



Rapport de stage en entreprise

**Développement d'un
module d'aide à la décision
et réalisation d'études :
Remisage en gare Toulouse
Matabiau**

Maître de stage : **Éric LEBEAU**

Chef du service Développement et Projets à la
direction régionale de la SNCF Midi-Pyrénées

Tuteur de stage : **Romain GUILLAUME,**
Professeur à l'Université Toulouse II, Le Mirail

Stage réalisé du 04/03 au 29/08 2014 à la
Direction Régionale SNCF Midi-Pyrénées

Pôle Développement et Projets TER



Promotion M2 ISMAG : année 2013-2014

Par : **Clément TOURNÉ**



Remerciements

- Thierry BAUCHET : Directeur Délégué TER Midi-Pyrénées pour m'avoir permis d'intégrer son activité et rendu ce stage possible.
 - Philippe LASCAUX : Directeur Délégué TER Midi-Pyrénées durant la seconde partie du stage, pour sa confiance prolongée.
 - Éric LEBEAU : Chef du service Développement et Projets et tuteur en entreprise de ce stage, également pour sa confiance mais aussi son encadrement et son soutien quotidien.
 - Philippe BLANC : Membre du service Développement et Projets, pour son aide et son suivi des projets.
 - Maxime PERRIN : Membre du service Développement et Projets, qui fut mon premier contact avec l'entreprise et m'a fait part de la mission.
 - Katia BENHELLAL et Aurélien PERILHOU : Membres du service Développement et Projets, avec qui j'ai pu collaborer.
 - Pascal SARDA : Directeur du Master ISMAG à l'UTM, pour m'avoir permis d'intégrer sa formation durant ces deux années.
 - Romain GUILLAUME : Enseignant-chercheur à l'UTM et enseignant référant de ce stage, pour sa disponibilité et son aide au cours de la mission.
 - L'équipe enseignante et la promotion ISMAG 2012-2014 pour l'intérêt et le dynamisme donnés à ces deux années de formation.
-

Sommaire

Introduction	6
I. Présentation de l'entreprise.....	7
A) Présentation générale	7
B) La branche SNCF Proximités	7
C) Le TER au sein de la SNCF	8
II. Outils et environnement de travail	9
A) Excel et VBA.....	9
1. Microsoft Excel	9
2. Les macros VBA.....	10
3. Pourquoi cette utilisation ?.....	10
B) NetBeans et Java	11
1. L'environnement de développement NetBeans	11
2. Le choix du langage Java	11
C) GLPK utilisé avec GUSEK et Java	13
1. La bibliothèque GLPK 4.8	13
2. SciTE et GUSEK	13
3. Utiliser GLPK avec Java.....	14
III. Travail à réaliser	15
A) But et description de la mission	15
B) Déroulement de la mission.....	15
C) Solutions mises en place ou envisagées.....	16
D) Planning prévisionnel	16
IV. Choix de l'algorithme d'optimisation	17
A) Evaluation de la taille du problème.....	17
1. Définition des objectifs	17
2. Nombre de sites et de voies	18
3. Trains et matériels à remiser	18
B) Réorganisation du problème	20
1. Fragmentation du problème	20
2. Utilisation de la méthode Tabou	20
C) Optimisation a une heure fixée	21
D) Optimisation sur une période	22

V.	Description technique du programme	23
A)	Objectifs et méthodologie générale.....	23
B)	Code JAVA.....	24
1.	Les classes du programme :	24
2.	Déroulement du programme.....	32
C)	Fichiers Excel et code VBA.....	36
1.	Préparation à l'optimisation	36
2.	Graphiques et résultats.....	38
D)	Modèles linéaires	41
1.	Optimisation du créneau de pointe.....	41
2.	Optimisation d'une période.....	41
VI.	Résultats d'études menées	43
A)	Viabilité des infrastructures actuelles à Matabiau.....	43
1.	Objectifs de l'étude.....	43
2.	Démarche et calculs réalisés.....	43
3.	Résultats obtenus	45
B)	Trois scénarios d'évolution.....	46
1.	Scénario peu coûteux	46
2.	Scénario large	47
3.	Scénario intermédiaire	48
4.	Préconisations en fonction des résultats.....	48
VII.	Intégration à l'outil SIMUTER	49
	Conclusion.....	50
	Résumé.....	51

Introduction

Ce stage s'est déroulé dans le cadre de la finalisation du Master ISMAG proposé par l'Université Toulouse II (Le Mirail) au sein de l'entreprise SNCF. Durant une période de 6 mois, celui-ci avait pour but de se centrer sur la Recherche Opérationnelle, mais aussi d'autres disciplines également enseignées dans le cursus, que sont la Programmation Informatique ou la Gestion de Projets.

Au sein du pôle Développement et Projets TER Midi-Pyrénées de la SNCF, la mission présentait deux enjeux : le développement d'un outil automatisé d'aide à la décision et la réalisation d'études en parallèle.

Ces deux objectifs étaient liés à la résolution de problématiques de remisage en gare Toulouse Matabiau. Dans le monde ferroviaire, le remisage correspond au rangement des matériels/trains lorsque ceux-ci ne sont pas en circulation. L'évolution des dessertes, de l'offre des transporteurs ferroviaires et le renouvellement des matériels entraînent une augmentation des engins à remiser et donc de nouvelles contraintes d'infrastructures.

En utilisant des méthodes de calculs de Recherche Opérationnelle telles que la modélisation et la résolution de problèmes linéaires ou les métaheuristiques, une première étude devait être menée sur la viabilité de l'organisation du remisage actuelle. En portant ces méthodes vers un outil informatique, il serait alors possible de dire si cette organisation pouvait être optimisée et si oui, une solution optimisée suffirait-elle pour pallier aux nouveaux besoins dans un futur proche.

En se basant ensuite vers des horizons plus lointains, ces méthodes de calculs devaient être réutilisées pour pouvoir émettre des préconisations sur des travaux et des ajouts de voies nécessaires au fil du temps.

Enfin, une fois l'ensemble des calculs automatisés, le dernier point de la mission était d'intégrer cet outil informatique en tant que module au logiciel Simuter. Ce logiciel d'aide à la décision développé pour l'activité TER Midi-Pyrénées vise à s'étendre à un maximum de domaines et à devenir adaptable pour les autres régions de France. Ainsi le développement du module pourrait servir à évaluer les besoins d'infrastructures de remisage dans le futur et également étayer les calculs de bilans économiques déjà proposés par le logiciel.

I. Présentation de l'entreprise

A) Présentation générale

La SNCF, Société Nationale des chemins de FER, a été créée le 1^{er} janvier 1938 et a obtenu le statut d'EPIC (Etablissement public à Caractère Industriel et Commercial) en 1983. En 1997, un nouvel EPIC chargé de gérer l'infrastructure voit le jour, il s'agit du Réseau Ferré de France (RFF). RFF devient alors propriétaire de l'infrastructure et décideur en matière d'aménagement, de développement et de valorisation du réseau. La SNCF, gestionnaire délégué de l'infrastructure, s'occupe de la gestion de la circulation et l'entretien du réseau pour le compte de RFF.

C'est une entreprise experte du transport et de la logistique.

Elle est présente à toutes les échelles du transport de voyageurs et de marchandises et se montre performante dans de multiples activités : Ingénierie, conseil, logistique, exploitation, travaux...

La SNCF fait partie du GROUPE SNCF qui se décompose, comme on peut le voir sur le schéma ci-dessous, en 5 activités :



Le groupe SNCF comprend aussi de nombreuses filiales. En effet, fort de son expérience, le groupe exporte son savoir-faire sur les 5 continents.

B) La branche SNCF Proximités

SNCF Proximités est le numéro 2 mondial de la mobilité collective et le premier partenaire des collectivités françaises. Son chiffre d'affaires s'élève à plus de 12,8 milliards d'euros (c'est le plus élevé des cinq branches). Chaque jour, 10 millions de voyageurs utilisent les trains régionaux, bus, cars, métros, tram-trains, vélos en libre-service, sans oublier les offres de parking et de covoiturage.

La branche Proximités rassemble l'ensemble des activités conventionnées du groupe dont l'activité TER.

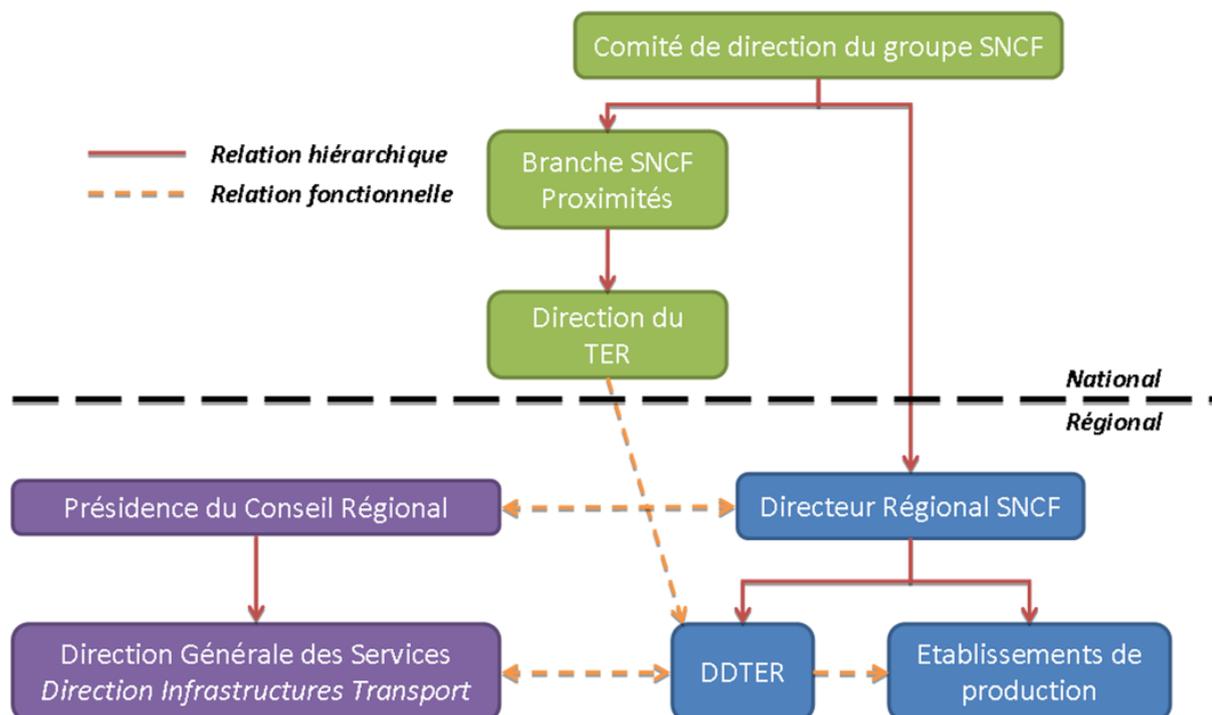
C) Le TER au sein de la SNCF

SNCF Proximités s'appuie notamment sur l'activité TER (Transport Express Régional) pour le transport ferroviaire régional de province, sur le TRANSILIEN (transport régionaux d'Ile-de-France) et sur la filiale KEOLIS (transport urbain, interurbain routier et ferroviaire à l'étranger).

L'organisation de la SNCF pour assurer les services TER des différentes régions est construite sur une logique de proximité avec les autorités organisatrices. Chacune des 20 activités régionales étant pilotée localement, sous l'autorité du Directeur Régional de la SNCF, par la Direction Déléguée TER – DDTER qui assure la gestion du service (conception du plan de transport, mise œuvre de l'offre tarifaire, suivi de la qualité, gestion contractuelle et financière).

L'exploitation au quotidien est pour sa part réalisée par les différents établissements de production régionaux (maintenance du matériel roulant, de l'infrastructure, vente et assistance en escale, conduite, accompagnement des trains, gestion des circulations).

Chaque activité TER fonctionne ainsi dans une relative indépendance, avec des équipes dédiées, au profit des relations avec les Conseils Régionaux. Au niveau national la Direction du TER encadre l'ensemble des activités régionales.

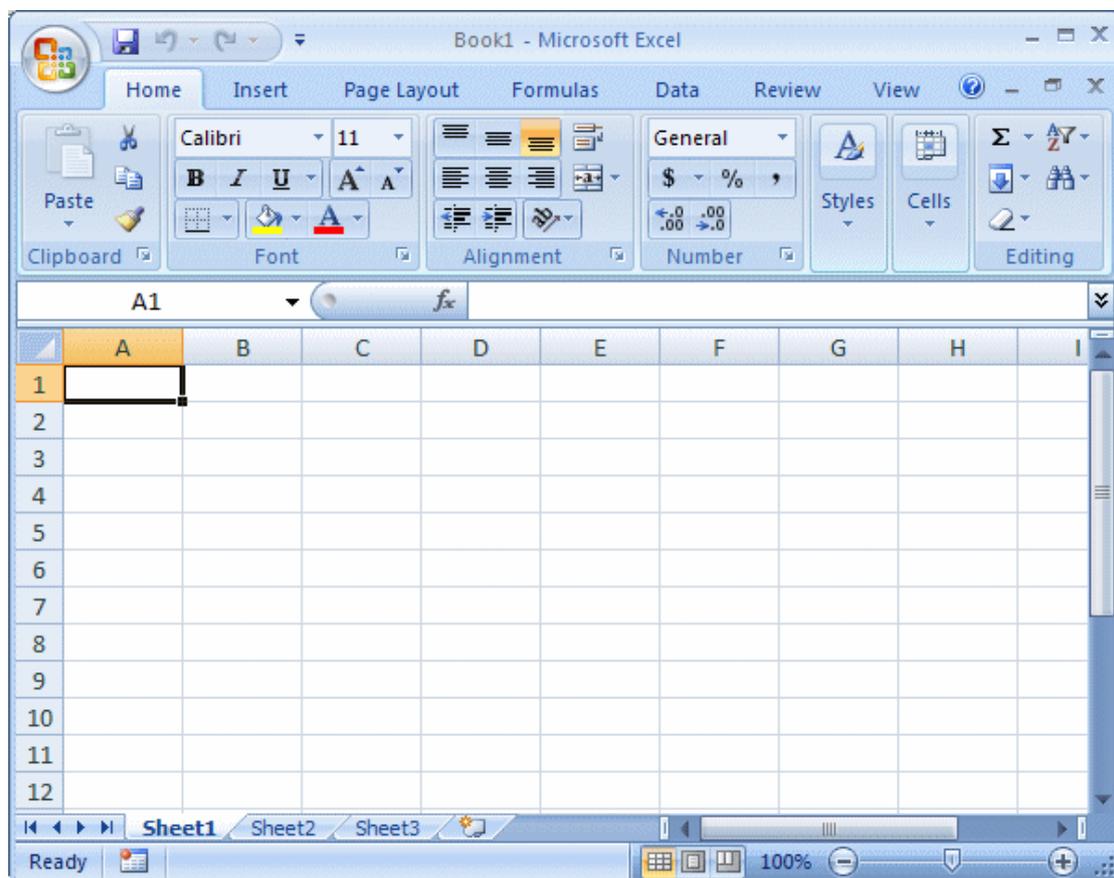


II. Outils et environnement de travail

A) Excel et VBA

1. Microsoft Excel

Faisant partie de la suite de logiciels de bureautique Microsoft Office, Excel est un logiciel tableur. Selon les machines de l'entreprise, les versions proposées étaient différentes. Les fichiers de données utilisés visent à être lisibles avec n'importe quelle machine et sont donc enregistrés au format compatible avec les anciennes versions du logiciel, avec l'extension « .xls ». En revanche, les développements automatisés se font eux uniquement sur les nouvelles versions pour un gain de temps d'exécution, et utilisent donc les extensions « .xlsx » et « .xlsm ».

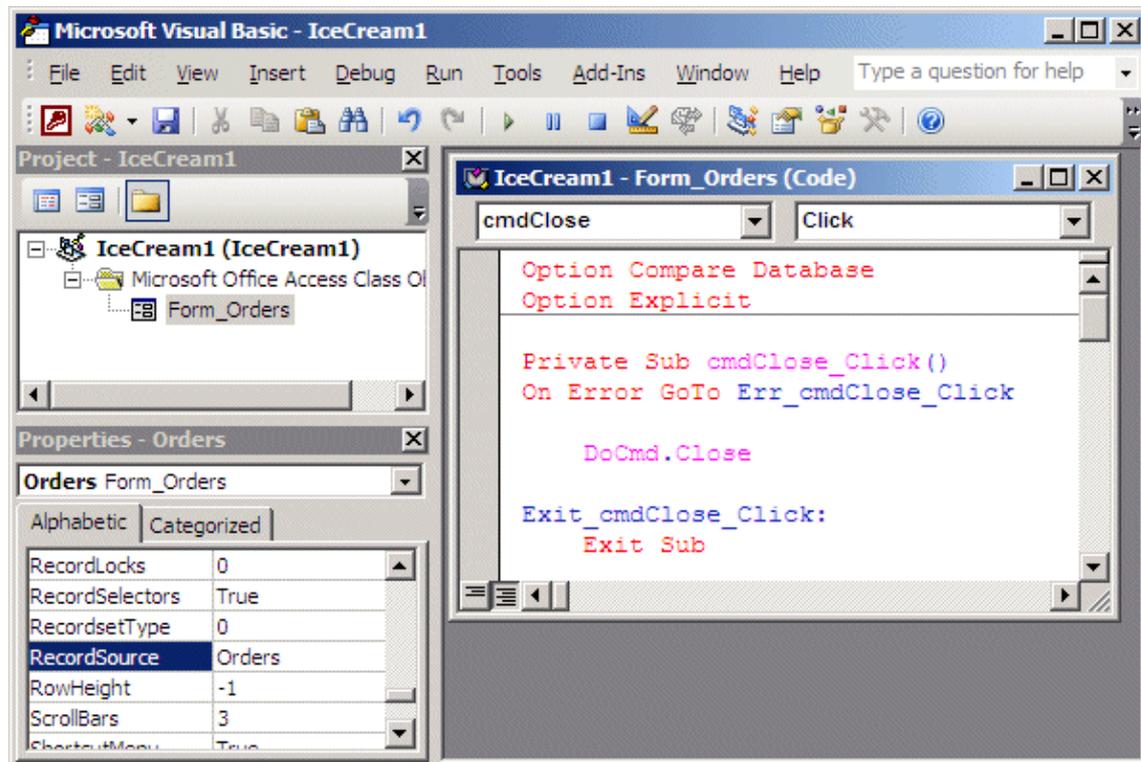


Au-delà de la simple saisie de données, l'interface principale du logiciel permet de réaliser de nombreux calculs automatiques notamment grâce à son onglet « formule ». Les données saisies peuvent également être interprétées sous de nombreuses formes, comme par exemple une chaîne de caractère, un nombre décimal ou une heure. La définition du format des données saisies est importante pour l'automatisation de calcul ou la lisibilité du fichier par d'autres programmes.

Pour le développement de calculs automatisés, Microsoft Excel possède l'avantage de posséder un système de sécurité solide. En effet, il offre de nombreuses options permettant de limiter les accès à un fichier et ainsi éviter des modifications intempestives.

2. Les macros VBA

Visual Basic for Applications est une implémentation du langage Microsoft Visual Basic pour l'ensemble des logiciels de la suite Microsoft Office. Sous Excel, son interface est accessible depuis l'onglet « Développeur » et permettra la programmation de macros.



Les macros sont des fonctions écrites en langage VBA. Elles visent à automatiser des traitements ou des calculs directement intégrés au document. Elles peuvent être soit écrites manuellement, soit résulter de l'enregistrement d'une suite de commande dans l'interface du logiciel et ensuite être traduite automatiquement en code VBA.

3. Pourquoi cette utilisation ?

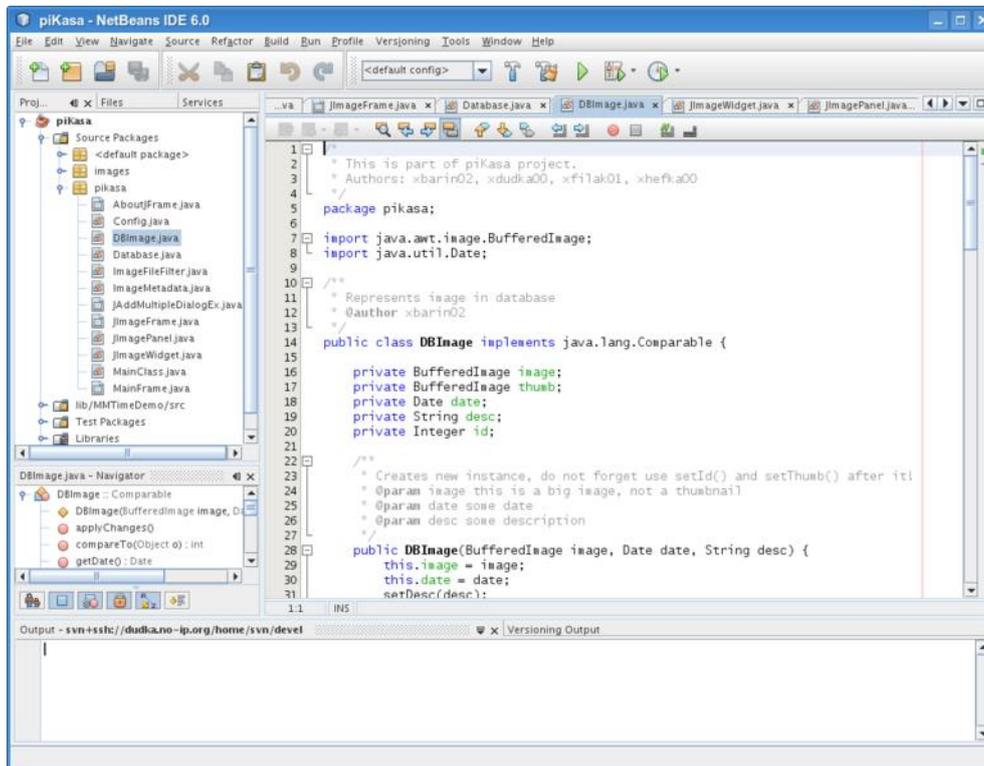
Au-delà de la sécurité proposée par Microsoft Office, Excel est un logiciel très populaire. Les données qu'il permet d'être enregistrées pourront donc être facilement transmises. Bon nombre de données utilisées par l'entreprise sont donc des fichiers Excel, ce qui entraîne naturellement à réaliser des développements en langage VBA.

Le pôle TER Développement et Projets de Midi-Pyrénées s'est lancé il y a quelques années, dans le développement d'une interface Excel permettant l'automatisation de nombreux calculs. Ce logiciel codé en VBA a pour but la simulation de bilans économiques à partir de la projection d'évolutions d'offres. Cet outil vise à devenir un outil d'aide à la décision exportable vers d'autres pôles régionaux de la SNCF.

B) NetBeans et Java

1. L'environnement de développement NetBeans

NetBeans est un Environnement de Développement Intégré, ou EDI, Open Source depuis 2000, mais dont la première version est parue en 1996. Il permet de supporter de nombreux langages tels que Java, C++, Ruby, XML, HTML, JavaScript...



Sa principale utilisation demeure le développement d'applications en Java, en utilisant un Java Development Kit (JDK). Le logiciel a lui-même été conçu en Java. Son environnement visuel est similaire à d'autres IDE comme Eclipse.

2. Le choix du langage Java

a) *Le choix d'un langage « objet »*

La mission reposant sur le développement d'un programme d'optimisation, le langage VBA, peu optimisé et gourmand en ressource ne s'avérait pas approprié. Pour une telle programmation un langage plus flexible et plus rapide à l'exécution était donc nécessaire.

