Ning B, Tang T (2019) Big Data Analytics in Intelligent Transportation Systems: A Survey. *IEEE Trans Intell Transp Syst*. <https://doi.org/10.1109/TITS.2018.2815678>
24. Rault R, Trentesaux D (2018) Artificial Intelligence, Autonomous Systems and Robotics: Legal Innovations. In: Borangiu T, Trentesaux D, Thomas A, Cardin O (eds) *Serv. Orientat. Holonic Multi-Agent Manuf. Proc. SOHOMA 2017*. Springer International Publishing, Cham, pp 1–9
25. Kravets AG, Bolshakov AA, Shcherbakov M (eds) (2020) *Cyber-Physical Systems: Advances in Design & Modelling*. <https://doi.org/10.1007/978-3-030-32579-4>
26. Chaudhuri A (2018) Predictive Maintenance for Industrial IoT of Vehicle Fleets using Hierarchical Modified Fuzzy Support Vector Machine.
27. Lee J, Azamfar M, Singh J, Siahpour S (2020) Integration of Digital Twin and Deep Learning in Cyber-Physical Systems: Towards Smart Manufacturing. <https://doi.org/10.1049/iet-cim.2020.0009>
28. Djurdjanovic D, Lee J, Ni J (2003) Watchdog Agent—an infotronics-based prognostics approach for product performance degradation assessment and prediction. *Adv Eng Inform* 17:109–125
29. Raza J, Liyanage JP, Al Atat H, Lee J (2010) A comparative study of maintenance data classification based on neural networks, logistic regression and support vector machines. *J Qual Maint Eng* 16:303–318
30. Garcia MC, Sanz-Bobi MA, del Pico J (2006) SIMAP: Intelligent System for Predictive Maintenance: Application to the health condition monitoring of a windturbine gearbox. *Comput Ind* 57:552–568
31. Niu G, Li H (2017) IETM centered intelligent maintenance system integrating fuzzy semantic inference and data fusion. *Microelectron Reliab* 75:197–204

Bibliographie

32. Caesarendra W, Widodo A, Yang B-S (2010) Application of relevance vector machine and logistic regression for machine degradation assessment. *Mech Syst Signal Process* 24:1161–1171
33. Krenek J, Kuca K, Blazek P, Krejcar O, Jun D (2016) Application of Artificial Neural Networks in Condition Based Predictive Maintenance. In: Król D, Madeyski L, Nguyen NT (eds) *Recent Dev. Intell. Inf. Database Syst.* Springer International Publishing, Cham, pp 75–86
34. Jegadeeshwaran R, Sugumaran V (2015) Fault diagnosis of automobile hydraulic brake system using statistical features and support vector machines. *Mech Syst Signal Process* 52–53:436–446
35. Elghazi Y, Chaouche ARY Implémentation d'un Digital Twin des stations commandées par automates de la MPS500. 73

ANNEXE A – TRAVAUX PRATIQUES SOUS FLEXSIM

TP 0 : initiation à Flexsim

- **La fenêtre de Flexsim :**

Sur la fenêtre de Flexsim, on peut distinguer plusieurs compartiments :

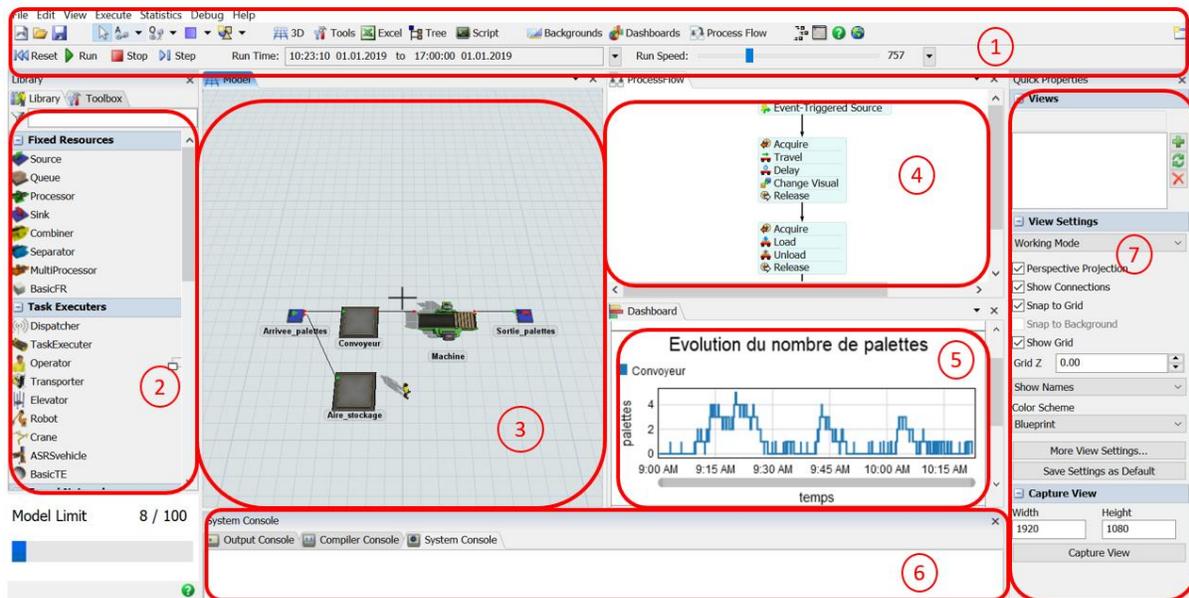


Figure 0.1 Interface de Flexsim

1. Menu principal

2. Bibliothèques de modélisation :

Flexsim contient plusieurs bibliothèques de modélisation :

