
Master 2 Sciences Technologie Et Métaverse

Rapport de stage

**Développement d'Outils de Simulation
Ferroviaire pour la Génération et la Détection
de Défauts de Shuntage**

Présenté et soutenu par : **BENCHEKROUN Mohammed**

Date de soutenance : **Le 12/09/2025**

Entreprise : Institut de Recherche Technologique RAILENiUM

Tuteurs :

Entreprise : Issam MALLOUK

Université : Meriem KHADHAR

Mars 2025 – Aout 2025

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué, de près ou de loin, au bon déroulement de mon stage et à l'enrichissement de mon parcours académique et professionnel.

Je remercie chaleureusement **l'équipe de Railenium**, et plus particulièrement **M. Issam Mallouk**, mon tuteur en entreprise, pour son accompagnement attentif, ses conseils techniques précieux, ainsi que sa disponibilité tout au long de ce stage. Je remercie également **M. Smail Ziani**, manager de l'équipe, pour sa confiance, son encadrement et les échanges enrichissants que nous avons pu avoir.

Je souhaite également exprimer ma reconnaissance à **l'INSA Hauts-de-France**, et tout particulièrement à **M. Mestiri Makram**, responsable de la formation, pour la qualité de l'enseignement dispensé et le cadre qu'il offre aux étudiants pour évoluer dans des projets concrets. Mes remerciements vont également à **Mme Meriem Khadhar**, ma tutrice universitaire, pour son suivi régulier, ses conseils avisés et son engagement à m'accompagner tout au long de cette expérience.

Je n'oublie pas **mes collègues de stage** et l'ensemble des membres du laboratoire, pour leur esprit d'équipe, leur soutien, ainsi que les moments de partage et d'entraide qui ont rythmé mon quotidien.

Enfin, j'adresse mes **remerciements les plus sincères à mes parents**, pour leur soutien indéfectible, leur patience, leur confiance et leur présence constante. Leur appui a été essentiel tout au long de ce parcours, tant sur le plan personnel que professionnel.

Table des matières

| | |
|--|-----------|
| Résumé | 6 |
| 1. Introduction | 8 |
| 1.1 Contexte du stage..... | 9 |
| 1.2 Présentation de l'institut de recherche..... | 10 |
| 1.2.1 Activités | 10 |
| 1.2.2 Organisation structurelle | 11 |
| 2. Projet TELLi | 13 |
| 2.1 Description du projet TELLi..... | 13 |
| 2.2 Objectifs du stage..... | 14 |
| 2.2.1 Shuntage et circuit de voie | 15 |
| 2.2.2 Problématique de Déshuntage | 16 |
| 2.2.3 Cause de Déshuntage..... | 17 |
| 2.2.4 Risque ferroviaires..... | 18 |
| 3. Spécifications fonctionnelles et techniques de l'application | 21 |
| 3.1 Besoins exprimés..... | 21 |
| 3.2 Fonctionnalités principales de l'application..... | 22 |
| 3.3 Optimisation et gestion des scénarios..... | 23 |
| 3.4 Architecture technique et composants développés | 24 |
| 3.4.1 Extraction des données initiales..... | 24 |
| 3.4.2 Intégration dans la base MySQL | 25 |
| 3.4.3 Déploiement d'une API pour l'exploitation des données..... | 25 |
| 3.4.4 Mise en œuvre du modèle multi-agents sous AnyLogic | 27 |
| 3.4.5 Passerelle de communication entre Python et MATLAB/Simulink..... | 28 |
| 3.4.6 Mise en œuvre d'une Co-simulation AnyLogic – MATLAB/Simulink..... | 29 |
| 4. Tests et Validation | 32 |
| 4.1 Préparation des données..... | 32 |
| 4.2 Définition et mise en place d'un scénario opérationnel | 33 |
| 4.3 Simulation MATLAB/Simulink..... | 34 |
| 4.4 Exploitation et transfert des résultats | 35 |
| 4.5 Analyse comparative et intégration des résultats | 36 |
| 4.6 Validation et perspectives..... | 37 |
| 5. Conclusion et Perspectives | 38 |

Liste des tableaux

| | |
|---|----|
| Tableau 1- Optimisation et gestion des scénarios | 23 |
| Tableau 2- Fonctionnalités de l'API | 26 |
| Tableau 3- Composants de l'interface dans AnyLogic..... | 27 |
| Tableau 4- données 'indice et valeur' envoyés à Matlab/Simulink | 28 |
| Tableau 5- Architecture | 29 |
| Tableau 6- Données 'indice, valeurs' de la ville de test | 34 |

Table des figures

| | |
|---|----|
| Figure 1- Coordination du projet..... | 9 |
| Figure 2- Chiffre-clés | 10 |
| Figure 3- Organigramme de la structure..... | 12 |
| Figure 4- Prototype du Train Léger Innovant (TELLi) | 13 |
| Figure 5- Shuntage d'essieu..... | 15 |
| Figure 6- Schéma d'un circuit de voie..... | 16 |
| Figure 7- Fonctionnement Voie occupé avec ($i_{CdV} = i_{Essieux} + i_{Résiduel}$)..... | 17 |
| Figure 8- Défauthèque de déshuntage..... | 18 |
| Figure 9- Collision Nez à Nez..... | 18 |
| Figure 10- Collision par rattrapage | 19 |
| Figure 11- Collision par prise en écharpe..... | 19 |
| Figure 12- Collision avec un obstacle | 19 |
| Figure 13- Déraillement | 20 |
| Figure 14- Fonctionnalités principales | 23 |
| Figure 15- Base de données fichier CSV | 24 |
| Figure 16- Base de données MySQL | 25 |
| Figure 17- Résultats de 1er API..... | 26 |
| Figure 18- Objectifs de 2ème API..... | 28 |
| Figure 19- Co-simulation AnyLogic/MATLAB | 30 |
| Figure 20- étape pour initialiser la simulation | 31 |
| Figure 21- Base de données incluant la colonne des impédances | 33 |
| Figure 22- Base de données MySQL de la ville de test..... | 33 |
| Figure 23- Résultats après la simulation MATLAB | 35 |
| Figure 24- Résultats transférés à la base de données Influx DB..... | 36 |

Résumé

En Français

Dans le cadre du projet **Telli**, porté par **Railenium**, mon stage s'est inscrit dans une démarche d'optimisation du réseau ferroviaire français via le développement d'un **train léger** destiné à circuler sur des voies peu ou pas exploitées par les TGV ou les TER. Le projet est structuré en plusieurs lots. Mon travail a été centré sur le **lot 3.2**, dédié à l'étude du **phénomène de Shuntage**.

Le Shuntage est un phénomène électrique fondamental pour la **détection des trains sur les voies**. Il repose sur le fait que les essieux des trains relient électriquement les deux rails, fermant ainsi un circuit détecté par les systèmes de signalisation. Cependant, ce processus peut être altéré par des facteurs comme l'**oxydation des rails**, la **présence de feuilles mortes**, ou d'autres **contaminants**, menant à un **déshuntage** : le système ne détecte plus le train, ce qui présente un risque de **collision**.

L'objectif principal de mon stage a été de créer une application web pour **contribuer à la modélisation, la simulation et le diagnostic du phénomène de déshuntage**, en développant une chaîne d'outils techniques permettant l'exploitation et l'analyse de données issues de scénarios simulés.

Ce stage m'a permis de renforcer mes compétences en développement web, notamment **Python** (Flask, API, automatisation), **MySQL** (stockage structuré des données), **AnyLogic** (modélisation multi-agents), **MATLAB** (analyse et simulation physique), **Excel** (données sources).

La compréhension du **shuntage** et la capacité à simuler ses défaillances constituent une contribution précieuse à la **sécurité ferroviaire**, en particulier sur des lignes peu fréquentées, où les conditions de circulation sont plus complexes à modéliser.

Mots clés :

Train léger, Lignes de dessertes fines (LDFT), Lot 3.2, Shuntage, Déshuntage, Détection des trains, Sécurité ferroviaire. Application web, Modélisation, Simulation, Diagnostic, Chaîne d'outils techniques, Exploitation et analyse de données, Scénarios simulés, Développement web, Python (Flask, API, automatisation), MySQL, AnyLogic (modélisation multi-agents), MATLAB (simulation physique), Excel (données sources), Optimisation du réseau ferroviaire.

En Anglais

As part of the Telli project, led by Railenium, my internship was part of an effort to optimize the French rail network through the development of a light rail system designed to run on tracks rarely or not at all used by TGV or TER trains. The project is structured into several work packages. My work focused on work package 3.2, dedicated to the study of the shunting phenomenon.

Shunting is a fundamental electrical phenomenon for train detection on tracks. It relies on the fact that train axles electrically connect the two rails, thus completing a circuit detected by signaling systems. However, this process can be impaired by factors such as rail oxidation, the presence of dead leaves, or other contaminants, leading to de-shunting: the system no longer detects the train, which poses a risk of collision.

The main objective of my internship was to create a web application to contribute to the modeling, simulation, and diagnosis of the shunting phenomenon, by developing a technical toolchain enabling the exploitation and analysis of data from simulated scenarios.

This internship allowed me to strengthen my skills in web development, particularly Python (Flask, API, automation), MySQL (structured data storage), AnyLogic (multi-agent modeling), MATLAB (physical analysis and simulation), and Excel (source data).

Understanding shunting and the ability to simulate its failures constitute a valuable contribution to rail safety, particularly on less-frequented lines, where traffic conditions are more complex to model.

Chapitre 1

1. Introduction

Ce rapport de stage s'inscrit dans le cadre de mon projet de fin d'études réalisé au sein de **RAILENIUM**, dans le domaine de la **simulation et de la sécurité ferroviaire**. Le stage a porté sur le projet **TELLi**, un projet innovant qui vise à concevoir un train léger capable de circuler sur des lignes ferroviaires peu ou non exploitées, avec une attention particulière portée à la **problématique du Shuntage**, un enjeu crucial pour la détection des trains par les circuits de voie.

Ce document est structuré en quatre chapitres principaux, chacun décrivant une étape essentielle du travail accompli durant le stage.

- **Le premier chapitre** est consacré au **contexte général du stage**. Il présente d'une part les objectifs et le cadre dans lequel s'inscrit le stage, et d'autre part l'institut d'accueil, son fonctionnement, ses activités principales, ainsi que son organisation structurelle.
- **Le deuxième chapitre** introduit le **projet TELLi** en fournissant une description globale de ses objectifs et de son architecture. Ce chapitre approfondit également la **problématique technique sur laquelle mon stage s'est concentré**, à savoir le **lot 3.2**, portant sur le phénomène de **déshuntage**. Sont abordés dans cette partie les concepts de Shuntage et de tension résiduelle, les causes de déshuntage, les incidents liés à ce phénomène ainsi que les solutions palliatives et les méthodes d'analyse des risques.
- **Le troisième chapitre** constitue le cœur technique du rapport. Il détaille le **cahier des charges fonctionnel** de l'application web et les **besoins exprimés par le consortium TELLi**. Ensuite, il décrit en profondeur l'**architecture technique**, les **composants développés**, les différentes **API utilisées**, ainsi que l'**intégration des simulations MATLAB/Simulink avec la plateforme AnyLogic**. Enfin, cette section inclut la création d'un **scénario de test** pour optimiser les performances de simulation.
- **Le quatrième chapitre** est dédié à la **validation du système mis en place**. Il présente les différentes **stratégies de test**, les **résultats obtenus**, ainsi qu'une **synthèse des scénarios simulés**, permettant d'évaluer la fiabilité, la performance et la robustesse de l'architecture globale.

Enfin, le rapport se termine par une **conclusion générale** récapitulant les acquis du stage, les contributions apportées au projet, ainsi que les perspectives d'évolution et d'amélioration possibles.

1.1 Contexte du stage

Le stage s'inscrit dans le cadre du **projet TELLi¹ (Train Léger Innovant)**, un projet national ambitieux regroupant **11 partenaires industriels et académiques** sous la coordination de la **SNCF**. Parmi ces partenaires figurent **Thales (GTS France), Texelis, CAF, Wabtec-Faiveley Transport, Railenium, Ferrocampus, Alstom, Capgemini, Cerema et Ektacom**. L'objectif principal du projet est de **redynamiser les lignes ferroviaires à faible trafic**, aussi appelées **lignes de dessertes fines du territoire (LDFT)**, en développant un système de transport **léger, modulaire et frugal**.