Les langages à programmation orientée objet permettent une meilleure gestion des ressources et sont également bien plus adaptés à la mise en place et l'utilisation d'algorithmes complexes. De plus, l'utilisation d'entités telles que les trains ou les voies de remisage dans le programme correspond tout à fait à la logique « objet ».

b) Communication avec Excel

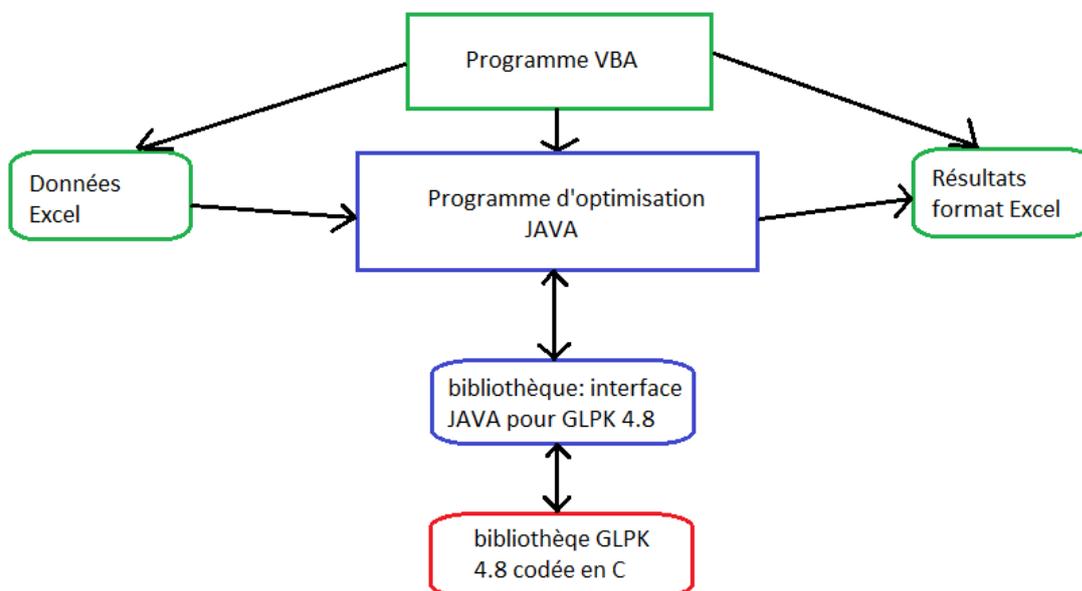
Un des points important ayant joué en faveur du Java est l'existence de bibliothèques faciles d'utilisation et efficaces permettant l'écriture et la lecture de fichiers aux formats Microsoft Excel. Le but final étant la communication avec l'application VBA Simuter, l'utilisation des données à ce format était irrémédiable.

JExcelApi est une bibliothèque Open Source mettant à disposition de nombreuses fonctions pour la lecture et l'écriture de fichiers Excel en langage Java. Ainsi, son utilisation permet au programme Java de charger l'ensemble des données nécessaires à l'optimisation directement depuis les fichiers aux extensions « .xls » ou « .xslm », puis extraire les résultats en fin de calcul.

c) Utilisation de solveur linéaire

Le programme allait utiliser l'optimisation linéaire. Afin de gagner du temps de développement, l'objectif était d'utiliser des bibliothèques de résolution de modèles linéaires déjà existantes. L'une d'entre elles, GLPK, est une bibliothèque libre qu'il est possible de faire communiquer avec Java.

Avec l'aide de l'ouvrage « Programmation Linéaire Avancée » de Gérard Fleury et Philippe Lacomme et l'interface Java pour GLPK 4.8 développée par Frank Björn, l'utilisation de Java a alors définitivement été établie.



C) GLPK utilisé avec GUSEK et Java

1. La bibliothèque GLPK 4.8

GNU Linear Programming Kit (GLPK) est un package pour la résolution de problèmes linéaires de grande échelle. Les problèmes utilisés doivent être sous le format décrit par le langage GMPL (GNU Mathematical Programming Language) pour être directement interprétés.

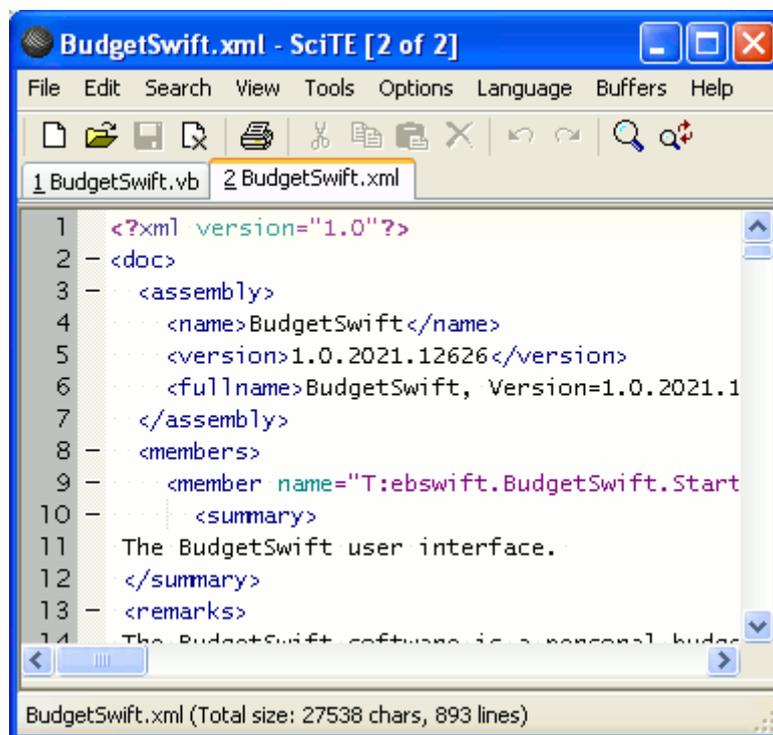
La bibliothèque GLPK 4.8 se présente sous la forme d'une dll. Elle résulte donc de la compilation d'un programme écrit en C, pour un obtenir du code natif directement interprétable par le processeur de la machine.

Les fonctions proposées par la bibliothèque permettent l'interprétation de modèle linéaire puis des algorithmes de résolution. La méthode du simplexe ou du branch-and-bound sont notamment implémentées afin de pouvoir résoudre des problèmes par des solutions décimales ou entières.

2. SciTE et GUSEK

a) *L'environnement de développement SciTE*

SciTE (Scintilla Text Editor) est un éditeur de texte également gratuit et Open Source. Sa version de base permet l'interprétation de nombreux langages, facilitant l'édition du code.



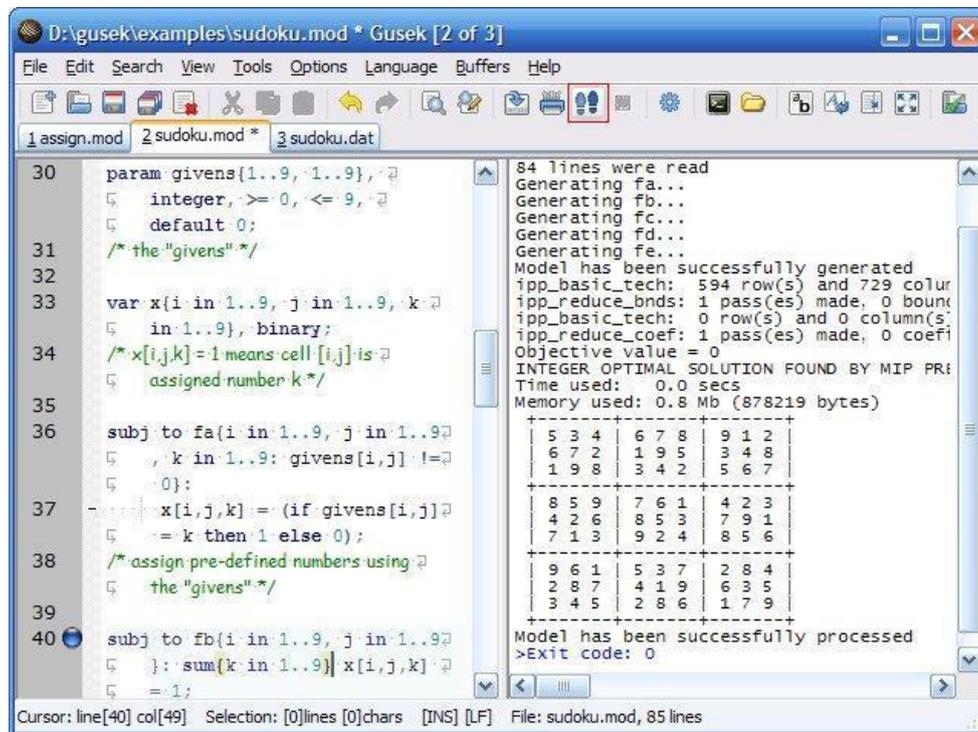
```
1 <?xml version="1.0"?>
2 <doc>
3 <assembly>
4   <name>BudgetSwift</name>
5   <version>1.0.2021.12626</version>
6   <fullname>BudgetSwift, Version=1.0.2021.1
7 </assembly>
8 <members>
9   <member name="T:ebswift.BudgetSwift.Start
10 <summary>
11   The BudgetSwift user interface.
12 </summary>
13 <remarks>
14   The BudgetSwift software is a personal budg
```

BudgetSwift.xml (Total size: 27538 chars, 893 lines)

L'ajout de nombreux plug-ins peut permettre l'interprétation d'autres langages et le lancement d'exécutions ou de compilations.

b) L'extension GUSEK

GUSEK (Glpk Under Scite Extended Kit) est une extension dédiée à ScITE afin de pouvoir interpréter des modèles écrit en langage GMPL et à faire appel à la bibliothèque GLPK comme solveur.



```
30 param givens{1..9, 1..9},
    integer, >= 0, <= 9,
    default 0;
31 /* the "givens" */
32
33 var x{i in 1..9, j in 1..9, k
    in 1..9}, binary;
34 /* x[i,j,k] = 1 means cell [i,j] is
    assigned number k */
35
36 subj to fa{i in 1..9, j in 1..9,
    k in 1..9: givens[i,j] != 0};
37 x[i,j,k] = (if givens[i,j]
    = k then 1 else 0);
38 /* assign pre-defined numbers using
    the "givens" */
39
40 subj to fb{i in 1..9, j in 1..9,
    k in 1..9: sum{k in 1..9} x[i,j,k]
    = 1;
```

```
84 lines were read
Generating fa...
Generating fb...
Generating fc...
Generating fd...
Generating fe...
Model has been successfully generated
ipp_basic_tech: 594 row(s) and 729 colou
ipp_reduce_bnds: 1 pass(es) made, 0 boun
ipp_basic_tech: 0 row(s) and 0 column(s)
ipp_reduce_coef: 1 pass(es) made, 0 coef
Objective value = 0
INTEGER OPTIMAL SOLUTION FOUND BY MIP PR
Time used: 0.0 secs
Memory used: 0.8 Mb (878219 bytes)
-----
 5 3 4 | 6 7 8 | 9 1 2
 6 7 2 | 1 9 5 | 3 4 8
 1 9 8 | 3 4 2 | 5 6 7
-----
 8 5 9 | 7 6 1 | 4 2 3
 4 2 6 | 8 5 3 | 7 9 1
 7 1 3 | 9 2 4 | 8 5 6
-----
 9 6 1 | 5 3 7 | 2 8 4
 2 8 7 | 4 1 9 | 6 3 5
 3 4 5 | 2 8 6 | 1 7 9
-----
Model has been successfully processed
>Exit code: 0
```

Si sa communication difficile avec d'autres programmes le rendent obsolète pour le développement du programme souhaité, son utilisation est très utile pour vérifier la validité d'un modèle linéaire avant de traduire celui-ci dans un autre langage.

3. Utiliser GLPK avec Java

La bibliothèque Java écrite par Frank Björn permet de faire appel aux fonctions natives de la dll GLPK 4.8 en traduisant les données en langage GMPL. Après avoir paramétré le modèle (nom, problème entier ou décimal, temps d'exécution maximal...), les variables de décisions, contraintes et coefficients qui y sont liés doivent être introduit sous la forme d'une matrice. Une fonction objectif peut également être ajoutée au modèle.

Saisir le modèle, fournir les données nécessaires et faire appel au solveur est bien plus complexe en passant par Java malgré tout, que d'utiliser directement le langage GMPL. Il est donc préférable de d'abord réaliser les modèles avec un interpréteur GMPL comme GUSEK avant de les traduire en Java. Enfin, la bibliothèque pour interface Java de GLPK 4.8 ne permet pas l'accès aux fonctions natives de gestion de mémoire. Ceci implique donc une grande vigilance à la gestion des ressources et la création d'objets faisant appel à GLPK, dans le code Java.

III. Travail à réaliser

A) But et description de la mission

Entre les trajets, les trains ne peuvent rester immobiles en gare pour ne pas perturber la circulation. Des lieux, appelés « sites de remisage », sont donc dédiés au stockage des véhicules avant un nouveau départ. L'organisation du rangement de ces trains est réalisée en suivant une semaine « type » décrivant les heures d'arrivée, de départ et les longueurs de matériel à remiser. Le but est d'étudier l'organisation actuelle de ces sites, d'évaluer des besoins futurs aux horizons 2017 et 2023 et de proposer des solutions en optimisant les plans de remisage de Toulouse. Ainsi, des besoins de places supplémentaires seront anticipés, et des demandes de travaux à RFF (Réseau Ferré de France) pourraient être avancées.

Enfin, en parallèle à mon travail, une étude sur la modélisation et l'optimisation du trafic en gare est aussi réalisée. Le but final est de lier ces deux travaux afin de pouvoir réaliser des plans de circulation et de remisage optimums grâce à des outils automatisés.

B) Déroulement de la mission

Dans un premier temps, mon travail consiste à récupérer l'ensemble des données et les formaliser afin de pouvoir réaliser des calculs automatisés grâce à Excel et VBA. Ainsi, des valeurs et des graphiques de saturations au fil peuvent être récupérés pour analyser la situation actuelle. Dans un second temps, ce travail doit être porté sur les prévisions des années 2017 et 2023 où le trafic TGV et le matériel TER auront évolué. Grâce aux calculs automatisés, il est ainsi rapide de vérifier si le plan de remisage actuel est toujours valable.

Il est supposé que ce plan ne sera plus suffisant et que dans tous les cas, celui-ci n'est pas optimisé. Une optimisation doit donc être réalisée pour les plans de 2017 et 2023 afin de voir si des solutions sont possibles avec les capacités actuelles des sites de remisages. Dans ces prévisions, des conditions de robustesse sont à prendre en compte afin de pouvoir gérer les incidents et imprévus. Si des dépassements de capacité ou des manques de robustesse se dégagent, des réorganisations des voies de stockages sont à proposer (allongement de voies ou ajout de voies).

En parallèle à ces calculs de prévisions et d'optimisations, des outils informatiques doivent être développés pour permettre l'automatisation des calculs avec des données différentes dans le futur. Un logiciel (SIMUTER) est actuellement développé en VBA afin de simuler différents processus, comme par exemple le calcul de bilans économiques, la création automatique de plannings de transports optimisés en fonction des catégories de trains... L'intégration des calculs sur le remisage à cette application est envisagée et est souhaité si le temps de travail le permet.

C) Solutions mises en place ou envisagées

Les différentes données n'étant pas centralisées et les notations étant différentes selon leur provenances (entre la gare de Toulouse Matabiau et les sites de remisages notamment), la première étape est de créer des fichiers Excel pour les regrouper. Pour cela des macros VBA sont à utiliser lorsque c'est possible, sinon les données seront réécrites manuellement.

Une fois l'ensemble des données nécessaires regroupées, les calculs sur l'existant et sur les prévisions 2017/2023 en réutilisant le plan 2014 seront à réaliser par des macros VBA. Les résultats seront présentés de manière la plus visuelle possible afin d'être clairs et lisibles par n'importe qui.

L'optimisation des prévisions sera réalisée grâce à des outils de résolution linéaire. Pour cela l'utilisation du logiciel Gusek est envisagée, mais un autre logiciel pourrait le remplacer pour faciliter les communications avec les données Excel. Afin d'utiliser un solveur, un modèle doit être réalisé pour optimiser les organisations de trains à l'intérieur de chacun des sites. Si ceci s'avère insuffisant, un modèle regroupant les différents sites sera alors à son tour construit.

Une fois l'ensemble des optimisations réalisées, des conclusions seront à tirer des résultats. La possibilité de la mise en place du plan obtenu en premier lieu. Pour cela, des critères de robustesse sont à ajouter, mais aussi en mettant les résultats en relation avec les optimisations réalisés au sein de la gare.

Enfin, un outil VBA le plus complet possible, récapitulant le travail à effectuer, sera réaliser pour être intégrer au logiciel SIMUTER.

D) Planning prévisionnel



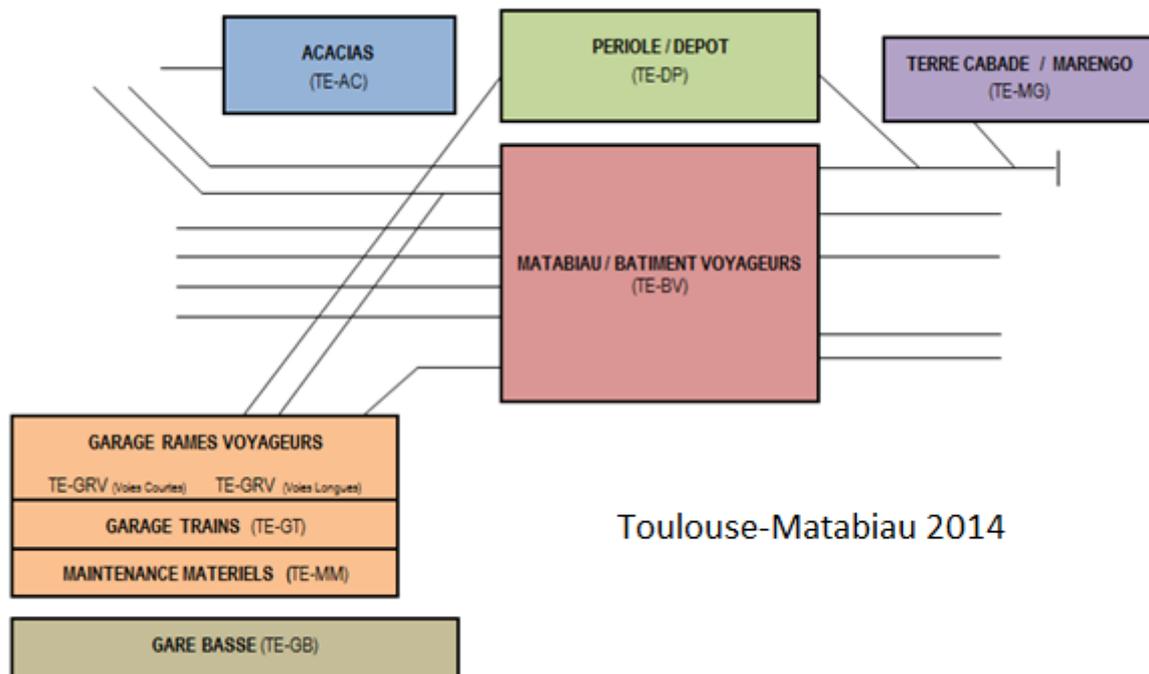
IV. Choix de l'algorithme d'optimisation

A) Evaluation de la taille du problème

1. Définition des objectifs

L'objectif principal de l'outil est de fournir une aide à la décision sur des plans d'investissements sur plusieurs années. Permettre de réaliser des simulations prenant en compte plusieurs sites de remisages, eux même composés de plusieurs voies peut permettre d'évaluer l'impact de travaux importants.

Pour ce qui est de la gare de Toulouse Matabiau, sur laquelle des études devaient être menées parallèlement au développement du module remisage, nous comptons actuellement 6 sites de remisage en service : Acacias (TE-AC), Périole (TE-DP), Terre Cabade (TE-MG), Garage Rames Voyageurs (TE-GRV), Garage Train (TE-GT) et Maintenance Matériels (TE-MM).



Durant la journée, et surtout la nuit, des trains sont garés à l'intérieur de ces 6 sites lorsqu'ils ne sont pas en circulation. La nuit du vendredi au samedi est souvent considérée comme le moment le plus critique de la semaine en remisage. Afin d'évaluer des besoins futurs, les études se réalisent donc souvent sur cette période qui est la plus dimensionnante.

Les études précédemment réalisées ne prenaient en compte que le cumul des longueurs de matériels à remiser en fonction de la somme des longueurs des voies de remisage. Ceci peut fournir un indicateur sur le taux de saturation des infrastructures de remisage, mais est bien insuffisant pour vérifier la validité d'un scénario d'évolution. En effet des contraintes telles que le fait que les trains sont composés d'éléments insécables sont oubliées par ce calcul.

2. Nombre de sites et de voies

Les 6 sites de remisage en activité vu précédemment représentent 27 voies pour un total de 8156m disponibles pour le remisage.

Chantier	Voie	Linéaire	Chantier	Voie	Linéaire	Chantier	Voie	Linéaire
TE-GT	5	525	TE-GRV (L)	142	351	TE-AC	51	118
TE-GT	6	525	TE-GRV (L)	143	355	TE-AC	53	192
TE-GT	8	495	TE-GRV (L)	144	370	TE-AC	55	192
TE-GT	10	472	TE-GRV (L)	145	292	TE-AC	sous total	502
TE-GT	11	530	TE-GRV (L)	146	292			
TE-GT	sous total	2 547	TE-GRV (L)	147	277	Chantier	Voie	Linéaire
			TE-GRV (L)	148	278	TE-DP	30	103
Chantier	Voie	Linéaire	TE-GRV (L)	sous total	2 215	TE-DP	31	100
TE-MM	15	535				TE-DP	32	47
TE-MM	16	520	Chantiers	Nb voies	Linéaire	TE-DP	sous total	250
TE-MM	17	543	TE-GT	5	2 547			
TE-MM	sous total	1 598	TE-GRV (L)	7	2 215	Chantier	Voie	Linéaire
			TE-GRV (C)	3	699	TE-MG	65	120
Chantier	Voie	Linéaire	TE-MM	3	1 598	TE-MG	66	100
TE-GRV (C)	132	225	TE-DP	3	250	TE-MG	67	125
TE-GRV (C)	133	250	TE-MG	3	345	TE-MG	sous total	345
TE-GRV (C)	134	224	TE-AC	3	502			
TE-GRV (C)	sous total	699	TOTAL	27	8 156	TE-GT et TE-GRV disjointes		
						GRV-L et GRV-C individualisés		

Les hypothèses d'évolution à Toulouse Matabiau reposent souvent sur l'ajout de voies voir de sites supplémentaires pour palier à l'augmentation de l'offre des transporteurs ferroviaires. Ce nombre de 27 voies est donc un ordre de grandeur minimum pour les données du modèle à construire. De plus, si l'outil vise à être réutilisable partout, y compris dans des gares encore plus grandes, il est important de faire en sorte que le programme puisse fournir une optimisation pour des problèmes à très grande échelle.

3. Trains et matériels à remiser

a) TER à Toulouse Matabiau

Comme rappelé plus haut, la taille des données du problème pour la gare Toulouse Matabiau servent d'étalonnage à la réalisation du programme pour y réaliser les études les plus précises possibles.

La gare est la plus importante de Midi-Pyrénées. Elle est le centre du réseau, et la quasi-totalité des axes traversant la région traversent cette gare. Un engin, dans la définition ferroviaire, correspond à l'élément insécable composant un train (un automoteur, un wagon, une locomotive...). Les engins TER appartiennent à une région. Midi-Pyrénées possède environ 200 engins, dont 150 sont en remisage la nuit du vendredi au samedi (les engins en circulation, en maintenance ou les locomotives n'étant pas pris en compte).

Sur 150 engins remisés dans la région à la période la plus contraignante, 67 le sont à Toulouse Matabiau, soit 45% du total. Ces 67 engins peuvent être parfois attachés entre eux pour réaliser les circulations. Afin d'éviter des manœuvres inutiles très coûteuses, les trains ne sont pas découpés en remisage et les engins restent si possible attachés. Ces 67 engins représentent au final 40 trains à remiser.