- Ressources fixes : elles ne se déplacent pas durant la simulation, telles que les machines.
- Ressources mobiles : elles peuvent se déplacer durant la simulation. Il s'agit des ressources qui exécutent les tâches ou les transportent.
- Réseaux de déplacement : définissant les allées de circulation et les chemins à prendre.
- Convoyeurs

- AGV
- Navigation A star : Algorithme de recherche de chemin

On peut retrouver également les bibliothèques pour le Process flow et le tableau de bord

3. Un modèle 3D du système physique:

Des objets de simulation représentant des stocks, des machines, des chariots de transports, des convoyeurs, des robot, ...

4. Un ensemble de PROCESS FLOW

Des **Graphes d'activités** représentant l'enchaînement des activités décrivant le comportement logique du système.

5. Tableau de bord

Afin d'analyser les performances, Flexsim utilise des tableaux de bord.

La bibliothèque des graphes peut mesurer les :

- Les encours (WIP = Work in Progress)
- Les débits (Throughput) : pièce produite par heure...
- Les temps d'attente (staytime)
- Les états : au repos, occupé, en panne, en attente...

6. Propriétés rapides

Ces propriétés des objets définis dans le modèle nous permettent de modifier les paramètres rapidement.

7. Consoles

Les consoles permettent la communication durant une simulation, de suivre les événements et repérer les erreurs.

Flow Item Vs Token :

Les **Flow items** circulent dans le modèle 3D et modélisent le flux des produits dans le système.

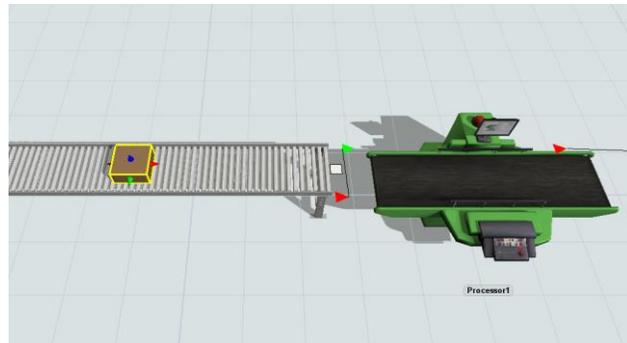


Figure 0.2 Flux d'une entité sur le modèle 3D de Flexsim

Des **Tokens ou jetons** circulent dans les Process Flow, leurs déplacements modélisent l'évolution des activités dans le modèle.

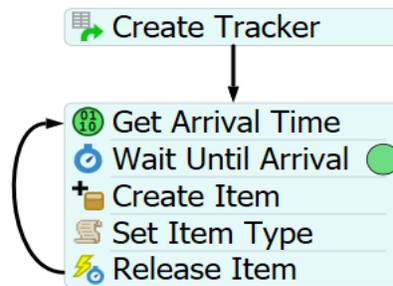


Figure 0.3 Le déplacement d'un jeton sur le Process Flow de Flexsim

TP 1 : Introduction aux concepts généraux de modélisation

Partie 1 : construire le modèle 3D

Objectif : se familiariser avec les concepts suivants :

- Item types
- Labels
- Distribution statistique
- Triggers

Description :

Les pièces arrivent à intervalles d'environ 30 s (loi exponentielle)

Il arrive 2 types de pièces dans les proportions suivantes: 60% de type 1, 40% de type 2

Il y a 2 machines, une pour chaque type de pièces après l'usinage, les pièces sont mises en attente pour être contrôlées. La durée du contrôle est constante: 10 s. Si la qualité de la pièce est bonne, elle quitte cet atelier. Les pièces défectueuses 10% sont renvoyées en production pour être rectifiées.