Dans ce contexte, le projet TELLi se structure en plusieurs lots techniques. Mon stage a porté plus spécifiquement sur le **lot 3**, intitulé « **Systèmes d'exploitation** », et plus précisément sur la **tâche de Shuntage**. Cette tâche vise à explorer des **solutions innovantes pour anticiper les problèmes de déshuntage** et proposer des **solutions techniques fiables pour garantir un Shuntage efficace**, adapté aux particularités des LDFT.

Pour structurer les travaux de recherche et développement, le projet TELLi est divisé en **plusieurs lots thématiques**, chacun regroupant des tâches spécifiques. Le **Schéma 1** ci-dessous illustre l'organisation générale des lots et des tâches correspondantes.

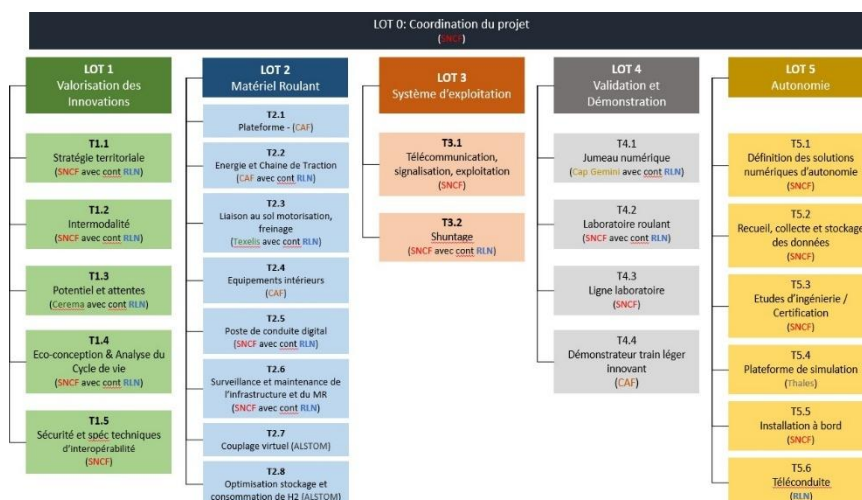


Figure 1- Coordination du projet

Mon stage est directement rattaché au **lot 3 – Systèmes d'exploitation**, et plus précisément à la **tâche T3.2 dédiée au Shuntage**. Cette tâche a pour but d'**anticiper les risques de déshuntage**, un phénomène critique pouvant compromettre la **détection fiable des trains** sur les circuits de voie, et de **développer des solutions techniques innovantes** adaptées aux particularités des LDFT.

Le **Shuntage²** est un mécanisme fondamental de la **sécurité ferroviaire**. Il permet de détecter la présence d'un train sur une portion de voie à l'aide des **circuits de voie**, en assurant la fermeture d'un circuit électrique par les essieux métalliques du train. Lorsqu'un train ne parvient pas à assurer ce contact électrique on parle alors de **déshuntage** cela peut provoquer de **graves incidents**, notamment en cas de mauvaise détection de sa position. La maîtrise du Shuntage est donc essentielle pour éviter des erreurs de signalisation, des collisions ou des défaillances de gestion du trafic.

¹ Livrable présenté par l'équipe d'énergie

² Wikipédia

Mon travail de stage s'inscrit dans cette problématique de **sécurité critique**, avec pour mission de participer au développement d'un **outil d'analyse et de simulation** permettant d'**étudier les risques liés au déshuntage**, de **modéliser des scénarios ferroviaires**, et de **tester l'efficacité des solutions proposées**.

Ce contexte exige une **approche pluridisciplinaire**, mêlant des **compétences en informatique, en simulation (AnyLogic, MATLAB/Simulink), en traitement de données**, ainsi qu'une **compréhension approfondie des systèmes ferroviaires**. Il s'agit également de proposer une interface technique permettant de faciliter la visualisation, l'interprétation et la validation des résultats issus des différentes simulations.

1.2 Présentation de l'institut de recherche

1.2.1 Activités

L'Institut de Recherche Technologique (IRT) Railenium³ conduit des projets collectifs de recherche et d'innovation réunissant le milieu universitaire, l'industrie et les autorités publiques. Ces initiatives ont pour objectif de concevoir, valider et tester des technologies de pointe ainsi que des solutions innovantes face aux enjeux sociétaux et technologiques majeurs de l'industrie ferroviaire, aussi bien en France qu'à l'international.

Fondé en 2012 dans le cadre du Plan d'Investissements d'Avenir, Railenium est un Institut de Recherche Technologique (IRT) spécialisé dans les systèmes ferroviaires. Grâce à l'expertise de ses équipes, Railenium offre des compétences précieuses pour franchir les obstacles technologiques et scientifiques. Établi dans les Hauts-de-France et en Île-de-France, Railenium a été crucial dans le financement, l'ingénierie et la gestion de projets collaboratifs, participant à plus de 80 projets avec plus de cent partenaires.



Figure 2- Chiffre-clés

³ Documents internes

1.2.2 Organisation structurelle

Railenium est piloté par une **Direction Générale**, à laquelle sont directement rattachés plusieurs responsables transverses : Communication, QHSE (Qualité, Hygiène, Sécurité, Environnement), et Système d'Information.

L'institut est structuré en **cinq grandes directions** :

- **Direction Scientifique** : supervise les projets de recherche fondamentale et appliquée, en lien avec les universités et centres de recherche partenaires.
- **Direction Développement** : dédiée à l'innovation technologique et à la création de nouveaux produits/services dans le domaine ferroviaire.
- **Direction Ressources Humaines** : en charge du recrutement, de la formation et de la gestion des carrières.
- **Direction Finance et Juridique** : gère les aspects financiers et contractuels des projets.
- **Direction Projets et Expertises (DPE)** : assurant la gestion opérationnelle des projets de recherche et le pilotage de l'expertise technique, sous la direction de **M. Arnaud DELARUE**.

La DPE est elle-même subdivisée en **trois domaines d'expertise** :

1. **Intelligence artificielle, science des données et optimisation ;**
2. **Comportement humain et sécurité des systèmes ;**
3. **Prédiction et gestion de l'énergie et de la durée de vie des actifs énergétiques.**

C'est dans ce **troisième domaine d'expertise**, orienté vers la **modélisation énergétique et la durabilité des systèmes ferroviaires**, que j'ai été intégré en tant que stagiaire. Ce domaine est encadré par un **Responsable de Domaine** et un **Référent d'Expertise**, et j'ai été supervisé plus spécifiquement par **Issam Mallouk**, expert du domaine.

Mon travail a ainsi contribué à l'analyse, la simulation et la compréhension des phénomènes liés au **Shuntage et au déshuntage**, éléments cruciaux pour la sécurité de détection des trains sur les lignes à faible trafic.

L'organigramme de la structure est illustré ci-dessous .

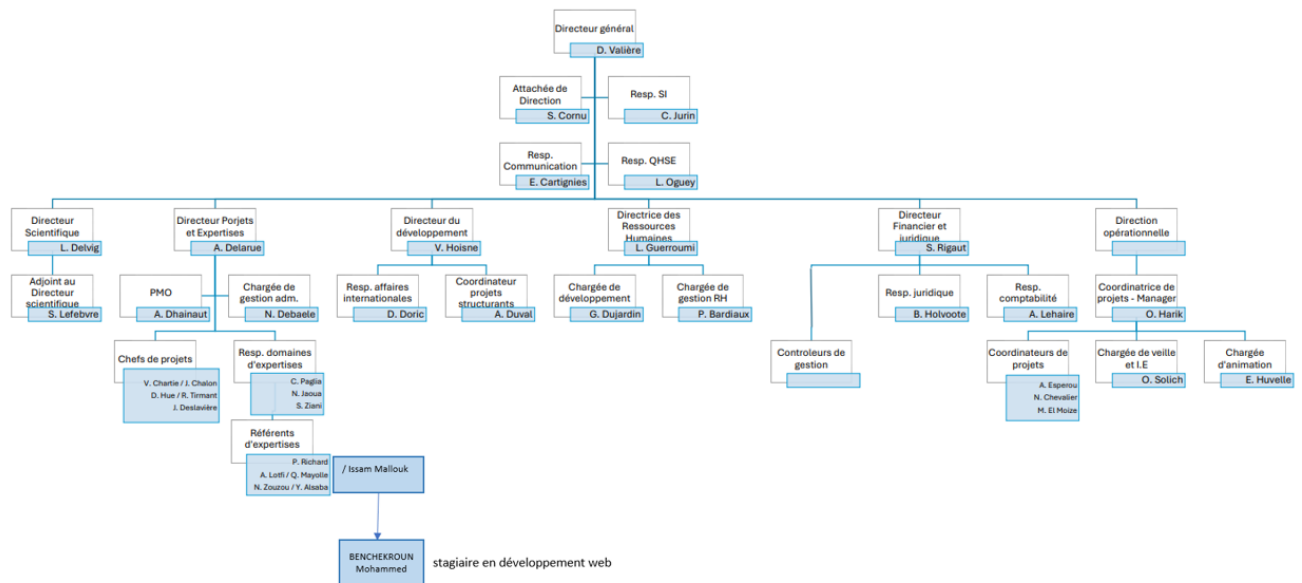


Figure 3- Organigramme de la structure

Chapitre 2

2. Projet TELLi

2.1 Description du projet TELLi

Le projet TELLi⁴ est une initiative innovante visant à revitaliser les lignes de desserte fine du territoire français, qui représentent un tiers du réseau ferré national. Conçu pour remplacer les trains diesel par des trains hybrides électriques, TELLi utilise des sources d'énergie telles que les batteries et l'hydrogène, offrant une solution de transport plus écologique et économique. Le projet se distingue par sa capacité à adapter le matériel roulant aux besoins spécifiques des régions, avec différentes versions disponibles selon la source d'énergie, incluant des configurations à 1500 V en courant continu, à 25 kV en courant alternatif, ainsi que des versions hydrogène et thermique. Alimenté par des batteries et construit avec des matériaux recyclables, TELLi est conçu pour être léger, minimisant ainsi la maintenance des voies tout en offrant une modularité d'aménagement intérieur pour diverses utilisations. Le **prototype du Train Léger Innovant (TELLi)** illustré dans la figure 2 démontre bien cette flexibilité et l'engagement du projet dans une mobilité durable. Actuellement, le projet progresse avec le soutien de la Région Nouvelle-Aquitaine et de partenaires comme le Cerema, avec un objectif de commercialisation prévu pour 2029 après une série de démonstrations et de tests sur une ligne laboratoire.



Figure 4- Prototype du Train Léger Innovant (TELLi)

⁴ Livrable sur TLI par Asmaa Barbata, à Valenciennes

2.2 Objectifs du stage

L'un des enjeux majeurs de ce stage résidait dans l'impossibilité de réaliser directement des campagnes de mesures en conditions réelles sur le terrain ferroviaire, en raison des contraintes de sécurité et d'accessibilité. Il a donc fallu concevoir une approche alternative reposant sur la **simulation numérique**.

Pour cela, les données exploitées sont constituées de **tensions résiduelles**, qui sont ensuite transformées en valeurs d'**impédance** à l'aide d'un modèle d'intelligence artificielle préalablement développé par l'équipe. Cette chaîne de traitement permet d'obtenir, au terme de la simulation, des indicateurs clés tels que les tensions, les courants et leurs valeurs efficaces sur le matériel roulant.

Ces résultats constituent une base essentielle pour l'**analyse du déshuntage**, en offrant une vision fiable et reproductible du phénomène sans nécessiter d'expérimentation physique risquée ou coûteuse.

Dans le cadre du **projet TELLi (Train Léger Innovant)**, mon stage s'inscrit au sein du **lot 3** dédié aux **systèmes d'exploitation ferroviaires**, et plus précisément dans la **tâche 3.2 portant sur le Shuntage**.