Gare	Engins	Trains
Bordeaux-St-Jean	6	3
Toulouse-Matabiau	67	40
Carcassonne	3	3
Tarbes	2	1
Foix	3	2
Brive-la-Gaillarde	5	3
Montréjeau-Gourdan-Polignan	4	2
Ax-les-Thermes	5	3
Carmaux	11	4
La Souterraine	1	1
Rodez	12	7
Auch	4	3
L'Isle-Jourdain	1	1
Narbonne	1	1
Aurillac	6	4
Latour-de-Carol-Enveitg	6	3
Cahors	1	1
Castres	2	1
Mazamet	3	2
Lourdes	2	1
Pau	5	2

Remisage du parc TER Midi-Pyrénées la nuit du Vendredi au Samedi

b) Prise en compte des autres transporteurs

Aux engins TER, il faut ajouter des trains des deux autres transporteurs que sont les TAGV (TGV et Renfe à Toulouse) et TET (Intercités). Si ils sont moins nombreux que les TER, ils présentent des engins plus longs et donc plus contraignants. Les TAGV sont de plus souvent en unité double (attachés par deux), ce qui en fait des trains de plus de 400m de long.

Site	ENGINS				TRAINS				LINEAIRE			
	TER	TET	TAGV	Total	TER	TET	TAGV	Total	TER	TET	TAGV	Total
TOTAL	67	48	7	116	40	6	5	51	2507,1	1267,2	1625,8	5400,1

C'est donc au total 51 trains qu'il faut actuellement organiser en remisage à la pointe.

B) Réorganisation du problème

1. Fragmentation du problème

Afin de simplifier le problème, il pourrait être supposé que durant la période de pointe qu'est la nuit du Vendredi au Samedi, aucun train de rentre ou ne sort du remisage. Cette hypothèse est proche de la réalité, car les circulations sont quasiment nulles la nuit et aucun train ne part ou ne rentre sur un site de remisage à Toulouse Matabiau actuellement la nuit, entre 2h30 et 5h30. Ceci permettrait d'éliminer les contraintes de manœuvre visant à ce qu'un train soit placé en bout de voie pour pouvoir quitter un site de remisage ou vienne se positionner en bout de voie quand il arrive. En prenant un plan fixe, n'évoluant donc pas dans le temps, ces contraintes d'ordonnement sur les voies n'existent plus.

Si l'on souhaitait modéliser le problème sous une forme linéaire, le nombre de variables de décisions engendrées correspondrait alors au produit du nombre de trains à remiser par le nombre de voie de remisage. Pour réaliser une optimisation du fonctionnement actuel, le problème ne générerait pas moins de 1377 variables de décision (51 trains pour 27 voies). Le nombre de solution possible à un tel problème serait donc de factorielle 1377, environ 2 fois 10 puissance 36 solutions. Dans un tel cas, même des algorithmes tels que le simplexe s'avèreraient inutiles pour trouver une solution dans un temps raisonnable.

La résolution du problème ne pouvait donc se faire par optimisation linéaire directement. La première idée fut donc de fragmenter le problème et de ne procéder à l'optimisation linéaire qu'à l'intérieur d'un site de remisage. La résolution allait ainsi prendre en compte au maximum 7 voies pour le plus grands des sites, et le nombre de train pouvant y être remisé ne pourrait jamais excéder l'ordre de grandeur de la vingtaine.

2. Utilisation de la méthode Tabou

La résolution du problème allait donc pouvoir s'organiser en deux couches d'optimisation : un algorithme proposerait des distributions des trains par sites, puis l'optimisation linéaire permettrait de dire si pour chaque site cette distribution est soluble.

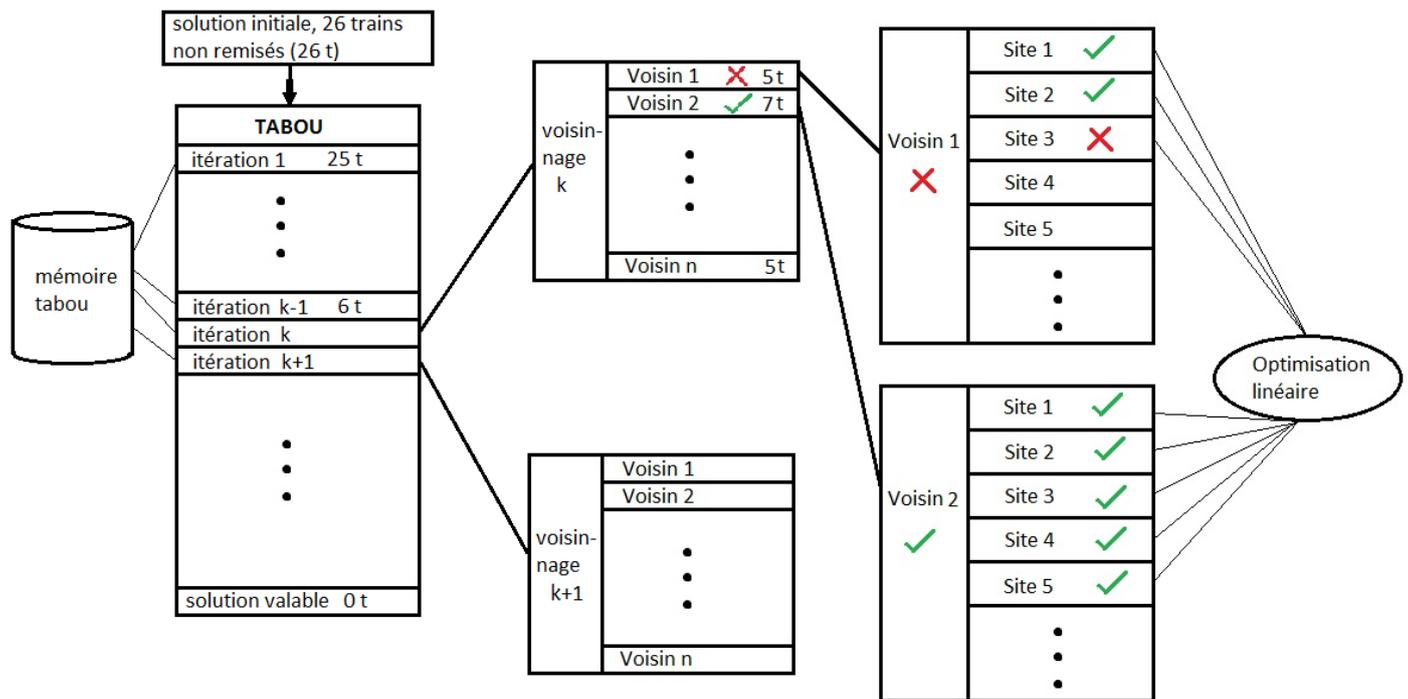
La couche supérieure de l'algorithme devait permettre le parcours dans un espace très grand de solutions, malgré le passage de la distribution par voies à la distribution par sites. Devant prendre en compte l'ensemble des trains à remiser, la recherche de la solution optimale était irréaliste. L'appel à une métaheuristique fut donc choisi afin de trouver une solution approchée de l'optimal mais dans un temps réduit.

La méthode Tabou fut choisie pour deux raisons :

- Son adaptabilité au problème. La méthode est en effet souple et peut être adaptée à énormément de type de problèmes.
- La minimisation des appels au solveur linéaire. L'algorithme ayant la particularité d'avoir deux couches, l'efficacité d'une méthode comme le Recuit-Simulé par exemple aurait été dégradée par un grand nombre d'appels aux modèles linéaires par sites à chaque itération.

C) Optimisation a une heure fixée

L'algorithme pouvait donc être premièrement appliqué à un créneau fixe, ne prenant en compte que les contraintes de capacités des voies par rapport aux longueurs des matériels qu'elles accueillent. Comme évoqué précédemment, pour des études de niveau macro, ce système est tout à fait adapté, car les dimensionnements se font par rapport à la nuit du vendredi au samedi, également période fixe pour le remisage.



- La méthode Tabou correspond à la première couche de l'optimisation. Elle répartie les trains à remiser par sites et essaie à chaque itération de diminuer le nombre de trains non remisés. La mémoire tabou empêche de revisiter une solution récemment explorée et ne pas tomber dans un minimum local.
- L'optimisation linéaire est la seconde couche. A partir de la distribution réalisée par la méthode Tabou, chaque site est optimisé par un modèle linéaire correspondant. Si le solveur GLPK trouve une solution au problème de chaque site, alors la solution voisine testée est réalisable. Tabou choisira ensuite un voisin parmi les solutions validées par l'optimisation linéaire.

La méthode Tabou étant une métaheuristique, tant que tous les trains n'auront pas été remisés par l'algorithme ou tant que celui-ci n'est pas bloqué par une liste vide de voisins possibles, il n'est pas garanti qu'une solution meilleure puisse encore être trouvée.

D) Optimisation sur une période

Si le dimensionnement se réalise surtout sur les périodes de pointe et fixes, la nuit, l'étude de la journée présente des enjeux différents. Le nombre de trains remisés simultanément étant toujours inférieur à celui de la nuit, les capacités de longueurs des voies ne sont plus un problème. Par contre, les manœuvres dues à l'arrivée et aux départs des trains sont bien plus fréquentes.

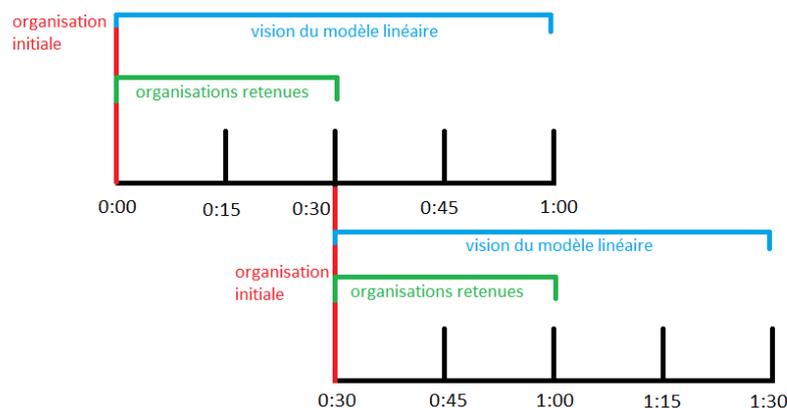
L'objectif d'une optimisation sur la journée n'est plus réellement pour une aide à la décision sur le dimensionnement des voies, mais cherche d'avantage à se rapprocher de la réalité pour fournir un plan de remisage réalisable. Bien plus lourd et complexe au niveau des calculs, cette optimisation se réalise de préférence sur un nombre de sites et de trains plus réduit que la pointe, afin de garantir une solution pertinente et dans un temps de résolution raisonnable.

Les voies de remisage sont utilisées en piles. Le dernier train arrivé sur une voie sera donc le premier à repartir. Si un train n'est pas en bout de voie et doit partir, alors les trains le bloquant devront être déplacés sur une autre voie de remisage afin de lui dégager la sortie.



Une optimisation similaire à celle de la pointe a été choisie. Un nouveau modèle linéaire plus complexe, prenant en compte l'ordre sur les voies, et rajoutant la dimension de temps permettrait l'optimisation dans un site sur une période donnée. Pour cela, le temps devait être discrétisé, et le choix se porta sur un arrondi au quart d'heure.

Une fois une période optimisée en réitérant l'algorithme à double-couche Tabou/modèle linéaire, les résultats seraient gardés pour pouvoir optimiser la période suivante. En partant ainsi de l'organisation optimisée obtenue à une heure fixée (de préférence à la pointe), les périodes suivant et précédant ce créneau pourraient être optimisées progressivement. Pour ne pas proposer des problèmes linéaires trop complexes, mais laisser tout de même une certaine liberté à l'optimisation linéaire pour l'ordonnancement des trains au fil du temps, la durée d'une période fut fixée à 1 heure (ou 4 créneaux). De plus, seule l'organisation la première demi-heure de chaque période ne serait retenue pour pouvoir anticiper les manœuvres à réaliser. L'optimisation avance ainsi par pas de demi-heures bien que les modèles prennent en compte une période d'une heure.



V. Description technique du programme

A) Objectifs et méthodologie générale

L'outil se décompose en deux parties essentielles :

- L'optimisation : recherche d'une solution optimale pour trouver une organisation au remisage de tous les trains fournis dans les données d'entrées. Cette partie de l'application est développée en langage Java (OptimisationRemisage.jar), utilisable simplement sur système 32bits.

- La représentation graphique : afin de pouvoir visualiser les résultats obtenus, des graphiques représentatifs des voies sont générés en VBA dans de nouveaux classeurs Excel.

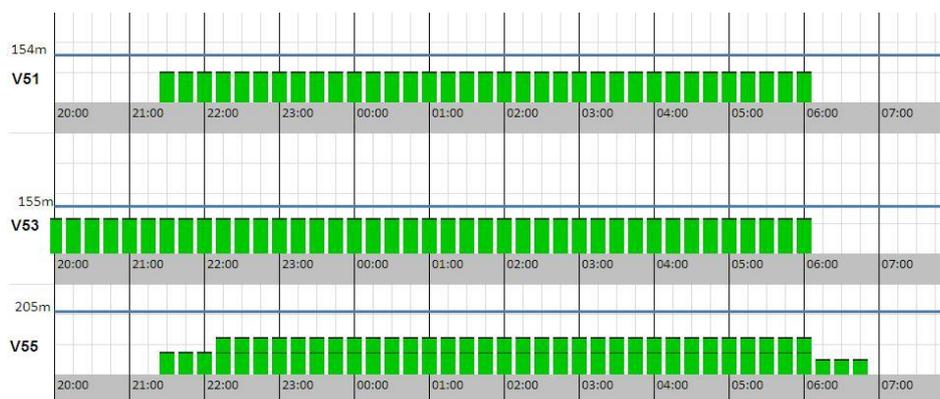
L'objectif du programme d'optimisation est de récupérer une liste de trains à remiser et de trouver une solution pour stationner l'ensemble de ces trains dans les paramètres d'infrastructures fournis, tout en limitant le nombre de manœuvres.

Données d'entrée :

NumTrain	Longueur	JourA	HeureA	JourD	HeureD	Série	Site	Electrique
9726	200.2	Vendredi	22:10	Samedi	06:43	TGV RESEAU		oui
8701	400.4	Samedi	11:59	Samedi	12:39	TGV RESEAU	TE-GT-RAY	oui
8703	400.4	Samedi	13:59	Samedi	14:39	TGV RESEAU	TE-GT-RAY	oui
871847	50.2	Vendredi	21:58	Samedi	16:39	Z7300		oui
14148	110.1	Vendredi	20:00	Samedi	10:05	ZPPg		oui
14142	220.2	Samedi	08:30	Samedi	12:12	ZPPg		oui
778239	95.5	Vendredi	22:45	Samedi	13:00	RIO 4C		oui

Résultats :

NumTrain	Série	NumVoie	HeureA	HeureD	Longueur	Site
8707	TGV RESEAU	5	23:59	18:24	200,2	TE-GT-RAY
8705	TGV RESEAU	8	21:24	7:24	400,4	TE-GT-RAY
8515	TGV RESEAU	11	22:59	5:24	400,4	TE-GT-RAY
6816	TGV RESEAU	10	22:47	5:27	200,2	TE-GT-RAY
3837	ZPPg	6	0:15	6:55	110,1	TE-GT-RAY
871420	PHDz	8	21:01	6:40	83	TE-GT-RAY



B) Code JAVA

1. Les classes du programme :

L'ensemble de la partie « optimisation » du programme est codée en Java. Ce code se décompose en 19 classes Java qui peuvent être classées en 3 groupes :

a) Gestion des données et résultats (5 classes)

. **ParamGlobaux** : Cette classe sera interrogée tout au long du programme Java afin de paramétrer les différents calculs. Elle possède de nombreux attributs publics (accessibles par les autres classes) qui une fois initialisés ne seront plus modifiés au cours du programme. Par exemple, le nombre de créneaux dans une semaine (672, car l'unité de temps choisie est le $\frac{1}{4}$ d'heure), le temps de résolution maximum accordé à un problème linéaire... Certains de ces paramètres sont fixés après la lecture des fichiers d'entrées du programme (voir description des classes FichierVoies et FichierEntree) comme le nombre de sites de remisage, la liste des trains fournie en donnée...

A son appel, la classe crée un site de remisage fictif permettant d'accueillir les trains pour lesquels l'algorithme n'a pas encore trouvé de voie d'affectation. Le but de l'algorithme sera au fur et à mesure de vider ce site fictif au profit des autres sites de remisages réels.

Enfin, cette classe sert également à initialiser les fenêtres Java du module, décrivant l'avancement de l'algorithme. De même, ces fenêtres sont des attributs publics de la classe, permettant ainsi aux autres classes de les modifier au fil du programme.

. **Calendrier** : Afin de faire fonctionner l'algorithme, les horaires doivent être discrétisés. Pour cela, chaque horaire sera arrondi au $\frac{1}{4}$ d'heure précédent. Ainsi, un train arrivant en remisage à 15h07 le Mardi sera considéré remisé à partir de 15h00.

Le calendrier fait donc correspondre un horaire de la semaine à un créneau identifié par un nombre entier. Le calendrier débute le dimanche à 20h00 et se termine le dimanche 19:59. Par exemple, « Dimanche 20:00 » correspond au créneau « 0 », « Lundi 01:32 » au créneau « 22 »... jusqu'au créneau 671.

La classe calendrier dispose donc de méthode permettant de récupérer l'heure ou le jour correspondant à un numéro de créneau, ou inversement donner un numéro de créneau en indiquant le jour et l'heure. Notons qu'un jour peut être écrit de deux façons, soit en toute lettre (ex : Lundi) ou avec les deux premières lettres en majuscules (ex : LU).

. **FichierEntree** : L'appel à cette classe a lieu en début d'algorithme afin de lire le contenu du fichier listant les trains à remisage et de permettre l'initialisation de la classe ParamGlobaux. Ce fichier étant un tableau Excel se présentant de la manière suivante :

NumTrain	Longueur	JourA	HeureA	JourD	HeureD	Série	Site	Electrique
8701	400.4	Samedi	11:59	Samedi	12:39	TGV RESEAU	TE-GT-RAY	oui
14148	110.1	Vendredi	20:00	Samedi	10:05	ZPPg		oui
778239	95.5	Vendredi	22:45	Samedi	13:00	RIO 4C		oui
872616	73	Vendredi	20:16	Vendredi	20:35	PPMz		oui

Après lecture, le nombre de trains total et la liste de ceux-ci sont stockés par la classe ParamGlobaux. L'indication d'un site dans la colonne dédiée permettra de fixer le train sur un site et empêcher l'algorithme de le placer ailleurs.

Cette classe donc a pour seul rôle la lecture de ce fichier Excel en utilisant la bibliothèque JExcelApi compressée dans le fichier « jxl.jar ».

FichierVoies : Tout comme FichierEntree, cette classe a pour simple but la lecture d'un fichier Excel afin d'initialiser ParamGlobaux. Le fichier concernant cette classe se présente sous la forme d'un classeur répertoriant les voies et caractéristiques d'un site par feuille Excel sous le format suivant :

Désignation	Longueur (m)	Distance Sécurité
142	351	6
143	355	Electrique
144	370	oui
145	292	
146	292	

Le nom de la feuille Excel correspond lui au nom du site et servira à l'identifier tout au long du programme.

FichierSortie : Utilisant également la bibliothèque JExcelApi, cette classe ne sert elle pas à la lecture mais à l'écriture d'un fichier Excel. Deux appels à cette classe ont lieu dans l'algorithme :

- Ecriture du fichier d'affectation des trains en période de pointe et leurs ordres sur les voies
- Ecriture du fichier de sortie qui sera ensuite mis en forme par VBA pour être visible par l'utilisateur

Numéro	Série	Voie	Arrivée	Départ		Longueur		Site
3837	ZPPg	143	06:30	6:55		110,1		TE-GT-GRV
14148	ZPPg	147	07:15	10:05		200		TE-GT-GRV
871400	PPMz	15	8:01	8:34		146		TE-MM
14148	ZPPg	143	20:00	07:15		200		TE-GT-GRV
870606	Z27500 3C	17	7:56	9:16		114,8		TE-MM
71206	X72500	16	07:50	12:40		107,2		TE-MM

Dans ce fichier écrit, nous retrouvons les trains traités dans le calcul (trains fournis par le fichier d'entrée et présents dans la période étudiée) et les résultats qui y sont associés. Un train du fichier d'entrée peut être retrouvé sur plusieurs lignes si celui-ci est déplacé pendant sa période de remisage et donc affecté à plusieurs voies (exemple du 14148 dans ce tableau).

b) Classes pour gestion d'objets (6 classes)

Train : Liste de ses attributs :

- id : son identifiant dans le fichier d'entrée (nombre entier).
- numCirculation : numéro de circulation stocké dans le fichier d'entrée (chaîne de caractères).
- longueur : longueur récupérée dans le fichier d'entrée (nombre décimal).
- site : nom du site sur lequel le train est placé (chaîne de caractères).
- estElectrique : indique si le train est purement électrique ou nom, cette propriété permettra d'indiquer s'il peut se positionner sur un site non électrifié (booléen).
- tabPresence : pour chaque créneau de la semaine, le train est-il présent ou non en remisage (tableau de booléen).
- estLocalise : vrai si le train est fixé sur un site, faux si le train doit encore être placé (booléen).
- série : série du train récupérée dans le fichier d'entrée (chaîne de caractères).
- jourA : jour d'arrivée en remisage, spécifié dans le fichier d'entrée (chaîne de caractères).
- heureA : heure d'arrivée en remisage, spécifié dans le fichier d'entrée (chaîne de caractères).
- jourD : jour de départ du remisage, spécifié dans le fichier d'entrée (chaîne de caractères).
- heureD : heure de départ du remisage, spécifiée dans le fichier d'entrée (chaîne de caractères).
- ordres : pour chaque créneau l'ordre sur la voie du train. 0 si non présent en remisage, 1 si en bout de voie, 2 si un seul train le bloque, 3 si 2 trains sont devant lui... (tableau d'entiers).
- nomVoie : nom de la voie sur lequel le train est remisé (chaîne de caractères).

Un train peut être identifié par son identifiant (id) croisé avec son heure d'arrivée en remisage (heureA, jourA) et son heure de départ (heureD, jourD). Ces 3 attributs garantissent l'unicité d'un train. Des trains avec un même identifiant montrent qu'il s'agit du même matériel (train dans le fichier d'entrée) mais qui a un moment a été déplacé de voie. Cette séparation est créée afin de réaliser le fichier de sortie nécessaire au tracé des graphiques.

VoieRemisage : Liste des attributs :

- nom : nom/numéro de la voie, unique par rapport aux autres voies de remisage (une non unicité peut créer des problèmes lors du tracé des graphiques VBA)(chaîne de caractères).
- longueur : distance utile au remisage (nombre décimal)

SiteRemisage : Liste des attributs :

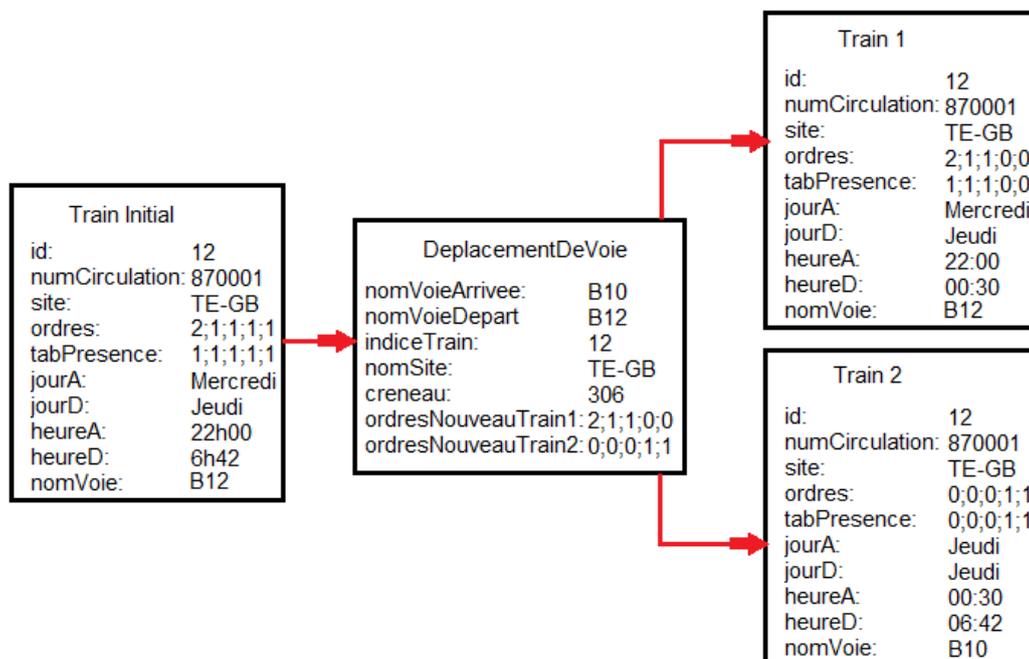
- nom : nom du site unique (chaîne de caractères).
- tabVoies : voies de remisages du site (tableau de VoieRemisage).
- nbVoies : nombre de voies sur le site (nombre entier).
- electrique : si le site est électrifié ou non (booléen).
- distanceSecurite : distance à respecter entre chaque train en mètre (nombre entier).