Les temps d'usinage sont identiques pour toutes les machines: uniforme (100,120). Pour une pièce défectueuse à rectifier, le temps d'usinage est plus long: uniforme (120, 130)

Les pièces défectueuses sont prioritaires sur les autres pièces

Disposition :

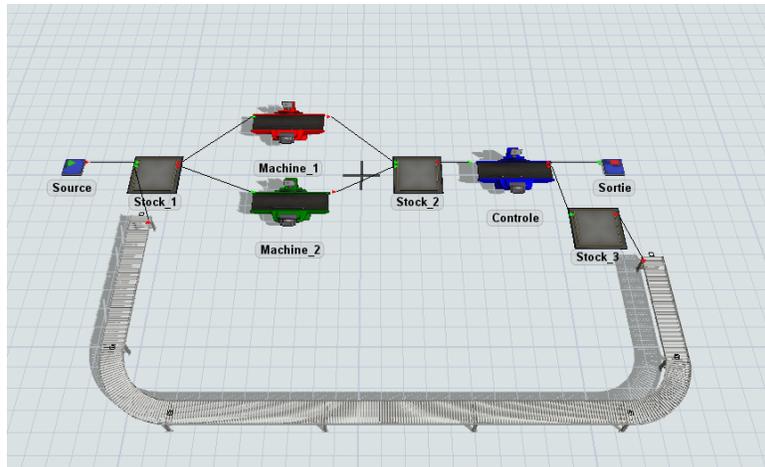


Figure 0.4 disposition du premier modèle

Étape 1 : création d'un nouveau modèle

Déterminer les unités les plus adaptées au système modélisé

- Unités de temps : secondes
- Unité de longueur : mètres, millimètres...

Fixer la date et l'heure du début du modèle

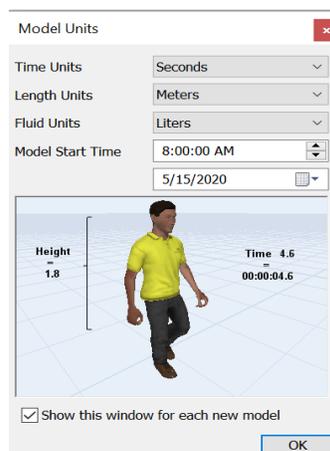


Figure 0.5 les unités d'un modèle de simulation sur Flexsim

Étape 2 : Choisir les objets Flexsim

Choisir les objets Flexsim qui représentent les éléments les plus adaptés à notre système

- Source : l'arrivée des pièces
- Processor : Les machines d'usinage et de contrôle
- Queue : stock tampon
- Conveyor : le retour des pièces défectueuses aux machines d'usinage

Il existe plusieurs façons afin d'insertion d'objets dans le modèle

1. Par glisser déposer dans la vue 3D
2. Par double clic dans la vue 3D puis sélectionner l'objet choisi
3. Mode rapide de création : 1 clic gauche sur un objet de la librairie puis à chaque clic dans la vue 3D

Étapes 3 : créer les connexions

Il existe deux types de connexion: Les connexions entre les ports d'entrée/sortie des ressources fixes et les connexions entre les ports centraux des ressources fixes et des ressources mobiles.

Une connexion **A** entre deux ressources fixes relie un port de sortie de l'objet amont à un port d'entrée de l'objet aval

Elle aide à créer le chemin que peuvent emprunter les pièces.

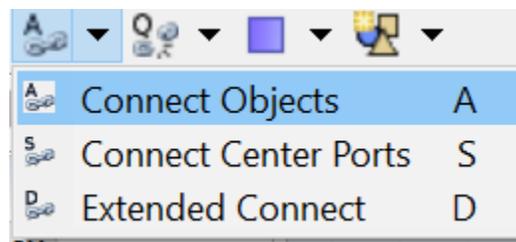


Figure 0.6 Connexions aux objets type

Étape 4 : éditer des propriétés des objets

- **Arrivée des pièces :**

Rappel : Les pièces arrivent à intervalles d'environ 30 s (loi exponentielle)

- Cocher Arrival at time 0 pour forcer l'arrivée de la première pièce à t 0
- Temps d'inter arrivée (Inter Arrival time) est durée qui s'écoule entre 2 arrivées de pièces

Cet intervalle peut être constant ou défini à l'aide d'une distribution statistique pour représenter la variabilité des arrivées.

Il existe différentes distributions qu'on peut retrouver sur le menu déroulant.

Pour une distribution statistique exponentielle, il existe 3 paramètres :

- Location : minimum
- Scale : facteur échelle
- Stream : germe aléatoire

La moyenne de la distribution et la somme du minimum et du facteur d'échelle

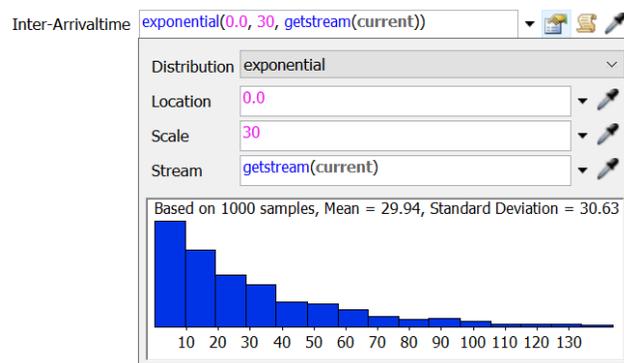


Figure 0.7 distribution statistique de la loi exponentielle

- **Déclencheurs (Triggers) :**

Une logique exécutée chaque fois qu'un événement se produit, elle aide à définir et personnaliser le comportement du système lors d'une occurrence d'événement spécifique, comme par exemple l'arrivée ou le départ d'un produit.