Le **Shuntage** est un mécanisme fondamental dans les systèmes de détection de présence des trains, permettant d'assurer la **sécurité du trafic ferroviaire**. Il repose sur le principe de court-circuiter électriquement un circuit de voie lorsqu'un essieu de train est présent, signalant ainsi la présence du train sur un canton donné. Toutefois, certaines configurations ou conditions d'exploitation, notamment sur les **Lignes de Desserte Fine du Territoire (LDFT)**, peuvent provoquer des **phénomènes de déshuntage**, c'est-à-dire l'incapacité d'un train à être détecté correctement.

L'**objectif principal de mon stage** est de contribuer à la **modélisation et à la simulation du phénomène de Shuntage**, afin d'étudier les conditions dans lesquelles le déshuntage peut se produire, d'identifier les paramètres critiques influençant ce phénomène, et de proposer des outils d'analyse permettant de mieux le prévenir.

Les missions spécifiques sont les suivantes :

- **Contribuer à la modélisation un système de Shuntage/Déshuntage** adapté aux lignes à faible trafic, en tenant compte des spécificités techniques et opérationnelles du projet TELLi ;
- **Créer un pipeline de simulation**, intégrant des données réelles (issues de fichiers Excel) et simulées, pour visualiser les signaux résiduels et leur comportement sur différentes portions de voie ;
- **Développer une interface de visualisation et de traitement** des résultats de simulation à l'aide de plusieurs outils interconnectés : **MATLAB/Simulink, AnyLogic, MySQL et API Flask** ;
- **Analyser les données collectées et simulées**, afin de repérer les cas critiques de déshuntage et d'alimenter une base de connaissances pour l'implémentation de solutions palliatives ;
- Participer à la **documentation** et à la **formalisation des travaux**, notamment par la rédaction de rapports techniques, de schémas de flux et de fiches de tests.

Ce travail s'inscrit dans une perspective plus large visant à **renforcer la fiabilité des systèmes de détection sur les futures infrastructures ferroviaires légères**, en proposant une approche de simulation multi-outils souple, évolutive, et orientée vers les cas réels.

2.2.1 Shuntage et circuit de voie

2.2.1.1 Définition de Shuntage

Par définition, le Shuntage⁵ est la mise en place d'un shunt (court-circuit) sur un circuit électrique. En vocabulaire ferroviaire, le Shuntage d'un circuit de voie est réalisé par la présence d'un essieu entre les deux rails. Il est possible ainsi de détecter la présence d'un véhicule sur un canton, à condition que l'essieu soit parfaitement conducteur et que le contact roue-rail soit effectif. Dans la norme NF EN 50617 du 01 Octobre 2015, le shunt d'un circuit de voie par un train est défini comme l'impédance totale du train, de la voie et de leurs interfaces, y compris l'impédance du contact entre la roue et le rail, l'impédance résultante des essieux montés d'un véhicule, les paramètres de charge par essieu, la qualité de la surface de roues, la qualité de la surface des rails, la nature de la traction (électrique ou autre), la nature des freins (la présence des freins à sabots affecte l'impédance résultante).

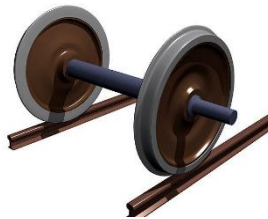


Figure 5- Shuntage d'essieu

2.2.1.2 Définition du circuit de voie

Un **CdV**⁶ est défini comme étant un circuit électrique, empruntant les rails d'une voie ferrée, utilisé pour détecter la présence d'un train dans la section de la voie considérée (qui constitue un canton) et qui commande automatiquement les signaux de protection placés aux bords de la voie pour indiquer aux agents de conduite les autorisations ou interdictions de circulation sur le canton concerné

Permet également de :

- Donner une indication de présence ou d'absence d'un train au TCO (Tableau de Contrôle Optique) du poste d'aiguillage ;
- Activer la fermeture automatique (FA) des signaux (Cantonnement) ;
- Empêcher l'ouverture des signaux (Protections aux bifurcations) ;
- Empêcher les manœuvres des aiguilles sur le parcours tracé ;
- Annoncer des passages à niveau ;
- Détecter une rupture de rail.

La plupart des fonctions de signalisation nécessitent de connaître l'état d'occupation des portions de la voie appelées « zones ». La détection d'une circulation sur une zone peut s'effectuer au moyen d'un « CdV » équipant cette zone.

⁵ Livrable partagé par Issam Mallouk

⁶ Article partagé par Issam Mallouk

Le CdV est constitué de :

- Un émetteur branché à l'une des extrémités de la zone, il délivre un courant qui peut être, selon les types de CdV, continu, alternatif sinusoïdal, alternatif modulé ou impulsionnel ;
- Une ligne de transmission, constituée par les rails ;
- Un récepteur branché à l'autre extrémité de la zone, Il assure le filtrage, l'amplification et la transformation du signal reçu via les rails, ce qui agit sur un relais appelé relais de voie. Les contacts de ce relais sont utilisés pour établir ou couper les circuits électriques des signaux d'entrée des cantons.

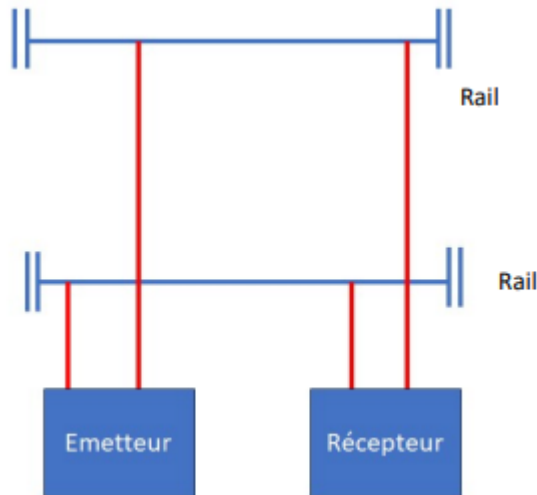


Figure 6- Schéma d'un circuit de voie

2.2.2 Problématique de Déshuntage

2.2.2.1 Définition du déshuntage

D'après la SAM S 004, un « **déshuntage** »⁷ est la non-occupation, l'occupation tardive ou la libération intempestive d'une zone de circuit de voie malgré la présence d'au moins un essieu dans la zone. Il est une cause potentielle de perturbations des installations de signalisation et a pour origine un mauvais contact roue-rail. Il s'agit d'un incident techniquement contraire à la sécurité. L'évènement redouté appelé « déshuntage » survient aussi lorsque la tension résiduelle au récepteur du CdV est supérieure au seuil d'attraction du circuit de voie et permet la non-désexcitation ou la ré-excitation du relais de voie.

2.2.2.2 Définition de la tension résiduelle

La « **tension résiduelle** »⁸ est la tension mesurée à l'entrée du récepteur du CdV alors que la voie est occupée ($U_{\text{résiduelle}} = Z_{\text{récepteur}} \times I_{\text{résiduel}}$). Si le contact roue-rail est optimal, la tension résiduelle tend vers zéro. Lorsque le contact se dégrade, le courant résiduel et donc proportionnellement la tension résiduelle augmentent. Si cette dernière dépasse le seuil de détection du CdV pendant une durée supérieure à la temporisation du récepteur, il y a alors attraction du relais de voie malgré la présence d'une circulation sur la zone isolée, on dit qu'il y a « déshuntage ».

⁷ Livrable rédigé par Asmaa Barbata

⁸ Livrable rédigé par Asmaa Barbata

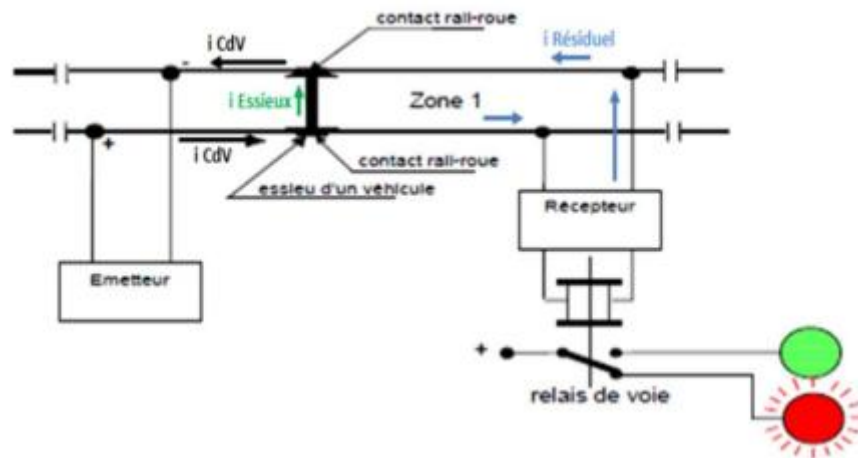


Figure 7- Fonctionnement Voie occupée avec ($i_{CdV} = i_{Essieux} + i_{Résiduel}$)

2.2.3 Cause de Déshuntage

La cause principale du "**déshuntage**" réside dans la détérioration de la qualité du contact électrique entre la roue et le rail. Cette dégradation est liée à la mauvaise qualité de la surface de la roue et du rail, ainsi qu'à la présence de polluants au niveau de ce contact.

Ces auteurs ont défini une période d'observation allant du 1er janvier 1992 au 31 décembre 2017, soit une durée de 26 ans, nous pouvons citer les causes répertoriées ci-dessous :

- Conditions d'environnement (feuilles, pluie, oxydation, pollution autre) ;
- Pannes diverses (graisseurs de boudins, Boucle Inductive d'Aide au Shuntage, ...) ;
- L'huile ;
- Engins ;
- Nombre de circulation ;
- Erreurs opérateurs ;
- Types de circuits de voie ;
- Sablage : excès de débit de sable amenant potentiellement à une perturbation des circuits de voie ;
- Une oxydation du champignon (rouille excessive) ;
- Un dépôt noirâtre dans les rails à gorge sur les zones de PN ;
- Un amalgame par mélange d'oxyde de fer, de feuilles, de silice, de pluie, etc....

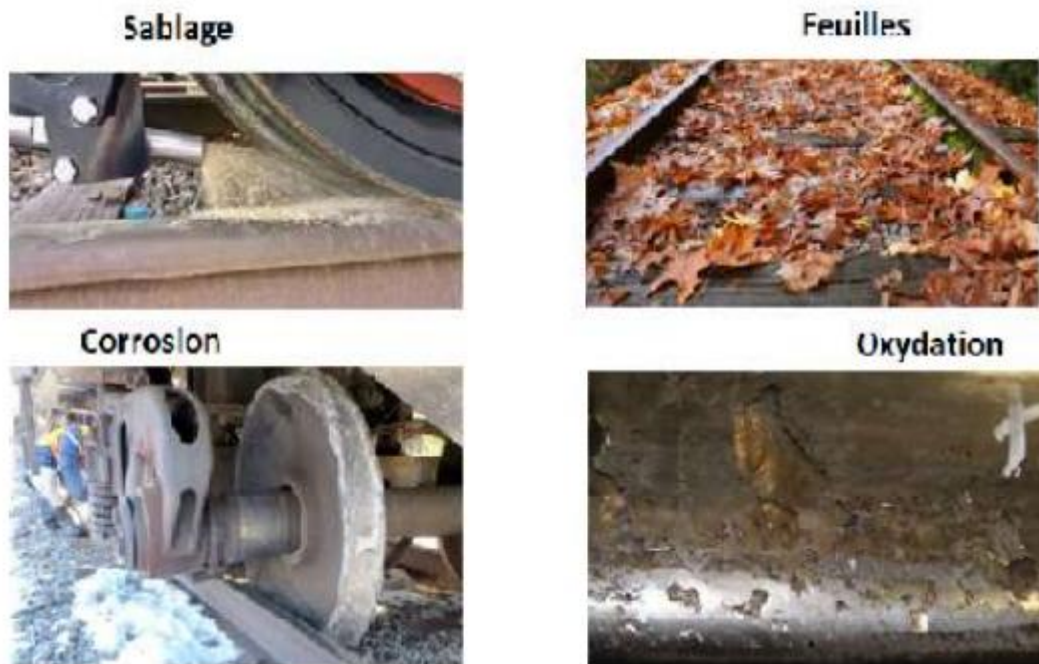


Figure 8- Défauthèque de déshuntage

2.2.4 Risque ferroviaires

Nous retrouvons 5 risques ferroviaires⁹ que nous proposons de présenter dans ce qui suit.