La classe possède des méthodes donnant par exemple le linéaire total disponible sur le site ou la longueur de sa voie la plus courte. Ces méthodes sont notamment utiles pour l'heuristique d'initialisation (voir classe HeuristiqueIni).

DeplacementDeVoie : Liste des attributs :

- indiceTrain : indice du train dans la liste initiale des trains obtenus dans le fichier d'entrée (nombre entier).
- nomVoieDepart : nom de la voie du train avant déplacement (chaîne de caractères).
- nomVoieArrivee : nom de la voie après déplacement (chaîne de caractères).
- creneau : creneau durant lequel a lieu le déplacement (nombre entier).
- ordresNouveauTrain1 : ordres sur la voie d'origine jusqu'au déplacement (tableau d'entiers).
- ordresNouveauTrain2 : ordres sur la voie d'arrivée depuis le déplacement (tableau d'entiers).

Un déplacement est un objet servant à la création de deux objets « Train » à partir d'un autre objet « Train ». Ainsi, un des deux objets créé aura pour attributs, la voie de remisage, les ordres sur la voie, l'heure d'arrivée en remisage du « Train » initial, mais prendra comme heure de départ, l'heure du déplacement. Le second objet prendra lui les attributs après déplacement (ordres, numéro de voie...) et aura donc la même heure de départ que le « Train » initial, mais aura pour créneau d'arrivée, le créneau auquel a lieu le déplacement.



EntreeModeleIni : Un objet de la classe sert à définir les paramètres d'entrées d'un modèle linéaire « ModeleLinIni ». Son rôle est d'éviter de rappeler le solveur linéaire pour un problème déjà résolu précédemment. Afin de définir un problème appelé par la classe « ModeleLinIni », il y a besoin de connaître le nombre de trains pris en compte, leurs longueurs et le site étudié. Un objet « EntreeModeleIni » mémoriserait ensuite la solvabilité du problème. Ainsi, si pour un même site, on souhaite étudier la capacité d'accueil de deux listes de trains de même taille, avec des trains de même longueur, cette classe permettra un appel supplémentaire au solveur et donc un gain de temps de calcul.

EntreeModeleLineaire : Similaire à la classe précédente, son rôle est de limiter les appels au solveur par la classe « ModeleLineaire ». Pour cela, un objet « EntreeModeleLineaire » a besoin de connaître le site concerné et la liste des trains associés. Les longueurs ne sont ici plus suffisantes, étant donné que le modèle linéaire prend en compte une période d'une heure. Ainsi, si les longueurs entre deux trains peuvent être identiques, leurs créneaux de présence en remisage ne le sont pas forcément.

Un objet de la classe « EntreeModeleLineaire » mémoriserait les résultats principaux obtenus par le solveur, à savoir la solvabilité ou non du problème, et si il est soluble, le nombre de déplacements de trains engendrés.

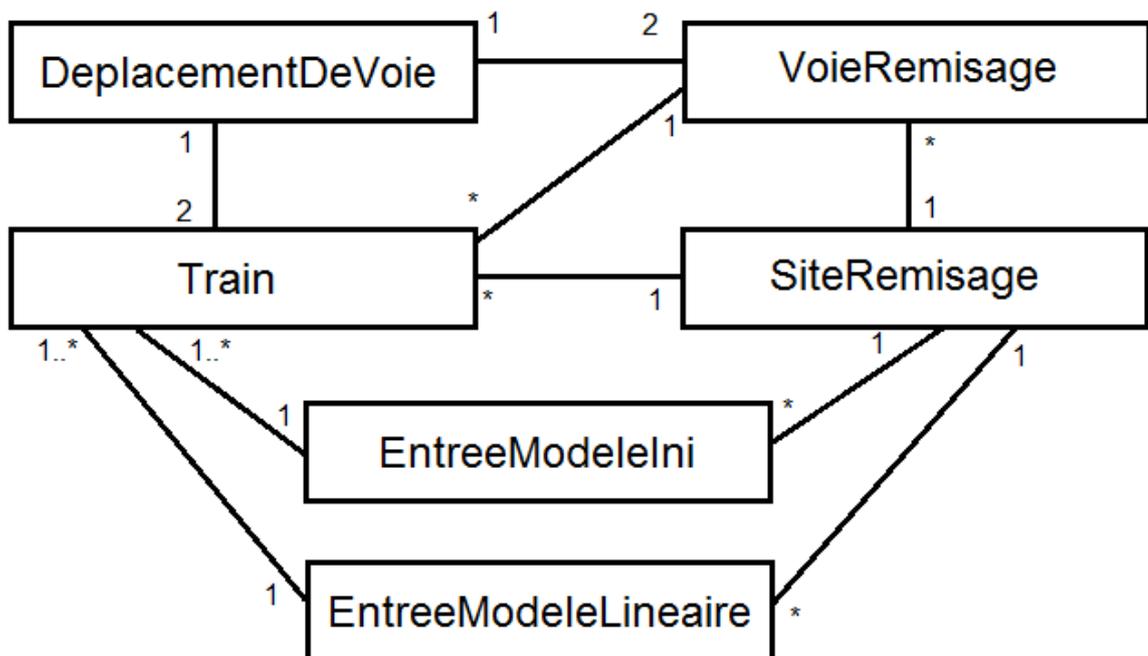


Diagramme de classes (classes pour la gestion d'objets)

c) Classes pour algorithme de calcul (8 classes)

OptimisationRemisage : Il s'agit de la classe principale du programme (classe incluant la fonction « main »). Son rôle est tout d'abord de récupérer les paramètres passés lors de l'appel au programme (noms des fichiers et périodes étudiées) puis ensuite de lancer l'algorithme.

Après avoir chargé les paramètres par l'appel à la classe « ParamGlobaux », l'algorithme de la période de pointe est lancé avec la classe « AlgoInitialisation » puis ses résultats sont récupérés. A partir de ses résultats, les périodes demandées pourront être optimisées avec la classe « AlgoPeriodes ».

Si aucune solution valable n'est trouvée, l'algorithme est relancé afin de chercher une solution avec une autre organisation à déterminer en période de pointe.

AlgoInitialisation : Cette classe a pour but d'organiser l'algorithme d'optimisation du créneau de pointe afin de trouver une organisation du remisage satisfaisante et de retourner les résultats obtenus. Etant donné que cette optimisation ne s'effectue qu'à un seul créneau, les déplacements de matériels ne sont pas pris en compte et seule l'occupation linéaire est ici contraignante. Afin de pouvoir ensuite optimiser les périodes autour du créneau de pointe, les ordres des trains sont déterminés selon le temps depuis leur arrivée en remisage ou avant leur départ et le créneau de pointe. Ainsi, plus un train est près de son heure de départ ou d'arrivée à ce créneau de pointe, plus celui-ci sera positionné au bout de la voie afin de faciliter son insertion ou sa sortie de la voie.

HeuristiqueIni : Afin de gagner du temps de calcul, cette heuristique a pour but de positionner un certain nombre de trains sans appel au solveur linéaire. Les méthodes de la classe permettent de placer tout d'abord les trains pour lesquels un site est indiqué dans le fichier d'entrée.

Les sites se voient ensuite affecter les trains les plus longs possibles jusqu'à 50% de leur capacité linéaire.

MetaheuristiqueIni : Après avoir appelé l'heuristique d'initialisation, la métaheuristique a pour but de positionner l'ensemble des trains restants. Pour cela la méthode Tabou est utilisée. Son but est de réaliser des itérations en cherchant à diminuer le nombre de trains non remisés prioritairement, et la longueur de ces trains remisés ensuite. Voici la décomposition des étapes d'une itération :

- Construction d'un voisinage à l'organisation actuelle : une solution voisine à une organisation a pour simple différence l'affectation d'un train de plus au site fictif ou un train de moins. Lors de la construction d'un voisinage, toute bascule d'un train du site fictif vers un autre site ou à l'inverse, toute bascule de l'affectation d'un train d'un site réel vers le site fictif constitue potentiellement une solution voisine. Cependant, pour chaque de ces voisins potentiel, le modèle linéaire « ModeleLinIni » est appelé afin de vérifier que le voisin construit est bel et bien soluble, sinon celui-ci n'est pas inclus dans le voisinage.
- Suppression de la mémoire Tabou : Afin d'éviter de boucler sur un même ensemble de solution et rester bloquer dans un minimum local, les dernières solutions parcourues sont mémorisées et éliminées du voisinage. La taille de cette mémoire est fixée dans « ParamGlobaux ».

- Choix d'une des meilleures solutions voisines : Après avoir filtré le voisinage, une solution peut être choisie. La solution proposant le moins d'affectations de trains au site fictif est définie comme meilleure solution. Si plusieurs solutions sont à égalité dans le voisinage, alors la longueur totale des matériels sur le site fictif est prise en compte. Plus cette longueur est courte, meilleure est la solution. Enfin, si l'égalité subsiste toujours, une des meilleures solutions est choisie au hasard.
- Préparation de l'itération suivante : Si la solution retenue ne permet pas encore de remiser l'ensemble des trains (au moins un train sur le site fictif), alors le travail de la méthode Tabou n'est pas terminé. La solution retenue est alors ajoutée à la mémoire taboue et une nouvelle itération est appelée en construisant un voisinage à partir de cette nouvelle solution.

La métaheuristique ne s'arrête pas tant qu'une solution où tous les trains sont remisés n'est pas trouvée. Si la résolution est impossible, le programme pourra alors boucler à l'infini. Il est également important de vérifier que la mémoire taboue ne bloque pas l'algorithme en éliminant l'ensemble des solutions du voisinage. La part d'aléatoire dans l'algorithme implique que la méthode ne retournera quasiment jamais deux solutions identiques.

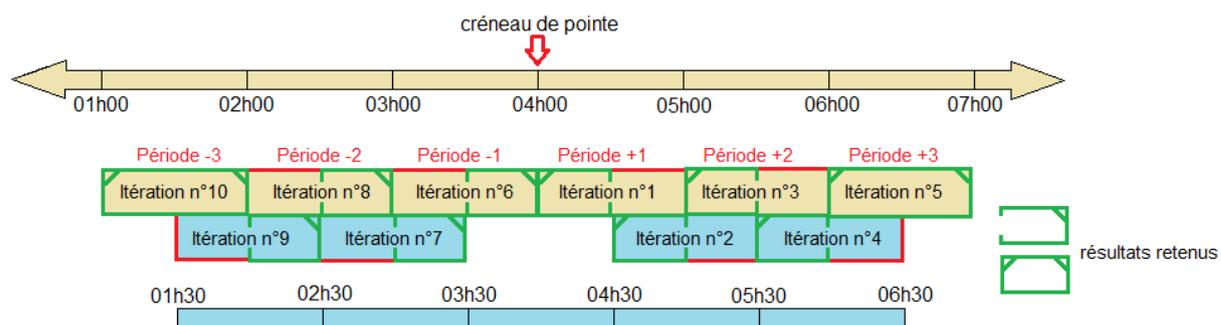
ModeleLinIni : Cette classe sert à l'appel du solveur linéaire GLPK. Le modèle GUSEK traduit est décrit dans l'annexe « Modèle linéaire : créneau de pointe » et le chapitre « Optimisation du créneau de pointe ».

Lors d'une itération, les organisations voisines sont testées afin de savoir si elles constituent des solutions valables. Pour cela, le solveur linéaire est appelé pour chaque des sites pour chaque organisation testée. Les modèles linéaires sont générés en fonction des affectations. Un temps maximum de résolution linéaire est fixé par la classe « ParamGlobaux ». Si pour l'un des sites, le solveur indique qu'aucune solution n'est possible obsolète ou si le temps de résolution est dépassé, alors cette organisation est invalidée.

Après l'appel à la classe « ModeleLinIni », les paramètres sont sauvegardés dans un objet de la classe « EntreeModeleIni » afin de minimiser les appels au solveur.

AlgoPeriodes : A partir de l'organisation obtenue au créneau de pointe, cet algorithme va dans un premier temps essayer de trouver une solution aux périodes postérieures à ce créneau puis aux périodes antérieures.

Pour chaque période, l'algorithme fera appel à la classe « MetageuristiquePeriodes » afin de déterminer si une solution est possible. Si une solution est dégagée, les résultats sont récupérés et réutilisés pour le calcul de la période suivante. Ces résultats sont prélevés à la moitié de la période afin de permettre un lissage de la résolution et ainsi diminuer les chances de bloquer l'algorithme :



MetaheuristiquePeriodes : Tout comme la classe « MetaheuristiqueIni », cette classe utilise la méthode Tabou avec la même forme de voisinage. Elle va elle essayer de placer l'ensemble des trains non remisés à la période optimisée précédemment. Pour cela elle fera appel au solveur GLPK par la classe « ModeleLineaire ».

Contrairement à la métaheuristique précédente, celle-ci est limitée à un nombre maximum d'itération. Si son algorithme est bloqué par un voisinage nul ou par un trop grand nombre d'itération, alors elle signalera l'absence de solution et permettra de redémarrer l'algorithme au créneau de pointe afin d'essayer une autre organisation.

ModeleLineaire : Cette classe fait appel au solveur GLPK en utilisant le modèle GUSEK traduit, décrit en Annexe « Modèle linéaire : optimisation période » et dans le chapitre « Optimisation d'une période ». Ce modèle a pour but de trouver une organisation de remisage par voies dans un site au cours d'une période. Ainsi, contrairement au modèle linéaire utilisé par la classe « ModeleLinIni », des contraintes liées à l'évolution au fil du temps sont présentes. En plus de se contenter de trouver une solution valable, le modèle possède une fonction objectif visant à minimiser les changements de voies pour chacun des trains.

Nous noterons notamment une limitation du déplacement entre chaque créneau horaire. Ainsi, sur une même voie, entre deux créneaux (par défaut un délai d'1/4 d'heure), un seul départ et une seule arrivée seront autorisés (soit deux déplacements). Cette contrainte vise à vérifier la faisabilité des manœuvres.

2. Déroulement du programme

a) Initialisation et paramétrage :

La partie Java du programme commence avec l'appel à la méthode « main » de la classe « OptimisationRemisage ».

Les arguments passés transmis par VBA sont alors récupérés : chemin du fichier listant les trains, chemin du fichier listant les sites et voies, numéro du créneau de pointe, nombre de périodes à optimiser avant le créneau de pointe et nombre de périodes à optimiser après.

Une fois ces premiers paramètres chargés, l'algorithme peut être lancé une première fois en faisant appel à la classe « ParamGlobaux ».

La classe va tout d'abord fixer le nombre de créneaux dans la semaine (672 si l'on fonctionne au ¼ d'heure), la taille en créneaux d'une période (4 si l'on prend une heure par période), la mémoire des métaheuristiques maximum sur les modèles linéaires déjà traités et le temps de résolution maximum accordé à un modèle linéaire (afin d'éviter au programme de bloquer sur un problème trop complexe). Puis, les appels aux classes « FichierEntree » et « FichierVoies » ont lieu, afin de récupérer les données des fichiers d'entrées et fixer les autres paramètres. La liste des sites et voies, des trains initiaux peuvent entre autre être construites, mais aussi des paramètres comme la taille de la mémoire tabou des métaheuristiques qui est fixée selon le nombre de trains et de sites fournis (plus les données sont grandes, plus la mémoire tabou sera grande).

Enfin, la classe « ParamGlobaux » initialise les fenêtres d'affichages décrivant l'état et l'avancement de l'optimisation et ajoute un site fictif à la liste des sites de remisages avec des longueurs de voies très grandes afin d'y positionner temporairement les trains non placés par l'algorithme.

b) Optimisation de la période de pointe :

La classe « HeuristiqueIni » récupère dans un premier temps l'ensemble des trains présents au créneau de pointe. Ceux dont le site est indiqué dans le fichier d'entrée sont les premiers positionnés et sont « bloqués » afin de ne pas être déplacés au cours de l'optimisation. Les trains restants sont ensuite triés du plus long au plus court et l'algorithme tente ensuite de les positionner sur les sites, en vérifiant qu'il existe des voies assez longues pour les accueillir et en ne dépassant pas une saturation supérieure à 50% du linéaire disponible par site. Les trains non positionnés à l'issue de cette heuristique sont eux affectés au site fictif.

Cette première organisation est transmise à la métaheuristique Tabou développée dans la classe « MetaheuristiqueIni » pour parvenir à déplacer tous les trains affectés au site fictif vers les autres sites. Une métaheuristique ne permet pas de savoir si le minimum global est atteint, et donc les itérations de la méthode Tabou continueront tant qu'il y aura au moins un train sur le site fictif. Afin d'évaluer un éventuel manque de linéaire, il est donc nécessaire de laisser l'algorithme tourner plusieurs heures pour garantir cette hypothèse.

Afin de rendre l'optimisation de ce créneau de pointe robuste, des trains fictifs peuvent être ajoutés au fichier d'entrée avec le numéro de circulation « 0 ». Ces trains seront ensuite supprimés de l'algorithme pour l'optimisation des périodes. Si leur ajout peut s'avérer contraignant pour le créneau de pointe, il garantit en contre-parti une plus grande flexibilité pour la suite de l'optimisation.

Une fois l'ensemble des trains positionnés au créneau de pointe, la voie de chacun est récupéré afin de déterminer l'ordre dans lequel les disposer. Si plusieurs trains sont présents sur une même voie, il est préférable de placer en bout de voie celui qui est le plus proche d'un départ ou qui vient juste d'arriver. Ainsi, pour chaque train le nombre de créneaux écoulés depuis son arrivée jusqu'à la pointe et le nombre de créneaux avant son départ sont calculés. Le minimum des deux résultats obtenus est conservé. Plus ce minimum sera petit, plus le train sera positionné proche de la sortie de la voie. Ainsi si trois trains sont présents sur une même voie, ils se verront chacun affecter un ordre entre 1 et 3 selon le calcul réalisé. Plus le nombre entier donné est petit, plus le train est positionné près de la sortie de la voie (ex : un train d'ordre 2 est séparé du bout de la voie par 1 train).

c) Ecriture d'un premier fichier de résultat :

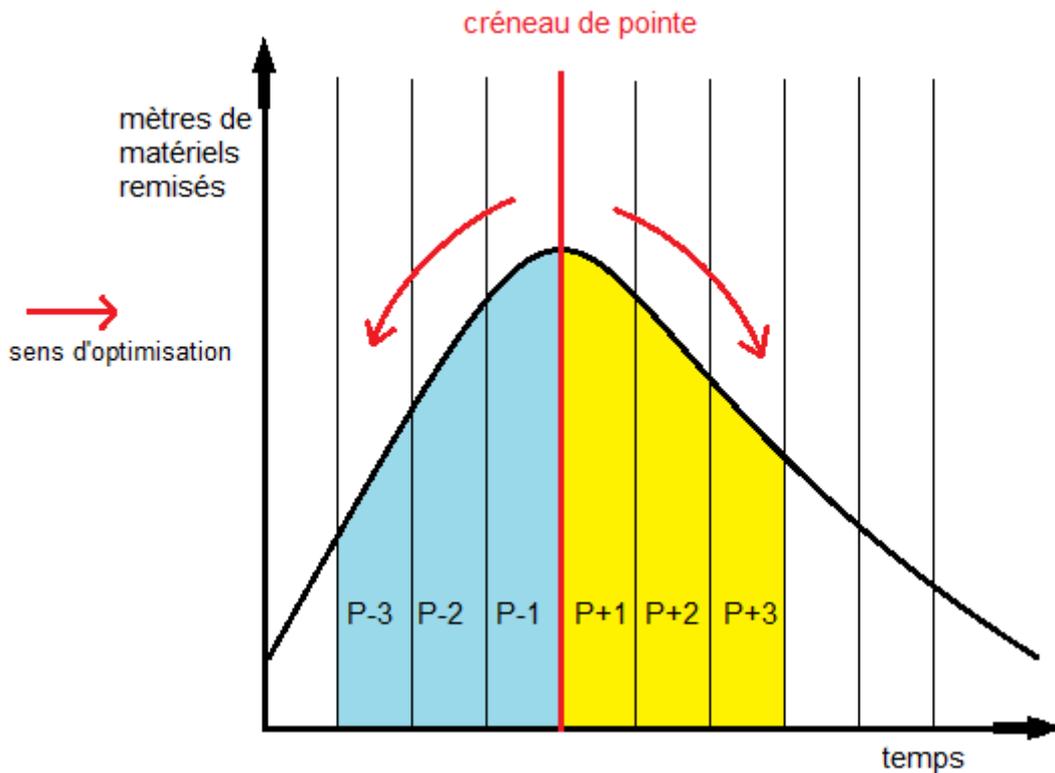
Un premier fichier de résultat est créé après l'optimisation de la pointe. Celui-ci s'appelle « ficResultatIni ». Il n'est normalement pas affichable par l'utilisateur à moins d'aller l'ouvrir manuellement.

Num Circul	Série	Voie	HeureA	HeureD			Longueur		Site	Ordre
3643	ZPPg	16	20:55	5:53			400		TE-MM	1
3663	ZPPg	17	2:37	15:55			400		TE-MM	1
70407	PPMb	15	20:00	8:24			146		TE-MM	4
72402	B81500	17	20:00	7:45			114,8		TE-MM	2
70303	Z27500	15	22:10	7:26			114,8		TE-MM	3
778239	RIO 4C	16	22:45	13:00			95,5		TE-MM	2
870312	X73500	15	20:00	19:59			87		TE-MM	5
871420	PHDz	15	21:01	6:40			83		TE-MM	2
70602	Z27500 4C	15	21:30	6:07			73		TE-MM	1
14150	ZPPg	145	21:42	8:16			200		TE-GT-GRV	2
14148	ZPPg	143	20:00	10:05			200		TE-GT-GRV	2
3837	ZPPg	147	0:15	6:55			110,1		TE-GT-GRV	1
78010	Z7300	146	21:20	6:07			100,4		TE-GT-GRV	1

Tous les trains présents en période de pointes sont récapitulés dans ce fichier avec des informations sous le même format que le fichier de sortie final. Seul leur ordre de positionnement au créneau de pointe est ici rajouté.

d) Optimisation des périodes :

La solution obtenue au créneau de pointe est récupérée afin de déterminer l'organisation des périodes alentours.



