THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ FÉDÉRALE TOULOUSE MIDI-PYRÉNÉES

Délivré par :
l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)

Présentée et soutenue le *04/12/2017* **par :**

Yun HE

PROBLÈMES DE TOURNÉES AVEC GESTION DE STOCK ET PRISE EN COMPTE EXPLICITE DE LA CONSOMMATION D'ÉNERGIE

Inventory Routing Problems with Explicit Energy Consideration

JURY

Nabil ABSI	Professeur	École des Mines St-Étienne
Cyril BRIAND	Professeur des universités	Université de Toulouse III
Nicolas JOZEFOWIEZ	Professeur des universités	Université de Lorraine
Dominique QUADRI	Maître de conférences	Université Paris Sud
Frédéric SEMET	Professeur des universités	Ecole Centrale de Lille
Maria Grazia SPERANZA	Professore Ordinario	University of Brescia

École doctorale et spécialité :
EDSYS : Informatique 4200018
Unité de Recherche :
Laboratoire d'analyse et d'architecture des systèmes (LAAS-CNRS)
Directeur(s) de Thèse :
Cyril BRIAND et *Nicolas JOZEFOWIEZ*
Rapporteurs :
Frédéric SEMET et *Maria Grazia SPERANZA*

Acknowledgement

First of all, I would like to thank Prof. Maria Grazia Speranza and Prof. Frédéric Semet for accepting to be the reporters of this thesis. Thank you for the reports and the time that you have dedicated for this thesis.

I would also like to thank Prof. Dominique Quadri and Prof. Nabil Absi for accepting to be members of the jury. Thank you for your work as examiners.

Next, I would like to thank my supervisors—Prof. Cyril Briand and Prof. Nicolas Jozefowiez for their guide and help during all these years. Thank you for your knowledge, your patience and your serious way for doing research. I have learnt a lot from you and I really appreciated working with you.

I would also like to pay special acknowledgement to Christian Artigues and Sandra Ulrich Ngeuveu. As a team with my two supervisors, we participated in the ROADEF 2016 Challenge and it has been a really unforgettable experience.

Thank you, all the members in the ROC team. Whether permanent or doctorant, you are all so kind and you have helped me a lot. I enjoyed many beautiful moments with you and I felt really lucky and happy being with you here in Toulouse.

Thanks to everybody who has helped in one way or another during the three years of thesis.

Finally, I would like to thank my parents and my sister, for all their supports and encouragement in all these years.

Je tiens à vous remercier tous. Merci!

Yun He
December 8, 2017

Abstract

The thesis studies the Inventory Routing Problem (IRP) with explicit energy consideration. Under the Vendor Managed Inventory (VMI) model, the IRP is an integration of the inventory management and routing, where both inventory storage and transportation costs are taken into account. Under the new sustainability paradigm, green transport and logistics has become an emerging area of study, but few research focus on the ecological aspect of the classical IRP. Since the classical IRP concentrates solely on the economic benefits, it is worth studying under the energy perspective. The thesis gives an estimation of the energetic gain that a better supplying plan can provide.

More specifically, this thesis integrates the energy consumption into the decision of the inventory replenishment and routing. It starts with a part supplying problem in car assembly lines, where the transported mass, the vehicle dynamics and the travelled distance are identified as main energy influencing factors. This result is extended to the classical IRP with energy objective to show the potential energy reduction that can be achieved. Then, an industrial challenge of IRP is presented and solved using a column generation approach. This problem put the limitations of the classical IRP model in evidence, which brings us to define a more realistic IRP model on a multigraph. Finally, a Lagrangian relaxation method is presented for solving this new model with the aim of energy minimization.

Keywords

Inventory Routing Problem, Logistics, Energy

Résumé

Dans le problème de tournées avec gestion de stock ou “Inventory Routing Problem” (IRP), le fournisseur a pour mission de surveiller les niveaux de stock d’un ensemble de clients et gérer leur approvisionnement en prenant simultanément en compte les coûts de transport et de stockage. Etant données les nouvelles exigences de développement durable et de transport écologique, nous étudions l’IRP sous une perspective énergétique, peu de travaux s’étant intéressés à cet aspect.

Plus précisément, la thèse identifie les facteurs principaux influençant la consommation d’énergie et évalue les gains potentiels qu’une meilleure planification des approvisionnements permet de réaliser. Un problème relatif à l’approvisionnement en composants de chaînes d’assemblage d’automobiles est tout d’abord considéré pour lequel la masse transportée, la dynamique du véhicule et la distance parcourue sont identifiés comme les principaux facteurs impactant la consommation énergétique. Ce résultat est étendu à l’IRP classique et les gains potentiels en termes d’énergie sont analysés. Un problème industriel de tournées avec gestion de stock est ensuite étudié et résolu, notamment à l’aide d’une méthode de génération de colonnes. Ce problème met en évidence les limitations du modèle IRP classique, ce qui nous a amené à définir un modèle d'IRP plus réaliste. Finalement, une méthode de décomposition basée sur la relaxation lagrangienne est développée pour la résolution de ce problème dans le but de minimiser la consommation énergétique.

Mots clés :

Problème de tournées de véhicules avec gestion de stock, Logistique, Énergie

Contents

List of Figures

List of Tables

Introduction

With limited natural resources, energy has become a critical factor for the sustainable development of human beings. Energy consumption has been closely related to the global economic development. Of all the world energy production, 81.2% is attributable to fossil fuels [87].

Nonetheless, the reserves for fossil fuels are quite limited. They could possibly run out by the end of this century with today's consumption rate. Besides, due to the massive consumption of the fossil energy resources driven largely by economic and population growth, the high emissions of Greenhouse Gases (GHGs) are extremely likely to have been the dominant cause of the climate change in recent decades [89].

This alarming energy situation requires a reflection on the use of energy and a change in our energy consuming behaviours. According to the World Energy Outlook 2016 of International Energy Agency (IEA), the energy efficiency improvement in various economic sectors could be the motor of change [88].

Industry and transport sectors account for more than 60% of the total final energy consumption worldwide [87]. However, in most of the time, attention is only paid to the economic benefits. The energy efficiency is considered less vital and the potential energy savings are often neglected compared to the economic ones. Therefore, the energy efficiency of the industrial production and transportation systems needs to be taken into consideration. Reports have shown that by adopting renewable energy consumption scenarios with emphasis on sufficiency and efficiency, it is possible to reduce by half the final energy consumption by 2050, while ensuring the same level of energy services [106].

To achieve a future with efficient energy usage, the first step is to study the current systems with energy consideration. This has led to concepts such as Energy Aware Manufacturing (EAM) [110], green logistics, sustainable supply chain management [68], to name just a few. In this thesis, the energy issues of the supplying systems with vehicles are studied explicitly.

In general, the problem studied in this thesis is a problem of *combinatorial optimization*. It belongs to a family of problems called Vehicle Routing Problems (VRPs). This family of problems is concerned with the distribution of products using a set of vehicles in a transportation network. The VRP has been studied for nearly 60 years and it is still an active area of research with many applications [47, 98]. It involves decisions on which route to take by each vehicle to serve a set of customers under certain restrictions such as the vehicle capacity. The demands of customers are assumed known in advance and are not part of the decisions. The objective is to minimize the total costs of routing. In most of the cases, it is the economic costs such as the distance or the total travel time that are considered.

Since the 1980s, a variant of VRP has emerged under the context of fast evolution of information systems, deriving the so-called Inventory Routing Problem

(IRP). In this variant of routing problems, the decisions on transportation and inventory management are taken simultaneously. It consists of monitoring the inventory levels of a set of customers. Deliveries are made whenever it is necessary and cost efficient. It usually expands to a horizon of several days or even longer. A particularity of this problem is that the demands of the customers are now considered as decisions. In addition, decisions made at one period of time influence the decisions to make later. The objective function often includes costs of both inventory storage and transportation. The combined decisions in the IRPs can improve the economic efficiency of the inventory management systems. However, the energy issue in the IRP was seldom studied.

Recently, as people are becoming more and more concerned with the energy issue, Green Vehicle Routing Problems (GVRPs) emerge as a new variant of VRP. It concerns a family of VRPs with a consideration for energy-efficiency, air-pollution or GHG emissions. In GVRPs, the ecological aspects are integrated into different variants of VRPs by setting new objectives, adding more constraints or new decision variables. It has been shown that the current transportation systems can be improved to be more energy-efficient [25, 64, 72].

Although GVRP is a popular area of study for the integration of energy issues into transportation systems, the study on the energy aspect of the IRP has just started. The problems studied in this thesis are an integration of energy consideration into the IRP. It models the energy as a cost that is linearly dependent on the transported mass, which links the inventory control and the product routing. It identifies the most influencing factors to vehicle energy consumption and shows the potential energy improvement in the current inventory routing systems. New features introduced to the IRP by the energy aspect are also highlighted.

According to the theory of complexity, the combinatorial optimization problems can be classified into P—problems that can be solved in deterministic polynomial time, and NP—problems that are solvable in non-deterministic polynomial time [77]. The category of VRPs is one of the most famous set of *NP-hard* problems and unless the condition P = NP is proved to be true, it cannot be solved by a polynomial algorithm. To deal with such problems, there are mainly two ways: one is *heuristic methods* that are fast but do not guarantee the optimality of the solution. The other is *exact methods* which ensure the quality of solution but are more computationally expensive.

With the ecological considerations, the problem usually gets much more complicated. Various approaches have been used to solve the GVRPs. Most of them are adapted from the methods for solving the VRPs. For the IRP, various heuristics exist, as well as a few exact methods. In this thesis, several mathematical models for the solution of the IRP with energy consideration are proposed. These models are helpful for developing advanced solution methods based on mixed integer programming. Decomposition methods based on column generation are also applied to solve a real-life IRP. Lagrangian relaxation based decomposition is provided for solving a simple version of the real-life IRP with energy consideration.

This thesis has been elaborated in the group “Operations Research, Combinatorial Optimization and Constraints (ROC)" of the Laboratoire d'analyse et d'architecture des systèmes of Centre national de la recherche scientifique (LAAS-CNRS) in Toulouse, France. It has been funded by the grant program called "Allocation Prioritaire Thématique" of the University of Toulouse III Paul Sabatier.

The thesis is divided into three parts, each part with two chapters:

Part I is the state of the art. Chapter 1 defines the combinatorial optimization problems in general, presents the VRPs and several variants that has been useful to integrate the energy aspect. It also explains the basic methods for the solution of these problems. Chapter 2 presents the IRP and the GVRP. Different types of IRPs are reviewed with regard to different application backgrounds. Common solution methods for the IRPs are presented as well as some well-known benchmarks. The GVRPs is then introduced with a discussion on energy modelling methods and energy influencing factors. Solution methods for the GVRPs are also presented with some of the benchmarks seen in the literature.

Part II is dedicated to the integration of the energy consumption into supplying and routing problems. Two problems are studied in details in this part. One is the Energy-Efficient Assembly-line Vehicle Supplying Problem (EEAVSP) and the other is the Inventory Routing Problem with Energy Consideration (IRP-EC). In Chapter 3, the EEAVSP is presented. It is a raw-material feeding problem encountered inside the assembly lines, with one fixed delivery route per period. The energy consumption of a vehicle is estimated according to the vehicle dynamics in relation to the supplying planning and the transported mass, assuming a fixed speed profile. Since the energy consumption is linearly dependent on the transported mass, two mathematical formulations based on the flow of mass are presented. The results show that the energy consumed by the transportation of spare parts in a supply chain is connected with the planning of the feeding activity. It can be highly reduced by making better decisions on the feeding quantity and time. Chapter 4 generalizes the EEAVSP to road networks and defines the IRP-EC. The energy cost is added as an objective for the inventory routing. Energy consumption of a vehicle is estimated in the same vein as for the EEAVSP by adding information on traffic conditions on road networks. The results of the experiments reveal the importance of energy consideration for the combined inventory management and routing systems.

Part III includes a real-life IRP in Chapter 5 and a simplified version with energy consideration in Chapter 6. The real–life IRP is adapted from the ROADEF 2016 Challenge problem [6]. It considers many industrial constraints and decisions that are not commonly dealt with in classic IRPs, such as the scheduling of the drivers, the timing of visits and the continuous monitoring of the customer inventory levels, among others. It is solved using a decomposition method based on column generation. In Chapter 6, a simplified version of the real-life IRP is presented with consideration on energy consumption. This version keeps the decisions on the timing of visits and inventory monitoring in continuous time, and adds the energy consideration. Since travel duration can influence both the energy consumption of the vehicle on a road segment and also the inventory variation of a customer, a multi-graph model is proposed, where each arc is characterised by the cost of energy as well as the travel duration. The solution method for this Multi-Graph Inventory Routing Problem with Energy Consideration (MG-IRP-EC) is a decomposition method based on Lagrangian relaxation. Some preliminary results are given at the end.

Finally, the works of the thesis are concluded with future research directions.

Part I

State of the Art

In combinatorial optimization, routing is an important part that intervenes in many applications. During all these years, with various application contexts, the problem has evolved and transformed into different variations.

With the development of the information systems, there is a trend to integrate different parts of a large system to achieve a better performance of the whole system. The *IRP* is a beautiful example of this integration. It considers inventory management and routing at the same time for a more efficient utilisation of resources.

On the other side, in a world with limited natural resources, energy issues are becoming more and more important. The integration of energy consideration is a new tendency among both practitioners and academic researchers. The *GVRP* is an emergent area of study that analyses the environmental aspect of the distribution systems.

This part is dedicated to the state of the art. In Chapter 1, we make a brief literature review of the general routing problems and principles of some solution methods that are relative to the works presented in this thesis. In Chapter 2, the IRP and the GVRP are presented in general and the integration of energy consumption into the IRP is introduced.

Chapter 1

Routing Problems and Their Solution Methods

Combinatorial optimization arises in everyday applications such as assignment, transportation, scheduling and so on. The Travelling Salesman Problem (TSP) is one of the most widely studied problem of combinatorial optimization. A natural generalisation of the TSP is the VRP, which concerns the distribution of products from central depots to final users.

This chapter introduces the general notion of the routing problems and their solution methods. We start with a formal definition of combinatorial optimization. Then the TSP is presented, which also serves as an example to show the modelling by graph and as an introduction to the notion of complexity. After that, several types of VRPs where energy issues can be integrated are presented. A brief review of some basic solution techniques concerned in this thesis is provided in the end.

1.1 Combinatorial Optimization

To optimize means to minimize (or maximize) a measure of performance of a system by choosing the value of certain variables under certain restrictions. A mathematical model in operations research is the system of equations and related mathematical expressions that describe the essence of the problem [86]. It usually contains a set of quantifiable *decision variables*, a set of *constraints*, and an *objective function*. The constraints are mathematical equations/inequalities or logical relationships for the restrictions of values that can be taken by each variable. The objective is typically a mathematical expression of the decision variables for the measure of performance. The constants (namely, the coefficients and right-hand sides) in the constraints and the objective function are called the *parameters* of the model. An *instance* of an optimization problem is a data set for the parameters. The mathematical model might then say that the problem is to choose the values of the decision variables so as to minimize (or maximize) the objective function, subject to the specified constraints [86].

Mathematically, a general optimization problem of n variables in $\mathbb{R}$ can be defined as:

$$\begin{aligned} \text{minimize} \qquad & c(x) \\ & x \in S \end{aligned}$$

where the set $S \subset \mathbb{R}^n$ is defined by some constraints and $x \in S$ is a vector of n variables. In particular, if the constraints and the objective are all linear and variables are all continuous, then the problem is called *linear* and can be solved by Linear Programming (LP).

The set S is called the *search space* or *feasible region* of the problem. If $S \neq \emptyset$, then the initial problem is *feasible* and S defines the set of *feasible solutions*. A solution x^* such that $\forall x \in S$, $c(x^*) \leq c(x)$ is called a globally *optimal solution*. For a subset $S_1 \subset S$, if x_1 has the property that $\forall x \in S_1$, $c(x_1) \leq c(x)$, then x_1 is called a *local optimum*.

If an optimization problem contains a set of $M \geq 2$ objective functions $(c_1, c_2, \ldots, c_M)$ it is a *multi-objective* optimization problem. In the minimization case, a solution x_1 is said to be *dominated* by a solution x_2 if and only if $\forall i \in \{1, 2, \ldots, M\}$, $c_i(x_1) \geq c_i(x_2)$ and $\exists j \in \{1, 2 \ldots, M\}$, $c_j(x_1) > c_j(x_2)$. A solution $x \in S$ is called *Pareto optimal* if it is not dominated by any other $x' \in S$. And the set of Pareto optimal solutions defines the *Pareto optimal set* and its mapping in the objective space is called the *Pareto front*.

The distinguishing feature of *discrete*, *combinatorial*, or *integer optimization* is that some of the variables are required to belong to a discrete set, typically a subset of integers. These discrete restrictions allow a mathematical representation of phenomena or alternatives where indivisibility is required or where there is not a continuum of alternatives [107].

Generally, let $N = \{1, \ldots, n\}$ be a finite set and $c = (c_1, ..., c_n)$ be an n–vector. For $F \subseteq N$, we define $c(F) = \sum_{j \in F} c_j$. Given a collection $\mathcal{F}$ of subsets of N, the *combinatorial optimization problem* is defined as [107]:

CP:

$$\text{minimize} \quad c(F) \qquad F \in \mathcal{F}$$

If the problem is linear and if the set N contains both continuous and integer values, then this problem is considered a Mixed Integer Linear Programming (MILP) problem.

In the following section, the TSP is introduced followed by the presentation of a category of combinatorial optimization problems—the VRPs, which is the basic problem of this thesis.

1.2 Vehicle Routing Problems

Vehicle Routing Problems (VRPs) is a family of problems encountered notably in transportation. These problems are to find the most efficient way to dispatch either passengers or goods from central origins to certain destinations under certain restrictions. In addition to the application in transportation and logistics, VRPs

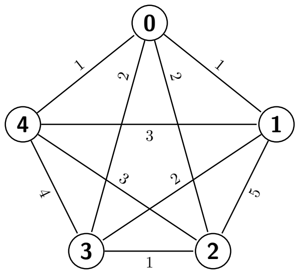


Figure 1.1: Example of TSP with five cities

are also widely applied to the design of electric circuit or computer networks, for instance.

1.2.1 Travelling Salesman Problem

Given a list of cities, a salesman, starting from his home city, is to visit each city on the list exactly once, and return home. Suppose that the salesman knows the distance between each two cities, the Travelling Salesman Problem (TSP) is to find the order of visit to each city to minimize the total travelled distance from all the possible combinations of the orders. The TSP is one of the most studied combinatorial problems. Figure 1.1 shows an example of TSP with five cities. Each city is represented by a node numerated from 0 to 4 and the cost between each pair of cities is marked by the number beside each arc of the graph. This example shows a symmetric TSP since the graph is not directed.

To solve such a problem is not easy. Actually, the TSP as expressed above can be categorized into a type of problems called "*NP-hard*". That is to say, all the problems in the class of *Non-deterministic Polynomial time (NP)* can be reduced to the TSP [82] and unless the condition P = NP is proved to be true, it is not sure to be solved by a polynomial algorithm.

A formal mathematical definition of the TSP can be given using some notations in the graph theory. Let $G = (V, A)$ be a graph (directed or undirected) with V the set of vertices and A the set of arcs. Each vertex represents a city and each arc $(i, j) \in A$ represents the way between two cities i and j. The distance D_{ij} between each pair of cities $i, j \in V$ is assumed known. The tour that visits each city in the list once and exactly once is also called a "Hamiltonian cycle" in the graph theory. Then the TSP is to find a tour (Hamiltonian cycle) in $G = (V, A)$ such that the total costs of edges on the tour is the smallest.