- Collision nez à nez : C'est une collision frontale entre deux trains



Figure 9- Collision Nez à Nez

⁹ Rapport sur les risques ferroviaires

- Collision par rattrapage : C'est une collision par l'arrière lorsqu'un train percute un autre train qui se trouve devant lui



Figure 10- Collision par rattrapage

- Prise en écharpe : C'est une collision latérale entre deux trains qui se produit à une intersection ou à une jonction de voies

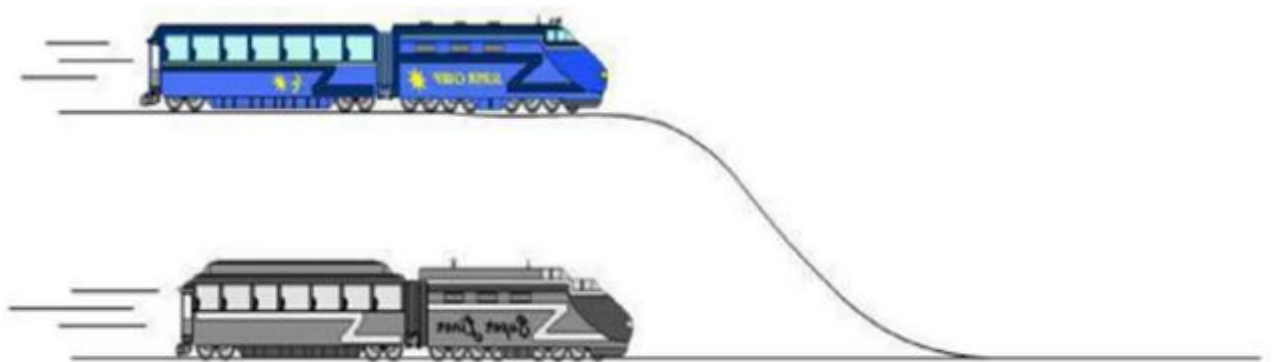


Figure 11- Collision par prise en écharpe

- Collision avec un obstacle : C'est le risque de rencontre avec un obstacle non ferroviaire (rocher, animal, véhicule...). Nous pouvons citer l'exemple du risque de collision sur un passage à niveau entre un train et une voiture. Ce risque est pris en charge par des dispositifs d'annonce des trains.



Figure 12- Collision avec un obstacle

- **Déraillement** : c'est un incident ou un accident dans lequel un véhicule ferroviaire sort des rails, totalement ou partiellement. L'origine peut être diverse (l'excès de vitesse, la manœuvre d'un appareil de voie au passage du train, une panne du système d'aiguillage, effets dynamiques de la voie et/ou du matériel roulant importants, défauts de la voie/rails et roues).



Figure 13- Déraillement

Chapitre 3

3. Spécifications fonctionnelles et techniques de l'application

3.1 Besoins exprimés

Dans le cadre du **lot 3.2 du projet Telli**, dédié à l'étude du phénomène de Shuntage ferroviaire, un besoin s'est clairement dégagé : celui de disposer d'un **outil informatique centralisé**, capable de **collecter, structurer, exploiter et simuler les données du projet**. Les fichiers d'origine étant multiples, hétérogènes et répartis sur plusieurs classeurs Excel, leur manipulation manuelle représentait un risque d'erreurs et un frein important à l'analyse rapide.

L'objectif principal est donc de **développer une application web et un ensemble de services techniques associés** permettant de :

- **Extraire automatiquement** les données pertinentes depuis les fichiers fournis par les équipes terrain ;
- **Centraliser ces données** dans une base relationnelle (MySQL) afin de les rendre exploitables via des requêtes structurées ;
- **Mettre à disposition Deux APIs web** pour faciliter l'accès aux données par d'autres outils, notamment AnyLogic et MATLAB/Simulink ;
- **Permettre le lancement dynamique de scénarios de simulation**, en filtrant les données selon certains critères (ville, date.) ;
- **Automatiser le calcul de certaines grandeurs physiques**, notamment l'impédance, à l'aide d'un modèle d'intelligence artificielle entraîné dans MATLAB.

Ce système vise à **faciliter l'analyse du déshuntage** en simulant différents cas issus de données réelles, tout en permettant une communication fluide entre les différents composants du projet (base de données, API, outils de simulation). Il s'agit également de rendre la solution **évolutive**, adaptable à de nouveaux jeux de données ou à d'autres types de scénarios dans le futur.

3.2 Fonctionnalités principales de l'application

L'application développée dans le cadre de ce projet repose sur une architecture modulaire conçue pour faciliter l'exploitation des données ferroviaires dans un contexte de simulation et de diagnostic. Elle intègre plusieurs **fonctionnalités principales**, répondant aux besoins identifiés dans l'étude du phénomène de Shuntage :

- **Extraction automatisée des données** : un script Python a été développé pour parcourir les fichiers Excel d'origine, extraire les données pertinentes et les structurer dans un format exploitable (CSV), permettant de gagner du temps et d'éviter les erreurs manuelles.
- **Stockage centralisé dans une base MySQL** : les données extraites sont importées dans une base de données relationnelle, offrant une structure cohérente et permettant l'exécution de requêtes SQL ciblées.
- **Accès aux données via une API Flask** : cette première API permet d'interroger la base de données selon des filtres spécifiques (ville, date, etc.) et de récupérer les données au format JSON. Elle facilite ainsi la communication avec des outils de simulation comme **AnyLogic**.
- **Interface de simulation dans AnyLogic** : les données récupérées via l'API Flask sont intégrées dans une interface utilisateur interactive permettant de sélectionner dynamiquement la ville et la date, et de lancer la simulation du scénario correspondant.
- **Transmission vers MATLAB via une seconde API** : une deuxième API Python (`api_matlab.py`) a été conçue pour recevoir des entrées physiques (tension, indice de temps) issues des simulations AnyLogic et les transmettre à MATLAB/Simulink. L'API retourne alors une **prédiction de l'impédance** calculée par un modèle d'intelligence artificielle entraîné.
- **Simulation complète du processus de Shuntage** : l'ensemble des composants permet de simuler le phénomène électrique dans MATLAB/Simulink, de valider les données physiques et de vérifier les conditions de Shuntage ou de Déshuntage.
- **Création d'un environnement de test** : pour améliorer les performances et limiter les temps de calcul, une base de test a été générée à partir d'un sous-ensemble des données réelles (d'une ville X), permettant des simulations rapides et représentatives.

Grâce à cette combinaison de fonctionnalités, l'application permet une **analyse complète du phénomène de Shuntage**, de l'extraction des données brutes jusqu'à la simulation multi-outils, en passant par le diagnostic automatisé via un modèle IA. Elle constitue ainsi un outil robuste et évolutif pour l'étude de scénarios ferroviaires critiques.

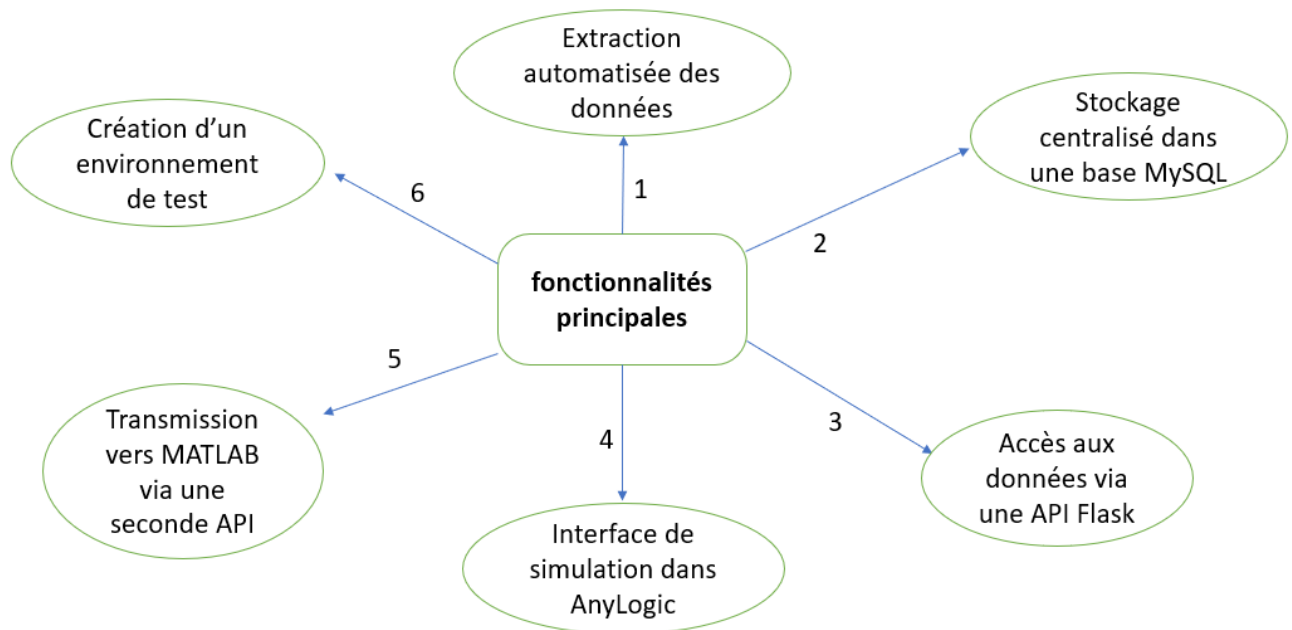


Figure 14- Fonctionnalités principales

3.3 Optimisation et gestion des scénarios

| Objectifs | Actions réalisées | Bénéfices |
|---|---|--|
| Réduire le temps de simulation | Création d'une base de test ville_test avec 10 premières lignes de d'une ville qui existe | Simulation plus rapide pour les phases de test et de validation |
| Faciliter l'expérimentation de scénarios ciblés | Adaptation de l'API Flask pour interroger cette base test spécifique | Contrôle précis des données utilisées pour chaque simulation |
| Automatiser les tests de bout en bout | Lancement simultané des deux APIs (Flask et api_matlab.py) | Validation fluide du fonctionnement global de la chaîne technique |
| Garantir la cohérence des résultats | Simulation sous MATLAB/Simulink avec fonction analyser_donnees | Vérification des grandeurs physiques : courant, tension, valeurs efficaces |
| Permettre une évolution des scénarios | Organisation des données sous forme structurée dans MySQL | Possibilité de créer facilement de nouveaux cas d'étude |
| Limiter les erreurs de manipulation | Création d'une interface utilisateur dans AnyLogic avec filtres simples | Sélection intuitive de la ville et de la date sans modifier le code |

Tableau 1- Optimisation et gestion des scénarios

3.4 Architecture technique et composants développés

L'architecture mise en place repose sur une série de composants interdépendants permettant de passer d'une source de données brute à un processus de simulation automatisé et interactif. Chaque étape de la chaîne a été développée pour assurer la fluidité du traitement, la fiabilité des échanges entre les outils et la scalabilité de la solution. La présente section décrit en détail les composants réalisés, selon une logique modulaire.

3.4.1 Extraction des données initiales

La première étape du travail a porté sur la **consolidation des données d'essais ferroviaires** issues du répertoire SSTR-SRTT UM71 2013. Ces données, dispersées dans plusieurs sous-dossiers au format Excel, regroupaient notamment des mesures de tensions résiduelles, des dates d'enregistrement et des identifiants de circuits de voie.

Afin d'en garantir la fiabilité et l'exploitabilité, un processus d'**automatisation du traitement** a été mis en place. Celui-ci a permis de collecter l'ensemble des fichiers, d'éliminer les doublons, de corriger les incohérences de format (notamment les dates) et de traiter les valeurs manquantes.