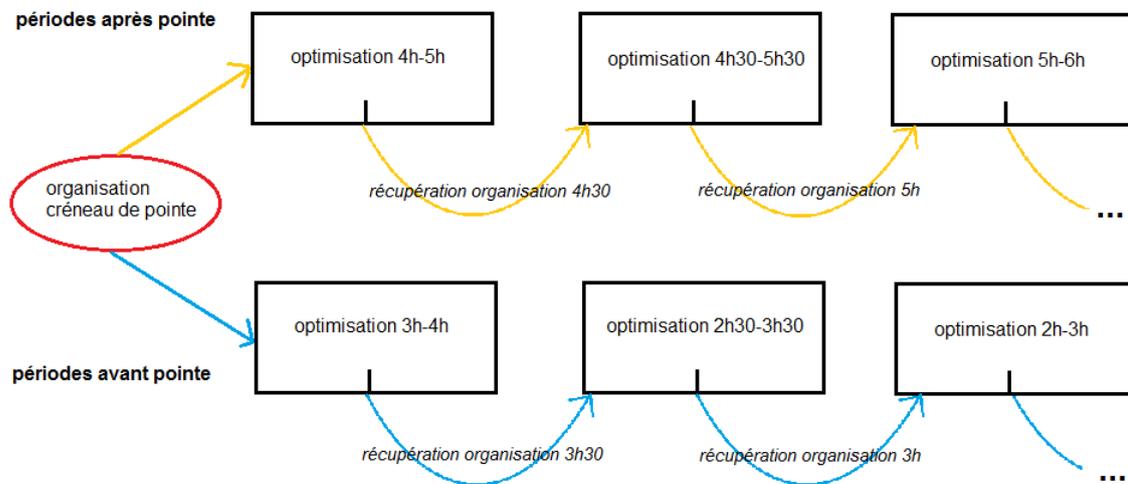
La classe « AlgoPériodes » va tout d'abord optimiser la période notée « P+1 » sur le schéma en partant des positionnements du créneau de pointe. Puis comme expliqué dans la description de la classe « AlgoPériodes » et le schéma correspondant, l'algorithme va récupérer les résultats obtenus dans la première moitié de cette période pour réitérer l'optimisation une demi période après.

Avancer par pas de demi-période permet une meilleure flexibilité de l'optimisation. Par exemple, un train partant à 15h15 et polacé en 3^{ème} position sur la voie ne pourra pas être sorti si l'optimisation a lieu sur la période 15h-16h. En revanche, en progressant par pas de ½ période, une optimisation aura d'abord lieu sur la période 14h30-15h30. Ici, à raison d'une sortie de train par créneau, le modèle linéaire permettra peut-être de trouver une solution pour dégager la sortie du train avant 15h15.

Une fois la dernière période après le créneau de pointe optimisée (ici P+3), les mêmes itérations seront réalisées en remontant le temps à partir de P-1. Ici, le recul par demi-périodes vise à donner de la flexibilité pour l'arrivée des trains et non plus leur départ. Si d'un point de vu logistique les enjeux ne sont pas les mêmes, d'un point de vu mathématiques, ces deux parties de l'optimisation sont symétriques.

Pour optimiser chacune des périodes, la classe « MétaheuristiquePériodes » sera appelée, utilisant la méthode Tabou tout comme « MétaheuristiqueIni ». Cependant cette métaheuristique sera arrêtée au bout d'un certain nombre d'itération ou si elle se retrouve bloquée sur une

organisation non valable (au moins un train non remis), redémarrant l'algorithme du tout début (période de pointe).



e) Ecriture du fichier de sortie :

Une fois une solution trouvée pour l'ensemble des périodes, l'algorithme d'optimisation est terminé. Il ne reste alors plus qu'à écrire les résultats dans un fichier lisible par le programme VBA afin de pouvoir afficher les résultats à l'utilisateur et éventuellement de tracer les graphiques correspondants.

Pour cela, la classe « fichierSortie » est appelée et écrira les résultats obtenus dans un fichier nommé « ficResultat.xls ».

La partie Java du programme est alors terminée, et les fenêtres et consoles se ferment, laissant de nouveau la main à l'utilisateur.

C) Fichiers Excel et code VBA

1. Préparation à l'optimisation

- Fenêtre principale :

The screenshot shows a Windows-style dialog box titled "Module Remisage" with a close button (X) in the top right corner. The main title is "Paramètres optimisation". The dialog contains the following elements:

- Two text input fields for file names: "Fichier, liste des trains:" and "Fichier, liste des voies:", each with a "Browse" button (represented by a folder icon) to its right.
- A "Créneau de pointe:" section with a "Jour:" dropdown menu set to "Samedi" and a "Heure:" dropdown menu set to "4".
- A "Nombre d'heures prises en compte autour de la pointe:" section with "Avant:" and "Après:" dropdown menus, both set to "0".
- A large "Lancer l'optimisation" button at the bottom.

La fenêtre principal du module « UserFormRemisage » est affichée afin de pouvoir saisir les paramètres nécessaires à l'optimisation Java. Les deux fichiers de paramétrages doivent être saisis ainsi que le créneau de pointe et les heures (ou périodes) à optimiser.

a) *Fichiers d'entrée des voies :*

Comme vu dans la partie Java, ce fichier décrit les caractéristiques des sites et voies de l'infrastructure. Ceci à raison d'un site par feuille Excel portant le nom du site. Le format descriptif des voies d'un site est le suivant :

Désignation	Longueur (m)	Distance Sécurité
142	351	6
143	355	Electrique
144	370	oui
145	292	
146	292	

La propriété « Electrique » permet d'établir une compatibilité entre le site et le type de matériel. Si dans les infrastructures toulousaines, l'électrification des sites et le mode des trains est une contrainte, ceci n'est pas forcément le cas partout. Cette propriété pourrait donc s'adapter à une

autre contrainte, comme la capacité à accueillir des trains de vieilles séries par exemple. Ceci n'est pas un problème du moment que les données soient mises en correspondance avec celles des trains.

b) Fichiers d'entrée des trains :

NumTrain	Longueur	JourA	HeureA	JourD	HeureD	Serie	Site	Electrique
870626	73	Vendredi	20:00	Samedi	06:48	PPMz		oui
870312	87	Vendredi	20:00	Samedi	19:59	X73500		non
778100	73	Vendredi	21:14	Samedi	16:36	Z27500 4C		oui
872708	73	Samedi	17:41	Samedi	19:03	PPMz		oui
3753	330.3	Samedi	07:22	Samedi	10:02	ZPPg	TE-GT-RAY	oui
3837	110.1	Samedi	00:15	Samedi	06:55	ZPPg	TE-GT-GRV	oui

Les informations sur les trains nécessaires à l'optimisation sont renseignées dans ce fichier. Seule les colonnes « numTrain » et « Série » ne sont pas indispensables mais simplement utiles pour les affichages des résultats.

Par défaut, le programme d'optimisation fonctionne par créneau d'1/4 d'heure. Afin d'éviter d'éventuelles erreurs de calcul, il est préférable d'éviter de renseigner des trains dont la durée de remisage serait inférieure à 15min. Les colonnes « JourA » et « HeureA » renseignent le moment d'arrivée en remisage et les colonnes « JourD » et « HeureD » le moment de départ.

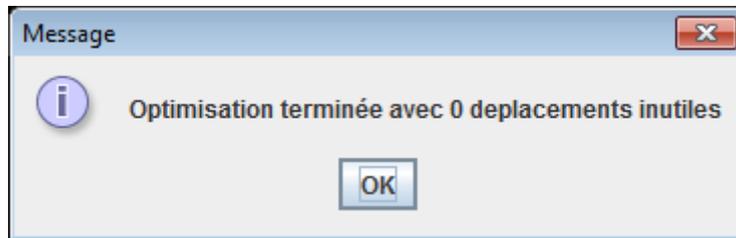
La colonne « Site » est importante et permet d'imposer un site de remisage à un train en indiquant son nom (nom de la feuille correspondante dans le fichier des sites et voies). Si l'on souhaite laisser le programme d'optimisation attribuer un site par lui-même, il faut alors laisser la case vide.

Enfin la dernière colonne correspond à la propriété à mettre en adéquation avec celle des sites. Dans le cas de Toulouse, elle correspond au mode du train (« oui » si le train est électrique, « non » si il est bi-mode ou thermique). Ainsi, sur un site où la propriété aura pour valeur « non », seuls les trains ayant également ce « non » pourront y être remisés.

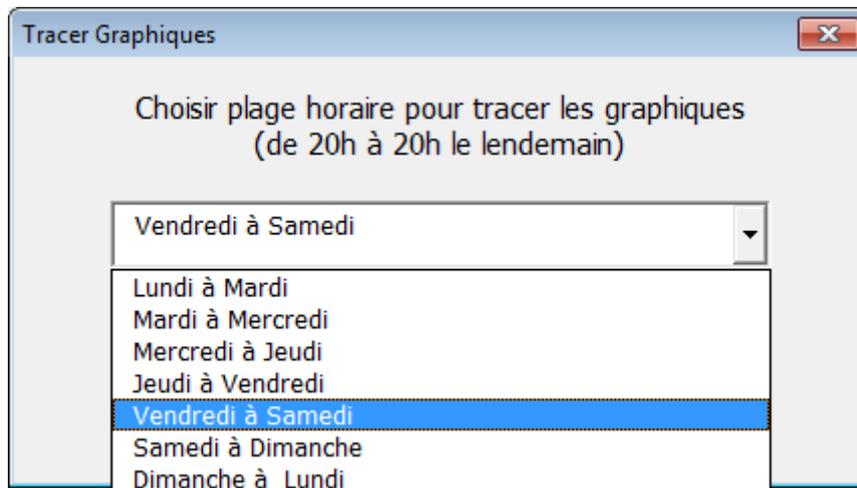
2. Graphiques et résultats

a) Tracés des graphiques :

Une fois l'optimisation terminée, une fenêtre indique le nombre de manœuvres supplémentaires engendrées :



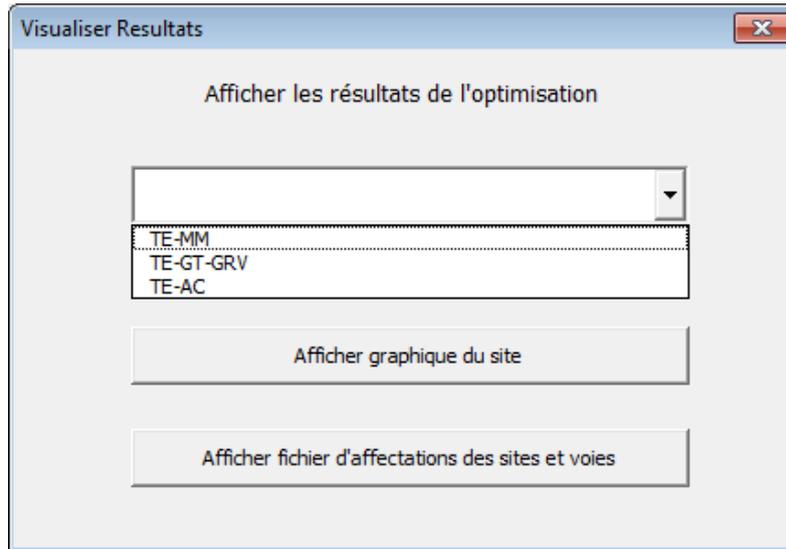
Une fenêtre est alors affichée afin de pouvoir tracer les graphiques correspondants.



L'utilisateur doit choisir la période sur laquelle il souhaite tracer les graphiques. Le fichier de résultats de précisant plus les jours d'arrivée et de départ des trains, seules les heures sont prises en compte. Ce choix doit donc correspondre au moment choisi pour le créneau de pointe pour l'optimisation. Cependant, après l'optimisation de la nuit du lundi au mardi par exemple, ce système permet de tracer les graphiques pour un autre JOB.

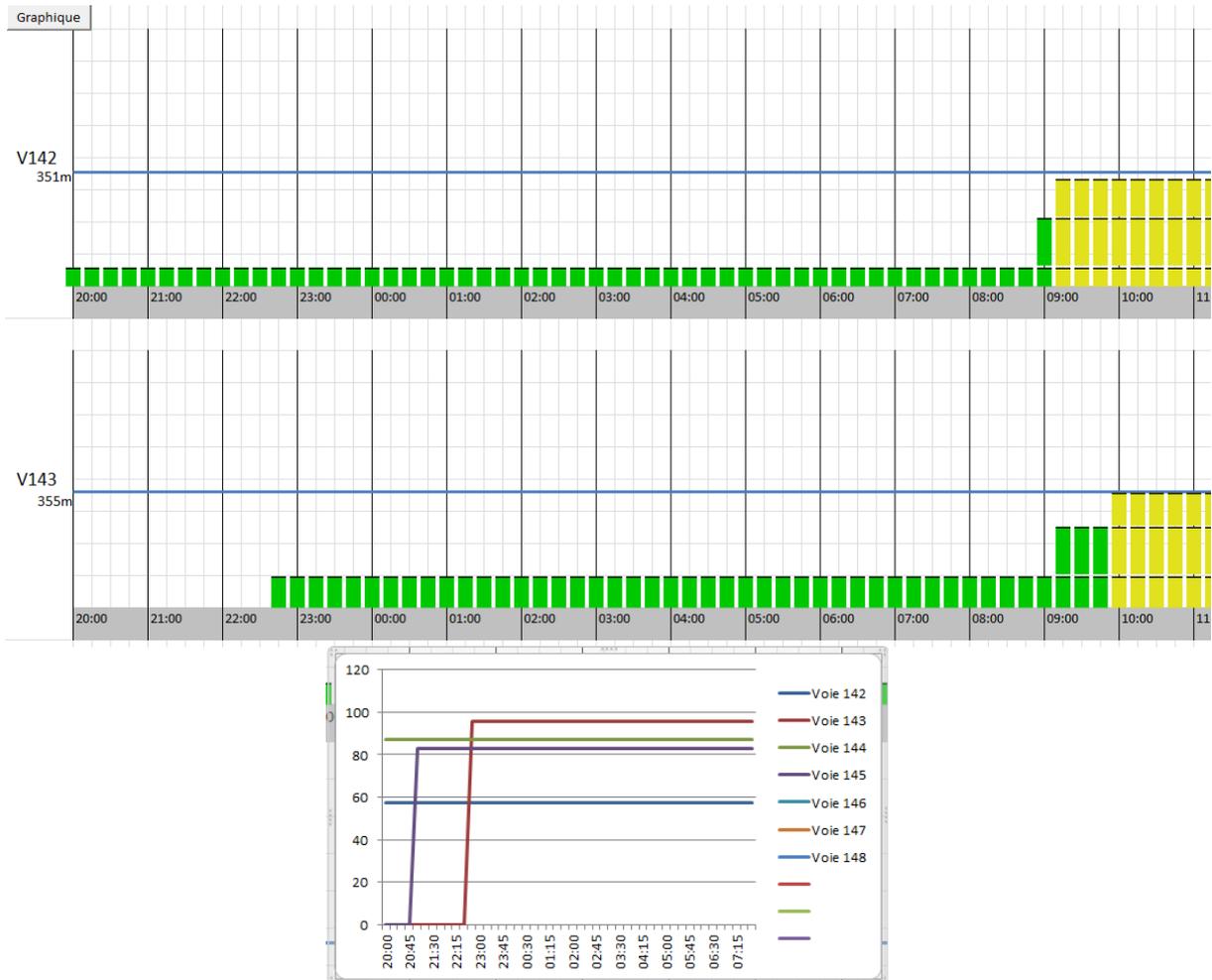
A ce niveau, il est important que les noms/numéros de voies soient uniques, en effet, lors du tracé de graphique, le programme va VBA va déterminer si un train se trouve sur un site en fonction de la voie qui lui a été attribuée. Site après site, le programme va tracer le graphique correspondant et enregistrer les classeurs dans le dossier « Saturations ». Les classeurs sont ensuite classés par dossiers correspondants à chacun des sites, puis les noms de fichiers indiquent eux la période à laquelle ils correspondent (ex : « Vendredi à Samedi »).

b) Fenêtre d'affichage des résultats et graphiques :

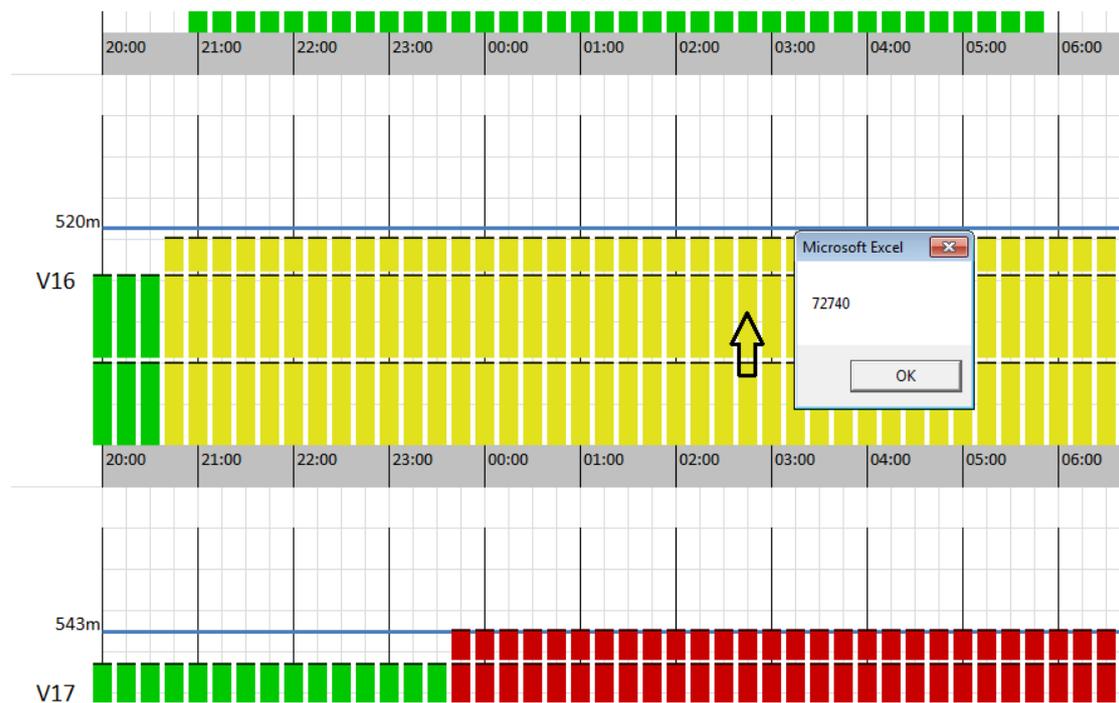


Pour terminer, l'utilisateur peut ensuite afficher les fichiers résultats qu'il souhaite :

- Il peut tout d'abord sélectionner un site et afficher le graphique correspondant, venant d'être tracé. Le bouton « Graphique » est attaché afin de tracer la saturation des voies au fil du temps, et la seconde feuille du classeur comporte les résultats détaillés/chiffrés.



Il est également possible d'afficher les numéros de en cliquant dans un graphique :



- Il peut afficher le fichier de résultat de l'optimisation afin de consulter la liste des affectations de trains par voies et sites :

Numéro	Série	Voie	Arrivée	Départ		Longueur		Site
871847	Z7300	146	21:58	16:39		50,2		TE-GT-GRV
14142	ZPPg	15	8:30	12:12		200		TE-MM
778239	RIO 4C	143	22:45	13:00		95,5		TE-GT-GRV
876702	Z27500 3C	146	8:57	12:50		57,4		TE-GT-GRV
3663	ZPPg	17	2:37	15:55		400		TE-MM
871617	PPMz	145	20:15	14:15		73		TE-GT-GRV
872740	PHDz	53	21:00	12:17		83		TE-AC
872732	PHDz	16	11:00	14:20		83		TE-MM
872706	PPMz	16	8:50	17:09		146		TE-MM
871641	Z27500 3C	15	10:05	19:23		57,4		TE-MM
871813	Z27500	142	9:12	19:59		114,8		TE-GT-GRV
871609	Z7300	16	10:39	15:00		50,2		TE-MM
872463	B81500	55	20:01	19:59		57,4		TE-AC
871609	Z7300	15	10:42	17:02		50,2		TE-MM

Le format de ce fichier correspond au format utilisé dans les données du Technicentre. Les colonnes vides correspondent à des données techniques simplement utiles aux techniciens. L'outil de tracé de graphiques a ainsi été conçu pour pouvoir directement représenter des plans de remisage du Technicentre.

D) Modèles linéaires

1. Optimisation du créneau de pointe

Ce modèle correspond au modèle GUSEK détaillé dans l'annexe I et est repris dans la classe Java « ModeleLinIni » du programme.

Son objectif est de trouver un plan d'affectation par voie d'une liste de train sur un site. Ses contraintes reposent donc uniquement sur la capacité linéaire des voies par rapport à la longueur des trains qui y sont remisés et les distances de sécurité.

Ce modèle est simple et ne comporte qu'un seul type de variable de décision : un train est affecté ou non à une voie.

Deux contraintes sont exprimées :

- La somme des longueurs de matériels et de distances de sécurités sur une voie ne peuvent excéder la capacité de la voie.
- Un train est affecté à une et une seule voie.

Le modèle ne possède pas de fonction objectif et le solveur cherchera donc simplement à fournir une solution valable. Le temps de résolution de ce modèle dépend aussi bien du nombre de voies du site que du nombre de trains que l'on souhaite remiser.

Les paramètres nécessaires la résolution du problème sont simplement les longueurs des voies et des matériels ainsi que la longueur d'une éventuelle distance de sécurité entre deux trains.

2. Optimisation d'une période

Ce modèle correspond au modèle GUSEK détaillé dans l'annexe II et est repris dans la classe « ModeleLineaire » du programme.

Ce problème reprend les contraintes et variables de décision du premier, mais rajoute à cela l'évolution au fil du temps. Ceci engendre la création de nouvelles variables de décisions et de nouvelles contraintes.

Il existe dans ce modèle 4 types de variables de décisions :

- Les variables d'affectation d'un train à une voie à un créneau comme dans le 1^{er} problème. Ces variables seront binaires (1 si affectation 0 sinon).
- Les variables de départ d'un train d'une voie. Il s'agit également d'une variable binaire indiquant si à un créneau un train part d'une voie ou pas.
- Les variables d'arrivées d'un train sur une voie. Similairement aux variables de départ, elles indiquent si un train vient se rajouter sur une voie ou non à un créneau donné.

- Les variables d'ordre permettent d'indiquer le rang d'un train sur une voie. La valeur de cette variable sera 0 si le train n'est pas présent sur la voie au créneau donné, vaudra 1 si le train est en bout de voie, 2 si un autre train est intercalé entre lui et la sortie, 3 si deux trains...

De nombreuses contraintes sont à exprimer en plus de celles du premier modèle :

- Le nombre d'arrivées sur une voie par créneau est limité à 1.
- Le nombre de départs sur une voie par créneau est limité à 1.
- L'ordre d'un train ne peut varier que de 1 entre deux créneaux. En effet, il doit être en bout de voie et donc d'ordre 1 pour sortir d'une voie et les limites d'arrivées et départs par créneau limite l'évolution de l'ordre entre 2 créneaux consécutifs à +1 ou -1.
- Si un train est présent sur une voie son ordre est forcément supérieur ou égal à 1, s'il n'est pas présent il vaudra 0. Ceci engendre des contraintes entre variables d'ordres et d'affectations par voies.
- L'ordre maximum d'un train sur la voie est forcément inférieur ou égal au nombre de trains présents sur sa voie.
- Lors d'un déplacement, un train ne peut partir et arriver sur la même voie à un créneau donné.

De plus, ce modèle possède une fonction objectif visant à limiter le nombre de départs et arrivées de voies. Ceci a pour but de minimiser le nombre de manœuvre et à si possible ne pas déplacer un train durant sa période de remisage.

Si l'on compare ce modèle au précédent, nous pouvons signaler l'ajout d'une dimension supplémentaire qu'est le temps. En plus des voies et des trains, les créneaux multiplient les variables de décisions et contraintes. Afin de limiter le temps de résolution du problème il est donc important de limiter le nombre de créneaux et par la même occasion le nombre de trains du modèle (plus la période étudiée dans le modèle est courte, plus le nombre de trains présents durant la période sur le site est petit).