Les déclencheurs permettent de :

Annexe A – Travaux pratiques sous FLEXSIM

- Écrire / lire des informations / depuis un flux.
- Modifier les propriétés visuelles des éléments de flux ou des objets.
- Envoyer des messages à d'autres objets.
- ...etc.

Rappel : Il arrive 2 types de pièces dans les proportions suivantes: 60% de type 1, 40% de type 2

On ajoute un déclencheur à l'arrivée des pièces afin de spécifier le pourcentage de chaque type.

Pour avoir un bonus visuel, on spécifie la couleur des types également.

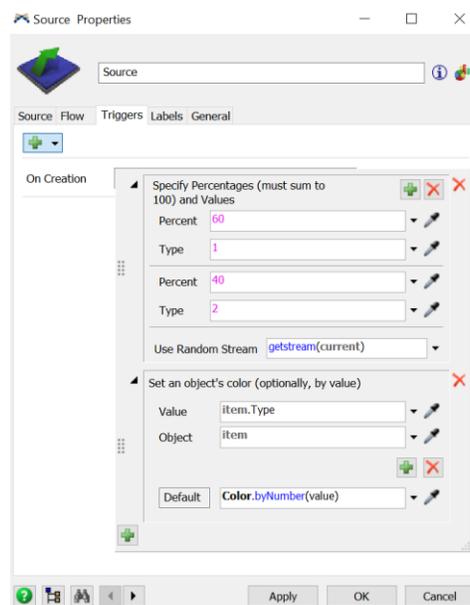


Figure 0.8 les évènements survenant lors de la création d'une pièce

Tous les éléments de flux ont un attribut (Label) appelé itemtype. Il s'agit d'un identifiant numérique utilisé pour distinguer un "type" de flux de données d'un autre dans les décisions de traitement et de routage dans un modèle.

Annexe A – Travaux pratiques sous FLEXSIM

- **Labels:**

Les labels ou les attributs sont les informations associées à un flow item (et d'autres objets de modélisation) sous forme d'étiquettes.

Les Flowitems et autres objets sur Flexsim peuvent avoir un nombre illimité d'étiquettes.

Les étiquettes ont un nom et des données. Ces données peuvent être numériques, chaîne, liste ou tableau.

Par exemple un flow item peut être caractérisé par une référence d'article, n°de commande, un type...etc.

- **Le stock :**

Le paramètre Send To Port permet de définir par quel port de sortie envoyer le produit

Il existe plusieurs règles de flux :

- Premier port disponible (First
- Vers la file d'attente la plus courte
- Par expression
- Aléatoirement ...etc.

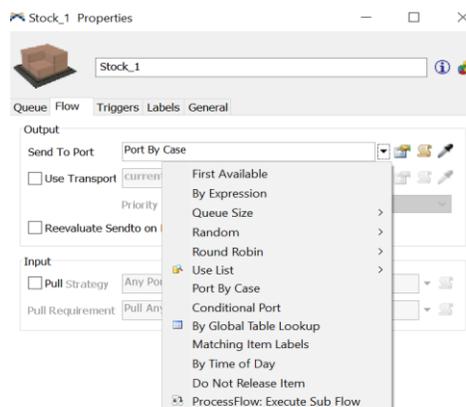


Figure 0.9 Propriétés du stock

Dans notre modèle, on va choisir un port de sortie en fonction du type de produit « Port By Case » cela signifie qu'on envoie type 1 sur machine 1 et type 2 sur machine 2

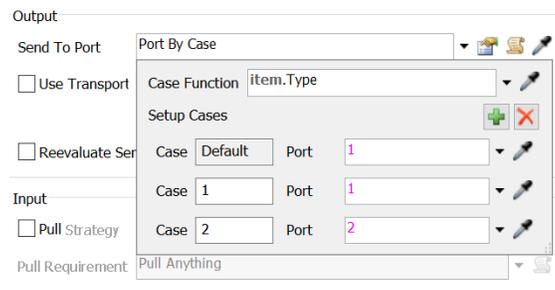


Figure 0.10 configuration des propriétés du stock

- **Pièces défectueuses :**

Rappel : 10% des pièces sont défectueuses et renvoyées en production pour être rectifiées

A la sortie du poste de contrôle, 10% des pièces sont défectueuses et sont renvoyées.

Les pièces défectueuses ont des propriétés spécifiques (priorité, temps d'usinage plus long), il est donc nécessaire d'étiqueter ces informations de défaut sur ces pièces en utilisant un **Label**.