Au terme de cette étape, un **jeu de données consolidé de 66 584 enregistrements normalisés** a été produit et converti en fichier unique. Cette base de données constitue le socle sur lequel reposent toutes les analyses et simulations ultérieures, en assurant une **cohérence et une homogénéité des informations** nécessaires aux traitements scientifiques et aux outils de modélisation.

| 1 | Ville | Date | Valeur | T | CdV | Commentaire |
|-------|------------------|-------------------------|--------|-------|-------|--|
| 2 | Électricité 1 TR | 2013-08-20 09:40:15.780 | 329.55 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 3 | Électricité 1 TR | 2013-08-20 09:40:15.800 | 329.71 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 4 | Électricité 1 TR | 2013-08-20 09:40:15.820 | 329.96 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 5 | Électricité 1 TR | 2013-08-20 09:40:15.840 | 329.83 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 6 | Électricité 1 TR | 2013-08-20 09:40:15.860 | 329.89 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 7 | Électricité 1 TR | 2013-08-20 09:40:15.880 | 329.87 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 8 | Électricité 1 TR | 2013-08-20 09:40:15.900 | 329.94 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 9 | Électricité 1 TR | 2013-08-20 09:40:15.920 | 329.87 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 10 | Électricité 1 TR | 2013-08-20 09:40:15.940 | 329.93 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 11 | Électricité 1 TR | 2013-08-20 09:40:15.960 | 329.97 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 12 | Électricité 1 TR | 2013-08-20 09:40:17.000 | 329.93 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 13 | Électricité 1 TR | 2013-08-20 09:40:17.040 | 329.96 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 14 | Électricité 1 TR | 2013-08-20 09:40:17.080 | 329.99 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 15 | Électricité 1 TR | 2013-08-20 09:40:17.120 | 329.94 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 16 | Électricité 1 TR | 2013-08-20 09:40:18.200 | 329.98 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 17 | Électricité 1 TR | 2013-08-20 09:40:18.400 | 329.98 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 18 | Électricité 1 TR | 2013-08-20 09:40:18.700 | 329.93 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 19 | Électricité 1 TR | 2013-08-20 09:40:18.900 | 329.8 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 20 | Électricité 1 TR | 2013-08-20 09:40:19.200 | 329.76 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 21 | Électricité 1 TR | 2013-08-20 09:40:19.400 | 329.98 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 22 | Électricité 1 TR | 2013-08-20 09:40:19.700 | 329.93 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 23 | Électricité 1 TR | 2013-08-20 09:40:19.900 | 329.97 | 1.000 | | Tension au passage du 2 ÂjA me train (X 73500) |
| 66565 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.40 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66566 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.40 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66567 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.5 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66568 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.50 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66569 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.7 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66570 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.4 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66571 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.4 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66572 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.75 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66573 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.67 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66574 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.76 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66575 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.40 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66576 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.94 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66577 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.75 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66578 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.73 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66579 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.73 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66580 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.73 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66581 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.73 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66582 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.73 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66583 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.40 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |
| 66584 | Nordmanning TR | 2013-03-04 14:11:00.000 | 290.70 | 1.000 | 73000 | Tension au passage du 14 ÂjA me train (B6 27000-37000) |

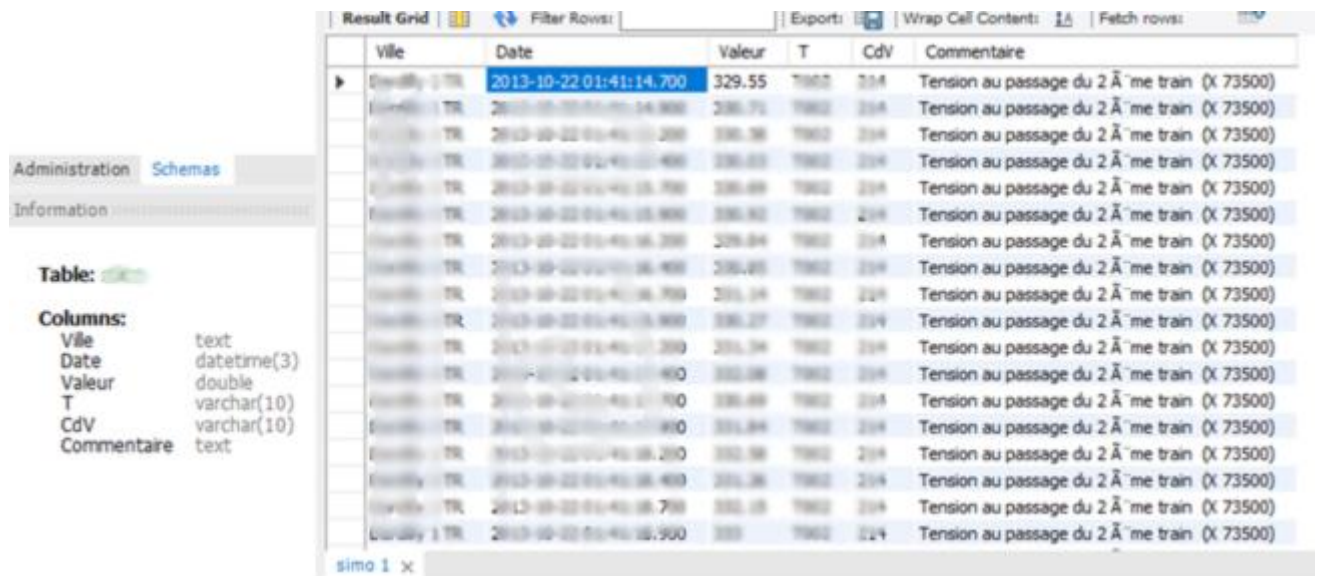
Figure 15- Base de données fichier CSV

3.4.2 Intégration dans la base MySQL

Après la phase de consolidation, les données ont été intégrées dans une **base relationnelle MySQL**, afin de garantir leur accessibilité et d'optimiser leur exploitation par les différents outils du projet.

Cette étape a permis de mettre en place une **infrastructure de stockage centralisée et structurée**, offrant un accès rapide, fiable et cohérent à l'ensemble des mesures collectées. La base a été conçue de manière à refléter la structure des données sources tout en facilitant leur interrogation selon des critères multiples (ex. : ligne, date, tension résiduelle, circuit de voie).

Grâce à cette organisation, les informations sont désormais **uniformisées, sécurisées et interrogeables efficacement**, constituant un socle technique solide pour les développements applicatifs ultérieurs, notamment les API et les modules de simulation.



The screenshot displays a MySQL database interface. On the left, a sidebar shows the database structure with a table named 'Table:'. The columns are listed as follows:

| Columns: | |
|-------------|-------------|
| Ville | text |
| Date | datetime(3) |
| Valeur | double |
| T | varchar(10) |
| CdV | varchar(10) |
| Commentaire | text |

The main window shows a 'Result Grid' with the following columns: Ville, Date, Valeur, T, CdV, and Commentaire. The data is as follows:

| Ville | Date | Valeur | T | CdV | Commentaire |
|---------|-------------------------|--------|------|-----|--|
| Orléans | 2013-10-22 01:41:14.700 | 329.55 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:14.800 | 329.71 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:14.900 | 329.86 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.000 | 329.93 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.100 | 329.99 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.200 | 329.90 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.300 | 329.84 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.400 | 329.85 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.500 | 329.74 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.600 | 329.27 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.700 | 329.34 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.800 | 329.38 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:15.900 | 329.49 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.000 | 329.84 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.100 | 329.86 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.200 | 329.96 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.300 | 329.96 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.400 | 329.96 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.500 | 329.95 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |
| Orléans | 2013-10-22 01:41:16.600 | 329.95 | 7000 | 204 | Tension au passage du 2 ^e train (X 73500) |

Figure 16- Base de données MySQL

3.4.3 Déploiement d'une API pour l'exploitation des données

Pour assurer un accès dynamique et centralisé aux données stockées dans la base MySQL, une **API REST a été développée avec le Framework Python Flask**. Cette API joue un rôle crucial dans la communication entre la base de données et l'application de simulation (AnyLogic), en permettant d'interroger les données de manière flexible via des requêtes HTTP.

AnyLogic¹⁰ est un outil de simulation développé par The AnyLogic Company. AnyLogic possède un langage de modélisation graphique et facilite également l'extension du modèle de simulation avec le code Java.

Objectifs de l'API :

- Permettre l'interrogation des données par **ville** et **date**.
- Fournir une **liste dynamique des villes disponibles** dans la base.
- Garantir une réponse rapide et fiable, avec une gestion claire des erreurs. Fonctionnalités implémentées :

¹⁰ Wikipédia

| Fonction | Description |
|-------------|---|
| /ville | Cette route accepte les paramètres nom (ville) et date (au format YYYY-MM-DD) via la méthode GET. Elle retourne l'ensemble des enregistrements correspondant à la ville et à la date spécifiée, avec un format temporel incluant les millisecondes. |
| /get_cities | Cette route retourne la liste de toutes les villes distinctes présentes dans la base (extraites de la table ville_test), ce qui permet d'alimenter dynamiquement une liste déroulante dans l'interface utilisateur |

Tableau 2- Fonctionnalités de l'API

Aspects techniques :

- La connexion à MySQL est établie via le module mysql.connector.
- L'API est lancée en local sur le port **5021**.
- Les réponses sont retournées au format **JSON**, assurant une bonne interopérabilité avec les autres composants.
- Un **formatage personnalisé des dates** a été mis en place pour conserver les millisecondes dans les horodatages, élément essentiel pour la précision de la simulation.
- Des messages d'erreurs explicites sont retournés en cas de mauvaise requête, d'erreur de connexion ou d'absence de résultats.

Cette API a été **testée manuellement** via des appels directs en navigateur ou avec Postman, mais elle a surtout été utilisée dans le cadre de l'intégration dans AnyLogic (voir section suivante), où elle permet de **filtrer les données à la volée** en fonction des choix de l'utilisateur (ville et date).

Résultats : on prend par exemple **ville : X** et **date : Y** : on récupère toutes les données de cette ville X et la date Y en format JSON

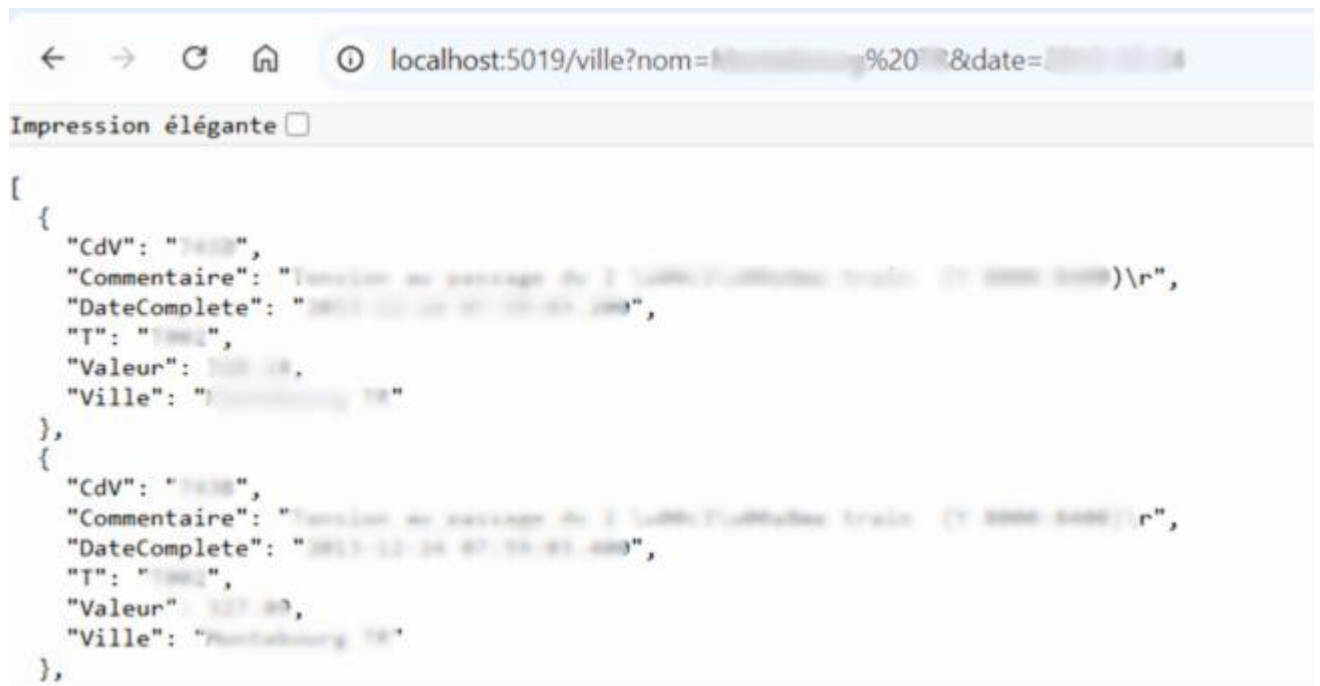


Figure 17- Résultats de 1er API

3.4.4 Mise en œuvre du modèle multi-agents sous AnyLogic

L'intégration de l'API Flask dans **AnyLogic** a permis d'établir une interaction dynamique entre les données de la base et la simulation du phénomène de déshuntage. L'utilisateur peut sélectionner une **ville** et une **date**, récupérer les données via une API, puis lancer une simulation contextualisée.