1.2.2 Capacitated Vehicle Routing Problem

In general, Vehicle Routing Problems (VRPs) are a family of problems that considers the distribution of goods to a set of *customers* in a given time period by a *fleet* (a set of *vehicles*). The vehicles are located in one or more *depots* and operated by a set of *drivers*. They move inside an appropriate *road network*. In particular, the solution of a VRP calls for the determination of a set of *routes*, each performed by a single vehicle that starts and ends at its own depot, such that all the requirements of the customers are fulfilled, that all the *operational constraints* are satisfied and that the global *transportation cost* is minimized. [124]

The Capacitated Vehicle Routing Problem (CVRP) is the simplest and most studied member of the family of VRPs. It first appeared in [47] under the name of "truck dispatching problem". In this problem, there is a set of customers $\mathcal{Z}$, each $i \in \mathcal{Z}$ having a known non-negative demand R_i to be delivered by a *fleet* $\mathcal{K}$ of identical vehicles, each with the same capacity B. Each time the vehicle traverses an arc (i, j), there is a cost $c_{i,j}$ that can be either the distance or the travel time on this arc. In the CVRP, each vehicle performs at most one route. Each route starts and ends from a central depot and visits each customer at most once. Each customer is visited by at most one vehicle route.

The CVRP is a generalisation of the TSP since in each vehicle tour a TSP has to be solved. What distinguish the CVRP and the TSP is the fact that $B \ll \sum_{i \in \mathcal{Z}} R_i$, so all the customers cannot be delivered in the same vehicle route. To ensure feasibility, it is assumed that $R_i \leq B$ for all $i \in \mathcal{Z}$. The total demands of customers visited in one route should not be larger than the vehicle capacity. The objective is to find the routes of vehicles that minimize the total costs of arcs belonging to the routes.

The CVRP is at the source of many routing problems. Please refer to the book [125] for the presentation of the problem and its solutions. A more recent book [127] presents the VRP with some new variants and new solution methods. In the remainder of this section, several variants are briefly presented as basic problems of the IRP and GVRP that we are going to present in Chapter 2.

1.2.3 Other Variations of Vehicle Routing Problems

Closely related to real-life applications, the VRP has attracted extensive research attention for over 50 years. Interests for the VRPs have never ceased and efforts are continuously being made to develop more realistic models and more efficient algorithms. Here are some variants of VRP that are mentioned in Chapter 2 to show the wide possibility of the integration of energy consideration into routing problems.

1.2.3.1 Vehicle Routing Problem with Time Windows

The Vehicle Routing Problem with Time Windows (VRPTW) is the extension of the CVRP where the service at each customer must start within an associated time window and the vehicle must remain at the customer location during service [45]. In addition to demand, each customer is associated with a visiting time window defined by an earliest and a latest visiting time, and also a service time smaller than the duration of the time window. Each visit of a customer by a vehicle must be included in the corresponding visiting time window. The VRPTW models can better reflect the situation where customers are not open all the time. The presence of time windows can also have an influence on the energy consumption on route. For example, in order to arrive in time to a customer, the vehicle might choose a fast route with more consumption than a slow one with less consumption.

1.2.3.2 Time-Dependent Vehicle Routing Problem

Traditional VRP considers constant travel distances or travel times. This is obviously not true in the real road network where traffic parameters change from the morning to the night. The distinctive characteristics of the Time-dependent Vehicle Routing Problem (TD-VRP) is that the travel time between any pair of points depends on the starting time of tour, the starting time of traversing an arc, or the distance between the points. An early review of the TD-VRP can be found in [104]. Due to its distinctive characteristics, TD-VRP is relevant and useful to account for the actual conditions such as urban congestion, where the speed of the vehicle is not constant (as well as the travel time) due to variation in traffic density as in [71]. Since speed and travel time is important for energy, the TD-VRP is often considered a starting point for energy integration.

1.2.3.3 Vehicle Routing Problem with Heterogeneous Fleet

The VRP with heterogeneous fleet can be seen as an example of “rich” VRP which is closer to the reality with multiple depots, multiple trips per vehicle, multiple vehicle types and so on. In the VRP with heterogeneous fleet, the vehicles are not identical. They are often associated with different capacities and different using costs. Together with the VRPTW, the problem could also account for the situation where the vehicle works in different times of a day. In addition, different types of vehicles could introduce different energy costs because of their different weights and engine powers. Please refer to the book chapter [19] for a detailed review of the VRP with heterogeneous fleet.

1.2.3.4 Vehicle Routing Problem with Pickup and Delivery

In the Vehicle Routing Problem with Pickup and Delivery (VRPPD), a heterogeneous fleet based at multiple terminals must satisfy a set of transportation requests. Each request is defined by a pickup point, a delivery point and a demand to be transported between these two points. There are often time windows to be satisfied at each stop [50]. In addition, the pickup and delivery points are to be visited exactly once in a route of the same vehicle without exceeding the capacity of the vehicle. Precedence constraints are also imposed to make sure that pickup points are visited before delivery points. The vehicles should return to the corresponding terminals at the end of each tour and there are resource restrictions on the number of drivers and vehicle types. The VRPPD can model the problem for the pickup and throw of wastes in urban areas [14, 54].

1.3 Solution Methods

For solving NP-hard problems in general, two strategies are applied in practice. The first defines the category of solution methods called "heuristics", which is fast and accepts the possibility of a suboptimal solution, while the second is called the "exact methods", which focus on the optimality of the solution but can be very time-consuming [100].

In this section, a brief review on these two categories methods is made with emphasis on exact methods. Then, decomposition methods for solving real size instances are particularly introduced to make clear some basic ideas of the solution methods applied in this thesis.

1.3.1 Heuristics

The heuristics are fast solution methods without optimality guarantees. There are many types of heuristics for the solution of combinatorial optimization problems. Simple heuristics are often applied to find an initial feasible solution in a short time for further improvement. The metaheuristics are methods that employ special strategies to explore the neighborhood structures of complex solution spaces.

Taking the VRP as an example, classical heuristic methods include constructive heuristics, two-phase heuristics and improvement methods [99]. The constructive heuristics gradually build a feasible solution while keeping an eye on the cost. The two-phase heuristcs decompose the solution into two parts, one aiming at clustering vertices into feasible routes and the other constructing the routes. The improvement methods upgrade a feasible solution by performing a sequence of edge or vertex exchanges within or between vehicle routes. Besides, there exist a wide variety of metaheuristics. The most popular and successful ones for the VRP are, among others, Simulated Annealing (SA), Tabu Search (TS), Genetic Algorithms (GA), Greedy Randomized Adaptive Search Procedure (GRASP), Variable Neighbourhood Search (VNS) and Adapted Large Neighbourhood Search (ALNS). For a literature review on the heuristics and metaheuristics for the CVRP, please refer to the book chapters in [125].

Recently, there is a new category called "hybrid metaheuristics" that integrate various solution methods and take advantage of each of the integrated methods. The hybrid methods may exclusively combine metaheuristic concepts, or also involve algorithmic ideas and modules from mathematical programming, deriving the so-called "matheuristics". Efficient solution methods can also be obtained by the joint effort of heuristics with constraint programming, tree-search procedures, to mention a few [131]. For a survey of such hybrid metaheuristics, please refer to [115]. More recently, a new category of heuristics called "hyper-heuristics" is emerging for the solution of industrial problems. Although not annonced officially, the ALNS can be seen as a simple version of hyper-heuristics. The hyper-heuristics operate on a search space of heuristics (or heuristic components) rather than directly on the search space of solutions [32], which tries to automatically design or select the adapting heuristic methods according to the structure of the problem.

1.3.2 Tools for Developing Exact Methods

Contrary to the heuristics that are content with finding rapidly a good solution even if it is suboptimal, the exact methods insist on the optimality of the solution, even though a lot of computation time may be spent. Classic exact methods for the VRP include direct tree search methods and Dynamic Programming (DP). They are usually derived from some special mathematical programming formulations.

In the following, we first introduce some of the mathematical formulations of CVRP , which have been used to derive exact methods in the literature and have also inspired some of the formulations in this thesis. Then, the principles of the Branch-and-Bound (B&B) are presented, which served as the general solution scheme for many exact solution methods. After that, the dynamic programming method are sketched. For a recent review of exact methods for CVRP please refer to [20, 119, 126].

1.3.2.1 Mathematical Programming

To apply the mathematical programming to an optimization problem in general, the first step is to formulate the problem using a mathematical model. Solution approaches vary a lot from one formulation to another. For the CVRP, there exist different types of formulations. In the following, we present three most common mathematical formulations for the CVRP ordered by the number of index of the principal variables.

Remember that the problem is modelled on a graph $G = (V, A)$. Node 0 denotes the depot. Other nodes denote customer locations. Let $\mathcal{Z}$ denote the set of customers, each with demand R_i. Let $\mathcal{K}$ denote the set of vehicles, each with the capacity B. The total number of vehicles is denoted by K.

One index formulation In this formulation, the CVRP is considered as a *Set Partitioning Problem.* Let Ω be the set of feasible routes. A feasible route is defined as a route starting and ending at the depot that make a hamiltonian tour among the set of customers to satisfy their demands. Binary parameter $a_{i,r}$ equal to 1 if the customer $i \in \mathcal{Z}$ is in the route r, 0 otherwise. Cost c_r for each route $r \in \Omega$ is computed as the sum of costs of arcs taken in the route. Binary variables λ_r equal to 1 if the route r is selected and 0 otherwise. The CVRP is then formulated as

$$\text{minimize} \quad \sum_{r \in \Omega} c_r \lambda_r$$

subject to

$$\sum_{r \in \Omega} a_{i,r} \lambda_r = 1 \qquad \forall i \in \mathcal{Z}$$

$$\sum_{r \in \Omega} \lambda_r = K$$

$$\lambda_r \in \{0,1\} \qquad \forall r \in \Omega$$

This formulation was first proposed in [21]. It is a binary integer formulation with potentially exponential number of variables. It is very general and can take into account various constraints on a route since route feasibility is implicitly considered in the definition of the set Ω. It can also serve as basic formulation for the column generation scheme detailed in section 1.3.3.1.

Two index formulation This type of formulation is also known as *vehicle flow formulation* or *assignment based formulation*. It contains an explicit arc-based binary variable $x_{i,j}$, which equals 1 if arc (i,j) is used in a solution, 0 otherwise. $x_{i,j}$ can be seen as a flow variable for the number of vehicles circulate on an arc (0 or 1). Let $c_{i,j}$ the cost of arc (i,j), and for each set $S \subset \mathcal{Z}$, $r(S)$ the minimum number of vehicles needed to serve the customers in S. The CVRP can be formulated as:

$$\text{minimize} \quad \sum_{(i,j)\in A} c_{i,j} x_{i,j}$$

subject to

$$\sum_{i\in\mathcal{Z}} x_{0,i} = K$$

$$\sum_{i\in\mathcal{Z}} x_{i,0} = K$$

$$\sum_{j:(i,j)\in A} x_{i,j} = 1 \qquad \forall i \in \mathcal{Z}$$

$$\sum_{j:(i,j)\in A} x_{j,i} = 1 \qquad \forall i \in \mathcal{Z}$$

$$\sum_{i\notin S}\sum_{j\in S} x_{i,j} \geq r(S) \qquad \forall S \subset \mathcal{Z} \qquad \text{(SEC1)}$$

$$x_{i,j} \in \{0,1\} \qquad \forall (i,j) \in A$$

The constraints (SEC1) is the set of subtour elimination constraints (SEC). It says that for all subset S of $\mathcal{Z}$, there is at least $r(S)$ vehicles entering or leaving the nodes in S. Since these constraints are for all the subsets of $\mathcal{Z}$, which are exponential in number, the formulation is not compact. In practice, there are various ways to eliminate the subtours. One way is to add auxiliary variables to generate compact formulation. Another way is the cutting plane method and the algorithm of Branch-and-Cut (B&C). A wide variety of B&C algorithms are derived from this formulations (see [20] for a review of these algorithms).

Three-index formulation Explicit arc-vehicle-based binary variable $z^k_{i,j}$ for the usage of each arc $(i,j) \in A$ and each vehicle $k \in \mathcal{K}$ is proposed in this formulation.

It is equal to 1 if arc $(i, j) \in A$ is traversed by vehicle $k \in \mathcal{K}$, and 0 otherwise.

$$
\begin{aligned}
\text{minimize} \quad & \sum_{(i,j)\in A} c_{i,j} z_{i,j}^k \\
\text{subject to} \quad & \\
& \sum_{i\in\mathcal{Z}} z_{0,i}^k = 1 && \forall k \in \mathcal{K} \\
& \sum_{j\in\mathcal{Z}} z_{j,i}^k - \sum_{j\in\mathcal{Z}} z_{i,j}^k = 0 && \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \\
& \sum_{j:(i,j)\in A} z_{i,j}^k = 1 && \forall i \in \mathcal{Z} \\
& \sum_{j:(i,j)\in A} z_{j,i}^k = 1 && \forall i \in \mathcal{Z} \\
& \sum_{i\in\mathcal{Z}} R_i \sum_{j:(i,j)\in A} z_{i,j}^k \le Q && \forall k \in \mathcal{K} && \text{(SEC2)} \\
& z_{i,j}^k \in \{0,1\} && \forall (i,j) \in A, \forall k \in \mathcal{K}
\end{aligned}
$$

This formulation is particularly useful when modelling heterogeneous vehicle fleet. Some formulations contain an additional set of binary variables y_i^k which equal to 1 if vehicle k serves customer i and 0 otherwise.

1.3.2.2 Branch and Bound

The Branch-and-Bound (B&B) is a useful tool for solving large scale NP-hard combinatorial optimization problems. It is often used as a basic scheme, in which different components can be developed according to the structure of the problem concerned.

Let us take a combinatorial optimization problem **(CP)** in general defined in Section 1.1. It is a minimizing problem with S denoting the feasible region of the problem, c denoting the objective function, $LB(S)$ and $UB(S)$ the lower and upper bound on the objective in the region S, c^* the best objective encountered for the best current solution x^*. A prototype of the B&B is given by Algorithm 1. In this algorithm, SL denotes a list of feasible regions to be explored.

The key to success of the B&B is often the quality of bounds used to guide the tree search. In addition, different branching strategies can be tried for the partition of the feasible region to speed up the search. Various search strategies can complement the solution process. Diverse bounds derived from different kinds of relaxations can be applied to the CVRP. For detailed presentation of the B&B algorithm applied to CVRP, please refer to the book chapter [119].

1.3.2.3 Dynamic programming

Dynamic Programming (DP) applies to problems with an optimal substructure and overlapping sub-problems. The basic idea of dynamic programming is to break

Algorithm 1 Branch and Bound

Initialize a set SL with a feasible region R_0, with associated bounds $-\infty, +\infty$
while $SL \neq \emptyset$ **do**
  Select a feasible region $R \in SL$ and $SL \leftarrow SL \backslash \{R\}$
  Bound the objective c in the region R: $LB(R) \leq c(x) \leq UB(R)$ for all $x \in R$
  if $LB(R) \geq c^*$ **then**
    Discard R
  else
    Branch R by dividing it into k subsets $R_1, \ldots, R_k$
    $SL \leftarrow SL \bigcup \{R_1, \ldots, R_k\}$
  if $UB(R) < c^*$ and the associated solution $x(R)$ is feasible (where $c(x(R)) = U(S)$) **then**
    $c^* \leftarrow UB(R)$ and $x^* \leftarrow x(R)$
return c^* and x^*

down the initial problem into a sequence of overlapping sub-problems, so that the optimal solution of the sub-problems leads to the optimal solution of the initial problem. It is often used as a subroutine in other algorithms. Let us consider a general combinatorial optimization problem with the same notations as given in Section 1.1. The DP can be applied, if the feasible region S can be divided into subsets $S_0 \subset S_1 \subset \ldots \subset S_k = S$, and the solution of the complete problem can be decomposed in to sequential solutions of each S_k with $f(S_{i-1}, S_i)$ something easy to solve depending on S_{i-1} and S_i for each $i \in \{1, \ldots, k\}$. A general presentation of the optimal solution $c*$ by dynamic programming can be resumed by Equation 1.1.

$$c* = \begin{cases} c(S_0) \\ c(S_i) = c(S_{i-1}) + f(S_{i-1}, S_i) \quad \forall i \in \{1, \ldots, k\} \end{cases} \tag{1.1}$$

1.3.3 Decomposition Methods Based on Mathematical programming

The idea behind the decomposition methods is to decompose the problem into several subproblems that can be easily solved. The solution of one sub-problem can provide information for the solution of other sub-problems or the original problem. By adding feedbacks or adjusting loops between sub-problems and the original one, a good quality solution is obtained in an acceptable time. Both the B&B and DP applies this idea by exploring special structures of the problem. Based on the mathematical programming, there exist other ways of decomposition. In the following, we present two classical decomposition methods that has served in this thesis: the column generation and the Lagrangian relaxation.

1.3.3.1 Principles of Column Generation

In this section, the one index formulation of CVRP (1.2)—(1.5) is taken as an example. This is what we call the *Master Problem (MP).*

$$\begin{aligned}
\text{minimize} \quad & \sum_{r\in\Omega} c_r\lambda_r & & & (1.2)\\
\text{subject to} \quad & \sum_{r\in\Omega} a_{i,r}\lambda_r = 1 & \forall i \in \mathcal{Z} & \quad (\alpha_i) & (1.3)\\
& \sum_{r\in\Omega} \lambda_r = K & & \quad (\beta) & (1.4)\\
& \lambda_r \in \{0,1\} & \forall r \in \Omega & & (1.5)
\end{aligned}$$

Recall that in this formulation, Ω is the set of feasible routes, which are exponential in number. Thus, this formulation contains exponential number of variables λ_r, $\forall r \in \Omega$ and the feasibility of a route r is hidden in the definition of the set Ω. To solve such a problem, one way is to start by solving the linear relaxation of the master problem by relaxing the integrality constraints (1.5).

The appealing idea of the *column generation* is to work only with a sufficient meaningful subset of variables, forming the so-called Restricted Master Problem (RMP) [53]. In our example, RMP is defined by (1.2)—(1.4). To do this, an iterative approach is needed. In each iteration, we have to solve the RMP to determine the current objective value and the dual values of each set of constraints (denoted by α_i, β on the right of constraints (1.3)—(1.5)). Then, we try to find a variable λ_r to enter RMP by solving a *pricing problem* defined by:

$$\bar{c}^* = \min_{r\in\Omega} c_r - \sum_{i\in\mathcal{Z}} \alpha_i a_{i,r} - \beta \qquad (1.6)$$

with $a_{i,r}$ a binary variable equal to 1 if customer i is in the route r, and 0 otherwise.

For a fixed route r, the value $c_r - \sum_{i\in\mathcal{Z}} \alpha_i a_{i,r} - \beta$ is called the *reduced cost.* Each iteration of column generation is to find a route (column) with negative reduced cost. If the solution of the pricing problem is negative, then the corresponding variable λ_r is added to the RMP and the process repeats. Otherwise, there is no improvement for the linear relaxation of the master problem. The problem is then converted to the initial MP with integer variables. Usually, a branching procedure is used to fix the integer values.

The pricing problem can be seen as a sub-problem of the initial problem. It can be solved by another scheme different from the mathematical programming, such as dynamic programming or heuristics. Thus, it is usually a point of integration when developing hybrid method.

Due to the fact that column generation only works with a sufficient subset of variables, this method is often used to solve large-scale combinatorial optimization problems. However, ensuring a good management of the set of columns is not an easy job. The way to generate columns of good quality within a reasonable size and computation time needs special care. Moreover, the solution of the final integer problem after the column generation stays also a big challenge.

The column generation is often implemented as a component of B&B, introducing the so-called Branch-and-Price (B&P). In B&P, the linear relaxation in each node of a branch-and-bound tree are solved by the method of column generation and the term "price" refers exactly to the solution of the pricing sub-probem. It can also be integrated into a method called Branch-and-Price-and-Cut, where additional cuts to strengthen the relaxation are inserted to the master during the column generation phase according to the information given by the generated columns. The book [51] gives more detailed information about the column generation.

1.3.3.2 Principles of Lagrangian Relaxation

The Lagrangian relaxation is a tool for solving large-scale combinatorial optimisation problems. It works by moving hard-to-satisfy constraints into the objective function by associating a penalty.