Afin d'attribuer ces labels, on va dans la fenêtre des FlowItem bin et on les ajoute sur l'entité box, qui sera créée ultérieurement par la source

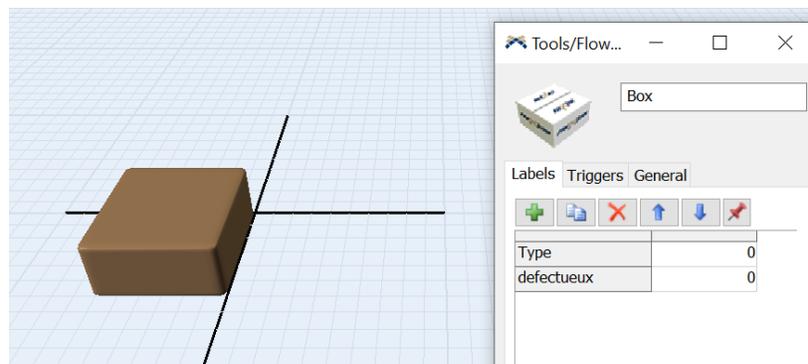


Figure 0.11 Fenêtre du Flow Item Bin

Annexe A – Travaux pratiques sous FLEXSIM

Après le contrôle, si la pièce est défectueuse, le label passe à 1

Sur le poste de contrôle, on choisit d'envoyer 90% à la sortie et 10% de pièces défectueuses à un stock tampon.

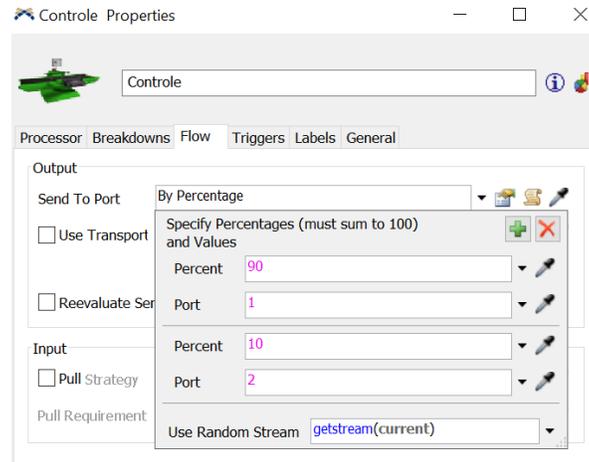


Figure 0.12 Propriétés du poste de contrôle

A leurs entrées au stocks, le label « défectueux » passe à 1 et la forme des pièces est changée en cylindre afin d'avoir un meilleur suivi visuel, à l'aide d'un déclencheur « On Entry »

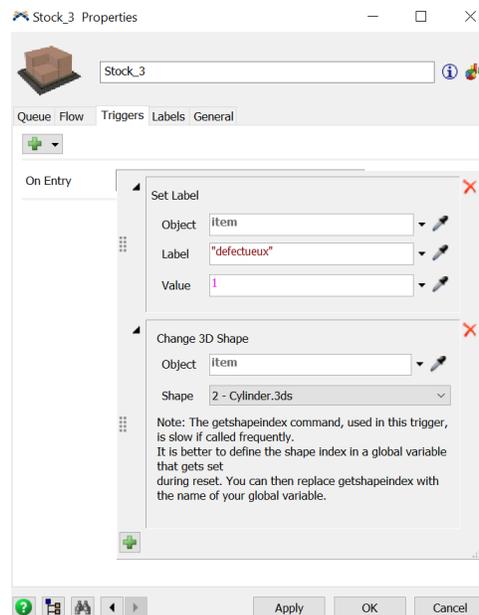


Figure 0.13 Configuration des Triggers

Rappel : Les temps d'usinage sont identiques pour toutes les machines: uniforme (100,120). Pour une pièce défectueuse à rectifier, le temps d'usinage est plus long: uniforme (120, 130)

Sur les machines d'usinage, on modifie les temps opératoires en fonction de la défectuosité des pièces.

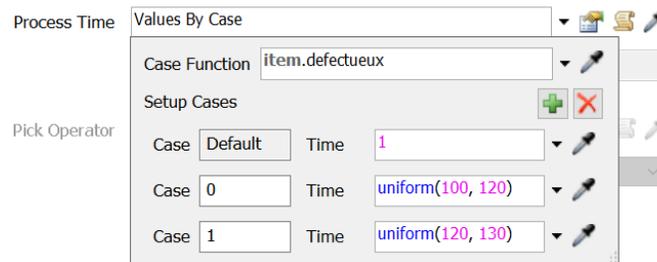


Figure 0.14 configuration des temps opératoires des machines

Afin de prioriser les pièces défectueuses lors de leur retour au premier stock tampon, on met un déclencheur « En Entry » qui trie les pièces entrantes en fonction de leur label « défectueux » en ordre décroissant.

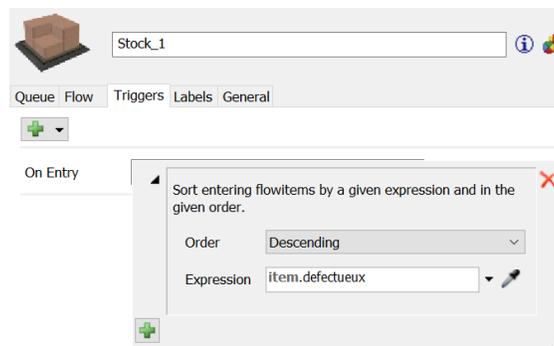


Figure 0.15 Configuration des Triggers à l'entrée du stock

Étape 5: Lancer la simulation

On lance le modèle à vitesse maximum pour collecter les résultats le plus vite possible et on visualise le déroulement.