En plus des longueurs de matériels, de voies et la distance de sécurité, ce modèle nécessite 3 paramètres supplémentaires sous forme de matrices :

- Une matrice de présence en remisage. Celle-ci est binaire et exprime pour chaque train à chaque créneau sa présence en remisage ou non. Ainsi, pour un « 0 » le modèle saura que le train n'est pas présent et donc ne devra être affecté à aucune voie à ce créneau. A l'inverse un « 1 » exprimera que le train doit être affecté à une et une seule voie à ce créneau.
- Une matrice d'initialisation des affectations de voies. Selon l'utilisation du modèle, cette matrice impose les affectations de trains par voies soit au premier créneau, soit au dernier. Les autres créneaux verront leurs affectations se réaliser en fonction de cette organisation initiale ou finale.
- Une matrice d'ordres des trains par voies. En correspondance à la matrice précédente, elle permet de détailler l'organisation initiale ou finale. Dans le programme d'optimisation, lorsque nous avançons dans le temps, chaque modèle partira de l'organisation initiale calculée précédemment. A l'inverse lorsque l'optimisation remonte dans le temps, une période est optimisée en cherchant à retrouver l'organisation finale (au dernier créneau).

VI. Résultats d'études menées

A) Viabilité des infrastructures actuelles à Matabiau

1. Objectifs de l'étude

Actuellement, l'organisation du remisage à Toulouse Matabiau n'est pas complètement optimisée. Les infrastructures telles qu'elles sont actuellement, permettent un fonctionnement satisfaisant, mais ceci ne sera pas forcément dans le futur. En effet, ayant connaissance des prévisions d'offres des transporteurs et l'arrivée de nouveaux matériels, la SNCF prévoit une évolution des contraintes de remisage à plusieurs horizons futurs :

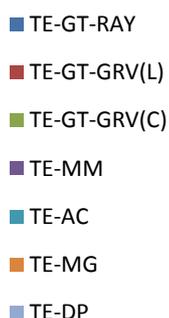
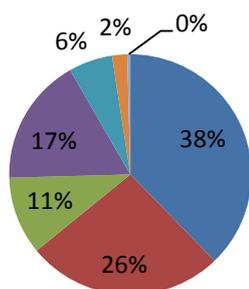
- 2017 : les séries de matériels TER devraient subir une évolution. L'arrivée des rames Régiolis entraîne un changement de longueur des rames. Dans le même temps, TGV prévoit une intensification de ses fréquences sur la ligne Paris-Bordeaux-Toulouse.
- 2024 : TER aura terminé l'intégration des séries Régiolis et Régio-2N et prévoit quelques fréquences supplémentaires engendrant 6 trains à remiser en plus. L'intensification des circulations TAGV devrait encore augmenter le nombre de rames à remiser et les rames TGV Atlantiques seront remplacées par des rames TGV Réseau. Enfin, TET changera également de matériels en délaissant les rames Corail.
- 2030 : il s'agit de l'horizon cible de l'étude. L'ensemble des changements de séries seront entièrement terminés. Les prévisions de l'augmentation des fréquences TAGV s'arrêtent également à cet horizon.

Le but est donc d'analyser l'organisation actuelle du remisage, essayer de reporter celle-ci aux horizons futur avec l'évolution du parc matériel, d'optimiser l'organisation si cela est possible et en tirer des conclusions sur la viabilité des sites de remisages tels qui sont aujourd'hui jusqu'en 2024 dans un premier temps.

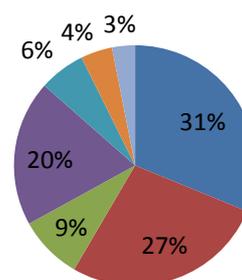
2. Démarche et calculs réalisés

Dans un premier temps l'étude visait donc à réunir l'ensemble des données nécessaires à la construction d'une vision globale du remisage à Toulouse Matabiau aujourd'hui.

Longueur matériels remisés



Capacité linéaire



Les simples chiffres sur les saturations globales et les capacités linéaires (longueurs de voies de remisage) ont permis de dégager 3 sites majeur que sont TE-GT-RAY (ou TE-GT), TE-GT-GRV(L) (ou TE-GRV(L)) et TE-MM. A eux trois, ils représentent près de 80% des capacités de remisage actuellement.

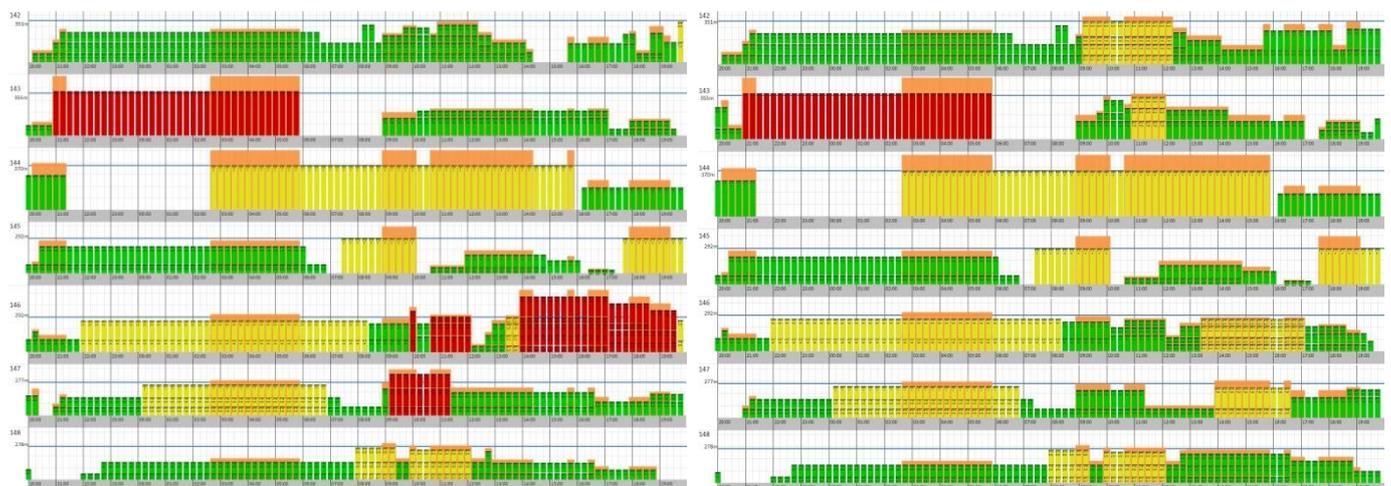
Afin de pouvoir mieux visualiser les saturations voies par voies à l'intérieur des sites, un outil de tracé de graphiques en VBA a été réalisé, également dans l'objectif d'être plus tard intégré au module d'optimisation :

Visualisation du site TE-GRV(C) La nuit du vendredi au samedi



L'outil permettait ainsi de visualiser sur une période de 24h (allant de 20h à 20h) la présence de trains sur les voies d'un site. De plus le code couleur permet de visualiser si une voie est complètement saturée voir en dépassement de capacité, ou si l'organisation du site est non robuste (incapacité d'accueillir une unité simple du matériel le plus long remisé sur le site).

Une fois l'ensemble des graphiques tracés pour 2014, le travail allait être réalisé de nouveau aux horizons 2017 et 2024. Pour ce qui est de l'organisation a ces horizons, celle d'aujourd'hui était reportée en ajoutant où cela semblait possible les trains supplémentaires dus à l'évolution. Si l'organisation s'avérait obsolète, en l'absence de programme d'optimisation global, l'optimisation linéaire était pratiquée site par site avec Gusek afin de rechercher une solution.

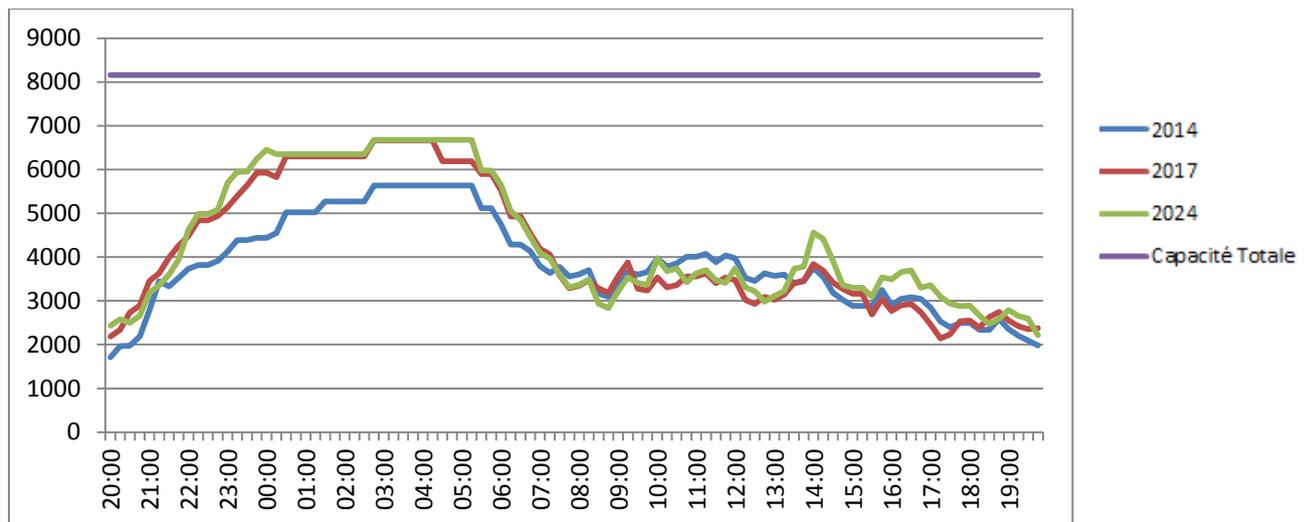


Visualisation du site TE-GRV(L) avant et après optimisation : élimination d'un maximum de dépassements de capacité grâce à l'optimisation linéaire.

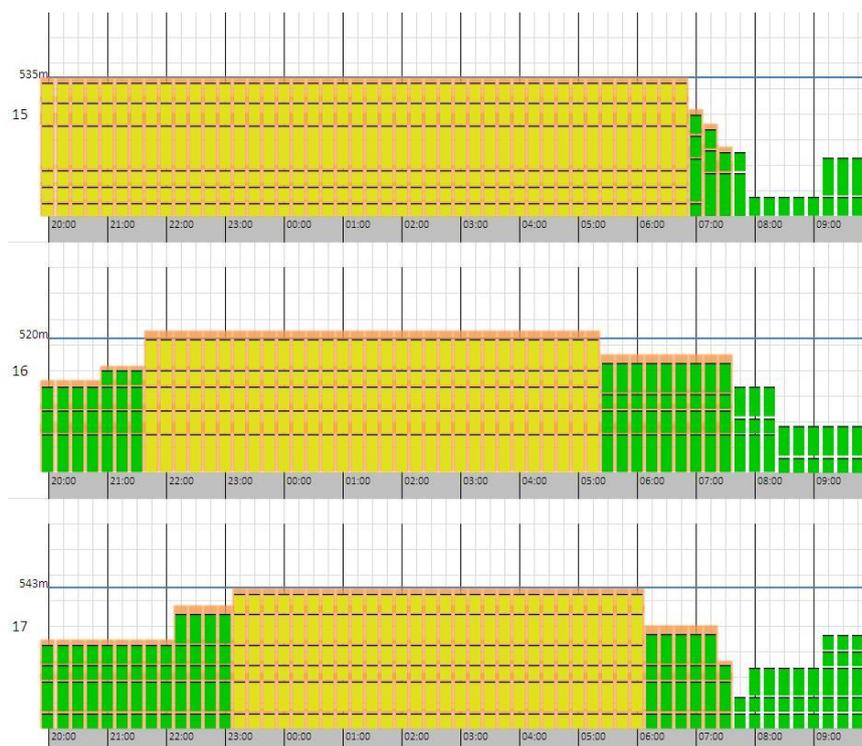
3. Résultats obtenus

Malgré que le programme d'optimisation n'était pas encore entièrement développé et que celle-ci n'était alors que linéaire et que la redistribution des trains par site se faisait manuellement, cette première étude a néanmoins permis de dégager des conclusions indiscutables.

En se référant aux graphiques, très visuels, le comparatif montre des différences flagrantes entre les trois horizons et révèle des problèmes de surcapacité dès 2017. Ces impressions dégagées par l'optimisation sont confirmées par les taux de saturations globaux.



Au maximum, les longueurs de matériels remisés représentent aujourd'hui 69% de la capacité linéaire des voies de remisage en prenant en compte les distances de sécurité entre les trains. En 2017, ce taux atteindrait 80% puis 81% en 2024, cumulé à l'allongement des rames, ce qui est encore d'avantage contraignant.



Des graphiques tels que celui obtenu après optimisation à TE-MM en 2017 montre la qualité des solutions obtenues, avec une saturation totale des voies. Si malgré une telle optimisation, des soucis de sursaturation ou même de robustesse subsistent, alors démonstration est faite que les capacités des infrastructures actuelles ne sera plus suffisante à cet horizon.

B) Trois scénarios d'évolution

1. Scénario peu coûteux

Après avoir démontré que des travaux devaient être envisagés pour augmenter les capacités de remisage à Toulouse Matabiau, plusieurs scénarios ont été imaginés. Ceux-ci visaient sur l'évolution des infrastructures aux 3 horizons, mais leur différenciation s'effectuait dès 2017. En effet, des travaux sont prévus pour l'intervalle 2024/2030 et ont donc été confirmés par la première étude. En revanche, pour pallier aux horizons 2017 et 2024, un autre site de remisage était prévu à être intégré : TE-GB (Gare-Basse). Mais celui-ci pourrait offrir plusieurs types de voies.

Le premier scénario, se voulant peu coûteux visait donc à l'introduction de 5 voies déjà existantes mais devant être rénovées :

Chantier	Voie	Linéaire	Chantier	Voie	Linéaire	Chantier	Voie	Linéaire
TE-GT	5	525	TE-GRV (L)	142	351	TE-AC	51	118
TE-GT	6	525	TE-GRV (L)	143	355	TE-AC	53	192
TE-GT	8	495	TE-GRV (L)	144	370	TE-AC	55	192
TE-GT	10	472	TE-GRV (L)	145	292	TE-AC	sous total	502
TE-GT	11	530	TE-GRV (L)	146	292			
TE-GT	sous total	2 547	TE-GRV (L)	147	277	Chantier	Voie	Linéaire
			TE-GRV (L)	148	278	TE-DP	30	103
			TE-GRV (L)	sous total	2 215	TE-DP	31	100
Chantier	Voie	Linéaire				TE-DP	32	47
TE-MM	15	535	Chantier	Voie	Linéaire	TE-DP	sous total	250
TE-MM	16	520	TE-GRV (C)	132	225			
TE-MM	17	543	TE-GRV (C)	133	250	Chantiers	Nb voies	Linéaire
TE-MM	sous total	1 598	TE-GRV (C)	134	224	TE-GT	5	2 547
			TE-GRV (C)	sous total	699	TE-GRV (L)	7	2 215
Chantier	Voie	Linéaire				TE-GRV (C)	3	699
TE-GB	B6	210	Chantier	Voie	Linéaire	TE-MM	3	1 598
TE-GB	B7	210	TE-MG	65	120	TE-DP	3	250
TE-GB	B8	210	TE-MG	66	100	TE-MG	3	345
TE-GB	B9	195	TE-MG	67	125	TE-AC	3	502
TE-GB	B11	215	TE-MG	sous total	345	TE-GB	5	1 040
TE-GB	sous total	1 040				TOTAL	32	9 196

Ensemble des sites en 2017, selon le scénario 1

En utilisant le programme d'optimisation global, dont le développement était terminé, ce scénario offrait des solutions pour 2017. Cependant, les offres transporteurs évoluant entre 2017 et 2024, et aucuns autres travaux majeurs n'étant prévus dans cette période, les infrastructures s'avéraient à nouveau insuffisante lors de la simulation de l'horizon 2024.

En effet, Parallèlement à une augmentation prévue de 400m des matériels au total, cumulé à l'allongement des séries utilisées, dans le même temps seul le chantier des Acacias (TE-AC) allait être mis à jour pour un gain de seulement 12 mètres au total.

En 2030, grâce aux nombreux travaux prévus, des solutions étaient de nouveau trouvées par le solveur, mais inintéressantes étant donné que le moment critique que constitue 2024 ne pourrait être passé.

VII. Intégration à l'outil SIMUTER

La Direction Déléguée du TER (DDTER) doit jouer un rôle de conseil auprès du Conseil Régional de Midi-Pyrénées à travers son expertise dans les différents métiers du transport ferroviaire. Cependant, la DDTER ne disposait pas d'outils permettant de répondre à des problématiques précises de manière rapide.



BILAN VOY

BILAN ECO

BILAN INFRA

BILAN PROD

Depuis trois ans environ, Eric LEBEAU, à travers son service, a souhaité apporter une solution à cette question à travers l'outil SIMUTER.

L'outil SIMUTER est composé de plusieurs modules développés en VBA, où chaque module fut créé à la suite d'études commandées par le Conseil Régional.

Par exemple, le Conseil Régional souhaitait savoir quel étaient les gares ayant des longueurs de quais trop court pour accueillir la nouvelle série de trains, les Régiolis (75m au lieu de 56m auparavant). La question permettait de savoir s'il fallait prévoir des travaux et quelles étaient les gares à traiter en priorité.

Un module en VBA fut réalisé afin de croiser les données de base récupérés à travers les différents pôles (longueurs des trains : TER, longueurs des quais : Infrastructure,...). De plus, l'outil apporte un gain de temps considérable de réponse et peut être réutilisable.

Voici la liste des modules composant SIMUTER :

- **Horaires et Sillons** : Représentation d'une grille horaire en graphique de circulation des trains.
- **Estimations de Trafic** : Estimation du trafic voyageur procuré par un projet d'offre TER.
- **Moyens et Ressources** : Conception d'une trame de roulement renseignée d'indicateurs de performance.
- **Parc et Parcours** : Allocation des ressources Matériel et mesure des effets (parc & maintenance).
- **Charges et Capacités** : Adéquation de la charge voyageur à la capacité des trains.

Comme pour le remisage d'autres modules devraient être développés et intégrés. C'est pour parvenir à cette intégration que l'interface utilisateur du module a d'ailleurs était réalisée en langage VBA.

Conclusion

Ces 6 mois de stage se sont révélés une très bonne conclusion aux enseignements suivis en Master ISMAG. De nombreuses matières m'ont été utiles lors de cette mission et j'ai pu de plus approfondir mes connaissances et mon intérêt pour la Recherche Opérationnelle notamment. Si la problématique me semblait à premier abord ardue, ceci ne fait que renforcer la satisfaction apportée par cette expérience professionnelle.

La mission m'a également permis de synthétiser les connaissances de l'ensemble de mes études supérieures. En effet, mon cursus initial en informatique m'a permis de gagner en efficacité et de directement mettre en œuvre les méthodes de calculs développées.

L'utilité directe des travaux a également été un facteur déterminant dans ma motivation. Le fait de pouvoir développer entièrement un outil, que ce soit réaliser sa conception, son algorithme mathématique et sa programmation puis l'appliquer à des problèmes réels et enfin défendre les résultats obtenus lors de réunions externes, fut d'un grand intérêt.

Ce fut également mon premier stage lié au service public, ce qui enrichit ma vision du monde des différentes entreprises. Le domaine ferroviaire était également une première pour moi. Ce dernier s'est révélé très pertinent pour pouvoir mettre en œuvre mes compétences. Les facteurs d'optimisation sont en effet très nombreux et les disciplines telles que la Recherche Opérationnelle, les Statistiques et l'Informatique sont au cœur de l'activité actuelle et future de l'entreprise.

La durée de la mission m'a également laissé le temps de m'affirmer et ainsi pouvoir participer aux réunions décisionnelles, une expérience nouvelle de plus. Pouvoir prendre part aux débats, que ce soit dans l'entreprise, ou face à des partenaires permet de prendre conscience des enjeux du travail réalisé. Ceci m'a aussi encouragé dans mes démarches et m'a motivé à fournir le travail le plus précis et le plus cohérent possible.

Au bilan, ce stage m'a donc énormément apporté que ce soit sur mes connaissances techniques et du monde de l'entreprise ou sur mon assurance. La confiance qui m'a été donnée par mes supérieurs au départ et que je pense avoir su renforcer avec le temps m'a permis de réaliser un stage épanouissant, avec un travail de recherche conséquent à réaliser, mais aussi en appliqué à des cas concrets et utiles.

Résumé

Après une adaptation rapide à l'entreprise et au monde ferroviaire, des premières études devaient être rapidement menées afin de pouvoir présenter des résultats en interne et aux partenaires de la SNCF que sont notamment Réseaux Ferrés de France (RFF) et le Conseil Régional de Midi-Pyrénées.

Les premiers calculs se sont centrés sur de l'optimisation linéaire en utilisant en particulier le logiciel Gusek et le solveur GLPK auquel il est associé. Ceux-ci, malgré des optimisations encore partielle ont permis de montrer les limites des sites et voies de remisage actuels de Toulouse. Ainsi, l'étude pouvait être poursuivie afin de trouver des solutions, une fois l'ensemble des entités convaincues que des travaux seraient rapidement à réaliser.

Pour pouvoir tester un maximum d'hypothèses et ceci le plus rapidement possible, un programme d'optimisation global du remisage a alors été développé en cumulant les concepts des métaheuristiques à la résolution linéaire. La programmation a été réalisée en utilisant les langages Java et VBA et en faisant de nouveau appel au solveur GLPK. Une fois terminé, le programme permettait d'optimiser l'organisation globale au sein de toutes les infrastructures de la gare Matabiau et de simuler l'impact de l'ajout ou de la modification de sites.

Trois scénarios d'évolution ont pu être testés à trois horizons futurs grâce à l'outil. Les résultats ont permis d'en éliminer un, laissant le choix aux décideurs entre les deux derniers. L'un permettant de garantir largement l'exploitabilité au fil du temps, mais avec un coup élevé, le second s'avérant juste suffisant aux prévisions mais avec un coup beaucoup plus restreint.

Enfin, le développement d'une interface utilisateur et la rédaction d'un manuel méthodologique pour le module ont permis de garantir son intégration au logiciel Simuter et son adaptation à d'autres contraintes pour le rendre utilisable par d'autres régions que Midi-Pyrénées.