Objectifs de l'intégration :

- Permettre à l'utilisateur de **sélectionner une ville et une date** via une interface graphique.
- **Interroger dynamiquement l'API Flask** pour récupérer les données spécifiques.
- Prétraiter les données en calculant les indices de temps.
- Intégrer les résultats dans la simulation AnyLogic.

Composants de l'interface utilisateur dans AnyLogic :

| Élément UI | Description |
|--------------|---|
| CityListBox | ComboBox contenant la liste des villes disponibles (récupérée via /get_cities). |
| TxtDate | Champ texte permettant de saisir une date au format YYYY-MM-DD . |
| BtnFetchData | Bouton qui déclenche l'appel à l'API Flask (/ville) pour récupérer les données de la ville et de la date spécifiées. |
| BtnOK | Bouton qui déclenche le traitement des données : calcul de l'indice de temps entre les échantillons et création d'une ArrayList contenant deux colonnes : → Colonne 1 : Indice (temps écoulé depuis la première mesure, en millisecondes) → Colonne 2 : Valeur de la tension résiduelle. |

Tableau 3- Composants de l'interface dans AnyLogic

Fonctionnement global :

1. **Initialisation** : la simulation récupère la liste des villes disponibles via /get_cities.
2. L'utilisateur **choisit une ville** et **saisit une date**.
3. BtnFetchData interroge l'API Flask et **stocke les résultats** dans une structure de données locale.
4. Lors du clic sur btnOK, les données récupérées sont **prétraitées** :
 - La **différence de temps** est calculée pour chaque ligne par rapport à la première mesure (DateX - Date0).
 - Une ArrayList est générée avec pour chaque ligne :
 - L'**indice de temps écoulé**
 - La **tension résiduelle** mesurée.
5. La simulation peut ensuite être **lancée avec ces données traitées**.

| Indice | Valeur |
|--------|--------|
| 0.0 | |
| 0.2 | |
| 0.5 | |
| 0.7 | |
| 1.0 | |
| 1.2 | |
| 1.5 | |
| 1.7 | |
| 2.0 | |
| 2.2 | |
| 2.5 | |
| 2.7 | |
| 3.0 | |
| 3.2 | |
| 3.5 | |
| 3.7 | |
| 4.0 | |
| 4.2 | |
| 4.5 | |
| 4.7 | |
| 5.0 | |

Tableau 4- données 'indice et valeur'
envoyés à Matlab/Simulink

Résultats :

Cette intégration permet une **expérience interactive et personnalisée**, en adaptant la simulation aux données historiques réelles de différentes villes et dates. Elle prépare également les entrées nécessaires à l'**envoi vers MATLAB/Simulink** pour des analyses complémentaires.

3.4.5 Passerelle de communication entre Python et MATLAB/Simulink

Dans cette étape, une API en Python utilisant Flask a été développée pour interagir automatiquement avec le moteur MATLAB et un modèle Simulink, afin d'automatiser la simulation à partir des données calculées dans AnyLogic.

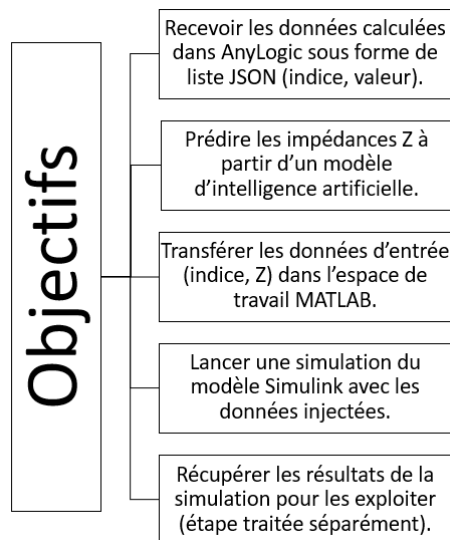


Figure 18- Objectifs de 2ème API

Architecture :

| Étape | Description |
|---|--|
| 1. Initialisation | Lancement d'une application Flask qui reste en écoute sur un port dédié (5049). Le moteur MATLAB est lancé une seule fois au démarrage de l'API. |
| 2. Chargement IA | Le modèle d'intelligence artificielle entraîné (fichier. pkcls) est chargé une seule fois en mémoire pour les prédictions futures. |
| 3. Réception des données | Le Endpoint /envoyer_donnees reçoit un tableau JSON contenant des paires { "indice": ..., "valeur": ... } correspondant aux tensions résiduelles. |
| 4. Prédiction avec IA | Pour chaque tension reçue, une impédance Z est prédite par le modèle Orange ¹¹ . |
| 5. Configuration de MATLAB | Le chemin vers le modèle Simulink est défini, puis MATLAB change de répertoire (cd) et efface les anciennes variables. |
| 6. Injection des données | Les vecteurs Vecteur_TempsRe (indices) et Reet (impédances) sont envoyés dans le workspace MATLAB. La variable stop_time est fixée automatiquement au dernier indice reçu. |
| 7. Simulation | Le modèle Simulink est ouvert visuellement et simulé en utilisant la commande sim(...). La simulation s'arrête au temps défini (stop_time). |
| 8. Récupération partielle des résultats | Les données de sortie (signaux SimOut) sont extraites depuis le workspace MATLAB. Ces signaux comprennent notamment Current, Current_RMS, Tension, Tension_RMS et le vecteur temps. (Leur traitement est abordé dans une autre section). |
| 9. Retour JSON | L'API retourne une réponse au format JSON contenant les paires { "indice": ..., "Z": ... }, permettant à AnyLogic de les exploiter. |

Tableau 5- Architecture

3.4.6 Mise en œuvre d'une Co-simulation AnyLogic – MATLAB/Simulink

Dans cette phase du projet, une Co-simulation entre AnyLogic et MATLAB/Simulink a été mise en œuvre afin d'associer la modélisation multi-agents ferroviaire à des calculs physiques avancés. L'objectif était de permettre un échange dynamique de données en temps réel : AnyLogic transmet les informations issues de la base de données (via l'API Flask) vers un modèle MATLAB, chargé de réaliser des prédictions à l'aide de réseaux de neurones. Cette intégration offre une meilleure représentativité des phénomènes physiques et renforce la pertinence des scénarios simulés.

¹¹ Orange est une bibliothèque utilisée pour lire les modèle IA

Étapes de la Co-simulation :

- **Lancement simultané des deux serveurs API :**
 - L'API Flask principale pour fournir les données issues de la base MySQL.
 - Une **seconde API Python** (nommée `api_matlab.py`) permettant d'envoyer les tensions et indices vers MATLAB/Simulink pour effectuer des prédictions sur l'impédance.
- **Communication entre AnyLogic et les APIs :**
 - L'utilisateur sélectionne une ville et une date.
 - Les données sont récupérées depuis l'API Flask et préparées (calcul de l'indice, création d'une ArrayList).
 - Ces données sont ensuite envoyées vers l'API MATLAB pour déclencher le processus de prédiction.
- **Simulation MATLAB/Simulink :**
 - Le modèle Simulink utilise les valeurs reçues (tension, indice) pour simuler le comportement électrique et retourner des valeurs comme le courant, la tension efficace ou l'impédance.
 - Les résultats sont ensuite affichés ou exploités dans AnyLogic.

Objectif de cette intégration :

Cette Co-simulation permet une interaction fluide entre la visualisation dynamique (AnyLogic) et les calculs physiques complexes (MATLAB/Simulink), facilitant une meilleure compréhension du phénomène de déshuntage ferroviaire dans un environnement interactif et modulaire.

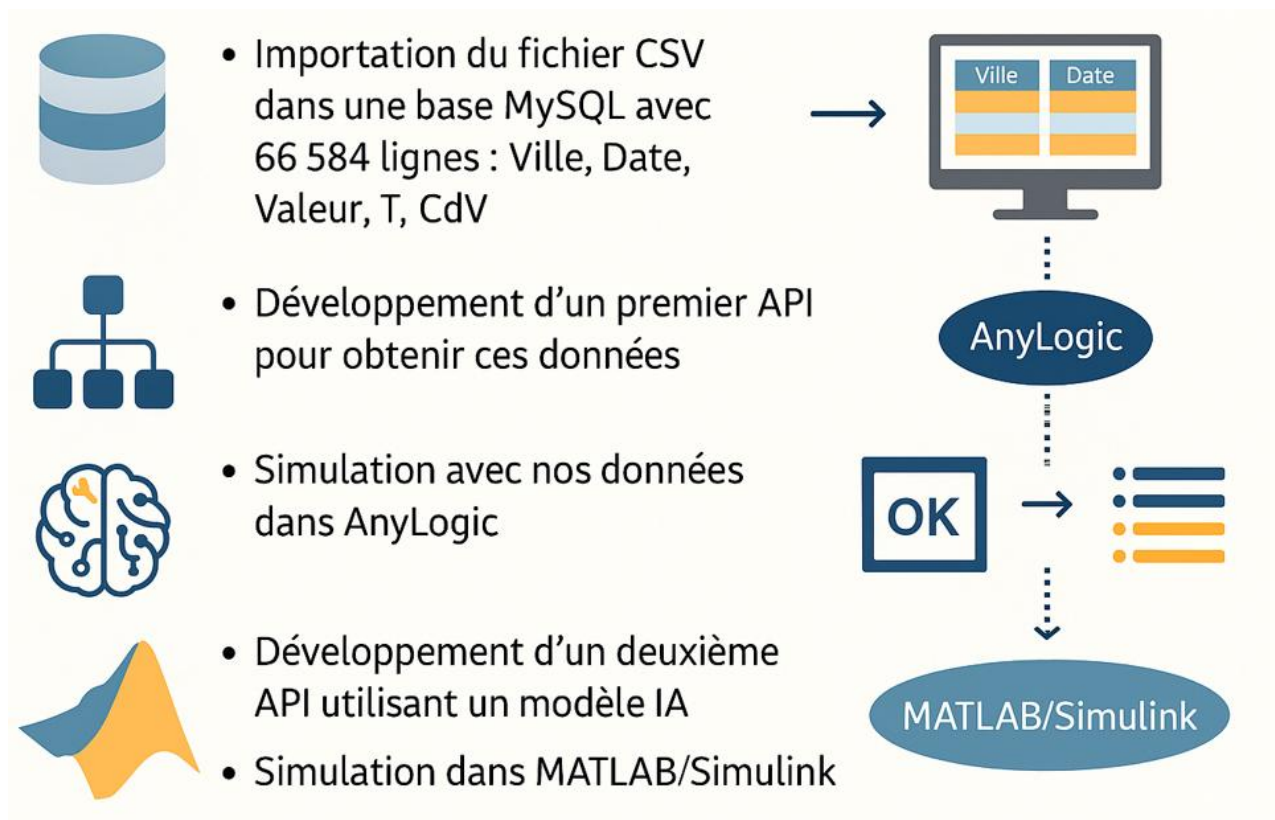


Figure 19- Co-simulation AnyLogic/MATLAB

3.4.6.1 Intégration dynamique avec MATLAB : Injection et simulation automatisée

Avant de passer à la simulation dans Matlab/Simulink, il faut établir une étape nécessaire :

Dans cette phase, un mécanisme d'automatisation a été mis en place afin d'assurer le transfert des données issues d'AnyLogic (via l'API Python) vers l'environnement MATLAB, où elles alimentent directement un modèle Simulink. Ce processus repose sur la préparation d'un espace de travail dédié, l'injection dynamique des vecteurs de temps et d'impédance, ainsi que la définition automatique de la durée de simulation. L'ensemble de ces étapes permet de déclencher et d'exécuter la simulation Simulink sans intervention manuelle, en s'appuyant sur un pilotage scripté depuis Python. Cette procédure de Co-simulation assure une orchestration fluide entre les deux environnements, garantissant un échange de données en temps réel et une exécution optimisée des calculs. Elle constitue un élément central du dispositif de diagnostic du déshuntage ferroviaire, en permettant d'analyser l'évolution des signaux électriques dans des conditions représentatives du terrain.