Questions :

1. Où est le goulot d'étranglement?
2. Pouvez-vous changer la couleur des pièces défectueuses?
3. Ajoutez un tableau de bord pour suivre les performances des machines

- **Acquisition des données de performance d'un modèle 3D**

Afin de visualiser les performances des machines, on utilise un Tableau de bord, on ajoute à partir de la librairie les éléments suivants :

- Taux d'occupation des postes en utilisant un diagramme à barre 

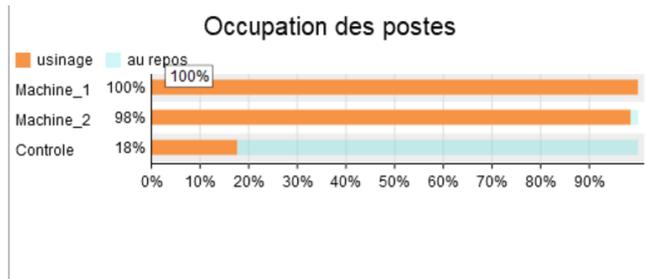


Figure 0.16 Taux d'occupation des postes

Les états d'utilisation des postes en utilisant un diagramme de Gantt 

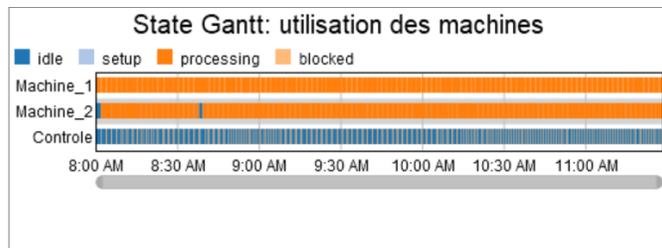


Figure 0.17 Diagramme de Gantt

TP 02 : Introduction au Process Flow

Objectif :

- Créer un Process Flow Général
- Se familiariser avec les ressources mobiles

Description :

Des pièces arrivent à intervalles de 20 s en moyenne (loi exponentielle) dans une zone de réception de 5 emplacements de 2m x 2m.

Un opérateur doit d'abord contrôler la pièce, cette opération dure 10 s

Une fois que cette dernière est contrôlée, l'opérateur peut l'emmener au poste de travail où elle sera traitée.

Modéliser cette zone de réception avec un Process Flow

Disposition :

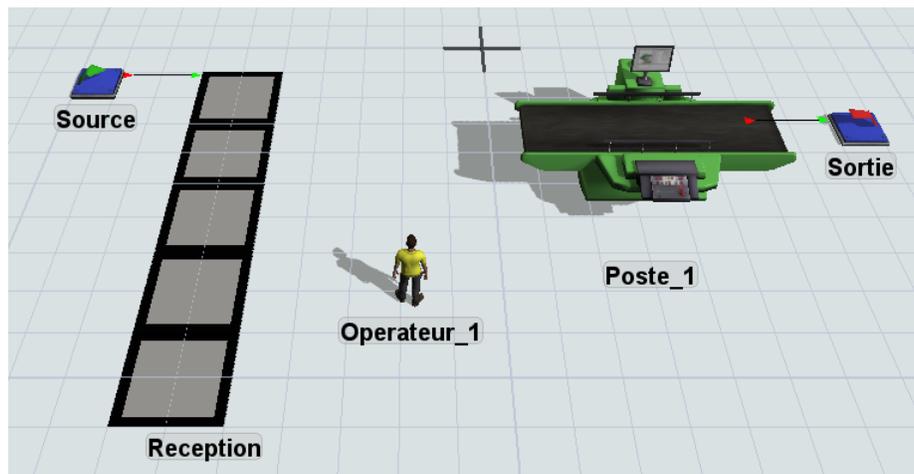


Figure 0.18 Disposition du deuxième modèle

- **Séquence de tâche :**

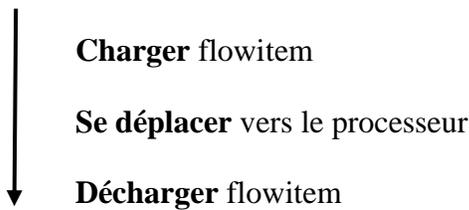
Une tâche (Task) - une seule instruction ou action à exécuter par **une ressource mobile** (TaskExecutor) telle que le chargement, le déchargement, le déplacement, le retard

Séquence de tâches (**Tasksequence**) - une série de tâches à exécuter dans l'ordre indiqué.

Lorsqu'un TaskExecutor reçoit une séquence de tâches, il commence à exécuter les tâches les unes après les autres jusqu'à ce que la séquence de tâches soit terminée ou préemptée par une séquence de tâches de préemption de priorité plus élevée.

Des séquences de tâches personnalisées peuvent également être créées et distribuées à l'aide des commandes du Process Flow.