Annexes

II. Modèle linéaire : Créneau de pointe

III. Modèle linéaire : optimisation période

VII. Rapport d'étude : Viabilité des infrastructures actuelles

Modèle linéaire : créneau de pointe

(Cet exemple porte sur un site à 4 voies sur lequel on souhaite remiser 5 trains durant 4 créneaux + 1 créneau d'initialisation)

```
/* Nombre de trains */
set I:=1..8;
/* Nombre voies */
set J:=1..4;

/* variable de décision, numéro de voie attribué au train */
var X{i in I, j in J}, binary;
/* 1 si le déplacement est attribué à la voie, 0 sinon */

/* trains */
param T { i in I };
/* Voies */
param V { j in J}, >=0;
/* Distance sécurité */
param Dsec, >= 0;

/* contraintes */
/* la somme des longueurs de trains sur une voie ne peut excéder
sa capacité*/
s.t.C1{j in J}: sum{i in I}(T[i]*X[i, j]) + Dsec*(sum{i in I}X[i,
j]-1) <= V[j];
/* un train est présent sur une et une seule voie */
s.t.C2{i in I}: sum{j in J}X[i, j] = 1;

solve;

for {j in J}
{
  printf "\n";
  printf "Voie %d", j;
  for {i in I}{
    printf "\t%d ", X[i,j];
  }
}
printf "\n";

/* valeurs des paramètres et données*/
data;
param T:= 1 90 2 80 3 150 4 80 5 20 6 50 7 200 8 150;

param V:= 1 300 2 300 3 100 4 200;

param Dsec := 6;
end;
```

Modèle linéaire : optimisation période

(Cet exemple porte sur un site à 3 voies sur lequel on souhaite remiser 8 trains)

```
/* Nombre de trains */
set J:=1..5;
set Jbis:=2..5;
/* Nombre voies */
set I:=1..3;
/* Nombre de périodes */
set T:=0..4;
/* Périodes de déplacement (nombre de périodes -1) */
set Tbis:=1..4;

/*variables de décision, numéro de voie attribué au train */
var X{i in I, j in J, t in T}, binary;
/* 1 si le train i est attribué à la voie j ,0 sinon */
/*variable de déplacements, exprime si un train a été déplacé par
rapport à la période précédente*/
var ARRIVEE{i in I,j in J, t in Tbis}, binary;
var DEPART{i in I,j in J, t in Tbis}, binary;
/*Matrice désignant l'ordre des trains sur la voie, seul le train
"1" peut sortir de la voie*/
var Ordre{i in I, j in J, t in T}, >=0;

/* plan de présences des trains en fonction du temps */
param M1{j in J, t in T }, binary;
/* longueur maximale des voies */
param V{i in I}, >=0;
/*longueur des trains */
param TRAINS{j in J}, >=0;
/* Dernière période */
param TMAX, >=0;
/*Placement des trains déjà présents à l'instant 0, avant
optimisation*/
param Ini{j in J, i in I}, binary;
/*initialisation des priorités */
param IniO {j in J, i in I}, >=0;
/* Numéros de circulation des trains */
param Num{j in J};
/*Distances de sécurité entre 2 trains*/
param DSEC, >=0;

/* contraintes*/
/* la somme des longueurs de trains et distances de sécurités sur
une voie ne peut excéder sa capacité*/
s.t.C1{i in I, t in T}: sum{j in J}(TRAINS[j] * X[i,j,t]) +
((sum{j in J}X[i,j,t])-1)*DSEC <= V[i];
```

```

/* un train présent sur le site à une période donnée est présent
sur une et une seule voie*/
s.t.C2{j in J, t in T}: sum{i in I}(X[i,j,t]) = M1[j,t];

/* calcul si un train est arrivé sur la voie ou en est parti*/
s.t.C3{i in I, j in J, t in Tbis}: ARRIVEE[i,j,t] = X[i,j,t] -
X[i,j,t-1] +DEPART[i,j,t];

/* on fixe les placements à la période initiale 0*/
s.t.C4{i in I, j in J}: X[i,j,0] = Ini[j,i];
s.t.C5{i in I, j in J}: Ordre[i,j,0] = IniO[j,i];

/* Un seul train ne peut partir ou entrer sur une voie par
période */
s.t.C6{i in I, t in Tbis}: sum{j in J}ARRIVEE[i,j,t]<=1;
s.t.C7{i in I, t in Tbis}: sum{j in J}DEPART[i,j,t]<=1;

/* évolution du dernier train en bout de voie */
/* l'ordre d'un train ne peut varier que de 1 d'une période à une
autre*/
s.t.C8{i in I, j in J, t in Tbis}: Ordre[i,j,t] - Ordre[i,j,t-
1]<=1;
s.t.C9{i in I, j in J, t in Tbis}: Ordre[i,j,t] - Ordre[i,j,t-
1]>=-1;

/*évolution de l'ordre */
/* l'ordre d'un train est supérieur à 1 si il est présent sur la
voie*/
s.t.C10{i in I, j in J, t in T}: Ordre[i,j,t] >= X[i,j,t];
/* l'ordre d'un train est nul si il n'est pas sur la voie */
s.t.C11{i in I, j in J, t in T}: Ordre[i,j,t]<= X[i,j,t]*100;

/* l'ordre d'un train ne peut dépasser le nombre de trains
présents sur la voie */
s.t.C12{i in I, j in J, t in Tbis}: Ordre[i,j,t]<= (sum{k in
J}X[i,k,t]);/******/

/* Partie pour les voies en pile */
/* l'ordre d'un train ne peut monter que si un train arrive sur
sa voie et qu'aucun ne part, ou si lui-même arrive sur la voie. Il
baisse si un train part et qu'aucun n'arrive */
s.t.C13{i in I, j in J, t in Tbis}: Ordre[i,j,t]<= Ordre[i,j,t-
1]+sum{k in J}ARRIVEE[i,k,t]-sum{k in J}DEPART[i,k,t]+ARRIVEE[i,j,t]+(1-X[i,j,t]);

/* en pile: l'ordre d'un train ne peut baisser que si un train
part de sa voie et aucun n'arrive, ou si il part lui-même de la
voie. Il monte si un train de plus arrive et que lui-même est sur la
voie (X[i,j,t-1] )*/
s.t.C14{i in I, j in J, t in Tbis}: Ordre[i,j,t] >=Ordre[i,j,t-1]
+sum{k in J}ARRIVEE[i,k,t]-sum{k in J}DEPART[i,k,t]-
DEPART[i,j,t]+(X[i,j,t]-1);

```

```

/* un train ne peut partir que si son ordre est 1 : en bout de
ligne */
s.t.C15{i in I, j in J, t in Tbis}: DEPART[i,j,t] <= 1-
(Ordre[i,j,t-1]-1)*0.01;

/* un train ne peut partir d'une voie et revenir sur la même lors
d'un déplacement */
s.t.C16{i in I, j in J, t in Tbis}: DEPART[i,j,t] +
ARRIVEE[i,j,t] <=1;

/* Partie pour les voies en file */
/* en file: les départs d'autres trains n'influent plus sur
l'ordre du train, seul son propre départ de la voie peut faire
baisse l'ordre d'un train */
/*s.t.C13{i in I, j in J, t in Tbis}: Ordre[i,j,t]<= Ordre[i,j,t-
1]+sum{k in J}ARRIVEE[i,k,t];
s.t.C14{i in I, j in J, t in Tbis}: Ordre[i,j,t] >=Ordre[i,j,t-1]
+sum{k in J}ARRIVEE[i,k,t]-DEPART[i,j,t]*100+(X[i,j,t]-1);*/
/* un train ne peut partir que si son ordre est égal au nombre
de trains sur sa voie */
/*s.t.C15{i in I, j in J, t in Tbis}: DEPART[i,j,t] <=
1+(Ordre[i,j,t-1]-sum{k in J}X[i,k,t-1])*0.01;*/

/* fonction objectif: minimiser le nombre de déplacements */
minimize F: (sum{i in I, j in J, t in Tbis}(ARRIVEE[i,j,t] +
DEPART[i,j,t]))- sum{j in J}((M1[j,0] - 1)*-1 + (M1[j,TMAX]-1)*-
1))/2;
/*minimize F: sum{i in I, j in J, t in Tbis}(ARRIVEE[i,j,t] +
DEPART[i,j,t]);*/

solve;
/* fonction d'affichage*/
for {t in T}
{
printf "\n\n\n\n";
printf "Période %d \n", t;

for {j in J}{
printf "\t%d", Num[j];
}

for {i in I}{
printf "\n";
printf "voie %d\t", i;
for {j in J}{
printf "%d;", Ordre[i,j,t];
}
}
}
}

```

```

/* valeurs des paramètres et données */
data;
param TMAX := 4;
param DSEC := 6;

/*longueur des voies*/
param V:= 1 200 2 100 3 60;

/*longueur des trains*/
param TRAINS:= 1 80 2 40 3 30 4 62 5 50.6;

/*numéros des trains (seulement pour affichage après
optimization)*/
param Num:= 1 870000 2 870011 3 870022 4 870033 5 870044;

/*matrice de presence des trains au fil du temps*/
param M1:=
[*,*]:0 1 2 3 4:=
1 1 1 1 1 1
2 1 1 1 1 0
3 1 1 0 0 0
4 0 0 0 1 1
5 0 0 0 0 1;

/*matrice d'affectations par voies de trains au créneau initial
(créneau d'indice 0)*/
param Ini:=
[*,*]: 1 2 3:=
1 1 0 0
2 1 0 0
3 1 0 0
4 0 0 0
5 0 0 0;

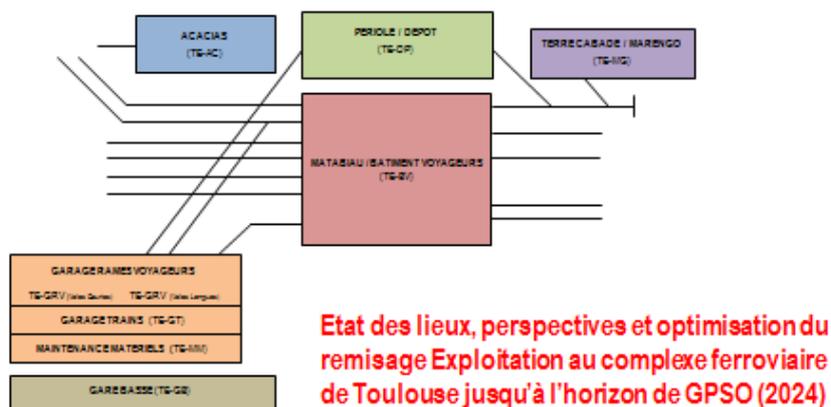
/*matrice d'ordres sur les voies au créneau initial*/
param IniO:=
[*,*]: 1 2 3:=
1 3 0 0
2 1 0 0
3 2 0 0
4 0 0 0
5 0 0 0;

end;

```

ETUDE TRANSPORTEUR SNCF

Rapport Phase 1
Identification des enjeux



Toulouse - 10 juin 2014



Sommaire

1- Motivation et objectifs

3- Organisation actuelle du remisage

3-1 Utilisation des différents chantiers de remisage

3-2 Construction d'une vision globale

4- Modélisation des besoins futurs de remisage

4-1 Projection du mode d'affectation actuel

4-2 Optimisation de l'allocation des capacités

5- Enseignements et compléments à réaliser



Eléments de cadrage 1/7

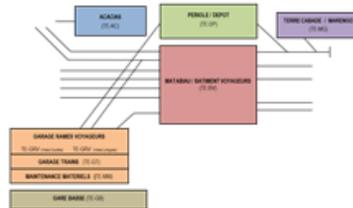
1- Calendrier et déroulement : l'étude se déroule sur 4 mois et comporte en 2 phases :

- Phase 1 (identification des enjeux) de mars à mai 2014
- Phase 2 (propositions de solution) de juin à juillet 2014

2- Périmètre et horizons : le périmètre d'étude intègre l'ensemble des trains de voyageurs (TER, TET et TAGV) remis sur les différents chantiers de remisage de Toulouse. Outre la situation actuelle (2014), l'étude prend en compte les horizons de mise en service des projets SEA (2017) et GPSO (2024)

3-1 Capacité de remisage : le complexe ferroviaire de Toulouse comprend plusieurs chantiers de remisage. Indépendamment de la longueur des voies, les caractéristiques majeures à retenir sont :

- voies non électrifiées au chantier « Terre Cabade » (TE-MG)
- chantier « GRV » différencié en faisceau « voies longues » et « voies courtes »
- voies équipées de caténaire renforcée et présence d'entrevoies larges permettant le nettoyage et le petit entretien des rames TGV au chantier « Garage des Trains » (TE-GT RAV)



3

Eléments de cadrage 2/7

3-2 Capacité de remisage : le linéaire des 7 chantiers est globalement de 8 156m (27 voies dédiées)

Chantier	Voie	Linéaire	Chantier	Voie	Linéaire	Chantier	Voie	Linéaire
TE-GT	5	525	TE-GRV (L)	142	351	TE-AC	51	118
TE-GT	6	525	TE-GRV (L)	143	355	TE-AC	53	192
TE-GT	8	485	TE-GRV (L)	144	370	TE-AC	55	192
TE-GT	10	472	TE-GRV (L)	145	292	TE-AC	sous total	502
TE-GT	11	530	TE-GRV (L)	146	292	Chantier	Voie	Linéaire
TE-GT	sous total	2 547	TE-GRV (L)	147	277	TE-OP	30	103
Chantier	Voie	Linéaire	TE-GRV (L)	148	278	TE-OP	31	100
TE-MM	15	535	TE-GRV (L)	sous total	2 215	TE-OP	32	47
TE-MM	16	520	Chantiers	Nbvoies	Linéaire	TE-OP	sous total	250
TE-MM	17	543	TE-GT	5	2 547	Chantier	Voie	Linéaire
TE-MM	sous total	1 598	TE-GRV (L)	7	2 215	TE-MG	65	120
Chantier	Voie	Linéaire	TE-GRV (C)	3	699	TE-MG	66	100
TE-GRV (C)	132	225	TE-MM	3	1 598	TE-MG	67	125
TE-GRV (C)	133	250	TE-OP	3	250	TE-MG	sous total	345
TE-GRV (C)	134	224	TE-MG	3	345	TE-GT et TE-GRV	disjoints	
TE-GRV (C)	sous total	699	TE-AC	3	502	GRV-L et GRV-C	individualisés	
			TOTAL	27	8 156			



4

Eléments de cadrage 3/7

4- Contraintes d'exploitation : en fonction des séries de matériel, des compositions prévues et de la distance de sécurité (ou non) entre 2 rames, on identifie, chantier par chantier, le besoin de remisage exploitation

Longueur hors tout des séries de matériel à remisageur Toulouse			
Série	Linéaire	UV automatisée	Distance sécurité
X2100	22,4	x	
R40GR (T)	66,0		
X72500	53,6	x	
X72500	29,0	x	
GR1 500	57,4	x	
E275003C	57,4	x	
E275004C	73,0	x	
E7300	50,2	x	
RRR21C	119,5		x
R104C	95,5		x
GRV1	73,0	x	
ERV1	73,0	x	
Z 1ND	63,0	x	
GR 2000	14,7	x	
GR 7000	17,5	x	
CORA L 4V	105,6		x
CORA L 5V	155,4		x
CORA L 5V	152,0		x
CORA L 7V	154,5		x
CORA L 5V	211,2		x
CORA L 12V	290,4		x
CORA L 14V	369,6		x
TGV ATLANTIQUE	237,6	x	
TSV NEREAU	200,2	x	

* : rame bleue composée de X2100 + X72500 + X2100

Distance d'attente entre rames (en m)							
Chantier	Voie	TGV	Autres trains	Chantier	Voie	TGV	Autres trains
TE-GT	5	30	6	TE-MM	15	6	6
	6	30	6		16	6	6
	8	30	6		17	6	6
	10	30	6		31	0	0
TE-GRV (L)	11	30	6	TE-AC	59	0	0
	142	6	6	55	0	0	
	143	6	6	65	0	0	
	144	6	6	TE-MG	66	0	0
	145	6	6	67	0	0	
	146	6	6	30	0	0	
7 Chantiers et 27 voies dédiées	147	6	6	TE-OP	31	0	0
	148	6	6	32	0	0	
				TE-GRV (C)	138	0	0
				139	0	0	



5

Éléments de cadrage 4/7

5-1 Hypothèses d'évolution de l'offre : seules les circulations Origine / Terminus Toulouse sont prises en compte, ce qui n'équivaut pas aux besoins de remisage nocturne (plage dimensionnante)

Offre TER Origine / Terminus Toulouse (Trains / JOB)				Offre TET Origine / Terminus Toulouse (Trains / JOB)			
Relation	Service 2014	SEA 2017	GPSO 2024	Relation	Service 2014	SEA 2017	GPSO 2024
Auch	94	94	94	Toulouse-Paris (Paléo)	6	6	6
Lezour	45	45	45	Toulouse-Paris (Occitan)	2	2	2
Pau	49	49	49	Toulouse-Hendaye	10	10	10
Rodez	41	41	41	Toulouse-Quimper	2	2	2
Figeac	12	12	12	Marseille-Toulouse	1	1	1
Mazamet	22	22	22	TOTAL	21	21	21
Agen	34	34	34	Offre TAGV Origine / Terminus Toulouse (trains / JOB)			
Brive	12	12	12	Relation	Service 2014	SEA 2017	GPSO 2024
Carcassonne	35	35	35	Toulouse-Paris	10	18	26
TOTAL	344	344	344	Toulouse-Barcelone	2	4	4
Bilan Offre Origine / Terminus Toulouse (Trains / JOB)				Toulouse-Strasbourg	2	2	2
Service 2014	344	344	344	Toulouse-Lille	2	2	2
SEA 2017	344	344	344	Toulouse-Dijon	2	2	2
GPSO 2024	344	344	344	TOTAL	18	28	36



6

Éléments de cadrage 5/7

5-2 Hypothèses d'évolution du parc : les données prises en compte proviennent des Transporteurs SNCF et sont celles qui sont privilégiées à date (mai 2014)

Longueur hors tout des séries de matériel à remiser sur Toulouse				
Série	Longueur	SA 2014	SEA 2017	GPSO 2024
02100	22,4	X		
REVER (*)	85,0	X		
07200	52,6	X	X	X
07300	29,0	X	X	X
08100	57,4	X	X	X
22700 3C	57,4	X	X	X
22700 4C	73,0	X	X	X
27200	80,2	X	X	
RRR 3C	118,5	X		
RIO 4C	95,5	X		
09PM	73,0	X	X	X
09PM	73,0	X	X	X
09PH	82,0	X	X	X
08 0000	14,7	X		
08 7000	17,5	X		
CORAIL 4V	105,6	X		
CORAIL 5V	158,4	X		
CORAIL 5V	152,0	X	X	
CORAIL 7V	184,5	X	X	
CORAIL 8V	211,2	X	X	
CORAIL 13V	290,4	X	X	
CORAIL 14V	388,6	X	X	
09P 0	110,0			X
TGV ATLANTIQUE	257,6	X	X	
TGV RESEAU	200,2	X	X	X

* : rame 6000000 de X2100 + X 80000 + X 2100

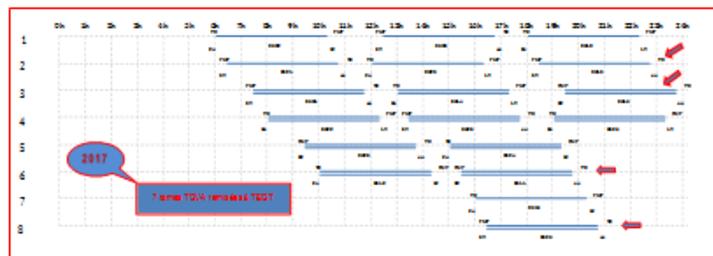


7

Éléments de cadrage 6/7

5-3 Hypothèses d'évolution du remisage TAGV 2017 : la prévision est de 13 AR entre Paris et Toulouse / jour en 2024 avec la mise en service de GPSO. A l'horizon de SEA, 9 AR sont pris en compte en 2017. 1 rame sur 2 est supposée simple (US), l'autre moitié étant double (UM). Avec une durée de crochet minimum de 50min (temps avant un nouveau départ), les roulements et la grille ci-dessous sont obtenus

Ainsi 7 rames TGVA sont à remiser à Toulouse pour la seule relation Paris-Bordeaux-Toulouse



	8601	8801	8603	8605	8803	8611	8805	8613	8616
Paris MIP	06:29	07:29	08:29	12:29	13:29	15:29	16:29	18:29	19:29
Toulouse	10:44	11:44	13:44	16:44	17:44	19:44	20:44	22:44	23:44
Compo	US	US	UM	UM	US	UM	UM	US	UM

	8600	8800	8610	8802	8614	8804	8618	8618	8808
Toulouse	08:04	08:04	10:04	12:04	13:04	15:04	16:04	18:04	19:04
Paris MIP	10:19	12:19	14:19	16:19	17:19	19:19	20:19	22:19	23:19
Compo	US	UM	UM	US	US	UM	US	UM	US

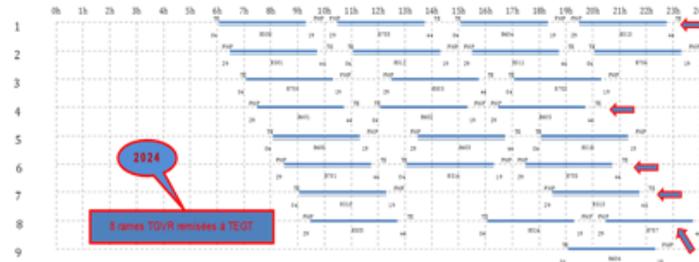


8

Éléments de cadrage 7/7

5-4 Hypothèses d'évolution du remisage TAGV 2024 : 13 AR /jour Paris-Bordeaux-Toulouse sont prévus avec GPSO. Avec un temps de trajet d'environ 3h15, 1 rame peut ainsi réaliser jusqu'à 2 AR dans la journée (avec crochets à 30min minimum). A cet horizon, les rames TGVA sont remplacées par des rames TGVR, passant de 237,6m à 200,2m. A noter qu'1 ligne de roulement sur 2 est envisagée en UM

Ainsi 8 rames TGVA sont à remiser à Toulouse pour la seule relation Paris-Bordeaux-Toulouse



	0701	0601	0701	0601	0701	0601	0701	0601	0701	0601	0701	0601	0701	0601
Paris-MP	06:28	07:29	08:29	09:29	10:29	11:29	12:29	13:29	14:29	15:29	16:29	17:29	18:29	19:29
Toulouse	08:44	10:44	11:44	12:44	13:44	14:44	15:44	16:44	17:44	18:44	19:44	20:44	21:44	22:44
Compt.	08:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12	09:12

	0700	0700	0600	0700	0700	0600	0700	0600	0700	0600	0700	0600	0700
Toulouse	06:04	07:04	08:04	09:04	11:04	12:04	13:04	14:04	15:04	17:04	18:04	19:04	20:04
Paris-MP	08:18	10:18	11:18	12:18	13:18	14:18	15:18	16:18	17:18	18:18	19:18	20:18	21:18
Compt.	07:46	08:46	09:46	10:46	12:46	13:46	14:46	15:46	16:46	17:46	18:46	19:46	20:46



Organisation actuelle du remisage 1/8

DEFINITION
Engins : élément de base du parc insécable (voiture Corail, remorque XR6000, automateur BB1500 ou rame RRP4C), hors engins moteur (remisés sur d'autres installations)
Trains : circulation commerciale (871400 pour TER, 141500 pour TET, 8503 pour TGAV), composée d'1 ou plusieurs engins en fonction de la capacité d'emport souhaitée

1-1 Répartition du remisage TER par gare, le nuit du vendredi au samedi : Les données utilisées pour les calculs sont principalement issues du GOV et des roulements Hiver 2014

Au total, la nuit du vendredi au samedi, 150 engins TER (équivalents à 88 trains) sont remisés dans 21 gares différentes. Pour mémoire, le parc total TER, engins moteurs compris, est de 200

En 2014, 45% du remisage TER Midi-Pyrénées est réalisé sur Toulouse

D'autres sites contribuent significativement au remisage (Rodez, Carmaux...)