Consider an integer problem in general:

P:

$$\begin{aligned} \text{minimize} \quad & cx && && (1.7)\\ \text{subject to} \quad & A^1x \geq b^1 && \text{(complicating constraints)} && (1.8)\\ & A^2x \geq b^2 && \text{(easy constraints)} && (1.9)\\ & x \in \mathbb{Z}^n_+ && && (1.10) \end{aligned}$$

The idea of the Lagrangian relaxation is to relax the complicating constraints (1.8) with the so-called *Lagrangian multipliers* (denoted by $\gamma \in \mathbb{R}^n_+$ here). Consequently, the relaxed problem, called the *Lagrangian relaxation* with respect to $A^1x \leq b^1$ becomes much easier to solve. It is defined as:

L(γ):

$$\begin{aligned} \text{minimize} \quad & cx + \gamma(b^1 - A^1x) && && (1.11)\\ & A^2x \geq b^2 && \text{(easy constraints)} && (1.12)\\ & x \in \mathbb{Z}^n_+ && && (1.13) \end{aligned}$$

It is obvious that the optimal solution of the **(L(γ))** with a fixed γ is a lower bound of the original problem.

If we set $z_D(\gamma) = \max_{x \in \chi}\{cx - \gamma(A^1x - b^1)\}$ with $\chi \subset \mathbb{Z}^n_+$ the domain defined by constraints (1.12). The maximization of $z_D(\gamma)$ defines the following problem:

L$_D$:

$$\begin{aligned} \text{maximize} \quad & z_D(\gamma)\\ & \gamma \in \mathbb{R}^n_+ \end{aligned}$$

The problem coincides with the Lagrangian dual of **(P)** with respect to constraints (1.8). Its optimal solution gives the best lower bound for the original problem. According to the duality theory, this bound is better than the linear relaxation bound except in the case with integrality property. For details of the theoretical demonstrations, please refer to [78].

The Lagrangian relaxation is often integrated in a B&B scheme. In each iteration, the easier relaxed problem with a fixed multiplier is solved to provide a lower bound. Then, the Lagrangian multipliers are updated. Additional component can be added to get a feasible solution from the solution of relaxed problem to update the upper bound. The process repeats until a certain stop condition is reached. In the B&B scheme, the lower bound is given by the best lower bound found by the Lagrangian relaxation, the upper bound is updated by feasible solutions found during the iterative process.

To apply such a method, several issues have to be kept in mind [69]. The first is how to choose the constraints that are to be relaxed. The method only works if the relaxed problem is really easy to solve, since the solution of the relaxed problem will be repeated many times in the whole solution process. The second is how to compute good multipliers. A common way is the subgradient method [69]. The third is how to deduce a good solution to the original problem from the solution of the relaxed problem.

The Lagrangian relaxation is often integrated with some heuristic components for the solution of the relaxed problem or to obtain a feasible solution [46]. It can also be used as a sub-routine for the solution of some large scale problems. [26, 37]

1.4 Conclusion

This chapter has defined the general routing problems and some of its variants. Exact solution methods can be developed from mathematical programming formulations. General solution methods such as Branch-and-Bound (B&B) and dynamic programming has been presented. Column generation and Lagrangian relaxation are two decomposition methods based on mathematical programming that are often used to solve large scale instances. These two methods are going to be applied later in Chapter 5 and 6 to solve a real IRP and its integration with energy consideration.

In the next chapter, the IRP and the GVRP are presented. The IRP is a generalization of the VRP in which the customer demands are not given by customers but also a decision variable. The compromise between the cost of routing and the cost of inventory storage has to be considered in the IRP. The GVRP is an integration of energy efficiency into the VRP, which is an emerging area of study in recent years.

Chapter 2

Inventory Routing Problems and Green Vehicle Routing Problems

The Inventory Routing Problem (IRP) has been introduced since the 1980s. It integrates both the vehicle routing and inventory management problems. The Green Vehicle Routing Problem (GVRP) has emerged in recent years covering the environmental issue met in the product distribution or public transportation. This chapter reviews these two problems in detail to give the general problem settings, different variants, well-known benchmarks as well as the solution approaches.

2.1 Inventory Routing Problems

The Inventory Routing Problem (IRP) is developed under the Vendor Managed Inventory (VMI) model. In this model, the vendor (or supplier) is usually the manufacturer but sometimes can be a reseller or distributor. He or she acts as a central decision maker and monitors the inventory level of each buyer (customer or retailer) physically or via electronic sensors and messaging. The vendor has to make periodic resupply decisions regarding delivery quantities and timing. In practice, with respect to the traditional Retailer Managed Inventory (RMI) where the buyers set orders for the suppliers, the VMI can results in reduced costs and improved services for both suppliers and buyers. For suppliers, smaller buffers of capacity and inventory are possible as the uncertainty of demands is mitigated. The coordination of services to several customers also allows for more efficient distribution and more predictable delivery planning. For the buyers, they devote less resources to inventory monitoring while having the guarantee that stock-out will never happen.

In this section, we start with a general presentation of the problem, followed by a classification and applications, and finish with some of the solution approaches. This is only a brief introduction to show the basic problems considered in this thesis. For a detailed explanation of the IRP, please refer to the tutorials [29, 30], which introduce the IRPs with examples, present different models and policies for different classes of the problems and explain the relations with classical routing problems. An early review is proposed in 1998 by [18] with a first classification, but the problem was named "dynamic routing-and-inventory problems (DRAI)" at that time. The review [10] describes the industrial aspects of combined inventory management and routing in maritime and road-based transportation and gives a literature survey of the state of the art in 2010. A more recent comprehensive literature review is provided by [43] with a new classification and more accentuation on solution methods.

2.1.1 Problem Presentation

The IRP is an integration of inventory management and vehicle routing and dispatching. In general, there are a set of suppliers and a set of customers. Each supplier or customer maintains a set of products in their inventory. Customers consume the products gradually. The suppliers monitor the inventory level of customers and make deliveries to make sure that no one is out of stock. The products are transported using a fleet of vehicles. Each vehicle has a capacity that should not be exceeded. Each unit of product stored in the supplier or customer has an inventory storage cost. Each unit of product transported on road has a transportation cost. The problem is to find the best distribution plan that minimizes the combined costs of inventory storage and transportation.

In summary, there are three simultaneous decisions to make: i) when to serve a customer; ii) how much to deliver when serving a customer; iii) how to route the vehicle among the customers to be served [34].

2.1.2 Applications

In industry, combined routing and inventory management problems are mainly found in road-based and maritime supply chains. An important road-based application is the distribution of industrial gases or petrol oil using trucks, where VMI model is implemented [35, 56, 61, 62, 118, 121, 136]. In fact, the first studies by Bell et al. in the 1980s [26] were to develop an integrated management system for the distribution of industrial gases. The main characteristics of this type of application are that the inventory monitoring is performed in a very fine granularity of time (often every few hours) during a very long period of time (weeks or years) and that the stock-out is strictly forbidden or costs a lot. The problem could also be integrated with the scheduling of drivers and the assignment of driver to trucks. Furthermore, constraints on the customer side can also complicate the problem, such as customer time windows or consistency of visits. Other road-based applications include, among others, the distribution of spare parts of the car manufacturing industry [31] and the distribution of perishable goods [12, 101, 8, 123].

The maritime inventory routing for Liquefied Natural Gas (LNG), called LNG-IRP is also an important area of applications [9, 79, 120]. It is different from road-based applications due to specialities met in the maritime context. Normally, the planning horizon of the maritime inventory routing is longer with bigger time granularity. There could be more types of vehicles in the maritime applications, too. Please refer to [10] for the comparison of road-based and maritime IRPs and [40] for a literature review of maritime routing and scheduling.

This thesis focuses on the road-based IRPs.

2.1.3 Classification of the Inventory Routing Problems

Due to the different possibilities of problem settings in practice, a wide variety of IRPs exist in the literature. The following classification is based on that given in [43] with some remarks on recently appeared features .

2.1.3.1 General Settings

Different supply chain structures and lengths of planning horizon generate different versions of IRPs

Types of supply chains According to the applications, different structures of supply chains are studied. Here we only list the types and some of the related papers. For detailed explanation of different types of supply chains and their impacts please refer to [10].

- one-to-one: there is only one central depot and one single customer. This case is studied in an early paper [57] to show the compromise between the transportation costs and the inventory costs.

- one-to-many: there is one central depot (the supplier) and a set of customers. The transporter makes a tour among the set of customers to deliver products. This is the most commonly studied version of the problem [15, 26, 42].

- many-to-many: there are a set of pick-up points (suppliers) and a set of delivery points (customers), resulting in a pick-up and delivery routing problem, which is often the case in maritime applications [9, 39, 79, 118].

Planning horizon The length of the time horizon can also change the nature of the problem. Even though the default setting is the inventory monitoring on a long time horizon, mainly two categories of time horizon are studied in the literature: infinite and finite.

- The problem with infinite horizon is to find the replenishment strategy with minimum costs in the long term. In this category of IRPs, the customer demands are usually given by a rate (fixed or variable) in each time unit. A common type of IRP in this category is the Cyclic Inventory Routing Problem (CIRP), in which the decision on replenishment and routing is defined by cycles that repeat infinitely [4, 12].

- With a finite horizon, the long-term problem is reduced to a short period of time and the adequate costs are chosen to reflect long-run inventory costs [57]. The time horizon can be divided into periods and a routing strategy is to be find for each period [16, 17].

2.1.3.2 Inventory Decisions and Policies

Based on inventory decisions and inventory policies, the problem can be divided into different categories.

Supplier production It can be fixed, variable or to be determined. Most of the papers on IRP do not deal with the production side. In general, the supplier is supposed to have a fixed production rate higher (or not) than the total demand rate of all the customers [16, 117]. In the LNG-IRP, the production rates of suppliers could be variable over time [116]. The integration of decisions on production, inventory and distribution is studied in [22]. If the production decisions are integrated, the problem becomes a Production Routing Problem [3].

Supplier capacity The supplier capacity can be unlimited or limited. In most of the literature of IRPs, it is assumed infinite but a cost related to the inventory storage at the supplier is taken into account.

Inventory decisions Concerning the case where inventory is not enough to cover the demands of customers, mainly two options are applied in practice: one is to make sure that there is always enough inventory in the customers; the other is to allow the shortage of inventory but to pay for it with a huge cost. They correspond to the following two strategies:

- *no stock-out* or *non-negative*: a *stock-out* happens when the customer or retailer does not have enough inventory in stock. In most IRPs, stock-outs are strictly forbidden. That is to say, the vendor (supplier) should make sure that the inventory level of each customer is never negative or below a safety level defined by the customer.

- *lost sales* or *back orders*: *lost sales* occur when the customer's inventory is not refilled in time. Customers that are not able to be refilled in time can also be *back logged* and treated later. Lost sales or backlogging can be allowed but with a penalty on the total cost [1, 109].

Inventory policy Among the various inventory policies applied in practice, two of them are often mentioned in the IRP literature: Maximum Level (ML) and Order-up-to Level (OU). Under the ML policy, the inventory level of each customer is flexible but bounded by his capacity. Under the OU policy, the inventory of each customer is automatically refilled to his capacity each time he is visited. The inventory policy can have an influence on the combined cost of transportation and inventory. Most early papers on IRP deal with ML policy. Recent papers usually test both policies to show the possible influence of inventory policy to the decisions of inventory routing. In papers regarding supply chain management, the inventory policy could also be tackled as a decision for optimizing the configuration of supply chains [60, 81], which aids to decide the size of the stock, the control strategies or the design of supply chains. In IRP literature, however, few paper deals with the optimization of inventory policy. The structure of the system is considered fixed and the inventory policy is decided in advance of the optimization procedure.

2.1.3.3 Customer Related Variations

Depending on the characteristics of the customers, several versions of IRPs are present in the literature.

Variation of customer demands over time The customer demands can be constant, periodic or variable over time. The constant demand rate case is very common [16, 117]. The periodic case is often seen in the CIRP [4] or Periodic Inventory Routing Problem (PIRP) [7]. Variable demands are closer to the reality but complicate the problem. In Chapter 5 of this thesis, we present a real problem where the demands of customers are totally random and can occur at any moment in the time horizon.

Availability of customer demand information According to the availability of customer demands when making decision, the IRP can be divided into three classes: deterministic, stochastic or dynamic. If the customer demands are totally known when making decision, the problem is deterministic [16]. If the probability distribution of the demands is known, then the problem is stochastic [12]. Dynamic IRP arises when demand is not fully known before the decision process but revealed over time during the decision making [42].

Customer composition The problem differs if all customers are managed with the VMI model or only part of them are. In the literature, the IRP only considers VMI customers. However, in reality, VMI customers can coexist with other customers that sets orders to the supplier. This is a type of problem that we met in Chapter 5.

Customer visit time The customers can have their own time windows. As time windows exist implicitly in IRP, most studies in the literature do not consider hard customer time windows. However, the customers do have visit time windows in practice (as shown in Chapter 5).

2.1.3.4 Routing Settings

On the routing side, different kinds of fleet and routing types are addressed in the literature.

Fleet size The fleet considered in the IRP can be unlimited, limited or to be determined in size. The IRP with unlimited fleet size is often studied in early papers [33, 12] where the fleet size is neglected to study in details the trade-off between transportation and inventory. Most recent studies focus on the problem under a fixed number of vehicles. The fleet could then contain one single vehicle [16] or multiple vehicles [17, 44]. The fleet size can also be a decision variable as in [112], where a fixed cost is related to each vehicle and a variable cost is related to the usage duration usage of each vehicle.

Fleet composition The vehicles can be homogeneous or heterogeneous. In homogeneous fleet, all the vehicles have the same characteristics [112, 117]. While in heterogeneous fleet, the vehicles have different capacity [1, 2].

Vehicle working time In classic IRPs, there are no time windows related to the vehicles. In CIRP [113], time windows of vehicles can be integrated to model the working time restrictions of the drivers.

Routing structure The routing of the vehicle can be direct, multiple or continuous. Direct shipping between the central depot and one single customer site has been studied in an early paper [33] as a simplification of the problem to look into the trade-off between transportation costs and inventory costs. Most studies in the literature consider multiple routing. That is to say, the vehicle starts and ends from the depot, and visits multiple customers in each tour. Each customer is visited at most once in a tour. IRP with continuous moves is proposed in [117, 118], where there is no central depot and the vehicle travels among several pick-up and delivery points continuously without stops. The LNG-IRP [9] is similar to the IRP with continuous moves, due to the fact that the vehicles make moves around pick-up sites and delivery sites continuously without a clear notion of "tour" as defined in common IRPs.

Routing frequency The frequency of routing can be unique, once in a period or to be determined. Unique routing is considered in [67] with stochastic demands. In this way, the combined decision of routing and inventory is simplified to a Vehicle Routing Problem (VRP). Most recent studies consider a horizon divided into multiple periods. In each period, the vehicle is routed at most once among a set of customers. Actually, the vehicle could also be routed multiple times in each period in practice, which leads to a multi-trip problem that is rarely studied in the literature. The routing frequency could also be a decision variable as is the case in CIRP [4].

Since 2012, the main stream of studies on IRPs focuses on the *one-to-many* version with one central depot (where resides the supplier) and several customers. The problem is set on a *finite time horizon* partitioned into several periods with a *homogeneous or heterogeneous* fleet composed of *one or several* vehicles. Customer demands can be *deterministic or stochastic*. Stock-outs are *not allowed* and several inventory policies are included. The objective is to find the routes of vehicles that start and end at the depot and visit several customers *once* in a period to minimize the total transportation and inventory costs. In this thesis, the energy consideration is integrated into this main stream version of IRP with deterministic demands.

2.1.4 Solution Approaches in the Literature

The solution approaches vary from one version to another depending on the length of the planning horizon and the geographical locations of customers. Due to the complexity of this combined problem of routing, scheduling and inventory management, the IRPs are often solved using a heuristic method or a hybrid method with a heuristic and an exact component [12, 33, 58, 59]. In general, heuristic methods perform a relatively limited exploration of the solution space through simple neighbourhood structures. They can produce a solution of acceptable quality in a reasonable time of computation. The following solution approaches reviewed are only for the deterministic version of the problem.

Early works usually contains two steps [58]. In the first step, the problem of minimizing annual distribution costs is reduced to an optimization problem over a limited planning period. A procedure is developed to select the customers for the distribution problem for each planning period. In the second step, the limited time horizon distribution problem is modelled and solved as a classical routing problem with some special features regarding customer inventory levels. Most of the classical heuristics for solving the VRP can then be applied for this short-term routing problem. These solution approaches are generally characterised by how to model the long-term effects of short-term decisions and how to decide which customers are included in the short-term problem. Please refer to [34] for an introduction of the approaches that have been taken before 2001.

Recent approaches are often hybrid methods of heuristics/metaheuristics and mathematical programming, yielding the so-called "matheurisctics". For example, [15] proposed a hybrid method of mathematical programming and tabu search, [42] uses a Mixed Integer Linear Programming (MILP) component inside the scheme of the Adapted Large Neighbourhood Search (ALNS), [46] developed a three-phase decomposition method, where a Lagrangian-based replenishment planning heuristic and a routing construction heuristic are given to a feedback mathematical model. [120] proposed a hybrid method combining Greedy Randomized Adaptive Search Procedure (GRASP) and mathematical programming.

There are not many pure exact methods in the literature. Archetti et al. [16] are the first to propose an exact Branch-and-Cut (B&C) algorithm for IRP. In [44], a B&C algorithm is developed for the exact solution of several classes of IRPs. In [2], several formulations and B&C algorithms together with a ALNS based heuristic are proposed and compared. A Branch-and-Price (B&P) method is proposed in [79] for the LNG. In the column generation approach, the master problem handles the inventory management and the port capacity constraints, while the sub-problems generate the ship route columns. A B&P guided search is applied to a real-world maritime IRP in [85], where a series of constrained MILP is solved to obtain heuristic solutions. In [52], an innovative mathematical formulation for the IRP is proposed and a branch-price-and-cut algorithm is developed, where know and new families of valide inequalities have been incorporated.

2.1.5 Benchmarks

There exist many types of benchmarks corresponding to different versions of the problem. The site of Coelho [41] contains instances for several types of IRPs. Instance set proposed in [16] is for the single vehicle single commodity IRP. Instance set proposed in [15] is for the same problem but is larger and more challenging. Both instance sets were used to evaluate multi-vehicle algorithms [2, 42, 44], by simply dividing the original vehicle capacity by the number of vehicles considered. The site [41] also contains small and large adapted instances from [16] for several multi-vehicle cases of IRP. Benchmark instances for the CIRP presented in [38] and [130] can be found on site [129].

2.2 Green Vehicle Routing Problem

Green Vehicle Routing Problems (GVRPs) is a general term for a category of Vehicle Routing Problems (VRPs) that considers the sustainable transportation issues from an operations research perspective. It is mainly investigated since 2006 and is becoming more and more popular these days. This category of routing problems comes often with an additional objective related to the environmental costs, such as energy consumption, CO_2 emission or the costs of wastes, etc. The GVRP tries to harmonize the environmental and economic costs by implementing effective routes to meet the environmental concerns and financial indices [102]. The main purpose of the study of the GVRP is to measure the energy efficiency of the current routing system, to provide an insight into the environmental factors met in the routing and to guide the practitioners in the practice of sustainable transportation. This category of routing can also be integrated into the Electric Vehicle Routing Problems (EVRPs), where the energy efficiency becomes crucial to the functioning of the whole system.

This section begins with a brief presentation of the problem that underlines the difference of GVRP from the classic VRPs induced by the ecological costs. Then, a short review of the energy modelling is given. Key factors affecting the energy related costs are also discussed. Finally, some solution approaches in the literature are presented. For a detailed literature review of the GVRPs, please refer to [102].

2.2.1 Problem Presentation

In general, GVRPs include consideration of wider objectives and more operational constraints that are concerned with sustainable logistics. Consequently, there have been new vehicle routing models and new application scenarios, which naturally lead to more complex combinatorial optimization problems. In the literature, the GVRPs are categorized into three groups [102]:

1. Green-VRPs (G-VRPs) for the optimization of energy consumption;
2. Pollution Routing Problems (PRPs) for the reduction of pollution, especially Greenhouse Gas (GHG) emissions;
3. Vehicle Routing Problem in Reverse Logistics (VRPRL) for the collection of wastes and end-of-life products.