Exemple : **Se déplacer** au stock



Une ressource mobile est automatiquement utilisée dans le modèle 3D lorsque la case « Utiliser le transport » est cochée sur une ressource fixe.

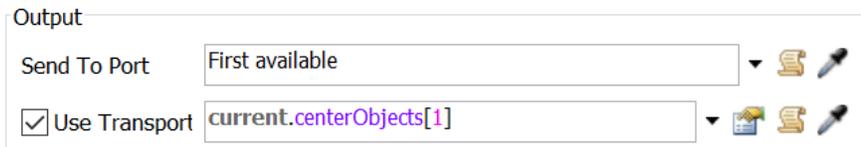


Figure 0.19 utilisation d'une ressource mobile

Pour cela, il est important de créer d'abord une connexion centrale entre la ressource fixe et la ressource mobile



Figure 0.20 établissement d'une connexion centrale

Nous retrouvons différentes ressources mobiles dans la librairie

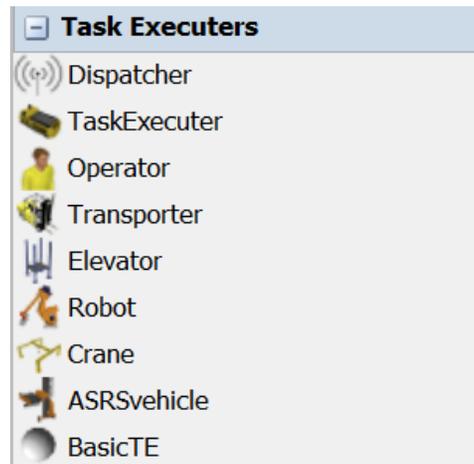


Figure 0.21 les différentes ressources mobiles

- **Opérateur :**

Un opérateur modélisera les tâches humaines telles que la configuration de la machine ou l'assemblage manuel.

L'opérateur peut également être utilisé pour transporter des FlowItems d'une ressource fixe à une autre.

L'opérateur peut se déplacer dans les directions x, y et z.

Comme tous les ressources mobiles, l'opérateur peut charger de nombreux FlowItems à partir de plusieurs emplacements, puis décharger les FlowItems séparément vers plusieurs emplacements.

- **Modélisation :**

On va modéliser la zone de réception par un stock appelé « Rack ».

Après avoir disposé les éléments essentiels sur le modèle 3D, on configure en premier lieu le temps d'arrivée et la taille du stock.

Rappel : Des pièces arrivent à intervalles de 20 s en moyenne (loi exponentielle) dans une zone de réception de 5 emplacements de 2m x 2m.

Cette fenêtre permet de modifier la largeur, la hauteur des cellules de stockage et le nombre de cellules souhaitées.

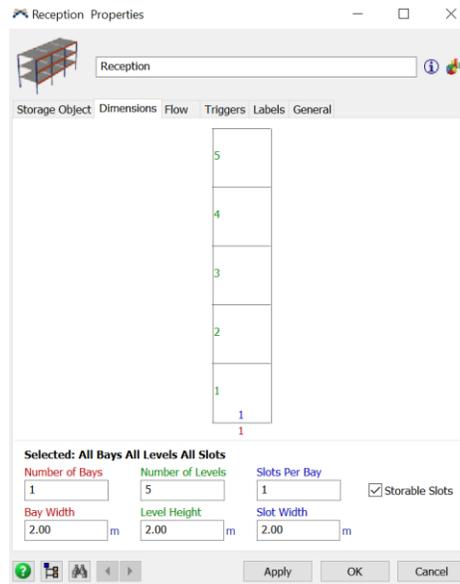


Figure 0.22 Dimensionnement du stock

Il n'est pas nécessaire de relier la zone de réception et le poste de travail avec une connexion, puisque c'est l'opérateur qui transportera les pièces et sera commandé par la logique du Process Flow.

- **Logique du Process Flow :**

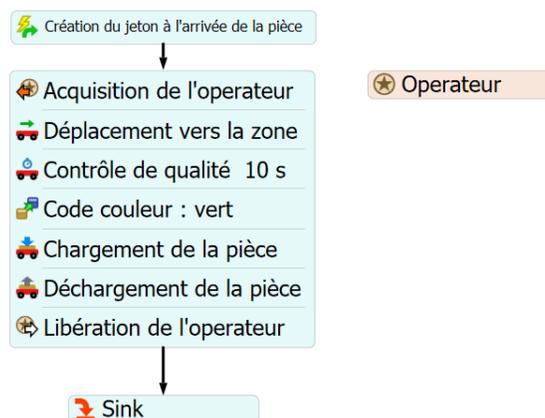


Figure 0.23 Process flow de l'opérateur

1. À l'aide de l'activité  **Event-Triggered Source** un jeton est créé dès l'entrée de la pièce à la zone de réception et un label «pièce» lui est attribué afin de l'utiliser dans le cheminement du Process flow