Gare	Engins	Trains
Bordeaux-St-Jean	6	3
Toulouse-Matabiau	67	40
Carcassonne	3	3
Tarbes	2	1
Foix	3	2
Brive-la-Gaillarde	5	3
Montréjeau-Gourdan-Pollignan	4	2
Ax-les-Thermes	5	3
Carmaux	11	4
La Souterraine	1	1
Rodez	12	7
Auch	4	3
L'Isle-Jourdain	1	1
Narbonne	1	1
Aurillac	6	4
Latour-de-Carol-Enveitg	6	3
Cahors	1	1
Castres	2	1
Mazamet	3	2
Lourdes	2	1
Pau	5	2

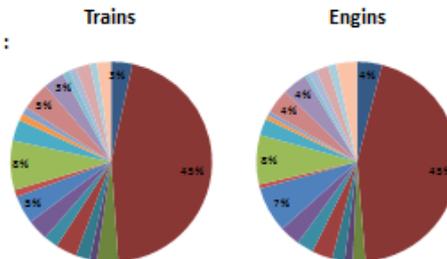


10
70

Organisation actuelle du remisage 2/8

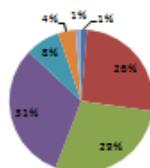
1-2 Part du remisage TER à Toulouse :

Gare	Engins	Trains
Aurillac	6	4
Bordeaux-St-Jean	6	3
Carmaux	11	4
Latour-de-Carol-Enveitg	6	3
Rodez	12	7
Toulouse-Matabiau	67	40

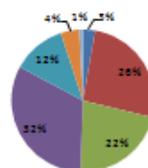


Si l'on décompose l'analyse par site à Toulouse : 80% du linéaire occupé par le matériel TER se trouve sur 3 sites : TE-MM, TE-GRV(L) et TE-GRV(C)

Nombre engins TER



Longueur matériels TER



11
77

Organisation actuelle du remisage 3/8

1-2 Répartition par transporteurs du remisage à Toulouse :

Sans prise en compte à ce stade d'analyse des distances de sécurité entre rames, le matériel TER représente moins de la moitié du linéaire de remisage sur les chantiers du site Toulousain. Ceci s'explique essentiellement en raison de la longueur des trains TET (Corail 11 ou 14V) et TAGV (UM TGV ou TGVA)

Site	ENGINES				TRAINS				LINEAIRE			
	TER	TET	TAGV	Total	TER	TET	TAGV	Total	TER	TET	TAGV	Total
TE-GT	1	14	7	22	1	1	3	7	73	369,6	1625,8	2068,4
TE-GRV(L)	20	28	0	48	8	4	0	12	694,7	739,2	0	1433,9
TE-GRV(C)	16	6	0	22	4	1	0	5	422,4	158,4	0	580,8
TE-MM	24	0	0	14	18	0	0	11	839	0	0	839
TE-AC	6	0	0	6	4	0	0	4	324	0	0	324
TE-MG	3	0	0	3	3	0	0	3	111,6	0	0	111,6
TE-OP	1	0	0	1	1	0	0	1	22,4	0	0	22,4
TOTAL	71	48	7	116	39	6	5	43	2507,1	1267,2	1625,8	5400,1

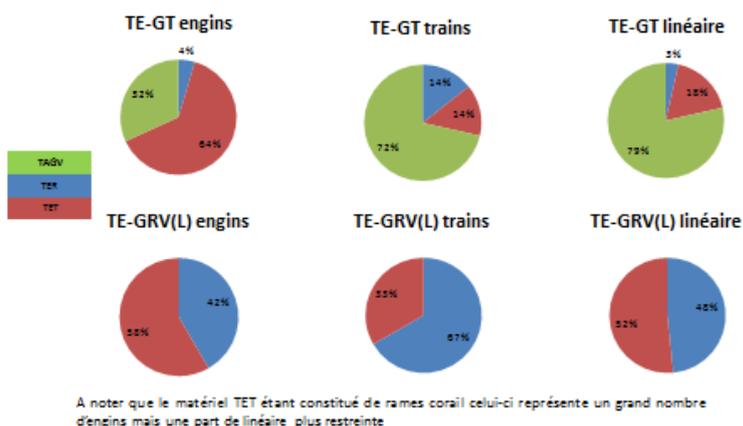
Pour une capacité totale de remisage de 8 156m, le linéaire brut (hors distance de sécurité) est de 5 400m, soit un taux d'occupation global de 66%



12
T3

Organisation actuelle du remisage 4/8

1-3 Utilisation actuelle des différents chantiers de remisage : 2 grands sites sont à ce jour occupés par plusieurs transporteurs, TE-GT principalement par TAGV et TE-GRV(L) partagé entre TER et TET



13
T3

Organisation actuelle du remisage 4/8

1-4 Occupation des différents chantiers de remisage :

A partir de données provenant du TCMP (Technicentre) et du GOV de Matabiau, des affectations par voie de remisage ont été réalisées. Ainsi, tous les 1/4h, l'occupation des voies du site de remisage est représenté. Chaque rectangle représente 1 train, permettant ainsi de visualiser l'état de saturation de la voie (périodes de 15min), le nombre de trains remisés et les distances de sécurité (si nécessaire)

Site TE-AC, du vendredi 20h au samedi 9h (voies 51,52,53)



Définition de l'indicateur de robustesse :

Une situation est considérée robuste lorsque sur au moins 1 des voies de remisage du chantier, la réception d'1 US de la série la plus longue (acceptée sur ce chantier) est possible

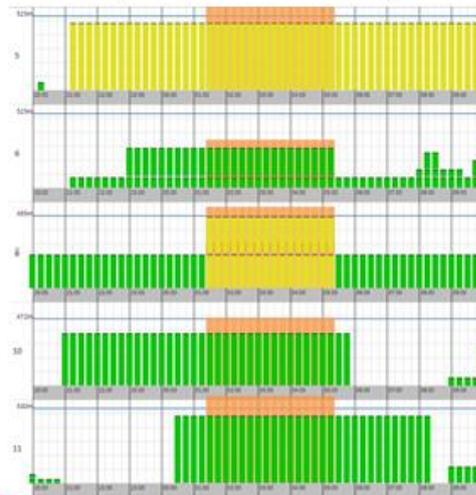
Le non respect du critère de robustesse est représenté en cadre orange autour des trains durant la période « non robuste »



14
T3

Organisation actuelle du remisage 5/8

1-6 Occupation des différents chantiers de remisage :



- situation non robuste
- voie non saturée
- voie saturée
- dépassement de capacité
- capacité de la voie

Site TE-GT, du vendredi 20h au samedi 10h (voies 5,6,8,10,11)

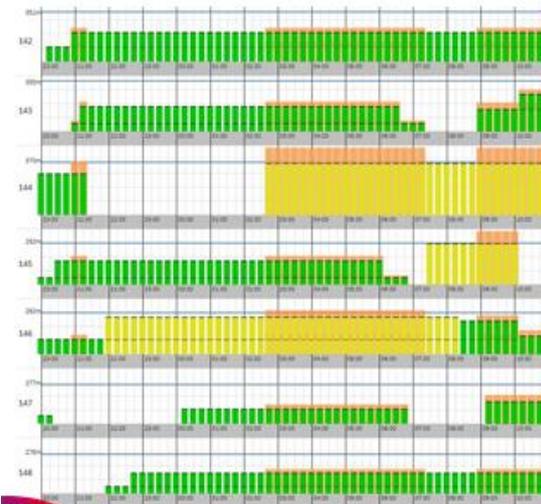
A ce jour, l'organisation du site TE-GT n'est pas en robustesse la nuit du vendredi au samedi car, entre 1h30 et 5h30, le site n'est pas en capacité d'accueillir 1 US TGVA supplémentaire sur l'une de ses voies



15
T2

Organisation actuelle du remisage 6/8

1-7 Occupation des différents chantiers de remisage :



- situation non robuste
- voie non saturée
- voie saturée
- dépassement de capacité
- capacité de la voie

Site TE-GRV(L), du vendredi 20h au samedi 11h (voies 142 à 148)

De même que pour TE-GT, le site TE-GRV(L) n'offre pas de robustesse la nuit du vendredi au samedi. Il est en effet impossible durant plusieurs heures d'accueillir sur le site un Corail 11 ou 14V sur n'importe quelle voie



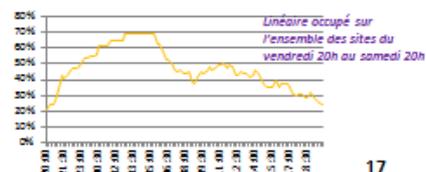
16
T2

Organisation actuelle du remisage 7/8

2-1 Construction d'une vision globale : En calculant le taux d'occupation des trois principaux sites (TE-GT, TE-GRV(L) et TE-MM) du vendredi soir au lundi soir, on remarque que la période nocturne la plus contraignante est bien celle du vendredi au samedi



En prenant en considération la totalité des 7 sites toulousains, une occupation linéaire de quasiment 68% durant la nuit du vendredi au samedi est constatée



17
T3

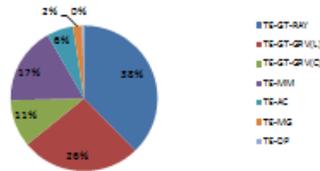
Organisation actuelle du remisage 8/8

2-2 Construction d'une vision globale :

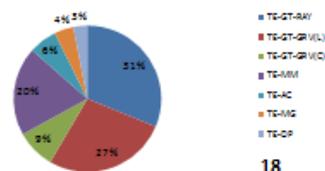
Site	Longueur trains	Distance sécurité	Total	Capacité	Taux de remplissage
TE-GT	2088,4	20	2088,4	2347	82%
TE-GRV(L)	1433,9	22	1433,9	2213	66%
TE-GRV(C)	380,8	0	380,8	699	83%
TE-MM	839	90	949	1598	39%
TE-AC	324	0	324	302	65%
TE-MG	111,6	0	111,6	343	32%
TE-DP	22,4	0	22,4	230	9%
TOTAL	5400,1	132	5532,1	8156	68%

Le taux de remplissage global est de 68% en prenant en compte les distances de sécurité. On remarque que les sites TE-GT, TE-GRV(L) et TE-MM représentent quasiment 80% des capacités linéaires de remisage

Longueur matériels remisés



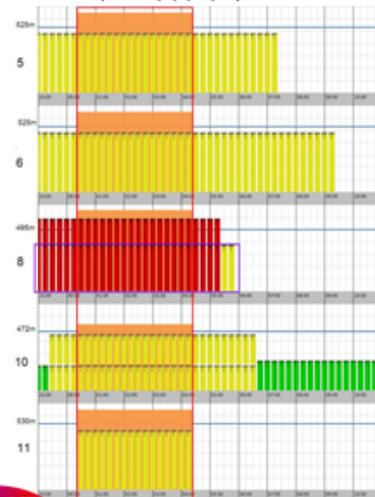
Capacité linéaire



18
T8

Modélisation des besoins futurs de remisage 1/8

Projection 2017 TE-GT du vendredi 23h au samedi 11h (voies 5,6,8,10,11)

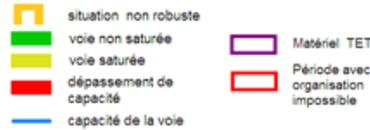


1-1 Projection du mode d'affectation actuel à TE-GT en 2017 :

Conformément aux hypothèses retenues, 9 rames TAGV sont à remiser en 2017 :

- 7 rames US TGVA (Paris-Toulouse)
- 1 rame US TGVR (Toulouse-Lyon)
- 1 rame US TGVR (Toulouse-Barcelone)

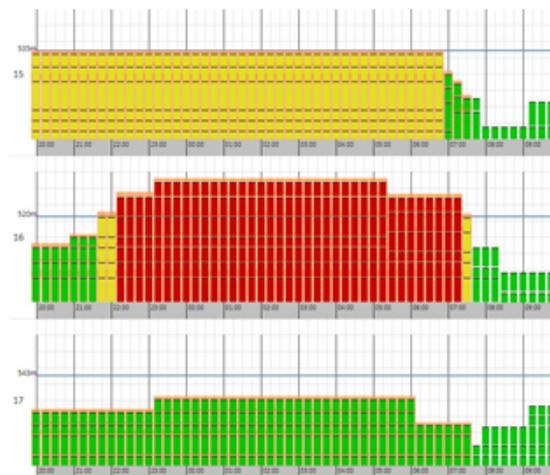
L'augmentation du nombre de rames TGV à remiser ne permet plus de fonctionner avec les allocations de voies actuelles. La première piste est de reporter le matériel TER sur un autre site, mais cette résolution est insuffisante. En effet, le remisage de matériel TET (corail 14V) rend, de fait, impossible le remisage des rames TAGV



19
T8

Modélisation des besoins futurs de remisage 2/8

1-3 Projection du mode d'affectation actuel en 2017 à TE-MM



TE-MM en 2017 en suivant l'organisation actuelle (voies 15,16,17)

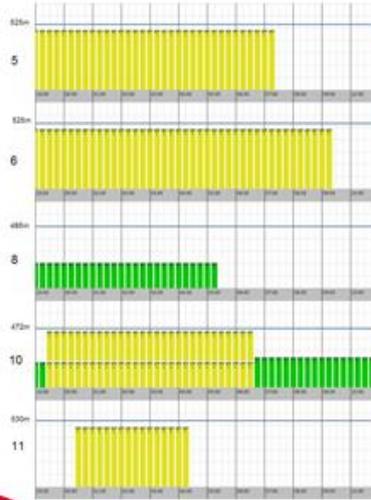
De même que pour TE-GT l'organisation actuelle n'est plus valide dès 2017 sur TE-MM. Intégralement occupé par TER, le site subit le changement de matériel. Le cumul du linéaire occupé atteint même 1 642m alors que la capacité totale du site est de 1 598m. Des engins doivent donc être déplacés vers un autre site



20
S0

Modélisation des besoins futurs de remisage 3/8

2-1 Optimisation de l'allocation des capacités 2017 TE-GT



En choisissant une solution de remisage où seul le matériel TAGV est remisé sur les voies du site TE-GT, le plan d'allocation obtenu est alors satisfaisant. Une solution est en effet possible en respectant les paramètres de robustesse.
La place pour 1 rame US TGVA est en effet disponible sur la voie 8 durant toute la nuit.

Vision du site TE-GT en 2017 du vendredi 23h au samedi 11h sans matériels TET et TER (voies 5,6,8,10,11)

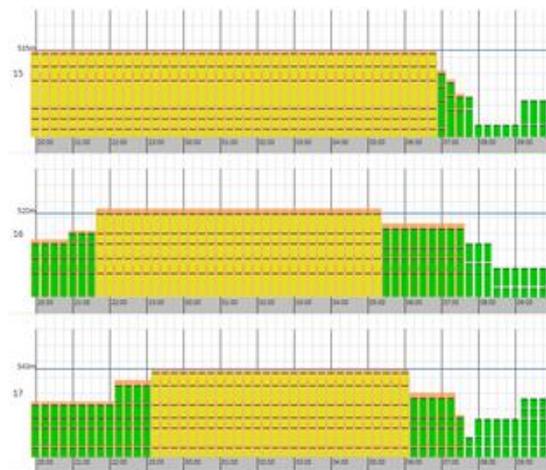
- situation non robuste
- voie non saturée
- voie saturée
- dépassement de capacité
- capacité de la voie



21
57

Modélisation des besoins futurs de remisage 4/8

2-2 Optimisation de l'allocation des capacités 2017 TE-MM



Le site TE-MM en 2017 du vendredi 20h au samedi 10h, après report d'un train sur le site TE-GRV(L) et optimisation (voies 15,16,17)

Afin de pouvoir trouver une organisation valable, le report d'un train (1 rame US ZPPM) vers le site TE-GRV(L) est obligatoire. Avec ce report, le logiciel d'optimisation Gusek a permis de trouver un nouveau plan valable mais où le site est entièrement saturé.

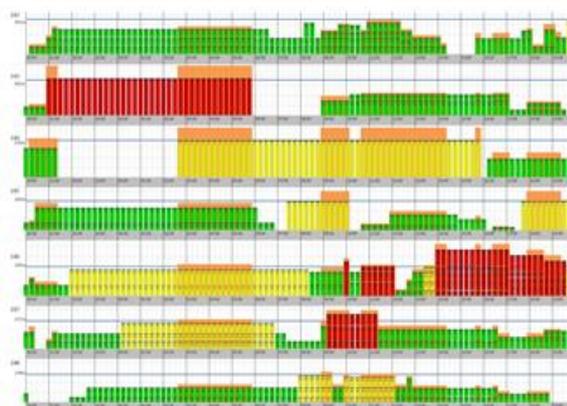
- situation non robuste
- voie non saturée
- voie saturée
- dépassement de capacité
- capacité de la voie



22
55

Modélisation des besoins futurs de remisage 5/8

2-3 Optimisation de l'allocation des capacités 2017 TE-GRV(L)



- situation non robuste
- voie non saturée
- voie saturée
- dépassement de capacité
- capacité de la voie

En utilisant le système d'allocation par voies actuel du site TE-GRV(L), il est impossible d'accueillir les engins TET et TER « exclus » de TE-GT et TE-MM, que ce soit durant la nuit ou même durant la journée de samedi. Un autre schéma doit être trouvé.

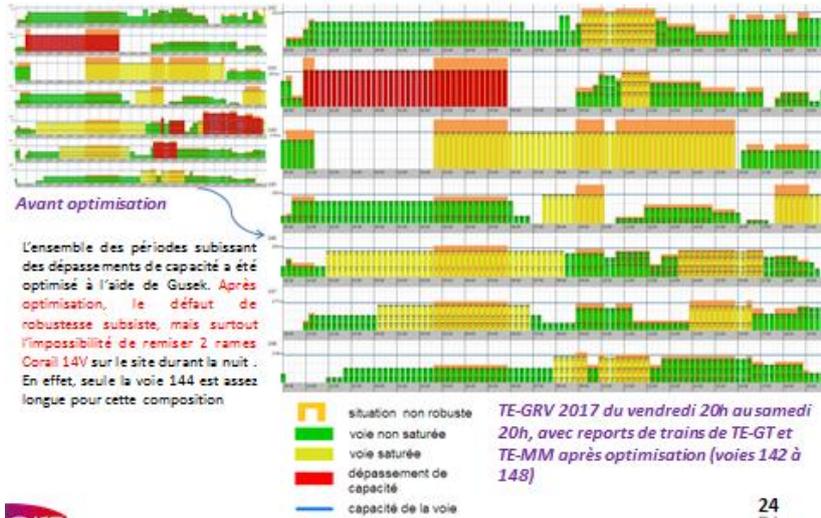
Vision du site TE-GRV voies longues (voies 142 à 148) en 2017 du vendredi 20h au samedi 20h, avec report des engins TET et TER des sites TE-GT et TE-MM, sans optimisation



23
53

Modélisation des besoins futurs de remisage 6/8

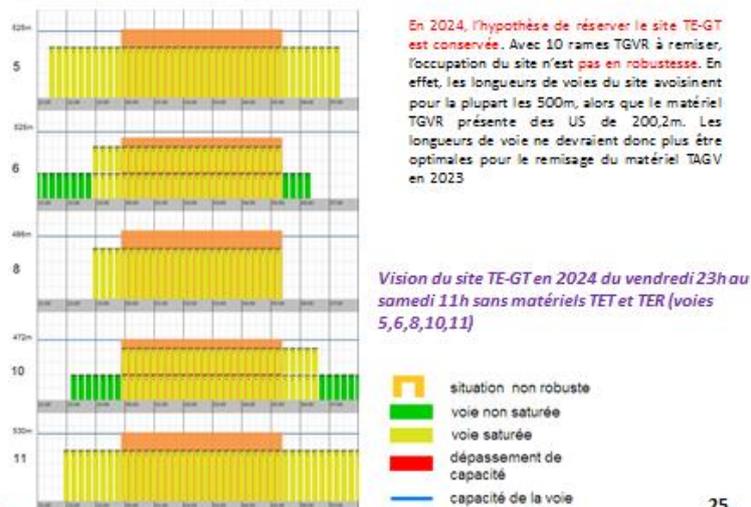
2-4 Optimisation de l'allocation des capacités 2017 TE-GT-GRV(L)



24

Modélisation des besoins futurs de remisage 7/8

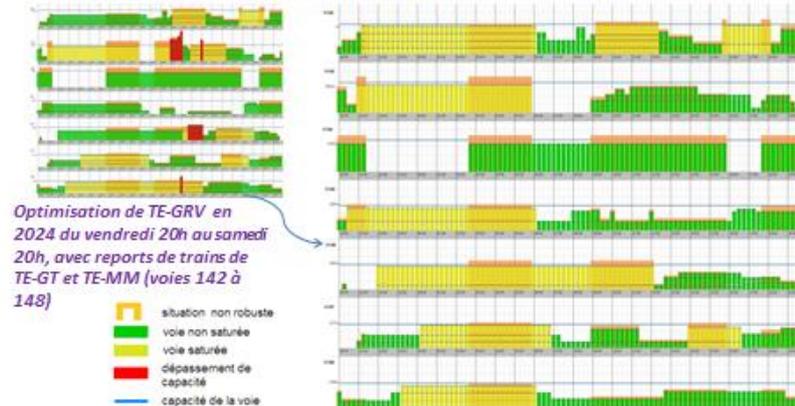
2-6 Optimisation de l'allocation des capacités 2024 TE-GT



25

Modélisation des besoins futurs de remisage 8/8

2-7 Optimisation de l'allocation des capacités 2024 TE-GRV(L)



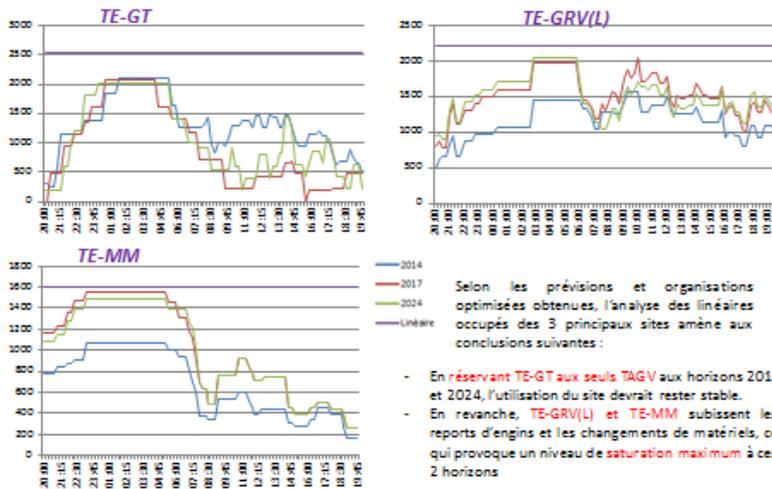
Comme pour l'horizon 2017, des reports de trains TER et TET des sites TE-GT et TE-MM doivent être faits. Si l'on reporte ces trains vers TE-GRV(L), une organisation est désormais possible sans dépassement de capacité. Cependant cette organisation est toujours sans robustesse et le linéaire remisé est même en augmentation : 2 045m de linéaire occupés en 2023 contre 1 971m en 2017, pour une capacité maximale du site de 2215m



26

Enseignements et compléments à réaliser 1/3

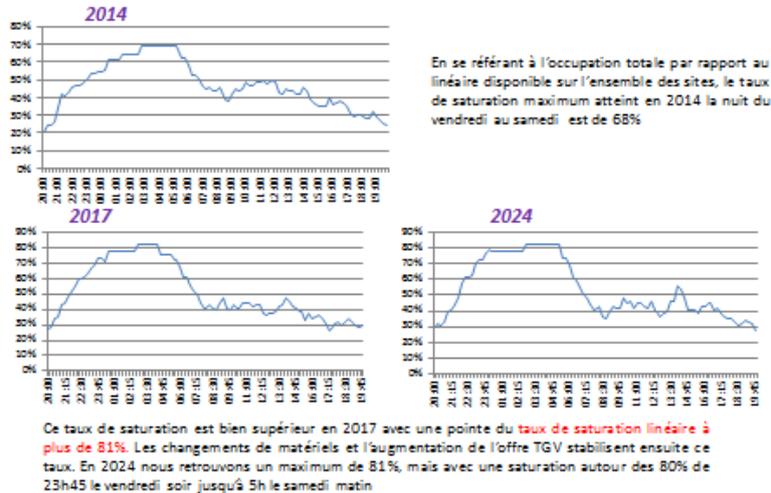
1 Linéaire occupé aux 3 horizons, par site, du vendredi 20h au samedi 20h



27
53

Enseignements et compléments à réaliser 2/3

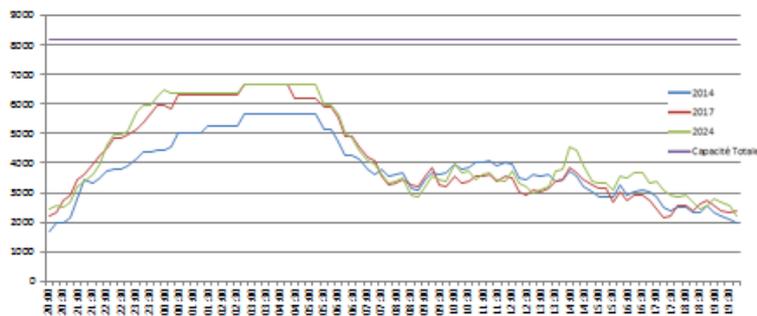
2 Linéaire total occupé par horizon, du vendredi 20h au samedi 20h



28
58

Enseignements et compléments à réaliser 3/3

3 Récapitulatif linéaire total occupé par horizon, du vendredi 20h au samedi 20h



On retrouve bien cette évolution en comparant le total maximum de mètres occupés de chaque horizon :

- Un maximum de 5 532m de matériel et de distance de sécurité atteint en 2014
- Un maximum de 6 667m en 2017, soit une progression de 1029 mètres ou 18%
- Un maximum de 6 681m en 2024, soit seulement 14m de plus qu'en 2017 mais toujours une augmentation de 18% par rapport à 2014



29
59