Here, we do not repeat what is stated in the literature, but try to make a summary of the current state of the art to show how the sustainable issues can be integrated into the VRPs.

2.2.1.1 Basic Problems

According to different application scenarios, different kinds of VRPs can serve as the basic problem, which have already been introduced in Section 1.2 of Chapter 1. Some examples of the integration of environmental aspect into the routing problem are summarized below.

In cases where speed is considered as a decision variable, service time windows of customers can be important, and the basic problem would be Vehicle Routing Problem with Time Windows (VRPTW) [25]. In the time-dependent cases, the starting point of the study is often Time-dependent Vehicle Routing Problem (TD-VRP), where the cost of traversing an arc depends on the beginning time at the origin node [72]. To study the influence of the composition of vehicle fleet, the ecological aspects are added to the VRP with heterogeneous fleet [96, 134]. In addition, Vehicle Routing Problem with Pickup and Delivery (VRPPD) is considered a basic problem for many VRPRLs [14, 54].

2.2.1.2 Objectives

The GVRP often comes with an ecological objective. It can be the energy consumption as in the case of Energy Minimizing Vehicle Routing Problems (EMVRPs), where the cost is a function of the carried load of the vehicle, the wage of the driver, and the distance travelled [93, 74]. This has led to the study of a new variant of VRP that is the Cumulative Vehicle Routing Problem (CumVRP) [94].

The objective can also be the fuel consumption as in [135], in which the Fuel Consumption Rate (FCR) is defined as the cost of fuel consumption relative to the weight and load of the vehicle. GHG emission costs are considered in the PRP [25, 72]. In reverse logistics, there can also be a profit related to the collection of wastes [14].

Ecological objective is often compared with the traditional economical ones to show the impact of energy consideration to the decision process [25, 123]. Multi-objective studies of combined economical and ecological objectives are also gaining attention [49, 73]. Most papers in the literature contain a discussion about the trade-off between these different objectives.

2.2.1.3 Decision Variables

In addition to the routing decision variables, different variables are defined in the GVRP depending on the basic variant of VRP concerned. For example, in the EMVRP [93], flow formulation of the basic Capacitated Vehicle Routing Problem (CVRP) is used to integrate the load dependency of the energy consumption. In the time dependent case [72, 64], the start time of each tour and the start time of traversing each arc are both decision variables since the beginning time of travelling an arc can have an impact on the cost of the arc. This is to reflect the real situation where the traffic condition on a road can depend on the time of the travelling. Finally, the vehicle speed on each arc can be regarded as a set of decision variables [25, 65]. The consideration of vehicle speed as decision variables could make the problem non-linear [25].

2.2.1.4 Constraints

The integration of energy issues usually does not modify the basic constraints of the corresponding VRPs. However, the time dependency can complicate the problem even more, with different scenarios considered when the starting time of each tour or traversing time of each arc is different [64, 72, 91, 123, 134]. In fact, both travel time or travel speed can be considered as time-dependent. An upper bound may also limit the total emission in the routing [128].

In general, the GVRP is more complicated than basic routing problems. The resulting mathematical formulations can be non-linear. In the PRP [25], for example, the basic integer programming formulation is non-linear since the energy function contains the speed as a component. The calculation of driver time wage could introduce non-linearity, too. One solution to the non-linearity is to linearise with some additional assumptions or by simplification of the model [25].

2.2.2 Energy Estimation

For the estimation of energy consumption of the vehicles, there are two ways in general. The first is to use the physics of motion by making simplifying hypothesis. Some parameters in the formulae are fixed while other parameters are left as decision variables. The second way is to apply existing energy estimation models directly to the problem to be studied. These models come from the automotive industry. They describe the energy consumption or CO_2 emission of certain types of vehicles based on tests with some pre-defined driving cycles.

The first method is simple and easy to understand, but it can be too general and might lose some subtleties of the problem. The estimation using this method is often very vague and not precise. The second method is better suited to reflect the real case, but these industrial models are usually complicated with a lot of parameters and they are dependent on the types of vehicles. One model might work for a certain type of vehicles but not applicable to another type. In addition, since these models come from experiments of the automotive industry, it usually demands a lot of analysis to understand the relationships between the parameters before studying the real optimization problem. One of the energy models that is widely used in the community of PRP is a Comprehensive Modal Emissions Model (CMEM) for heavy-good vehicles [23]. The reader is referred to [48] for a comprehensive review of vehicle emission or fuel consumption models and their inclusion into the existing optimization methods.

2.2.3 Key Factors Affecting Energy Consumptions

Most of the current studies of G-VRP try to extract the key factors influencing the energy consumption in routing process. The most recognized factors in the literature are:

- Distance travelled. It is a common sense that the longer the vehicle travels, the more energy is consumed.
- Vehicle load. The more the vehicle carries, the more energy is needed.
- Vehicle speed. The faster the vehicle runs, the more energy is consumed.
- Traffic congestion. In the literature, the most common way to model the influence of traffic congestion to energy consumption is to study the time-dependent speed. The speed varies a lot depending on the time when the road is travelled. Since vehicle speed influences the energy consumption of the vehicle directly, the energy consumption can be time-dependent as well.

In [25], by comparing the result given for a small example with different objectives, the authors give an insight into what happens when energy consumption is integrated into classic CVRP or VRPTW. They show that even with the same distance travelled, the energy consumption could be different if the order of visits to customers is changed, which is due to the load dependency property of the problem. It is also pointed out that simply minimizing a weighted load objective is not enough to minimize energy because of the effect of variable speed. When time windows are considered, speed plays a more important role. Thus, taking speed as a variable could induce much more energy savings in a VRPTW. With consideration on vehicle and driver costs, it is shown that the solution that minimizes total costs is not necessarily the most ecological one.

In the time-dependent PRP [72], the impact of considering traffic congestion on the routing and scheduling planning activities can be important. Congestion is likely to increase costs or even lead to an infeasible solution when customer nodes have delivery time windows. The authors add strategic wait times at the depot or at the customer nodes to avoid congestion. Different wage policies for drivers could also have an impact on the total route duration and the visit time, and consequently the total energy consumption.

2.2.4 Solution Approaches

Most of the solution approaches presented in the literature are derived from the basic VRPs. Since this is an emerging area of study, most papers have a modelling part and then propose a mathematical model. In general, the mathematical models could not be solved with solutions of satisfiable quality in an acceptable time. Thus, various heuristics or metaheuristics are applied. In the time-dependent case, a tabu search procedure is often adapted [64, 91, 111]. In [71], the time-dependent PRP is solved with an improved Adapted Large Neighbourhood Search (ALNS) metaheuristic. In [134], a hybrid algorithm combining mathematical programming and iterated neighbourhood search is proposed to solve the heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. A bi-objective Inventory Routing Problem (IRP) with both economical costs and emission costs is solved in [73] by a column generation approach with Non Inferior Set Estimation (NISE) algorithm for the multi-objective solution.

2.2.5 Benchmarks

Situations change from one application to another. Case studies exist for the city of London [111], for the city of Stuttgart [64] and for a supply chain linking several cities in the Netherlands [122]. A library for PRP instances can be found on the website [13], which contains nine groups of instances. These instances are based on real distances collected from randomly chosen UK cities. Simulation or generated instances are also widely used in different studies, but there is no standard set of benchmarks.

2.3 Conclusion

This chapter has generally reviewed the state of the art of the IRP and the GVRP. Different versions of the problems have been presented with their solution methods. These two categories of routing problems are combined in this thesis. This integration started to attract attention at the beginning of the thesis in 2014 [8, 128]. It is now becoming more and more popular in the community e.g. see [36, 73, 114, 123].

The global management of inventory in the IRP gives very good properties for energy minimization, which cannot be addressed under the VRP setting. The integrated problem studied in this thesis associates the three principal decisions of the classic IRP with the energy consumed by vehicles in transportation. Firstly, the transported mass links the energy consumption of the vehicles and the inventory levels of the customers. The mass flows on the transportation network are identified as a significant set of variables that balance the demands of the customers and the consumption of the vehicles. Chapters 3 and 4 discuss this issue more in details.

Secondly, the routing of the vehicle is important for the total energy consumption due to the load dependency property of the problem as revealed in the PRP [25]. In addition to that, routing decisions also influence the timing of each visit to customer if the travel time on each segment of road is considered. The timing of visits to customers can be influential, since both delivering quantities and energy consumption could be dependent on travel time or vehicle speed. This motivate us to model the road network as a multi-graph in Chapter 6, where both time and energy consumption values are attributes of the arcs.

Part II

Mass Flow MILP Formulations and Experimentations

In this part, two problems related to inventory management and routing are presented with consideration on energy consumption by vehicles. One is a study of the energy efficiency in the assembly lines by electric tow trains. The other is a generalisation of this problem to the Inventory Routing Problem (IRP). Energy estimation models are proposed for each of the two problems. One for tow trains in the assembly lines and the other for trucks in the road networks. The main energy influencing factors are identified.

Chapter 3

Energy-Efficient Assembly-line Vehicle Supplying Problem

In most supply chains, raw material supplying is governed by the Just In Time (JIT) paradigm. The philosophy of the JIT is to deliver the perfect parts and services at exactly the time they are required. Although JIT has been a key factor of performance for the last decades, it concentrates solely on economic performances. It tends to favour frequent deliveries with less than truckload shipments, small production runs, and multiple regional warehouses, which from an ecological point of view, could have as much of an impact on the carbon footprint of a firm as the energy efficiency of individual units deployed in production or distribution. [27].

With the need of confronting new regulations, environmental sustainability is considered as the most recognized dimension of sustainable supply chain management [68]. To achieve energy savings inside manufacturing industry, common perception has been focused on the update and replacement of technology or equipments. Recently, a new effort has been made in raising the awareness of the amount of energy used in the manufacturing processes, consumed by the machines and lost in different components of the production lines. This introduces the idea of Energy Aware Manufacturing (EAM). The EAM operations are defined as manufacturing operation management systems that consider the energy as a decision variable, an objective or as a constraint in a predictive or reactive way, and in addition to usual production decision variables, objectives and constraints (e.g. time based or time/ quantity based) [110]. The first step of the EAM has been to recognize energy usage patterns, to identify energy related parameters and to clarify energy optimization requirements within the manufacturing.

The work of this thesis began with a realistic problem met in the car manufacturing industry. Focusing on the internal logistics of car manufacturing factories, the Eco-Innovera project "Energy-Aware feeding SYstems (EASY)" aimed at favouring energy-aware practices. This project was a cooperation between the University of Navarra (Navarra, Spain), the University of Skövde (Skövde, Sweden) and the Laboratory of Analysis and Architecture of Systems (LAAS) (Toulouse, France). The industrial partners were Volvo Sweden and Volkswagen Spain. The project identified major energy influencing factors in raw material supplying inside assembly lines. It also integrated energy consumption factors explicitly into optimization procedures. The objective was to improve the energy efficiency, while keeping the manufacturing effectiveness under control. In this project, tactical/operational decision-aiding tools were developed based on simulation and optimization techniques, so as to highlight low-energy decisions and to favour energy awareness in the decision makers.

The Energy-Efficient Assembly-line Vehicle Supplying Problem (EEAVSP) is a theoretical operations research problem studied under the context of the EASY project, with the idea of improving energy efficiency in the feeding systems of assembly processes. In this problem, raw materials are transported by electric tow trains over a certain distance from a central depot (the "supermarket") to the assembly workstations. From a physical point of view, this transportation through the factory consumes significant amount of energy. By using energy wisely, one could obtain a better physical performance of the batteries in the toll-trains and consequently important savings in battery usage.

From a management point of view, one of the purpose of the study is to demonstrate that JIT and EAM paradigms can cohabit. Under the JIT paradigm, the performance of a feeding system is only measured by economical costs. A common criterion considered is the number of tours. However, the energy efficiency of this criterion is unknown. The negligence of environmental aspect in the JIT paradigm might be cancelled out by the energy aware decisions made in EAM paradigms. By putting financial objectives with environmental ones, the assembly lines could be both energy efficient and economically beneficial.

This chapter starts with a general presentation of the background of the problem. After that, an energy consumption estimation of the electric tow trains is introduced. Then, the problem is formally defined and its complexity is proved. Two mathematical models are proposed, followed by the solution methods and the experimentation results. This chapter finishes with a discussion on the main impacting factors for the energy consumption in feeding systems of assembly lines.

3.1 Background

The EEAVSP is motivated by a real situation encountered at and automotive assembly plant in Spain (Volkswagen) and an engine assembly line in Sweden (Volvo VLE). In this problem, the feeding line is composed of a supermarket, some workstations and some tow trains with several wagons. Figure 3.1 shows the position of the supermarket and the workstations linked by the rail of the tow trains. The following paragraphs explain the functionality of each part of the feeding system.

Supermarket The supermarket is the decentralizing area where raw materials of the whole assembly line are to be dispatched to each workstation. In other words, it contains all the components that are going to be used in the assembling activities of the line.

Workstation Each workstation is responsible for a certain number of assembly activities, putting components received from the supermarket into the final product.

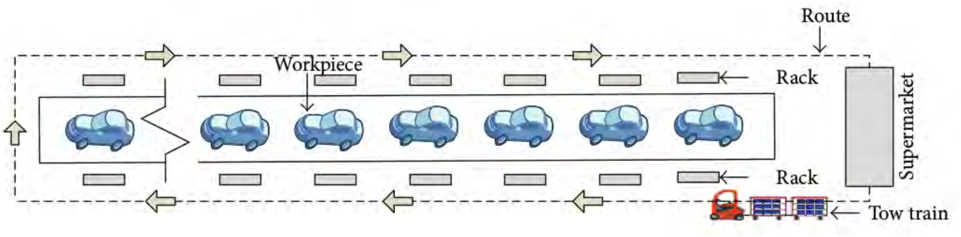


Figure 3.1: A straight assembly line view with the supermarket-concept [66]

The component demand of each workstation in a period is the total number of components consumed in this period of time.

Tow train A set of tow trains with wagons is in charge of delivering each component to a corresponding workstation. At the beginning of each period, a tow train is charged with the components at the supermarket. Once charged, it follows a predefined route and stops at workstations to provide the appropriate amount of corresponding components. The route is prefixed and closed. Finally, the empty tow train goes back to the supermarket to be recharged and gets ready for the next tour.

The objective is to minimize the energy consumed by the tow trains in the feeding system.

3.2 Energy Consumption of a Tow Train in an Assembly Line

This section characterizes the energy consumption of an electric tow train without regenerative braking effect. It rolls on a flat and concrete-made floor.

As discussed in [76], the main forces that influence the energy consumption of a tow train are: the traction force (F_{traction}), the friction (F_{friction}) and the air-drag force (F_{airdrag}), all in Newtons (N). The air-drag force can be neglected in comparison to the others because the mass of the tow train is relatively high and its speed pretty low, so that the vehicle aerodynamics are not very significant in this context. At each instant t, the friction varies in function of the total mass (payload and curb weight) of the vehicle $m(t)$ in kilograms (kg), the rolling coefficient C_r and the gravity g (in m/s^2), according to formula $F_{\text{friction}}(t) = m(t)gC_r$, with g and C_r considered constant over time. The traction force generates a motion between an object and a tangential surface against the friction. At each instant t, it depends on the total mass ($m(t)$) and the acceleration ($a(t)$ in m/s^2), with respect to the relation $F_{\text{traction}}(t) - F_{\text{friction}}(t) = m(t)a(t)$. Note that in each vehicle tour, the total mass $m(t)$ of the vehicle and its payload is a piece-wise constant and decreasing function over time, since the total mass stays the same between two workstations and components are unloaded at each workstation.

Therefore, the energy consumption, which is the integration of the forces multiplied by the vehicle speed along the time horizon H, can be expressed as follows:

$$\mathscr{E} = \int_0^H m(t)(a(t) + gC_r)v(t)dt \tag{3.1}$$

In the remaining of this section, it is assumed that the distance $D_{i,j}$ between two stations i and j is long enough for the tow train to reach its nominal speed. The motion of the tow train between two consecutive stops from a workstation (or the supermarket) i to another workstation (or the supermarket) j can be decomposed into three phases. The first phase corresponds to the acceleration when the vehicle starts from station i with speed 0 and accelerates continuously until the speed of the vehicle reaches the nominal value. The second is the cruise phase when the vehicle goes on with the nominal speed V_{max}. Finally, the deceleration phase precedes the stop at station j.

In the context of assembly line supplying, the acceleration or deceleration (A_{acc} or A_{dec}) and the nominal speed V_{max} are assumed known and constant as they are managed automatically by the vehicle controller. The rolling coefficient C_r is also constant since it is the same concrete floor in the whole factory. We refer to D_{acc} (or D_{dec}) as the distance travelled during the acceleration phase (or deceleration phase, respectively). They are independent of the origin and destination stations with fixed nominal speed and acceleration or deceleration ($D_{\text{acc}} = \frac{V_{\text{max}}^2}{A_{\text{acc}}}$ and $D_{\text{dec}} = \frac{V_{\text{max}}^2}{A_{\text{dec}}}$). The rest of the distance $D_{\text{cru}}^{i,j} = D_{i,j} - D_{\text{acc}} - D_{\text{dec}}$ is the distance travelled during the cruise phase. Similarly, the travel time is $T_{i,j} = T_{\text{acc}} + T_{\text{cru}}^{i,j} + T_{\text{dec}}$ with

$$T_{\text{acc}} = \frac{V_{\text{max}}}{A_{\text{acc}}} \tag{3.2}$$

$$T_{\text{cru}}^{i,j} = \frac{D_{i,j}}{V_{\text{max}}} - \frac{V_{\text{max}}}{2A_{\text{acc}}} - \frac{V_{\text{max}}}{2A_{\text{dec}}} \tag{3.3}$$

$$T_{\text{dec}} = \frac{V_{\text{max}}}{A_{\text{dec}}} \tag{3.4}$$

One can apply the energy formula (3.1) for each phase and obtain the energy expression $E_{i,j} = E_{\text{acc}}^{i,j} + E_{\text{cru}}^{i,j} + E_{\text{dec}}^{i,j}$. In the following of this chapter, we consider an immediate stop of the vehicle at station j, then the energy consumption of the vehicle going from workstation i to j during each phase can be expressed by:

$$E_{\text{acc}}^{i,j} = \frac{1}{2} m_{i,j} A_{\text{acc}} (A_{\text{acc}} + gC_r) T_{\text{acc}}^2 \tag{3.5}$$

$$E_{\text{cru}}^{i,j} = m_{i,j} g C_r V_{\text{max}} T_{\text{cru}}^{i,j} \tag{3.6}$$

$$E_{\text{dec}}^{i,j} = \frac{1}{2} m_{i,j} A_{\text{dec}} (A_{\text{dec}} - gC_r) T_{\text{dec}}^2 \tag{3.7}$$

With time in seconds (s), distance in meters (m), mass in kilograms (kg), accelerations in meters per square second (m/s^2), speed in meters per second (m/s), the energy obtained is in Joule (J) and it is converted into Kilowatt-hour (KwH).

If the supermarket is denoted by 0 at the beginning and by $Z+1$ at the end of a tour, and the visted workstations is denoted by the sequence $1, 2, \ldots, Z$, then the total energy consumption in the tour is:

$$\mathscr{E} = \sum_{i=0}^{Z} E_{\text{acc}}^{i,i+1} + E_{\text{cru}}^{i,i+1} + E_{\text{dec}}^{i,i+1} \tag{3.8}$$

Formulae (3.5)–(3.7) and (3.8) shed light on the energy influencing factors in the assembly line context. Firstly, it is observed from (3.5)–(3.7) that the energy consumption between two consecutive workstations i and j visited in a tour varies linearly in function of the carried mass $m_{i,j}$. In the case where the weights of the component bins are heterogeneous, a supplying strategy that delivers the heavy bins first can be more energy efficient, because heavy mass would be carried in a shorter distance in the total sum in Equation (3.8). Secondly, it is also reflected that the number of vehicle stops directly impacts the energy consumption, especially in assembly lines where the tow train stops very often. Minimizing the vehicle stops reduces the number of acceleration/deceleration phases, hence tends to minimize the energy consumption. Thirdly, the travelled distance $D_{i,j}$ is also directly linked to the energy as it impacts the cruise duration $T_{\text{cru}}^{i,j}$.