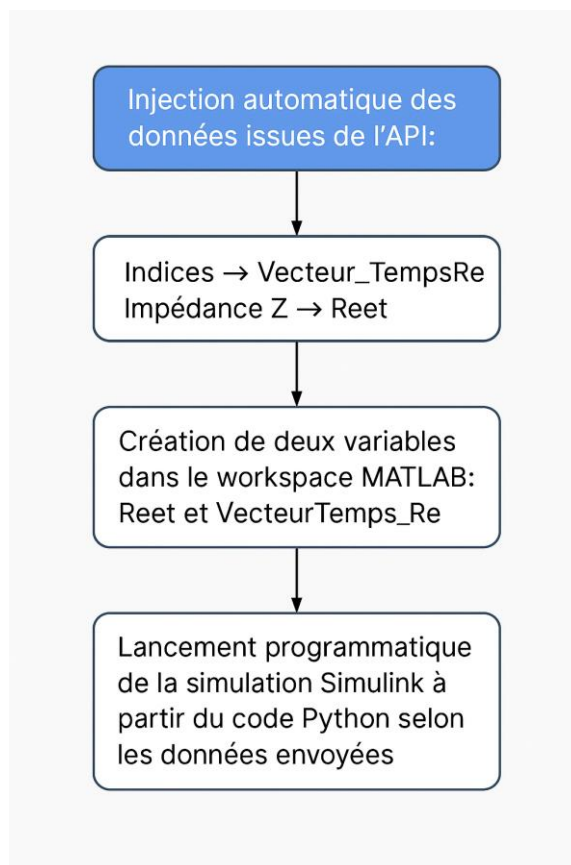


Figure 20- étape pour initialiser la simulation

Chapitre 4

4. Tests et Validation

Ce chapitre est consacré à la phase de test et de validation, étape essentielle pour garantir la fiabilité et la pertinence du modèle développé dans le cadre de ce projet. L'objectif est de confronter les hypothèses théoriques et les choix de conception aux conditions pratiques de simulation, tout en s'assurant de la cohérence des résultats obtenus.

La démarche adoptée s'articule autour de plusieurs volets complémentaires. Dans un premier temps, une préparation des données a été réalisée, notamment par l'enrichissement de la base avec une nouvelle colonne d'impédance (Z) calculée pour chaque valeur de tension. Cette étape préalable assure une meilleure représentativité des phénomènes électriques à simuler.

Ensuite, un scénario opérationnel de test a été défini et mis en place afin de reproduire, dans un cadre simplifié mais réaliste, les conditions d'utilisation du matériel roulant. Ce scénario a servi de base aux simulations conduites sous MATLAB/Simulink, permettant d'obtenir des grandeurs physiques clés telles que le courant, la tension et leurs valeurs efficaces.

Enfin, les résultats générés par la simulation ont été exploités et transférés vers une base de données de séries temporelles (Influx DB), offrant un environnement adapté à la gestion de volumes massifs de mesures et à leur visualisation. L'ensemble de ce processus permet non seulement de valider le fonctionnement du pipeline de Co-simulation, mais aussi d'identifier les limites actuelles et les pistes d'amélioration pour les travaux futurs.

En vue de lancer la phase expérimentale, il est nécessaire de préparer en amont les données de simulation afin d'assurer leur cohérence et leur exploitabilité. La première étape porte ainsi sur l'enrichissement de la base de données initiale par l'ajout d'informations indispensables au bon déroulement des tests.

4.1 Préparation des données

La qualité et la structuration des données constituent un facteur déterminant dans le succès d'une Co-simulation. Dans ce cadre, une contribution spécifique a été réalisée par le développement d'un script Python permettant d'automatiser l'ajout d'une nouvelle colonne d'impédance (Z) associée à chaque valeur de tension enregistrée dans la base.

L'intégration de cette grandeur électrique répond à un double objectif : d'une part, disposer d'une vision plus complète du comportement du système électrique simulé, et d'autre part, offrir une base de calcul plus robuste pour les étapes de modélisation et d'analyse ultérieures. Grâce à cette étape, la base de données devient non seulement plus représentative des phénomènes physiques à reproduire, mais également mieux adaptée aux besoins des outils de simulation et de validation.

| | A | B | C | D | E | F | G |
|----|-------|-------------------------|---------|---|-----|-------------|-------------------|
| 1 | Ville | DateComplete | Tension | T | CdV | Commentaire | Impedance_Predite |
| 2 | | 2013-10-22 01:41:14.700 | 329,55 | | | | 653,5427194 |
| 3 | | | | | | | 655,8354978 |
| 4 | | | | | | | 655,1832419 |
| 5 | | | | | | | 654,4914553 |
| 6 | | | | | | | 655,7959671 |
| 7 | | | | | | | 656,2505697 |
| 8 | | | | | | | 654,115914 |
| 9 | | | | | | | 656,1122124 |
| 10 | | | | | | | 656,685407 |
| 11 | | | | | | | 654,9658232 |
| 12 | | | | | | | 657,0807136 |
| 13 | | | | | | | 658,5433482 |
| 14 | | | | | | | 655,7959671 |
| 15 | | | | | | | 658,0689802 |
| 16 | | | | | | | 659,5316147 |
| 17 | | | | | | | 656,922591 |
| 18 | | | | | | | 658,6817055 |

Figure 21- Base de données incluant la colonne des impédances

4.2 Définition et mise en place d'un scénario opérationnel

Lors du développement de la Co-simulation, un problème majeur est rapidement apparu : **le temps de traitement excessif des données**. En particulier, lors de l'utilisation de la ville **1** par exemple, issue de la base de données **initiale**, le volume d'informations à transmettre et à traiter devenait extrêmement lourd. Ce jeu de données comprenait environ **13 000 indices** et **13 000 valeurs de tension résiduelle**, avec certains indices de temps atteignant **plus de 22 000**. Cette complexité se traduisait par un temps de traitement estimé à **plus de 5 jours**, rendant impossible une simulation fluide sans disposer d'une **machine très puissante** ou d'une **solution d'optimisation via machine virtuelle**.

Pour **contourner cette contrainte technique**, une solution pragmatique a été mise en place : la **création d'une ville de test allégée**. Concrètement, cela a consisté à **extraire les 10 premières lignes** de la ville **1** depuis la base de données **MySQL Mypro**, tout en conservant la structure originale des données (colonnes : Ville, Date, T, CdV, Commentaire).

Ces données ont été insérées dans une **nouvelle table ville_test**, permettant d'exécuter des tests rapides sur l'ensemble du pipeline (API, interface AnyLogic, injection MATLAB et Co-simulation). Cette approche a permis de valider la logique de traitement sans subir les ralentissements dus au volume, tout en gardant un jeu de données représentatif.

| Ville | Date | Valeur | T | CdV | Commentaire |
|-------|-------------------------|--------|------|-----|--|
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |
| 1 | 2013-10-22 01:41:14.700 | 329.55 | T002 | 214 | Tension au passage du 2 1/2 me train (X 73500) |

Figure 22- Base de données MySQL de la ville de test

4.3 Simulation MATLAB/Simulink

Pour finaliser le processus de simulation, la mise en place d'un second serveur API dédié à l'environnement MATLAB/Simulink a été nécessaire. Cette architecture a permis d'assurer l'exécution conjointe de deux services distincts et complémentaires :

- L'API Flask, chargée de l'extraction et de la mise à disposition des données issues de la base MySQL,
- L'API MATLAB, dédiée à l'injection des données et au pilotage des scénarios de simulation dans Simulink.

Une fois ces deux services opérationnels, la plateforme AnyLogic a été mobilisée comme interface centrale de test et de validation. À travers cette interface, un scénario expérimental a été configuré, prenant comme paramètres la **ville 1** et une date de référence **X**. L'objectif était de vérifier la continuité et la cohérence du flux de données, depuis leur stockage en base relationnelle jusqu'à leur exploitation dans l'environnement de simulation.

Les résultats obtenus ont confirmé le bon acheminement des informations et la capacité du pipeline à restituer, en temps réel, les grandeurs issues de MATLAB/Simulink directement dans AnyLogic. Cette étape représente une validation essentielle de l'orchestration technique, démontrant la robustesse de l'intégration multi-outils (MySQL – Flask – MATLAB – AnyLogic) et ouvrant la voie à des scénarios d'analyse avancée dans le cadre du phénomène de déshuntage.

| Indice | Valeur |
|--------|--------|
| 0.0 | 330.95 |
| 0.2 | 330.71 |
| 0.5 | 330.38 |
| 0.7 | 330.03 |
| 1.0 | 329.68 |
| 1.2 | 329.32 |
| 1.5 | 328.84 |
| 1.7 | 328.35 |
| 2.0 | 327.14 |
| 2.2 | 326.27 |

Tableau 6- Données 'indice, valeurs' de la ville de test

Puis après les tensions sont prédites en Z impédance et injectés dans la variable Reet créé dans le workspace de Matlab, avec stop_time = 2.2

➤ Simulation commencée

Après la fin de la simulation obtient des résultats dans l'objet SimOut :

- Current (Courant)
- Current_RMS (Courant efficace)
- Tension (Tension)
- Tension_RMS (Tension efficace)

Avec le retour de la fonction **analyser_donnees** qui nous permet de savoir le nombre d'éléments traités