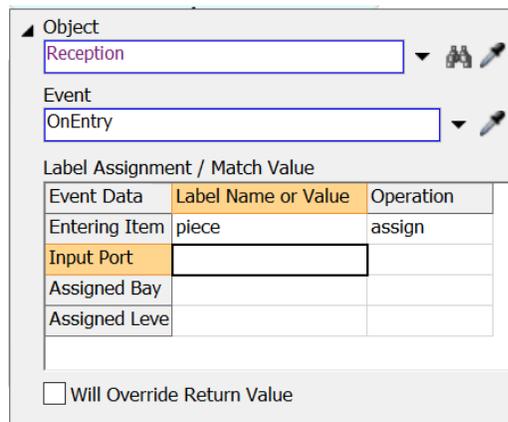


Figure 0.24 création d'un jeton

2. Dans l'activité  **Resource**, on crée la ressource « operateur » et on fait référence à l'opérateur sur le modèle 3D

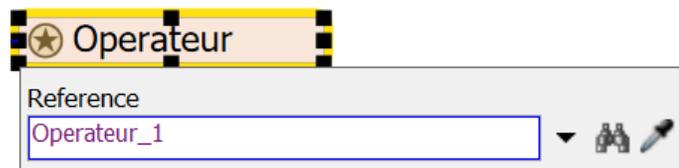


Figure 0.25 affectation d'une ressource

L'opérateur va dès lors exécuter la séquence de tâches afin de se déplacer à la zone de réception, contrôler les pièces, les charger à partir de la zone de réception et les décharger dans le poste de travail.



Figure 0.26 Séquence de tâches de l'operateur

Dans les activités, on fera référence au jeton de l'opérateur dans « Executer » et du jeton de la pièce lorsqu'on parle de cette dernière.

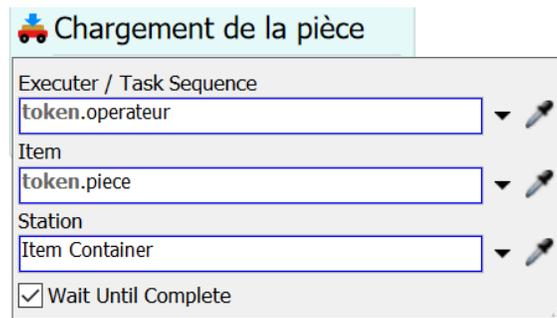


Figure 0.27 configuration de l'activité de chargement de pièce

Pour avoir un bonus visuel, la couleur des pièces est changée en vert après avoir été contrôlées.

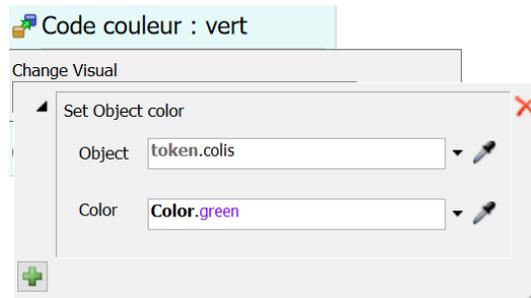


Figure 0.28 changement de la couleur de la pièce

À la fin de la séquence, on libère l'opérateur

RÉSUMÉ

Les systèmes de production sont de plus autonomes et imprévisibles. Dans ce projet, un jumeau numérique basé sur des techniques d'analyse de données et d'IA vise à élaborer une maintenance prédictive et réactive des CPPS est implémenté, en utilisant le logiciel de simulation de flux Flexsim. Afin d'évaluer les performances de l'outil proposé, une étude de cas sur les stations de la chaîne de production flexible de FESTO Didactique MPS500 de l'université de Tlemcen ainsi que la ligne d'assemblage de l'IUT de Nantes est proposée.

Mots-clés: Jumeau numérique, maintenance prédictive, système de production cyber-physique, intelligence artificielle.

ABSTRACT

Production systems are more autonomous and unpredictable. In this project, a digital twin based on data analysis and AI techniques aims to develop predictive and reactive maintenance of CPPS is implemented, using Flexsim, a flow simulation software. In order to prove the efficiency of the proposed tool, a case study on the MPS500 flexible manufacturing system of FESTO at the University of Tlemcen as well as the assembly line of the IUT of Nantes is proposed.

Keywords: Digital Twin, predictive maintenance, cyber-physical production system, artificial intelligence.

تلخيص

الأحداث والاضطرابات الغير متوقعة التي يمكن أن تحدث في أي وقت في البيئات الصناعية الحقيقية تزيد من تعقيد أنظمة الإنتاج. يهدف هذا المشروع الى تطوير الصيانة التنبؤية والتفاعلية عن طريق انجاز توأم رقمي باستعمال تحليل البيانات وتقنيات الذكاء الاصطناعي، باستخدام برنامج محاكاة التدفق فليكسيم. من أجل تقييم أداء الأداة المقترحة، تمت دراسة حالة على محطات الإنتاج لجامعة تلمسان بالإضافة إلى سلسلة التجميع لجامعة نانت.

الكلمات الرئيسية: التوأم الرقمي، الصيانة التنبؤية، نظام الإنتاج المادي السيبراني، الذكاء الاصطناعي