In conclusion, any strategy that aims at minimizing the energy consumption should take these three parameters (the carried mass between two workstations, the number of vehicle stops and the travelled distance) into account simultaneously.

3.3 Problem Definition and Complexity Analysis

This section gives a formal definition of the EEAVSP. First, the assembly line supplying environment defines parameters related to different parts of the feeding system.

Definition 1. Assembly line supplying environment
An assembly-line supplying environment is defined as:

- *an ordered set $\mathcal{Z} = \{1, \ldots, Z\}$ of workstations;*
- *a supermarket denoted by 0 or $Z+1$;*
- *a fleet $\mathcal{K}$ of K identical vehicles with mass capacity B and weight W;*
- *a set $\mathcal{H} = \{1, \ldots, H\}$ of periods;*
- *a vector $\mathcal{I}^0 = (I_1^0, \ldots, I_Z^0)$, which specifies for each workstation $i \in \mathcal{Z}$ its initial inventory $I_i^0 \in \mathbb{R}$ expressed as the number of bins of components;*
- *a vector $\overline{\mathcal{I}} = (Q_1, \ldots, Q_Z)$, which specifies for each workstation $i \in \mathcal{Z}$ the maximum number $\overline{I}_i \in \mathbb{N}$ of bins of components that can be left in front of the workstation;*

- *a vector* $\mathcal{W} = (W_1, \ldots, W_Z)$*, which specifies for each workstation* $i \in \mathcal{Z}$ *the weight* W_i *of one bin of components;*

- *a vector* $\mathcal{R} = (D_i^1, \ldots, D_i^H)$*, which gives for each workstation* $i \in \mathcal{Z}$ *its demand* $R_i^t \in \mathbb{R}$ *for period* $t \in \mathcal{H}$ *expressed as a portion of a bin.*

Note that in Definition 1, the capacity B of one vehicle is expressed as a maximum mass in kg, which differentiates our approach from others that express it as a maximum number of bins. Of course, both kinds of capacity can be considered simultaneously, but we chose the mass limitation that appears to be common for tow trains [63]. Now the notion of an assembly-line supplying strategy introduces the main decision variables $q_{i,k}^t$.

Definition 2. Supplying strategy
Given an assembly line supplying environment, a supplying strategy fixes the number of component bins $q_{i,k}^t \in \mathbb{N}$ *to be left at workstation* $i \in \mathcal{Z}$ *at period* $t \in \mathcal{H}$ *by each vehicle* $k \in \mathcal{K}$*.*

Note that in the case where $q_{i,k}^t = 0$, the vehicle k will not stop at workstation i during period t. Moreover, if $\sum_{i \in \mathcal{Z}} q_{i,k}^t = 0$, then the vehicle k will not make any tour at period t (it will stay at the supermarket). Also note that in the single vehicle case where $K = 1$, we will omit the index k.

On the basis of the previous definitions, the Feasible Assembly-line Vehicle Supplying Problem (FAVSP) is defined with its main constraints.

Definition 3. Feasible Assembly-line Vehicle Supplying Problem (FAVSP)
The FAVSP consists in finding a feasible assembly-line supplying strategy, if it exists. A supplying strategy is feasible, if and only if:

- *the total mass of the component bins carried at period* t *by vehicle* k *never exceeds the vehicle capacity* B*, i.e.,* $\sum_{i \in \mathcal{Z}} q_{i,k}^t W_i \leq B$ *for all* $k \in \mathcal{K}$ *and* $t \in \mathcal{H}$*;*

- *at each period* t*, the total number of non-empty bins located at workstation* i *never exceeds the workstation capacity* $\overline{I}_i$*, i.e.,* $I_i^0 + \sum_{k=1}^{K} \sum_{u=1}^{t} q_{i,k}^u - \sum_{u=1}^{t} D_i^u \leq \overline{I}_i$ *for all* $i \in \mathcal{Z}$ *and* $t \in \mathcal{H}$*;*

- *none of the workstations ever runs out of components, i.e.,* $I_i^0 + \sum_{k=1}^{K} \sum_{u=1}^{t} q_{i,k}^u - \sum_{u=1}^{t} D_i^u \geq 0$ *for all* $i \in \mathcal{Z}$ *and* $t \in \mathcal{H}$*;*

Finally, with minimizing total energy as the objective, the EEAVSP is defined based on the FAVSP with an additional set of variables $x_{i,j,k}^t$.

Definition 4. Energy-Efficient Assembly-line Vehicle Supplying Problem (EEAVSP)
The EEAVSP consists in finding a feasible supplying strategy that minimizes the energy consumption of the vehicle fleet, which is equal to:

$$\sum_{k=1}^{K}\sum_{t=1}^{H}\sum_{i=0}^{Z}\sum_{j=i+1}^{Z+1} x^t_{i,j,k} C^e_{i,j} (\sum_{l=j}^{Z+1} q^t_{l,k} W_l + W) \tag{3.9}$$

where:

- $x^t_{i,j,k}$ *equals* 1 *if vehicle* k *travels from* i *to* j *during period* t (0 *otherwise);*
- *workstations* 0 *and* $Z+1$ *are both referred to as the supermarket, assuming that* $q^t_{0,k} = s^t_{Z+1,k} = 0$ *for any* k *and* t*;*
- $C^e_{i,j}$ *with* $i < j$ *for each* $i, j \in \{0, Z+1\} \bigcup \mathcal{Z}$ *is the energy cost per mass unit for travelling directly from workstation* i *to workstation* j *without stops.*

Note that the objective (3.9) is not linear, but it can be easily linearised as shown in the objective function (3.10).

Complexity

The EEAVSP with a single vehicle remains NP-hard in the strong sense, even though the order of visit of the workstations is imposed. The proof is as follows.

Proposition 1. *EEAVSP is strongly NP-hard.*

Proof. The proof goes by showing that FAVSP is NP-Complete using a reduction from 3-Partition problem, which is known to be NP-complete in the strong sense [77]. A 3-partition problem consists in deciding whether a set $\zeta = \{a_1, \ldots, a_N\}$ of $N = 3n$ positive integers, such that $\sum_{i=1}^{N} a_i = nB$ and $a_i \in]B/4, B/2]$, can be partitioned into n subsets $\zeta_1, \ldots, \zeta_n$ such that the sum of integers in each subset equals B (note that, due to the bounds on the integer values, $\forall j \in \{1, \ldots, n\}$, $|\zeta_j| = 3$ in any YES-instance).

Obviously, FAVSP is in NP since, given a supplying strategy one can check in polynomial time whether it is feasible or not. From any 3-partition problem instance, we build up an instance of FAVSP as follows. We consider $K = 1$ vehicle, a set $\mathcal{Z}$ of $Z = N$ workstations and a time horizon $\mathcal{H}$ of n periods. The vehicle capacity is set to B. The initial workstation inventory I^0_i equals $n-1$ and the workstation capacity $\overline{I}_i = n$ for each workstation i. The component bin weights are such that $W_i = a_i$. It is also assumed that $R^t_i = 1$ for any period $t \in \mathcal{H}$ and for every workstation $i \in \mathcal{Z}$. As a consequence, exactly one component bin should be delivered to each workstation in the course of the n periods (no matter the period) to avoid any stock-out.

($\Leftarrow$) We first show that any YES-instance of 3-Partition can be associated to a solution of FAVSP. Clearly, any set ζ_t, with $t \in \{1, \ldots, n\}$ can be associated with a set of three component bins such that the sum of their weight a_i equals B, which is actually the maximum vehicle capacity. Therefore, a solution to FAVSP is given by taking $q_i^t = 1$ if $a_i \in \zeta_t$, 0 otherwise, $\forall t \in \{1, \dots, n\}$.

($\Rightarrow$) We now show that any solution to FAVSP gives a YES-instance of 3-Partition. As exactly one component bin has to be delivered to each workstation, all the $N = 3n$ component bins have to be brought to the workstations. They have to be partitioned among the n tours achieved by the vehicle. Due to the vehicle capacity B, a tour can bring at most three component bins to the workstation. Therefore, the 3-partition problem is a YES-instance only if the FAVSP admits one solution. □

3.4 Network Flow Representation

Any EEAVSP can be associated to a network $G = (V, A)$ such that:

- the set of vertices $V = V^1 \cup \cdots \cup V^H$ where a vertex $i^t \in V^t = \{0^t, (Z+1)^t\} \cup \mathcal{Z}^t$ is associated with a pair (i, t), for each period $t \in \mathcal{H}$ and each workstation or the supermarket $i \in \{0, Z+1\} \cup \mathcal{Z}$. Particularly, node 0^t or $(Z+1)^t$, associated with the supermarket, is the source or the sink nodes of the sub-graph defined by V^t;
- the set of arcs A is decomposed into two subsets A_I and A_V where:
  - subset A_I is made of the inventory arcs (i^t, i^{t+1}) with $t \in \{1, \dots, H-1\}$ and $i \in \mathcal{Z}$;
  - subset A_V is made of the vehicle arcs (i^t, j^t) with $t \in \mathcal{H}$, $i \in \{0\} \cup \mathcal{Z}$, $j \in \mathcal{Z} \cup \{Z+1\}$ and $j > i$.

This network is traversed by three categories of flow:

- the vehicle flow, which associates to each arc $(i^t, j^t) \in A_V$ a flow vector $X_{i,j}^t = (x_{i,j,1}^t, \dots, x_{i,j,K}^t)$ such that $x_{i,j,k}^t = 1$ if the elementary route (i^t, j^t) is used by vehicle k;
- the mass flow, which associates to each arc $(i^t, j^t) \in A_V$ a flow vector $M_{i,j}^t = (m_{i,j,1}^t, \dots, m_{i,j,K}^t)$ such that $m_{i,j,k}^t \leq B$ is the mass of the bins carried by vehicle k from workstation i^t to workstation j^t plus the vehicle mass W, provided that $m_{i,j,k}^t = 0$ if $x_{i,j,k}^t = 0$;
- the inventory flow, which associates to each arc $(i^t, i^{t+1}) \in A_I$ a flow $I_i^t \leq \overline{I}_i$ that equals the portion of bins still in the stock of workstation i at the end of period t (i.e., $I_i^t = I_i^0 + \sum_{k=1}^{K} \sum_{u=1}^{t} \left(\frac{(\sum_{j=0}^{i-1} m_{j,i,k}^u - \sum_{j=i+1}^{Z+1} m_{i,j,k}^u)}{W_i} \right) - \sum_{u=1}^{t} D_i^u$).

Figure 3.2 shows the network traversed by the mass and inventory flow. Each line in the figure corresponds to a fixed sequence route of assembly line in each period. Each column in the figure is the time variation of the inventory level of each workstation. It should be noted that the mass flow presented in the figure is not complete since we have omitted the arcs linking the workstations that are not directly linked (e.g. arcs $(0^1, 2^1)$, $(0^1, 3^1)$, $(0^2, 2^2)$, etc).

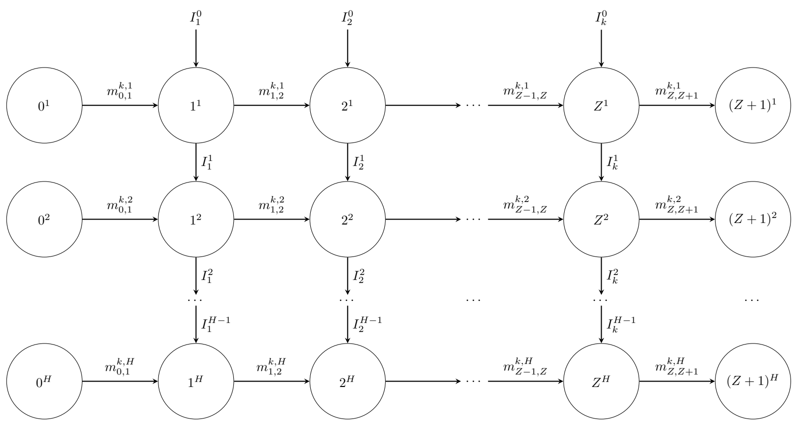


Figure 3.2: Network with mass and inventory flow for vehicle k

A cost $C^e_{i,j}$ is associated to each arc $(i^t, j^t) \in A_V$ that represents the energy required to bring one mass unit from station i to station j without any stop between. In the particular case where the arc $(0^t, (Z+1)^t)$ is used by vehicle k at period t (i.e., $x^t_{0,Z+1,k} = 1$), vehicle k does not move away from the supermarket at period t and the energy cost is null (i.e., $C_{0,Z+1} = 0$).

Under this particular network setting, EEAVSP amounts to find a mass flow that:

- is compatible with the vehicle flow (i.e., $m^t_{i,j,k} = 0$ if $x^t_{i,j,k} = 0$);
- satisfies the demands and the capacity of the workstations, i.e., $0 \leq I^t_i \leq \overline{I}_i$;
- minimizes the energy consumption that equals $\sum_{k=1}^{K}\sum_{t=1}^{H}\sum_{i=0}^{Z}\sum_{j=i+1}^{Z+1} C^e_{i,j} m^t_{i,j,k}$.

Reasoning in terms of mass is interesting since, as already highlighted in Section 3.2, the energy spent from one location to another location is proportional to the mass of the transported bins. Therefore, one can directly consider the cost related to the mass flow circulating in the network.

3.4.1 A First Mixed Integer Linear Programming Formulation

In this section, a Mixed Integer Linear Programming (MILP) model is presented for EEAVSP. This model is a basic flow formulation inspired from the network flow representation above. Instead of handling the flow in terms of number of components, the mass of the shipped components is considered, as also proposed in [105, 84].

As presented in 3.3, the decision variables are

- integer variables $q^t_{i,k}$ for the total number of components left at workstation i during period t by vehicle k;
- binary variables $x^t_{i,j,k}$ which equal 1 if the vehicle k travels from workstation i to workstation j at period t;
- continuous variables $m^t_{i,j,k} \in \mathbb{R}$ for the payload of vehicle k travelling from workstation i to workstation j during period t

Note that variables $m^t_{i,j,k}$ and $q^t_{i,k}$ are linked together by the relation

$$\sum_{j=0}^{i-1} m^t_{j,i,k} - \sum_{j=i}^{Z+1} m^t_{i,j,k} = W_i q^t_{i,k} \;\forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall t \in \mathcal{H}$$

Using the above decision variables, the energy minimization MILP model can be formulated as follows:

$$\text{minimize} \quad \mathscr{E} = \sum_{k=1}^{K}\sum_{t=1}^{H}\sum_{i=0}^{Z}\sum_{j=i+1}^{Z+1} C_{i,j} m^t_{i,j,k} + W \sum_{k=1}^{K}\sum_{t=1}^{H}\sum_{i=0}^{Z}\sum_{j=i+1}^{Z+1} C_{i,j} x^t_{i,j,k} \tag{3.10}$$

st:

$$I_i^0 + \sum_{k=1}^{K}\sum_{u=1}^{t} q^u_{i,k} - \sum_{u=1}^{t} D_i^u \geq 0 \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t \tag{3.11}$$

$$I_i^0 + \sum_{k=1}^{K}\sum_{u=1}^{t} q^u_{i,k} - \sum_{u=1}^{t} D_i^u \leq \bar{I}_i \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t \tag{3.12}$$

$$\sum_{(i^t,j^t)\in A_V} x^t_{i,j,k} - \sum_{(j^t,i^t)\in A_V} x^t_{j,i,k} = 0 \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t,\ \forall k \in \mathcal{K} \tag{3.13}$$

$$\sum_{(0^t,j^t)\in A_V} x^t_{0,j,k} = 1 \qquad \forall t \in \mathcal{H},\ \forall k \in \mathcal{K} \tag{3.14}$$

$$\sum_{(i^t,j^t)\in A_V} m^t_{i,j,k} - \sum_{(j^t,i^t)\in A_V} m^t_{j,i,k} = q^t_{i,k} W_i \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t,\ \forall k \in \mathcal{K} \tag{3.15}$$

$$m^t_{i,j,k} \leq B x^t_{i,j,k} \qquad \forall t \in \mathcal{H},\ \forall (i^t,j^t) \in A_V,\ \forall k \in \mathcal{K} \tag{3.16}$$

$$m^t_{i,j,k} \geq 0 \qquad \forall t \in \mathcal{H},\ \forall (i^t,j^t) \in A_V,\ \forall k \in \mathcal{K} \tag{3.17}$$

$$q^t_{i,k} \in \mathbb{N} \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t,\ \forall k \in \mathcal{K} \tag{3.18}$$

$$x^t_{i,j,k} \in \{0,1\} \qquad \forall t \in \mathcal{H},\ \forall (i^t,j^t) \in A_V,\ \forall k \in \mathcal{K} \tag{3.19}$$

The objective function (3.10) aims at minimizing the energy consumption $\mathscr{E}$, which is made up of the energy proportional to the mass $m^t_{i,j,k}$ traversing each arc $(i^t, j^t) \in A_V$, plus the energy consumed by transporting the vehicle mass. At each period $t \in \mathcal{H}$, constraints (3.11) and (3.12) ensure that the pallet bins in front of each workstation is enough to cover its demand and never exceed the workstation capacity, respectively. Constraints (3.13) and (3.14) ensure the conservation of the vehicle flow. The conservation of the mass flow in network G is imposed by constraints (3.15). Finally, constraints (3.16) enforce the respect of the maximum vehicle load capacity whenever arc $(i^t, j^t) \in A_V$ is used by vehicle k.

3.4.2 A More Powerful Mixed Integer Linear Programming Formulation

The quality of the linear relaxation of the previous flow-based formulation is affected by the large number of arcs in A_V, which increases the number of terms involved in constraints (3.13) and (3.15). This results in very high computation time for MILP solvers, because the search tree of their embedded Branch-and-Cut (B&C) procedure cannot be efficiently pruned when proving the optimality of one solution. In this section, we show how the flow network detailed in Section 3.4 can be reduced and translated into a new and more powerful MILP formulation.

Actually, in EEAVSP, the route of the vehicle is always fixed. To get from station i to j ($j > i+1$), the vehicle has to travel through all the stations $i+1, \dots, j-1$ between i and j. With fixed acceleration/deceleration rate (A_{acc} and A_{dec}) and fixed nominal speed ($V_{\max}$), the cost per mass unit for acceleration and deceleration is always the same. It is also assumed that the distance between each adjacent stations is enough long to perform the three phases from acceleration to deceleration. Consequently, the cost to travel between station i and $i+1$ with and without stop only depends on whether the distance for acceleration and deceleration (D_{acc} and D_{dec}) is travelled with the nominal speed $V_{\max}$ or not. In this way, the cost $C^e_{i,j}$ per mass unit associated to any arc $(i^t, j^t) \in A_V$ can be decomposed using equation (3.20), in which i and j are the stations of departure and arrival and the vehicle does not stop between these two stations. The cost $C^e_{i,j}$ is the sum of the cost to accelerate to nominal speed between stations i and $i+1$, plus all the cost to travel with nominal speed on each adjacent stations from $i+1$ to $j-1$, and the cost to decelerate to zero speed from station $j-1$ to j.

$$C^e_{i,j} = \alpha + \beta_{i,i+1} + \sum_{k=i+1}^{j-1} (\gamma + \beta_{k,k+1} + \delta) + \lambda \tag{3.20}$$

where:

- α is the acceleration cost paid for bringing one unit of mass from speed 0 to speed $V_{\max}$ starting from any workstation to cover the distance D_{acc}; it is constant for each pair of stations since the acceleration A_{acc} and the nominal speed $V_{\max}$ are constant;
- $\beta_{i,i+1}$ is the cost per mass unit for covering distance $D^{i,i+1}_{cru} = D_{i,i+1} - D_{\text{acc}} - D_{\text{dec}}$ at speed $V_{\max}$; it is dependent on the distance between adjacent stations i and $i+1$;

- γ is the cost per mass unit for travelling along distance D_{acc} at speed V_{max}; it is also independent from the pair of stations or period of time since D_{acc} is constant and the ground condition is the same;

- δ is the cost per mass unit for travelling along distance D_{dec} at speed V_{max}; similar to γ, it is constant for each pair of stations at each period, too; [1]

- λ is the cost per mass unit for decelerating along the distance D_{dec} from speed V_{max} to stop; same as in the accelerating phase, it is constant for each pair of stations for each period.