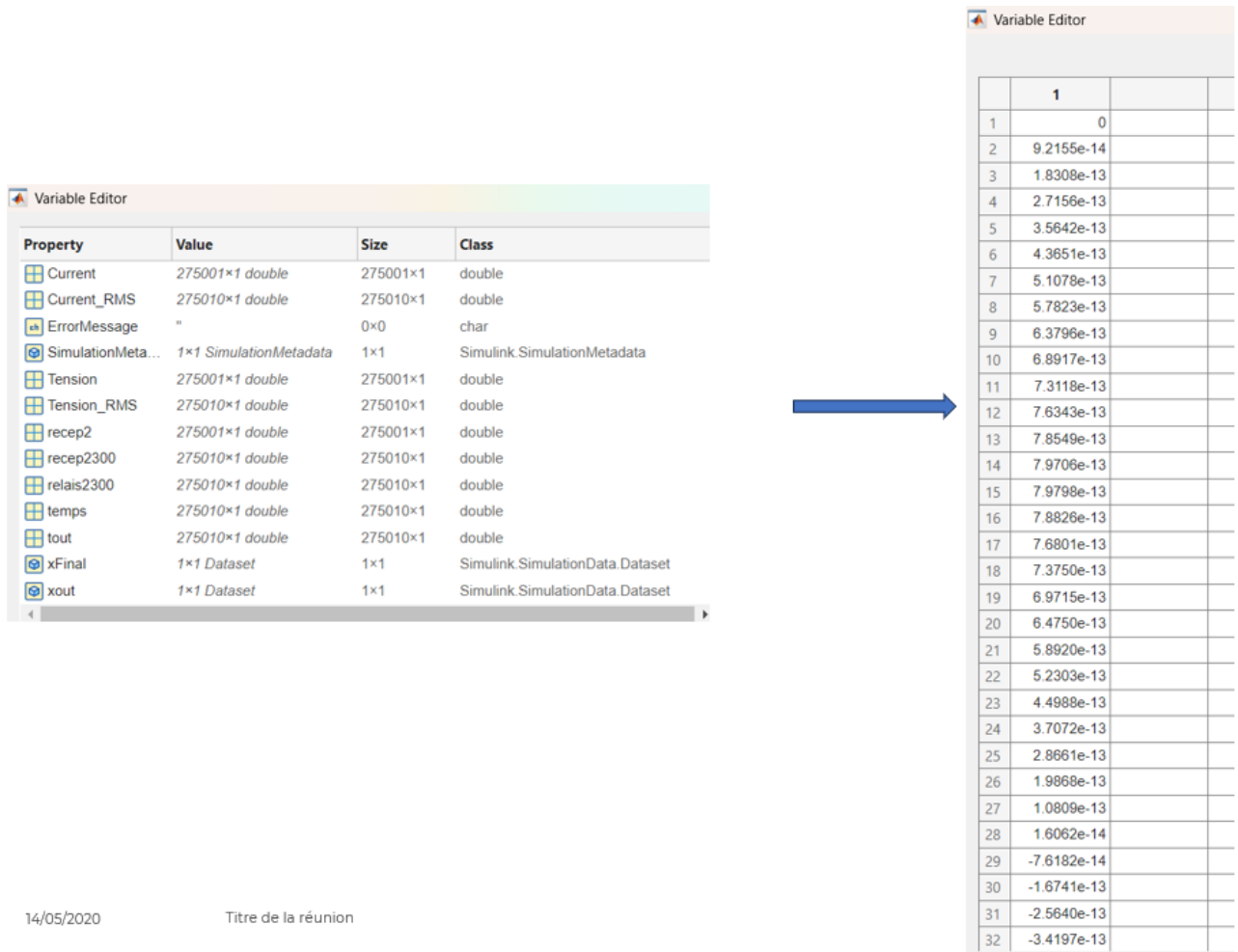


Figure 23- Résultats après la simulation MATLAB

4.4 Exploitation et transfert des résultats

À l'issue des simulations réalisées sous MATLAB/Simulink, plusieurs grandeurs électriques ont été obtenues, parmi lesquelles les courants, les tensions et leurs valeurs efficaces représentatives du comportement du matériel roulant. Ces résultats constituent des indicateurs essentiels pour l'analyse de la performance du système et la validation des hypothèses initiales.

Afin de faciliter leur gestion et leur exploitation, l'ensemble des données issues des simulations a été transféré vers une base de données de séries temporelles (Influx DB). Ce choix technologique se justifie par la nature volumétrique et continue des résultats générés : chaque simulation produit un ensemble conséquent de mesures évoluant dans le temps. Influx DB permet ainsi de stocker efficacement ces séries temporelles, d'assurer une structuration adaptée et de mettre à disposition des outils de visualisation et d'analyse en temps réel.

Cette intégration offre plusieurs avantages : elle garantit la traçabilité des résultats, favorise leur exploitation ultérieure pour des analyses comparatives et ouvre la voie à une interopérabilité avec d'autres environnements de traitement ou de visualisation. L'approche retenue contribue ainsi à renforcer la robustesse du processus de validation tout en posant les bases d'une architecture évolutive pour les expérimentations futures.

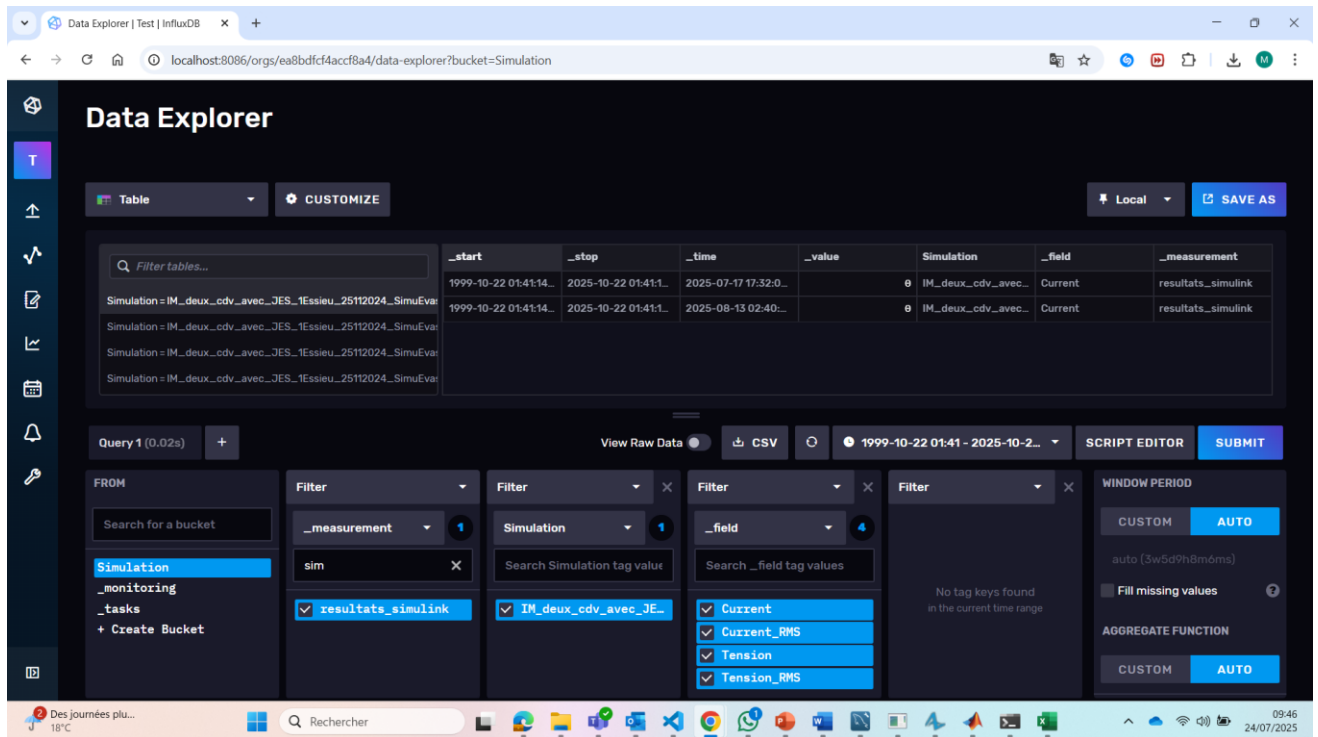


Figure 24- Résultats transférés à la base de données Influx DB

4.5 Analyse comparative et intégration des résultats

Une fois les simulations effectuées dans MATLAB/Simulink, une étape essentielle a consisté à confronter les grandeurs simulées aux données résiduelles initialement disponibles. Plus précisément, la comparaison a porté sur les valeurs de tension résiduelle en entrée et celles issues de la sortie du modèle de simulation appliqué au matériel roulant.

L'objectif de cette analyse comparative est de mesurer l'écart entre la réalité observée (tension résiduelle mesurée) et la réponse simulée (tension calculée), afin d'évaluer la fiabilité et la précision du modèle mis en place. Pour cela, un calcul de pourcentage d'erreur a été réalisé en s'appuyant sur la norme 2 (erreur quadratique moyenne relative), qui constitue une référence robuste pour estimer la qualité d'ajustement entre deux séries de données.

Cette approche permet de mettre en évidence les éventuelles divergences entre les résultats simulés et les valeurs de terrain, tout en quantifiant le degré de précision atteint. Les écarts identifiés fournissent ainsi des indications précieuses sur les axes possibles d'amélioration du modèle, notamment en termes de paramétrage ou d'hypothèses physiques intégrées.

Enfin, afin d'assurer une traçabilité complète et de faciliter les analyses ultérieures, l'ensemble des résultats obtenus (tensions simulées, écarts calculés, pourcentages d'erreur) a été intégré dans la base de données initiale. Cette capitalisation permet de constituer un référentiel exploitable pour de futures simulations, comparaisons ou optimisations.

4.6 Validation et perspectives

La phase de validation a permis de confronter les résultats de simulation aux objectifs fixés et d'évaluer la cohérence du pipeline mis en place. Les mesures de courant, de tension et de valeurs efficaces obtenues se sont révélées conformes aux attentes théoriques, confirmant la pertinence de l'approche adoptée. L'intégration dans une base de données temporelle dédiée a renforcé cette validation en offrant une exploitation claire et structurée des résultats.

Cette étape a également mis en évidence certaines limites, notamment liées au volume de données et au temps de traitement requis pour des simulations de grande envergure. Bien que l'utilisation d'un scénario de test allégé ait permis de contourner ces contraintes, une montée en charge reste à envisager afin d'anticiper les besoins réels d'une exploitation industrielle.

En termes de perspectives, plusieurs axes d'amélioration peuvent être considérés :

- L'optimisation du traitement des données pour réduire les délais de simulation ;
- L'automatisation de la chaîne de transfert entre les outils de simulation et les bases de données ;
- L'élargissement du périmètre des scénarios testés afin de couvrir des cas opérationnels plus variés et représentatifs.

Ces évolutions offriraient non seulement un gain d'efficacité mais également une meilleure fiabilité des analyses, consolidant ainsi la valeur ajoutée du système dans un cadre applicatif métier.

Et à l'aide de ces résultats nous pouvons détecter le **déshuntage**.

Chapitre 5

5.Conclusion et Perspectives

Ce projet a permis de concevoir, mettre en œuvre et valider une chaîne complète de Co-simulation intégrant différentes briques technologiques : base de données relationnelle (MySQL), API Python, modèle de simulation AnyLogic, intégration MATLAB/Simulink et stockage des résultats dans une base de séries temporelles (Influx DB).

L'approche adoptée a répondu à la problématique initiale en offrant une architecture cohérente et robuste, capable de traiter et de transférer des données complexes entre plusieurs environnements hétérogènes. L'introduction d'un scénario de test allégé a constitué une solution pragmatique pour contourner les contraintes de temps de calcul tout en assurant la validité des résultats. Le développement d'un script Python pour enrichir la base de données avec une colonne d'impédance (Z) a, par ailleurs, renforcé la représentativité physique des simulations.

Au-delà de l'aspect technique, ce projet m'a permis de réaliser des simulations numériques impossibles à mettre en œuvre physiquement dans le cadre de ce stage. Ces expérimentations virtuelles, basées sur des données réelles, ont offert la possibilité de reproduire le comportement électrique du matériel roulant et d'obtenir des grandeurs essentielles (tension, courant, valeurs efficaces). Ces résultats constituent un support précieux pour l'analyse et la détection du phénomène de déshuntage, contribuant ainsi directement aux objectifs de sécurité et de fiabilité du système ferroviaire. Dans l'ensemble, les résultats obtenus témoignent de la pertinence de la méthodologie mise en place, confirmant la faisabilité et la valeur ajoutée du pipeline de Co-simulation pour l'étude des phénomènes électriques liés au matériel roulant.

Perspectives

Au-delà des résultats obtenus, plusieurs perspectives peuvent être envisagées afin de consolider et d'étendre les travaux menés :

- **Optimisation des performances** : réduire le temps de traitement des données pour permettre la simulation de cas à grande échelle.
- **Automatisation complète de la chaîne** : mise en place de scripts et services permettant d'assurer un transfert fluide et transparent entre les différentes briques du pipeline.
- **Extension des scénarios de test** : enrichir la base de données et diversifier les cas simulés pour couvrir un plus large spectre de situations opérationnelles.
- **Interopérabilité accrue** : explorer l'intégration avec d'autres plateformes de simulation ou d'analyse, notamment pour une utilisation en contexte industriel.
- **Visualisation avancée** : développement d'interfaces interactives permettant une exploitation plus intuitive et dynamique des résultats de simulation.

Ces évolutions offriraient une continuité naturelle au projet en le rapprochant progressivement d'une solution opérationnelle, tout en renforçant son intérêt dans un cadre métier et scientifique.