The interest of such a decomposition is that $C^e_{i,j}$ now only depends on costs between adjacent workstations. There are 4 scenarios for the energy cost between adjacent workstations i and $i+1$ during a period t (as show in Figure 3.3 and 3.4):

Case 1 the vehicle starts from speed 0 at station i and decelerates and stops at station $i+1$, which corresponds to the energy cost $C_{i,i+1} = \alpha + \beta_{i,i+1} + \lambda$ (see Figure 3.3a and 3.4a);

Case 2 the vehicle continues at speed V_{max} from station i but decelerates and stops at station $i+1$, with energy cost $C_{i,i+1} = \gamma + \beta_{i,i+1} + \lambda$ (see Figure 3.3b and 3.4b);

Case 3 the vehicle starts from speed 0 at station i but not decelerates (or stops) at station $i+1$, with energy cost $C_{i,i+1} = \alpha + \beta_{i,i+1} + \delta$ (see Figure 3.3c and 3.4c);

Case 4 the vehicle continues at speed V_{max} from station i to station $i+1$ without stop, which induces an energy cost $C_{i,i+1} = \gamma + \beta_{i,i+1} + \delta$ (see Figure 3.3d and 3.4d);

Without loss of generality, the deceleration is supposed to be instantaneous and the corresponding energy cost is 0. The four cases can be reduced to only two cases:

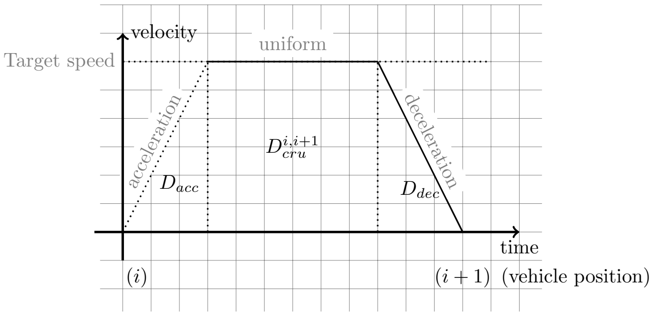
1. the vehicle starts from speed 0 at station i, with energy cost $C_{i,i+1} = \alpha + \beta_{i,i+1}$;

2. the vehicle continues at speed V_{max} from station i, with energy cost $C_{i,i+1} = \gamma + \beta_{i,i+1}$.

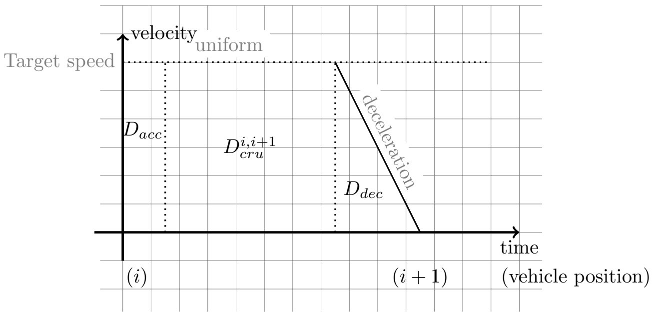
Graph $G(V, A_I, A_V)$ can then be transformed into multi-graph $G'(V, A_I, A'_V)$ as shown by Figure 3.5 and 3.6). The multi-graph G' only differs from G by subset A'_V composed by:

- the arcs $(0^t, 1^t)^{(0)}$ between vertex-pair 0^t and 1^t of V with unitary mass cost $\alpha + \beta_{0,1}$;

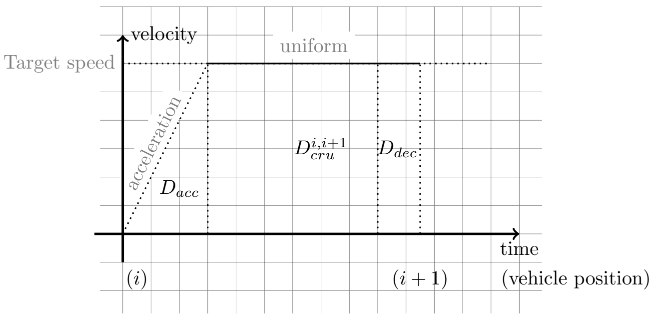
- the arc pairs $(i^t, (i+1)^t)^{(0)}$ and $(i^t, (i+1)^t)^{(1)}$ for each pair of adjacent vertices $i^t \in \{0\} \cup \mathcal{K}$ and $(i+1)^t$. Arc $(i^t, (i+1)^t)^{(0)}$ corresponds to case 1. It is associated with unitary energy cost $\alpha + \beta_{i,i+1}$. Arc $(i^t, (i+1)^t)^{(1)}$, associated with unitary energy cost $\gamma + \beta_{i,i+1}$, corresponds to case 2.



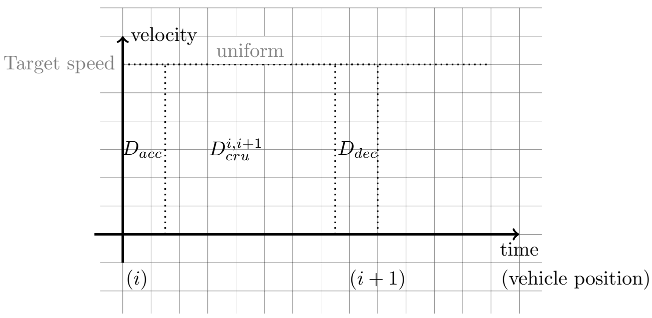
(a) Case 1



(b) Case 2

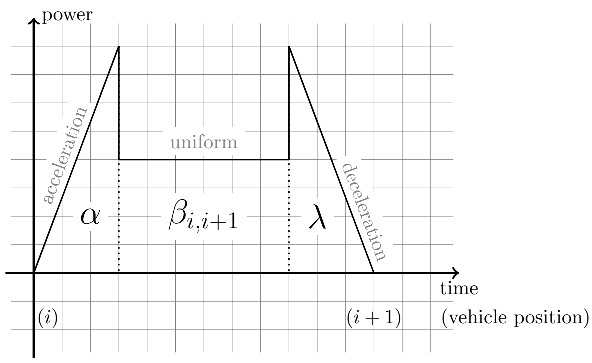


(c) Case 3

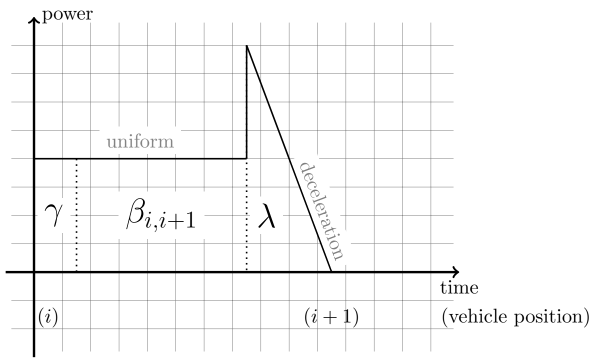


(d) Case 4

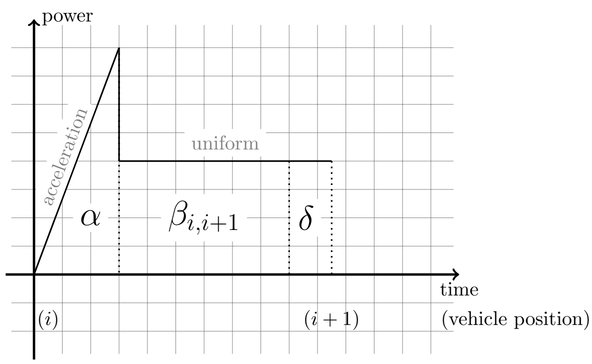
Figure 3.3: Vehicle speed profile scenarios



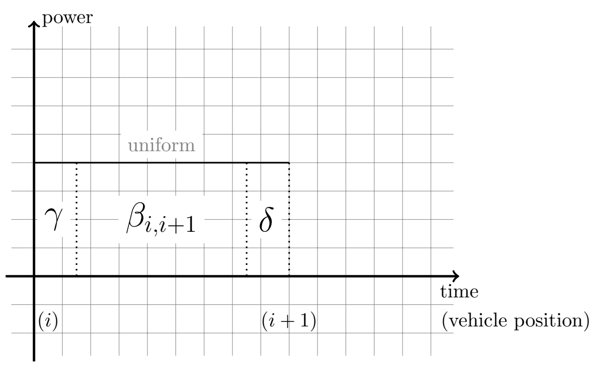
(a) Case 1



(b) Case 2



(c) Case 3



(d) Case 4

Figure 3.4: Vehicle power change scenarios

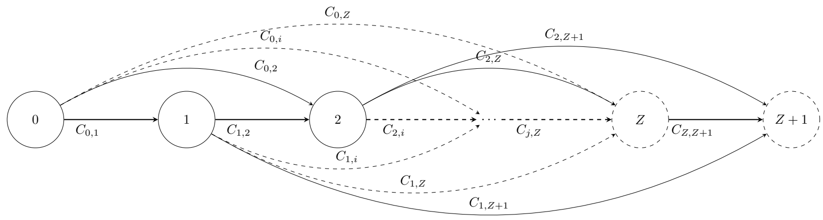


Figure 3.5: Simple graph for the supply chain with energy cost $C^e_{i,j}$ on arc (i, j)

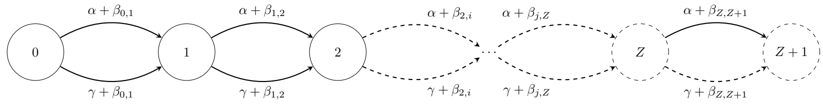


Figure 3.6: Simplified multi-graph for the supply chain with different energy costs on arcs

A vehicle flow associates to each arc $(i^t, (i+1)^t)^{(l)} \in A'_V$ (with $l \in \{0, 1\}$) a flow vector $Y^{(l),t}_{i,i+1} = (y^{(l),t}_{i,i+1,1}, \ldots, y^{(l),t}_{i,i+1,K})$ such that $y^{(l),t}_{i,i+1,k} = 1$ if vehicle k travels from i to $(i+1)$ with l the corresponding simplified case of stop.

Similarly, a mass flow vector $M^{(l),t}_{i,i+1} = (m^{(l),t}_{i,i+1,1}, \ldots, m^{(l),t}_{i,j,K})$ is associated to each arc $(i^t, (i+1)^t)^{(l)} \in A'_V$ (with $l \in \{0, 1\}$) such that $m^{(l),t}_{i,i+1,k}$ is the mass carried by vehicle k from i to $(i+1)$ during period t, provided that if the arc $l = 0$ is chosen, it stops at i, otherwise ($l = 1$) it does not stop at i.

Under this new graph setting, EEAVSP still amounts to finding a mass flow in G' which:

- is compatible with the vehicle flow (i.e., $m^{(l),t}_{i,i+1,k} = 0$ if $y^{(l),t}_{i,i+1,k} = 0$);
- satisfies the demands and the capacity of the workstations;
- minimizes the energy consumption.

This problem can be modelled by the following MILP model:

$$\text{Minimize} \quad \mathscr{E} = \sum_{k=1}^{K}\sum_{t=1}^{H}\left(\sum_{i=0}^{Z}(\alpha + \beta_{i,i+1})m^{(0),t}_{i,i+1,k} + \sum_{i=1}^{Z}(\gamma + \beta_{k,k+1} + \delta)m^{(1),t}_{i,i+1,k}\right)$$

$$+W\sum_{k=1}^{K}\sum_{t=1}^{H}\left(\sum_{i=0}^{Z}(\alpha + \beta_{i,i+1})y^{(0),t}_{i,i+1,k} + \sum_{i=1}^{Z}(\gamma + \beta_{k,k+1} + \delta)y^{(1),t}_{i,i+1,k}\right) \tag{3.21}$$

st: (3.11), (3.12)

$$\sum_{l=0}^{1} y0, 1, k^{(l),t} \leq 1 \qquad \forall t \in \mathcal{H},\ \forall k \in \mathcal{K} \tag{3.22}$$

[1]See Section 3.2 for the definition of D_{acc} and D_{dec}

$$\sum_{l=0}^{1} y_{i,i+1,k}^{(l),t} - \sum_{l=0}^{1} y_{i-1,i,k}^{(l),t} = 0 \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t,\ \forall k \in \mathcal{K} \tag{3.23}$$

$$\sum_{l=0}^{1} m_{i,i+1,k}^{(l),t} - \sum_{l=0}^{1} m_{i-1,i,k}^{(l),t} = q_{i,k}^t W_i \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t \tag{3.24}$$

$$m_{i,i+1,k}^{(l),t} \leq B y_{i,i+1,k}^{(l),t} \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t,\ \forall l \in \{0,1\},\ \forall k \in \mathcal{K} \tag{3.25}$$

$$m_{i,i+1,k}^{(l),t} \geq 0 \qquad \forall t \in \mathcal{H},\ \forall i^t \in V^t,\ \forall l \in \{0,1\},\ \forall k \in \mathcal{K} \tag{3.26}$$

$$q_{i,k}^t \in \mathbb{N} \qquad \forall t \in \mathcal{H},\ \forall i^t \in \mathcal{Z}^t \tag{3.27}$$

$$y_{i,i+1,k}^{(l),t} \in \{0,1\} \qquad \forall t \in \mathcal{H},\ \forall i^t \in V^t,\ \forall l \in \{0,1\},\ \forall k \in \mathcal{K} \tag{3.28}$$

The objective function (3.21) reformulates the energy consumption $\mathscr{E}$ in Equation (3.10) according to the energy cost composition (3.20), so that only the mass flows between adjacent workstations are taken into account. Constraints (3.11) and (3.12) are kept unchanged. Constraints (3.22) make sure that at most one arc outgoing from the supermarket is used in the solution. Constraints (3.23) and (3.24) ensure the conservation of the vehicle and mass flows (assuming that $y_{0,1,k}^{(1),t} = 0$ and $m_{0,1,k}^{(1),t} = 0$, i.e., vehicles k always start with speed 0 at the supermarket). Note that their left-hand side only involves four terms, which is significantly smaller than the number of terms Z in constraints (3.13) and (3.15).

Both MILP models were implemented but the performance of the more powerful one is significantly better in terms of computation time and the number of nodes explored in the B&C tree of the solver. In the following, only the experimentation results of the more powerful model are reported and discussed to see the energy influencing factors in the feeding system of assembly lines.

3.5 Experimentations and Remarks

3.5.1 Instance Generation

The instances used in the experiments are generated randomly based on the benchmark instances of Inventory Routing Problem (IRP) given in [16] by Archetti et al. There are 200 instances in total with a number of workstations varying from 5 to 50 and a time horizon varying from 3 to 12. For each value of Z and H, 5 problem instances were randomly generated. As presented in Section 1, an instance is defined by an assembly-line supplying environment, where:

- the size of the ordered set $\mathcal{Z}$ is an integer in the set $\{5, 10, 15, 20, 25, 30, 35, 40, 45, 50\}$;
- for each workstation $i \in \mathcal{Z}$, the weight W_i of its bin of parts is a random integer between 1 and 10 and is expressed in unit of mass (kg);

- for each workstation $i \in \mathcal{Z}$, its demand $R_i^t = R_i$ is a constant for each period $t \in \mathcal{H}$ and is expressed in terms of bins of parts randomly generated as an integer number in the interval $[10, 100]$ (the same as in [16]);
- the maximum number $\overline{I}_i$ of bins of parts that can be left in front of the workstation for each $i \in \mathcal{Z}$ equals $R_i^t g_i$ where g_i is randomly selected from the set $\{2, 3\}$ and represents the number of time units needed in order to consume the quantity $\overline{I}_i$ (see [16]);
- the initial inventory level I_i^0 for each workstation $i \in \mathcal{Z}$ is also the same as in the initial IRP instances given in [16] and equals $\overline{I}_i - R_i$;
- there is one single vehicle and its mass capacity B is equal to the vehicle capacity given in the initial IRP instance ($\frac{3}{2} \sum_{i \in \mathcal{Z}} R_i$) multiplied by the average bin mass of all the workstations ($\frac{\sum_{i \in \mathcal{Z}} W_i}{\|\mathcal{Z}\|}$);
- the distance between the supermarket and the first workstation is a random number between 50 and 500;
- the distance between each two consecutive workstations is a random number between 5 and 30;
- the vehicle mass W has been set to B, which appears to be the lowest weight value that can be chosen relative to the reality;
- the number of periods H varies in the set $\{3, 6, 9, 12\}$.

According to the technical report of the tow train used in the factory [103], the common nominal speed of the vehicle $V_{\max}$ is set to 15 km/h with a maximal payload of 6 tonnes and a vehicle weight of 1 tonne. Its acceleration A_{acc} is set to 0.5 m/s^2. Again, it is assumed that vehicle stops instantaneously at a workstation and the stop does not consumed any energy (i.e., $D_{\text{dec}} = 0$ and $\delta = 0$).

3.5.2 Implementation Details

The experiments has been conducted on a basic MacBook Air laptop having a 2 GHz process unit Intel Core i7 of 1600 MHz with 8 GB memory. We use the commercial MILP solver Gurobi (version 6.04). Only the results with the more powerful formulation are reported.

We highlight that our model allows to minimize either the total travelled distance or the energy consumption. The objective function $\mathscr{E}$ can be replaced with the total travelled distance $\mathscr{D} = D \sum_{k=1}^{K} \sum_{t=1}^{H} y_{0,1,k}^{(0),t}$, in which D is the fixed travel distance in one tour. Indeed, as $y_{0,1,k}^{(0),t}$ equals 1 only if vehicle k performs a tour at period t, minimizing $\mathscr{D}$ is equivalent to minimizing the number of tours. Recall the in the JIT paradigm, the number of tours is a crucial criterion for the economical efficiency of the supplying system. Therefore, as we are interested in the comparison between energy-efficient supplying strategies and economical-efficient ones, for each problem instance, the experimental strategy described by the high-level algorithm 2 is followed.

Algorithm 2 Experimental strategy

1: **for all** Instance $\mathcal{I}$ **do**
2: **for all** $H \in \{12, 9, 6, 3\}$ **do**
3: Initialize MILP model of section 3.4.2 according to $\mathcal{I}$ with time horizon H and objective function $\mathscr{D} = D \sum_{k=1}^{K} \sum_{t=1}^{H} y_{0,1,k}^{(0),t}$;
4: Run the solver and determine $\mathscr{D}^*_{\text{min}}$;
5: Replace objective function $\mathscr{D}$ by total energy consumption $\mathscr{E}$ and relaunch the solver to determine $\mathscr{E}^*_{\text{min}}$;
6: Fix the total distance at $\mathscr{D}^*_{\text{min}}$ and relaunch the solver once again but now maximize $\mathscr{E}$ to determine $\mathscr{E}^*_{\text{max}}$ imposing that

i the number of tours never exceeds $\mathscr{D}^*_{\text{min}}$;

ii the final inventory levels I_i^H equals 0;

In the algorithm step (6), we aim at determining a feeding strategy that maximizes the energy, provided that it remains optimal with regard to the number-of-tours criterion (i.e., we add the constraint $D \sum_{k=1}^{K} \sum_{t=1}^{H} y_{0,1,k}^{(0),t} \leq \mathscr{D}^*_{\text{min}}$). Moreover, in order to measure the energy gain in a fair way, we also constrain the final stock to be empty so that maximizing energy will never cause the increase of the number of delivered bins. In this way, we determine a feeding strategy that provide the workstations with the least number of components to avoid stock-out with the minimum number of tours (which is often the case if economical costs are taken as the only criterion in most of the assembly line feeding systems) and we estimate the worst energy consumption $\mathscr{E}^*_{\text{max}}$. By comparing it with $\mathscr{E}^*_{\text{min}}$, we show the largest energy saving that can be expected.

Finally, as decision-makers often give preference to fast solution procedures, we set up a maximum time limit of 60 seconds for each solver run. We are also interested in determining whether our formulation is able to achieve good solution quality in a short amount of time. In cases where the solver has failed in finding an optimal supplying strategy, we report the best upper and lower bound found so far.

3.5.3 Results

For the Control Processing Unit (CPU) Time required for the determination of $\mathscr{D}_{\text{min}}$ and $\mathscr{E}_{\text{min}}$, we observe that all the instances were solved to optimality in less than 9.3 seconds (0.93 seconds in average) for distance minimization and 1.6 seconds (0.27 seconds in average) for energy minimization. Thus, our model is quite efficient for both distance and energy minimization. Moreover, determining the worst energy strategy while maintaining the number of tours to its minimum does not significantly increase the CPU time.

Table 3.1 reports the computational effort required for the determination of $\mathscr{E}_{\text{min}}$. We distinguish between the minimum, mean and maximum CPU time required to solve the 5 instances having the same number of stations Z and the same number of periods H. For instances having no more than 10 workstations ($Z \leq 10$), a solution can be easily found in less than 10 seconds for up to 12 periods. For instances having only 3 periods, the optimality can be obtained in less than 60 seconds for up to 50 customers.

Table 3.2 shows the gaps to optimality in percentage of each set of instances. It is observed that the solver can find a solution with less than 10% gap for nearly all the instances. So in general, the model is very efficient and can be applied to real situations where a fast solution is needed.

		T															
		3				6				9				12			
		#OPT/5	Tcpu			#OPT/5	Tcpu			#OPT/5	Tcpu			#OPT/5	Tcpu		
			MIN	MEAN	MAX		MIN	MEAN	MAX		MIN	MEAN	MAX		MIN	MEAN	MAX
Z	5	5	0,0064	0,0171	0,03	5	0,0651	0,087	0,1027	5	0,1245	0,405	1,1151	5	0,3659	0,7879	1,1983
	10	5	0,0173	0,0412	0,0729	5	0,2005	0,3851	0,6354	5	0,4613	0,7176	1,0928	5	1,4107	4,5369	8,4942
	15	5	0,0748	0,1537	0,3564	5	4,2149	11,078	23,159	4	1,1577	19,024	60	1	42,774	56,555	60
	20	5	0,1064	0,1403	0,1631	4	11,024	35,512	60	4	3,6017	27,147	60	0	60	60	60
	25	5	0,073	0,1662	0,2328	0	60	60	60	2	20,91	45,434	60	0	60	60	60
	30	5	0,1538	0,3474	0,7104	0	60	60	60	0	60	60	60	0	60	60	60
	35	5	0,6221	4,8533	10,621	0	60	60	60	0	60	60	60	0	60	60	60
	40	5	0,2583	4,6085	16,212	0	60	60	60	0	60	60	60	0	60	60	60
	45	5	0,355	11,122	32,457	0	60	60	60	0	60	60	60	0	60	60	60
	50	5	5,2446	16,984	45,672	0	60	60	60	0	60	60	60	0	60	60	60

Table 3.1: CPU time in seconds for the determination of $\mathscr{E}_{\min}$.

		T											
		3			6			9			12		
		Gap			Gap			Gap			Gap		
		MIN	MEAN	MAX	MIN	MEAN	MAX	MIN	MEAN	MAX	MIN	MEAN	MAX
Z	5	0	0	0	0	0	0	0	0	0	0	0	0
	10	0	0	0	0	0	0	0	0	0	0	0	0
	15	0	0	0	0	0	0	0	0,0704	0,3518	0	1,7332	3,0947
	20	0	0	0	0	0,087	0,435	0	0,3235	1,6175	2,611	4,1806	7,5428
	25	0	0	0	1,1034	1,6687	2,643	0	0,7647	1,9952	2,4292	3,6698	4,9411
	30	0	0	0	2,2247	3,002	4,0024	0,9536	2,3587	3,2632	4,534	6,2751	7,7673
	35	0	0	0	3,2374	3,9889	4,7547	4,2027	5,4331	7,6618	7,5831	9,362	11,301
	40	0	0	0	3,2175	4,6497	6,0363	3,6058	5,2768	6,7736	7,7912	9,1473	10,342
	45	0	0	0	4,2817	5,4079	6,4933	3,3808	5,2807	6,1796	9,0561	10,999	13,866
	50	0	0	0	4,8608	5,7088	6,7068	5,2471	6,2623	7,5245	9,7775	10,944	12,749

Table 3.2: Gaps to optimality in percentage.

As to the energy gains, first of all, all the experiments show that minimizing energy reduces the number of tours to the minimum value. This can be explained by the energetic cost induced by the empty vehicle. Since the weight of the vehicle is non-negligible compared to the weight of products transported from one station to another, the weight of the vehicle contributes to a large portion of the total energy consumed. Minimizing total energy consumption means, in part, minimizing the energy consumed by the empty vehicle. As the distance travelled per tour is fixed in this problem, the energy consumed by the empty vehicle is a fixed value if the number of stops is fixed in a tour. Therefore, minimizing energy is not in contradiction with minimizing the number of tours as indicated by the JIT paradigm. Secondly, it is shown that even by fixing the number of tours at the minimum value, one can still improve the energy expenses a lot. This improvement is related to a better management of the distribution of bins to workstations.

Table 3.3 details the minimum, mean and maximum energy savings for each set of instances with the same Z and H. It is shown that the energy gain equals 35.5% in average and that it is often greater than 50%. It is also observed that, for $H \geq 9$, the energy savings tend to increase with the number of workstations Z. Therefore, in real situations where the tow train works for a long period of time and there is a lot of workstations to feed, the energy savings can be huge! This experimentation shows that the JIT paradigm is not in conflict with the energy-awared feeding. Instead, significant energy savings can be obtained without any economic sacrifice. One needs simply adapt the distribution or feeding strategy and the number of stops to a more energy efficient way.

		T											
		3			6			9			12		
		Energy gain			Energy gain			Energy gain			Energy gain		
		MIN	MEAN	MAX	MIN	MEAN	MAX	MIN	MEAN	MAX	MIN	MEAN	MAX
Z	5	4,0854	22,804	36,67	12,958	18,328	23,364	10,48	16,519	22,351	11,752	16,683	24,575
	10	4,4435	30,506	49,563	12,47	22,973	33,248	15,711	25,986	33,657	16,581	23,34	31,056
	15	40,878	44,583	47,779	25,543	28,769	32,873	20,977	27,443	33,709	20,813	27,781	35,018
	20	47,713	50,009	53,011	31,75	33,917	37,826	28,459	30,723	35,525	24,892	35,319	50,737
	25	3,7374	31,425	51,501	14,22	31,824	40,335	27,962	33,91	37,654	29,604	32,996	34,937
	30	3,3662	31,48	50,911	35,318	37,982	41,985	31,566	34,817	38,937	29,155	36,35	40,674
	35	51,548	52,978	54,13	38,149	39,732	41,401	35,022	39,393	47,713	36,268	42,148	44,626
	40	1,7177	32,349	55,022	37,073	40,761	44,213	34,969	40,8	46,441	41,966	47,704	53,074
	45	3,175	42,115	54,585	33,178	39,718	43,561	35,705	39,173	42,158	41,996	48,619	53,364
	50	51,294	52,374	54,733	36,94	39,628	43,493	36,312	45,782	51,43	42,742	48,686	53,547

Table 3.3: Energy savings in percentage.

3.5.4 Remarks

For assembly-line feeding systems, it is crucial to take the carried load, the number of stops and the total distance simultaneously into account in order to determine energy-efficient supplying strategies. A powerful MILP formulation is proposed, which integrates these parameters all together and determines energy-efficient strategies without loss of economic benefits. Moreover, it is shown that this formulation is quite generic as it also allows to find strategies that minimize the number of tours (which is the commonly-considered criterion under the JIT paradigm). In addition, strategies can also be found to maximize the energy cost with the minimal number of tours and delivered bins. By comparing the results of the energy consumption in minimizing energy, minimizing the number of tours and maximizing energy with the minimum number of tours and delivered bins, it is shown that minimizing energy can be complementary to the distance minimization.

On the other hand, the computational effort required for determining an energy-efficient strategy is always significantly greater than the one required for distance minimization (since the mass flows need to be taken into account). As this computational complexity will certainly get worse as the number of vehicles increases, additional researches are still needed in order to boost the MILP B&C procedure. Introducing new domination rules or specific problem-dependent cuts are among the directions of future research. More advanced optimization mechanisms such as generation of valid inequalities, variable fixing techniques, or decomposition approaches (column generation) could also be of interest.

Chapter 4

Mass-Flow Based Inventory Routing Problem with Energy Consumption

The problem met in the assembly lines can be generalised to larger supply chain networks, where the central depot monitors the inventory levels of a set of customers, provides products to each of the customers and makes deliveries by a set of vehicles. Under this setting, the IRP is recognized. In comparison with the EEAVSP, vehicle routing is now an additional decision.

By analysing the decisions of the IRP with energy consideration, we found that energy efficiency is an important but neglected issue that is highly worth studying under the setting of the IRP:

1. The visiting time to a customer is adaptable. We can choose a delivery time that is both convenient for the customers and also able to avoid rush hours, as congestion is one of the main causes of high energy consumption and CO_2 emissions.

2. Under the Vendor Managed Inventory (VMI) policy, the customer demands are flexible and can be distributed in different combinations. This inspires us to determine a set of delivery quantities that is the most effective for energy use while making sure that stock-out never happens.

3. The order of visit and the vehicle routes are to be determined. It is thus possible to design a routing strategy that considers the roads with the least energy costs.

In this chapter, energy issue is explicitly incorporated into the IRP. The compromise between the energy costs and the economical costs in the IRP is studied. First, an energy estimation method of a vehicle in a transportation network is introduced. Then, the Inventory Routing Problem with Energy Consideration (IRP-EC) is presented and the energy costs are integrated into the objective function of a flow formulation of IRP. In this way, the EEAVSP is generalized to the IRP-EC and a Mass-flow MILP formulation based on the existing IRP formulations is proposed. Finally, we discuss the possible influence of the distribution and routing strategy to the energy consumption of the inventory routing system.

4.1 Energy Consumption of a Vehicle in a Transportation Network

As mentioned in Section 2.2.2 in Chapter 2, the existing energy models reviewed in [48] resulting from automotive industrial test benches seems realistic and informative. However, most of them depend on vehicle type and focus only on fuel consumption. They are only applicable to certain driving cycles and can hardly be integrated with real driving situations. This can be partly explained by the fact that the main power source of vehicles used today is petroleum. This could also be due to the fact that driving systems integrated with real traffic condition are still under development. Nevertheless, with the emergence of electric and hybrid vehicles and the development of Intelligent Transportation System (ITS), we find it more appropriate to estimate the amount of energy independent of the type of vehicles, while integrating the road traffic information into this estimation.

In this thesis, a general simple model based on vehicle dynamics is proposed. It can give us a gross estimation of the energy required by a vehicle on a road segment with speed variation, independent of vehicle type or energy source. This model would be applicable to European suburban transportation network with short or medium distances and potentially high traffic intensity.

In our model, the stop rate τ, i.e., the number of stops per unit of distance is used to model the dynamics of the vehicle on a fixed segment of road. This parameter can also represent the traffic condition on the road. More precisely, with a traffic near free flow, τ takes a value near 0, which means that the vehicle goes through the road fluently without any stops. With congestion, this number is set to a bigger value to indicate a frequent speed variation. Usually τ takes a value between 0 and 4 depending on road types [11]. Moreover, there exists an interrelationship between the distance travelled, the stop rate, the speed and the acceleration of the vehicle, which is generally explained in Section 4.4.1.

Suppose a vehicle travelling from one location i to another location j. The path of the vehicle between two locations is supposed to be predefined with an average stop rate $\tau_{i,j}$, and the total distance travelled is $D_{i,j}$. So the vehicle stops $\tau_{i,j} \cdot D_{i,j}$ times during the trip. The coefficient of friction is a fixed parameter $C_r = 0.01$. The gravitational acceleration is $g = 9.81 m/s^2$. Road slopes, denoted by $\Delta_{i,j}$, is defined as the difference of altitude between the origin i and the destination j. If the vehicle climbs (the destination is higher than the origin), then $\Delta_{i,j}$ is positive. If the vehicle descends, then $\Delta_{i,j}$ is negative. The environmental effects of the road (wind, temperature etc.) as well as the viscosity of air are ignored. The only forces exerted on the vehicle are the gravity, the rolling resistance and the traction force of the engine.

The energy estimation is as follows: first, the speed profile of the vehicle between every two stops is defined and the energy between every two stops is computed; then, the energy of all the stops of a segment of road is summed up and this energy sum is defined as the energy cost of this segment of road; finally, the energy cost of a

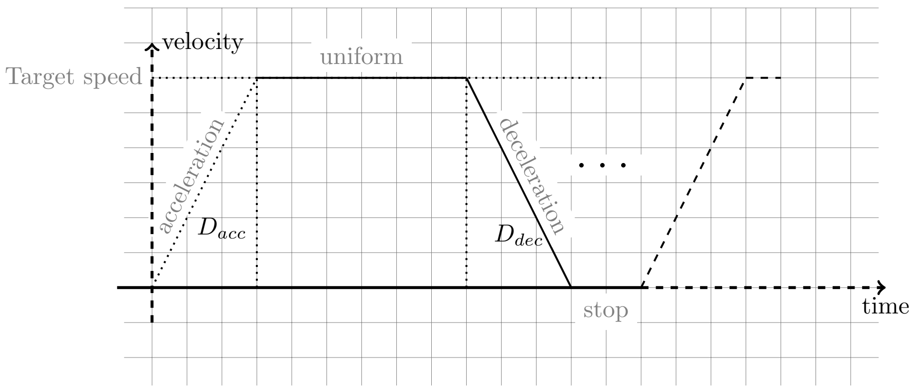


Figure 4.1: The speed variation of the vehicle with time

route is the summation of all the energy costs on each road segment included in this route.

4.1.1 Energy Cost between Every Two Stops

The energy estimation between every two stops is similar to the method presented in Section 3.2. Between every two stops, the vehicle speed is supposed to follow a fixed pattern of variation—acceleration, uniform speed movement and deceleration (Figure 4.1). First, the vehicle speeds up from 0 to the nominal speed $V_{\max}$ with a fixed acceleration A_{acc}. Then, it goes on at this speed for a while. And finally, it decelerates until it stops. The stops are not instantaneous anymore and there is a energy cost related to the deceleration, too.

The duration of the acceleration and deceleration phase of the motion of the vehicle is the same as given in (3.2) and (3.4). The total duration of the cruise phase is computed in the next part. Similar to the energy analysis in Section 3.2, the energy for each phase can be obtained by applying energy formulas (3.5)– (3.7).

4.1.2 Energy Cost on a Segment of Road

As explained before, on a road segment between two points i and j with distance $D_{i,j}$ and stop rate$\tau_{i,j}$, the pattern in Figure 4.1 is repeated $\tau_{i,j} \cdot D_{i,j}$ times. The duration of the cruising phase is the total duration to travel the additional distance which is longer than the total distance travelled in acceleration and deceleration. It can be computed using the following equation:

$$T_{\mathrm{cru}}^{i,j} = \frac{D_{i,j} - \tau_{i,j} D_{i,j}(D_{\mathrm{acc}} + D_{\mathrm{dec}})}{V_{\max}} \tag{4.1}$$

with $D_{\mathrm{acc}} = \frac{V_{\max}^2}{A_{\mathrm{acc}}}$ and $D_{\mathrm{dec}} = \frac{V_{\max}^2}{A_{\mathrm{dec}}}$.

The following assumption are made to simplify the computation of energy:

- the vehicle has no speed at both the starting and the ending points;

- the distance $D_{i,j}$ is long enough to allow the vehicle to accelerate to the nominal speed after each stop;
- after each stop, it speeds up again to the same target speed.

The kinetic energy consumption for the acceleration and cruise phase can be computed with formulas (3.5) and (3.7) with some adaptation for the number of stops:

$$\begin{aligned} E^{i,j}_{\text{acc}} &= \frac{1}{2}\tau_{i,j} D_{i,j} m_{i,j} A_{\text{acc}}(A_{\text{acc}} + gC_r) T_{\text{acc}} \\ E^{i,j}_{\text{cru}} &= m_{i,j} g C_r V_{\max} T^{i,j}_{\text{cru}} \\ E^{i,j}_{\text{dec}} &= \frac{1}{2}\tau_{i,j} D_{i,j} m_{i,j} A_{\text{dec}}(A_{\text{dec}} - gC_r) T_{\text{dec}} \end{aligned}$$

with $m_{i,j}$ the transported mass, T_{acc} the same as given in 3.2, $T^{i,j}_{\text{cru}}$ given in 4.1, and $T_{\text{dec}} = \frac{V_{\max}}{A_{\text{dec}}}$.

Finally, the total change of the potential energy $m_{i,j}\ g\ \Delta_{i,j}$ is added, which is the additional energy for the vehicle to climb the slope or the energy obtained while descending a slope. The total energy is thus:

$$\begin{aligned} E_{i,j} &= \frac{1}{2}\tau_{i,j} D_{i,j} m_{i,j} A_{\text{acc}}(A_{\text{acc}} + gC_r) T^2_{\text{acc}} \\ &+ m_{i,j} g C_r V_{\max} T^{i,j}_{\text{cru}} \\ &+ \frac{1}{2}\tau_{i,j} D_{i,j} m_{i,j} A_{\text{dec}}(A_{\text{dec}} - gC_r) T^2_{\text{dec}} \\ &+ m_{i,j} g \Delta_{i,j} \end{aligned} \quad (4.2)$$

From Fomula 4.2, the total energy cost per unit of mass when distance $D_{i,j}$ is travelled with stop rate $\tau_{i,j}$ with a climb of $\Delta_{i,j}$ is then:

$$C^e_{i,j} = \tau_{i,j} D_{i,j} V^2_{\max}(1 - \frac{gC_r}{A_{\text{dec}}}) + gC_r D_{i,j} + g\Delta_{i,j} \quad (4.3)$$

Formula 4.3 reflects the energy consumption per unit of mass on a segment of road between two points i and j. It is dependent on the distance travelled $D_{i,j}$, the stop rate $\tau_{i,j}$, the acceleration A_{dec}, the nominal speed $V_{\max}$ as well as the slope $\Delta_{i,j}$. It is recognized that the energy consumption is higher if the vehicle runs faster and longer, or when it climbs a slope. We can also see that as the deceleration increases, there will be energy used for braking, and the energy consumption becomes higher. Moreover, the less fluent the traffic flows ($\tau_{i,j}$ takes a bigger value), the higher the energy would be consumed.

4.1.3 Energy Cost for Each Edge

For each edge (i, j), $C^e_{i,j}$ given by Formula 4.3 is the energy cost per unit of mass. If a mass $m_{i,j}$ (kg) is loaded on the vehicle when traversing from i to j and the vehicle weighs W (kg), the energy consumption is then:

$$C^e_{i,j}(m_{i,j} + W) \tag{4.4}$$

Note that the energy cost for each edge cannot be decomposed as in Section 3.4.2, because on each edge, there are several repeating driving cycles and the energy consumption could not be simply divided into acceleration—uniform speed driving—deceleration on each edge. In the following, the Inventory Routing Problem with Energy Consideration (IRP-EC) is presented.

4.2 Problem Statement

The problem studied in this section is based on a *multi-period single-vehicle deterministic* IRP with *one depot and several customers*. Stock-out and back-orders are not allowed. In addition to the distance and inventory minimization, energy minimization is taken as objective. Both the Maximum Level (ML) and the Order-up-to Level (OU) policies are considered to see the influence of different replenishment strategies on the energy consumption.

The problem is constructed on a *complete undirected* graph $G(V, E)$ with V the vertex set and E is the set of undirected edges. The set V includes one depot denoted by 0 and Z customers to visit denoted by the set $\mathcal{Z} = \{1, \ldots, Z\}$.

There are H replenishment planning periods. Each period can be a day, a week or even a month according to the applications. A set of vehicles is based at the depot. One single vehicle leaves the depot at most once per period. If a tour is done in a period, the vehicle starts from the depot, makes a tour around the customers that need to be refilled and returns to the depot at the end of the period.

To facilitate the energy estimation, two units are used to measure inventory components—the number of components and the weight in kilograms (kg). The number of components is used by the customers to represent their inventory levels and to count the number of packages of delivered goods. The weight is used by the transporters. It is the physical mass of the components transported by the vehicle. The weight of one unit of component is not always the same for each customer and it is denoted by W_i in kilograms (kg) for customer $i \in \mathcal{Z}$. The vehicle has a capacity B expressed as a mass limit in (kg). The empty vehicle mass, or curb weight (kg) of the vehicle is W.

Inventory levels at customers and depot are monitored during the whole planning time horizon. They are summarised at the end of each replenishment period. The customer demands are described as a deterministic constant demand rate per period denoted by R_i units of components for customer $i \in \mathcal{Z}$. In particular, R_0 is the number of components made available at the depot in each period. Each customer $i \in \mathcal{Z}$ has a stocking capacity $\overline{I}_i$ while the depot is supposed to have an unlimited stocking capacity. It is also assumed that the minimum inventory level of each customer is 0.

The costs are:

- the inventory storage costs, with c_i^{inv} the unitary cost of component stored at customer i or the depot for one period of time;

- the transportation costs, which in classical IRP is estimated as the total distance travelled for the delivery of all the periods;

- the energy costs defined in Section 4.1.

The problem is to decide a replenishment plan for each customer and to find tours of the vehicles in each period to minimize the costs, while making sure that the customers are never out of stock. The problem is NP-hard since it is a generalization of the EEAVSP by add the routing part. It is also a generalization of the IRP by adding the energy consideration.

Two variables are defined for the inventory management:

- $I_i^t \in \left[0, \overline{I}_i\right]$ is a continuous variable representing the inventory level in number of components at the depot 0 or at the customer $i \in \mathcal{Z}$ at the end of period t.

- $q_{i,k}^t$ is the number of components delivered to customer $i \in \mathcal{Z}$ during period $t \in \mathcal{H}$.

Three sets of decision variables z_i^t, $x_{i,j}^t$ and $y_{i,j}^t$ are for the vehicle routing:

- For each $i \in \mathcal{Z}$, $t \in \mathcal{H}$, z_i^t is a binary variable indicating whether customer i is served at period t. It equals 1 if customer i is served and 0 otherwise. Particularly, z_0^t indicates whether the tour at period t is performed (equals 1) or not (0).

- For each arc $(i,j) \in V \times V$ and each period $t \in \mathcal{H}$, variable $x_{i,j}^t$ is a binary variable to indicate the direction of the vehicle route. It equals 1 if the vehicle travels from node i to j at period t and 0 otherwise.

- For each edge $(i,j) \in A$ and each period $t \in \mathcal{H}$, $y_{i,j}^t$ is an integer variable indicating the number of times that edge (i,j) is used in the tour of period t.

4.3 Mass-Flow MILP Formulation

A flow formulation is given in [17] to model the inventory flows inside the transportation network. Instead of the flow in terms of number of components as in [17], it is the mass flow of the shipped components that is considered in our model. Once the mass transported on each edge of the network at each period is decided, the number of components left at each customer vertex can be deduced. Inversely, if the number of components delivered to each customer at each period are known, the order of visits can be decided and a flow of mass in the transportation network that minimizes the energy consumption can be obtained.

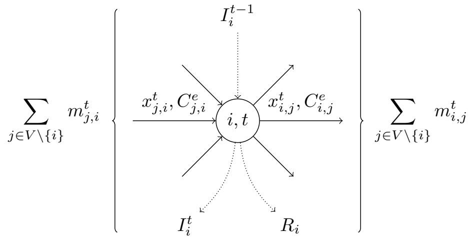


Figure 4.2: The flows passing through customer i at period t

4.3.1 Network Flow Representation

Figure 4.2 details the various flows traversing customer i at period t, similar to the one proposed in 3.4. Three kinds of flows inside the network are to be decided:

- the vehicle flow is represented by variables $x_{i,j}^t$, indicating the routes of the vehicle at each period;
- the mass shipped by the vehicle from i to j at period t is denoted by variables $m_{i,j}^t$. They are linked with the vehicle flow variables $x_{i,j}^t$. If the vehicle does not go from i to j at period t ($x_{i,j}^t = 0$), $m_{i,j}^t$ is equal to 0. The mass flow (associated with the solid arcs in Figure 4.2) corresponds to the mass of the incoming and outgoing products. They are used to estimate the potential energy consumption, with $C_{i,j}^e$ the energy cost per unit of mass on edge (i, j). The difference $\frac{1}{W_i}(\sum_{j \in \mathcal{Z} \setminus \{i\}} m_{j,i}^t - \sum_{j \in \mathcal{Z} \setminus \{i\}} m_{i,j}^t)$ gives the number of components $q_{i,k}^t$ delivered by the vehicle to customer i during period t, provided that it is integer.
- the inventory flow (associated with the dotted arcs in the figure) denoted by I_i^t for each customer $i \in \mathcal{Z}$ in each period $t \in \mathcal{H}$, indicates the change of the inventory level with the demand R_i at the customer i from one period to another.

4.3.2 Mathematical Model

With the variables and the network representation detailed above, a mathematical model is presented below.

Objectives Two objectives are defined. The Objective (4.5) is for inventory and distance optimization, which is the sum of the total distance travelled plus the sum of the inventory storage costs over all the periods.

$$\min \sum_{t \in \mathcal{H}} \sum_{(i,j) \in V \times V} D_{i,j} x_{i,j}^t + \sum_{t \in \mathcal{H}} \sum_{i \in V} c_i^{inv} I_i^t \tag{4.5}$$

The Objective (4.6) is for minimizing the total energy consumed in the inventory routing over all the periods. It contains two terms: the first one is a flexible cost related to the transported mass of the vehicle $m^t_{i,j}$, and the second one is a fixed cost induced by the vehicle curb weight W.

$$\min \sum_{t\in\mathcal{H}} \sum_{(i,j)\in V\times V} C^e_{i,j} m^t_{i,j} + W \sum_{t\in\mathcal{H}} \sum_{(i,j)\in V\times V} C^e_{i,j} x^t_{i,j} \tag{4.6}$$

Constraints Compared with the basic flow formulation in [17], flows are considered in terms of mass to link the energy consumption with the inventory management and transportation.

Inventory Management Constraints (4.7) to (4.11) are for monitoring the inventory levels of each location at each period.

$$I^t_0 = I^{t-1}_0 + D_0 - \sum_{i\in\mathcal{Z}} q^t_i \qquad \forall t \in \mathcal{H} \tag{4.7}$$

$$I^t_i = I^{t-1}_i - R_i + q^t_i \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.8}$$

$$q^t_i \geq \overline{I}_i z^t_i - I^{t-1}_i \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.9}$$

$$q^t_i \leq \overline{I}_i - I^{t-1}_i \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.10}$$

$$q^t_i \leq \overline{I}_i z^t_i \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.11}$$

Constraints (4.7) and (4.8) ensure that the inventory levels of each station are coherent from one period to another. The OU inventory policy is ensured by constraints (4.9) and (4.10)—after each delivery, the inventory level of each visited customer is fulfilled to the maximum. If we delete Constraints (4.9), the ML policy is considered, where the replenishment level is flexible but bounded by the stocking capacity of each customer. Constraints (4.11) ensure that if a customer i is not visited at a period t ($z^t_i = 0$), the delivered quantity q^t_i equals 0 and if the customer is visited, the delivered quantity never exceeds his capacity.

Commodity Mass Flow Management Constraints (4.12) and (4.13) are the mass flow constraints.

$$\sum_{j\in\mathcal{Z}} m^t_{0,j} = \sum_{i\in\mathcal{Z}} q^t_i W_i \qquad \forall t \in \mathcal{H} \tag{4.12}$$

$$\sum_{j\in V} m^t_{j,i} - \sum_{j\in V} m^t_{i,j} = q^t_i W_i \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.13}$$

Constraints (4.12) ensure that at period t, the mass leaving the depot is equal to the total mass transported to all the customers. Constraints (4.13) ensure that for each customer i at each period t, the quantity received is equal to the difference between the incoming and the outgoing mass flow.

Vehicle Routing Constraints (4.14) to Constraints (4.19) are typical routing constraints.

Undirected routing

$$\sum_{j \in \mathcal{Z}} y_{0,j}^t = 2z_0^t \qquad \forall t \in \mathcal{H} \tag{4.14}$$

$$\sum_{\substack{j \in V \\ j<i}} y_{j,i}^t + \sum_{\substack{j \in \mathcal{Z} \\ j>i}} y_{i,j}^t = 2z_i^t \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.15}$$

Directed vehicle flow

$$\sum_{j \in \mathcal{Z}} x_{0,j}^t = z_0^t \qquad \forall t \in \mathcal{H} \tag{4.16}$$

$$\sum_{j \in V} x_{i,j}^t = z_i^t \qquad \forall t \in \mathcal{H}, i \in \mathcal{Z} \tag{4.17}$$

$$\sum_{j \in V} x_{j,i}^t = z_i^t \qquad \forall t \in \mathcal{H}, i \in \mathcal{Z} \tag{4.18}$$

$$y_{i,j}^t = x_{i,j}^t + x_{j,i}^t \qquad \forall t \in \mathcal{H}, (i,j) \in E \tag{4.19}$$

Constraints (4.14) and (4.15) define the non-directed route of the vehicle in each period. Constraints (4.16)–(4.18) restrain the direction of the vehicle flow. They link y and z variables to make sure that in each period at most one tour is performed and that each customer is visited at most once in each period. Constraints (4.19) link variables y and x to ensure that each edge is used at most once in each period.

Vehicle Capacity Constraints (4.20) and (4.21) guarantee that the vehicle capacity is never exceeded both in number of components and in unit of mass.

$$\sum_{i \in \mathcal{Z}} q_i^t W_i \leq B z_0^t \qquad \forall t \in \mathcal{H} \tag{4.20}$$

$$m_{i,j}^t \leq B x_{i,j}^t \qquad \forall t \in \mathcal{H}, (i,j) \in V \times V \tag{4.21}$$

Constraints (4.21) also link the mass flow and the vehicle flow on the graph. They enforce the direction of the vehicle flow to be the same as that of the mass flow.

Variable Domains Constraints (4.22)–(4.28) are the variable domains.

$$0 \leq I_i^t \leq \overline{I}_i, I_i^t \in \mathbb{N} \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.22}$$

$$0 \leq q_i^t \leq \overline{I}_i, q_i^t \in \mathbb{N} \qquad \forall i \in \mathcal{Z}, t \in \mathcal{H} \tag{4.23}$$

$$0 \leq m_{i,j}^t \leq B \qquad \forall (i,j) \in V \times V, t \in \mathcal{H} \tag{4.24}$$

$$x_{i,j}^t \in \{0,1\} \qquad \forall (i,j) \in V \times V, t \in \mathcal{H} \tag{4.25}$$

$$y_{i,j}^t \in \{0,1\} \qquad \forall (i,j) \in E, i<j, t \in \mathcal{H} \tag{4.26}$$

$$y^t_{0,j} \in \{0, 1, 2\} \qquad \forall j \in \mathcal{Z}, t \in \mathcal{H} \qquad (4.27)$$

$$z^t_i \in \{0, 1\} \qquad \forall i \in V, t \in \mathcal{H} \qquad (4.28)$$

All the variables except $m^t_{i,j}$ take integer values. Note that for variables x^t_{0j}, since direct routing is possible, they can be assigned with value 2.

4.4 Experimentations and Results

The existing IRP instances proposed in [16] are adapted for energy estimation. The MILP model is constructed and solved using the adapted instances. An analysis of the obtained results is presented.

4.4.1 Data Generation

Benchmark instances proposed in [16] are taken to generate new test instances. Information on stop rates τ and vehicle nominal speeds $V_{\max}$ relative to the travelling distance on different arcs is first added. The correlation within these parameters is determined based on empirical data of delivery trucks on real routes provided in [133]. The following part explains how the data set is generated.

Two types of road are considered: highway and national route. The target speed and the number of stops for different types of roads are generated using different methods according to the road types. On a highway, the nominal speed is fixed at 110 km/h, and the number of stops is fixed at 2 stops per edge no matter how long is travelled. On a national route, the vehicle speed is fixed at 80 km/h and the number of stops is linearly dependent on the distance with a random error. For all types of road, the average acceleration and deceleration rate is fixed at 1.01 m/s^2. For each edge between two locations, the type of road is generated randomly. The instances generated contain two categories of road type proportion: one is with $\frac{2}{3}$ edges among all the edges defined as highway and $\frac{1}{3}$ as national route; the other is with $\frac{1}{3}$ edges among all defined as highway and $\frac{2}{3}$ as national.

Then, a random number between 1 and 10 is generated for each customer i to represent the mass of one unit of its components W_i. Vehicle weight and mass capacity are correlated according to vehicle information provided in [63]. A random number between 0 to 500 is also generated as the altitude of each location.

There are 64 cases in total. Each case contains 5 instances. The cases are categorized by the number of periods (3 or 6), the proportion of the inventory storage cost in relation to the transportation cost (high or low), the inventory replenishment policy (OU or ML), the proportion of each type of road in the whole map (highway:national = 1 : 2 or 2 : 1) and the number of customers in the map.

4.4.2 System Settings

The model was coded in C++ with *IBM® ILOG® CPLEX 12.6.1.0* and solved by the default B&C algorithm with one thread. The operating system is *Ubuntu 14.04*

	ML policy					OU Policy				
T	n	status1	status2	time1	time2	n	status1	status2	time1	time2
3	5	Optimal	Optimal	0.137	0.0892	5	Optimal	Optimal	0.0992	0.0844
	10	Optimal	Optimal	1.78	1.60	10	Optimal	Optimal	1.70	1.34
	15	Optimal	Optimal	12.5	61.5	15	Optimal	Optimal	16.6	35.0
	20	Optimal	Optimal(13)	199	976	20	Optimal	Optimal(17)	65.6	749
	25	Optimal	0.084	67.01	1800	25	Optimal(12)	0.058	787	1800
	30	Optimal	0.12	310	1800	30	Optimal(12)	0.10	1017	1800
	35	Optimal	0.15	183	1800	35	Optimal(8)	0.15	1248	1800
	40	Optimal(16)	0.17	624	1800	40	Optimal(2)	0.19	1714	1800
	45	Optimal(14)	0.18	756	1800	45	0.054	0.22	1800	1800
	50	Optimal(5)	0.23	1649	1800	50	0.10	0.27	1800	1800
6	5	Optimal	Optimal	2.53	0.401	5	Optimal	Optimal	0.489	0.465
	10	Optimal	Optimal	45.0	54.9	10	Optimal	Optimal	29.2	55.3
	15	Optimal	Optimal(1)	429	1790	15	Optimal	Optimal(3)	169	1630
	20	Optimal(4)	0.098	1639	1800	20	Optimal(7)	0.10	1487	1800
	25	Optimal(4)	0.14	1575	1800	25	Optimal(6)	0.16	1515	1800
	30	0.077	0.20	1800	1800	30	0.075	0.21	1800	1800

Table 4.1: Solution status and solving time

LTS with *Intel® Core® i7-4790 3.60GHz* processor and *16 G* memory.

Since the transportation and inventory minimization can give a good start for the energy minimization in a reasonable time, the solution process is divided into two phases. In the first phase, the objective is to minimize the combined cost of transportation and inventory as given in objective (4.5). In the second phase, starting with the solution of the first phase, the same model is solved to minimize the total energy consumption as computed in objective (4.6).

A time limit of 1800 seconds is set for each of the two phases. All the other settings of CPLEX are as default. The results of both of the two phases are compared in the next part.

4.4.3 Performance

The dimension of an instance is determined by the number of periods and the number of customers. The inventory policy (OU or ML) changes the constraint set of the model. The combination of these three parameters defines a category of instances. Each category contains 20 instances. In Table 4.1, computation time in seconds of each solution phase ("time1" and "time2") and the solution status within the time limit ("status1" and "status2") are listed for each category. Information marked with 1 corresponds to the inventory/transportation cost minimization and that marked with 2 is for energy minimization. The values for computation time are average values over all the instances of the same category. If all the instances of a category can be solved to optimality by CPLEX, the status is noted "Optimal". If part of the instances of a category can be solved to optimality, then the status is noted "Optimal(n)" with a number n in parentheses indicating the number of instances solved to optimality in this category. Otherwise, if no optimal solution is found in the time limit by CPLEX, then the average relative gap after 1800 seconds of computation is reported in the status column, and the time value is noted 1800.

As we can see from Table 4.1, energy minimization is much more difficult to solve than inventory and transportation cost minimization (time2 $\gg$ time1). This may result from the large possible number of combinations of the mass flows. It becomes more difficult as the dimension of the instances increases. For both OU and ML policies, instances larger than 20 customers with 3 periods or 15 customers with 6 periods can hardly be solved to optimality for energy optimization within the time limit. The influence of the inventory policy on the computation time when minimizing energy is not very obvious.

4.4.4 Energy Impacting Factors

Let us note the energy consumption in solution phase 1 (inventory/transportation minimization) by $\mathscr{E}_1$ and the consumption in solution phase 2 (energy minimization) by $\mathscr{E}_2$. The energy reduction in the following paragraphs is defined as the ratio $r = \frac{\mathscr{E}_2 - \mathscr{E}_1}{\mathscr{E}_1}$ in percentage. In general, the energy reduction can achieve 35% in average. It is at least 21% and can reach as high as 46%.

Several factors have an impact on the energy reduction. First, the size of the instance can influence the potential energy reduction. Larger instances tend to induce higher energy conservation. Figure 4.3 shows the variation of the average energy reduction in relation with the number of customers in average.

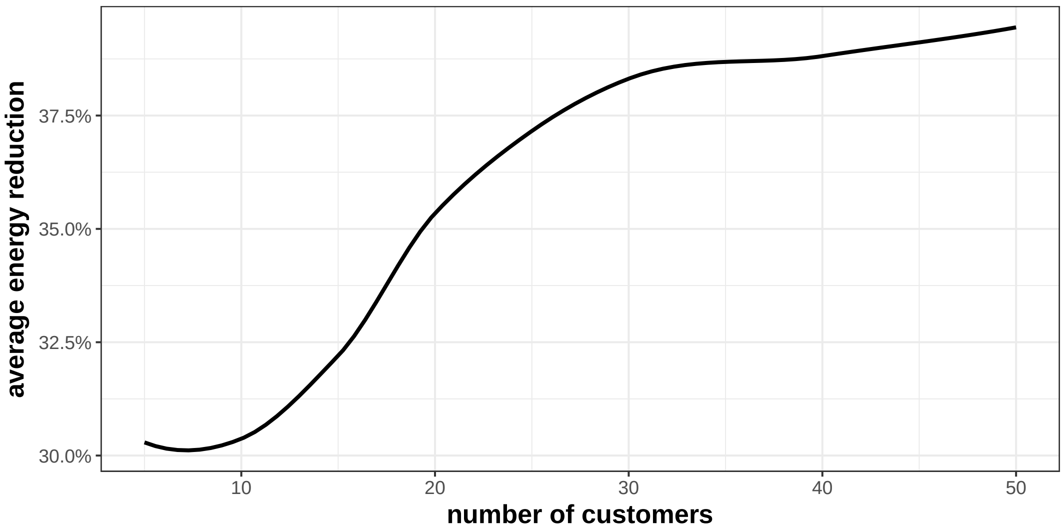


Figure 4.3: Number of customers and energy reduction

Second, a compromise between the inventory/distance cost and energy cost exists since all the energy reduction necessitates an augmentation of distance and inventory costs whatever policy or planning horizon is considered. This is shown in Figure 4.4, where road type “A2N1” means that 2/3 arcs of all the arcs are highway and 1/3 are national route, similar for “A1N2”. It is observed that under the configuration with more national routes (the case of “A1N2” where the number of stops is more variable), the compromise becomes more obvious as energy reduction requires more augmentation of inventory/transportation costs compare to the case with "A2N1".

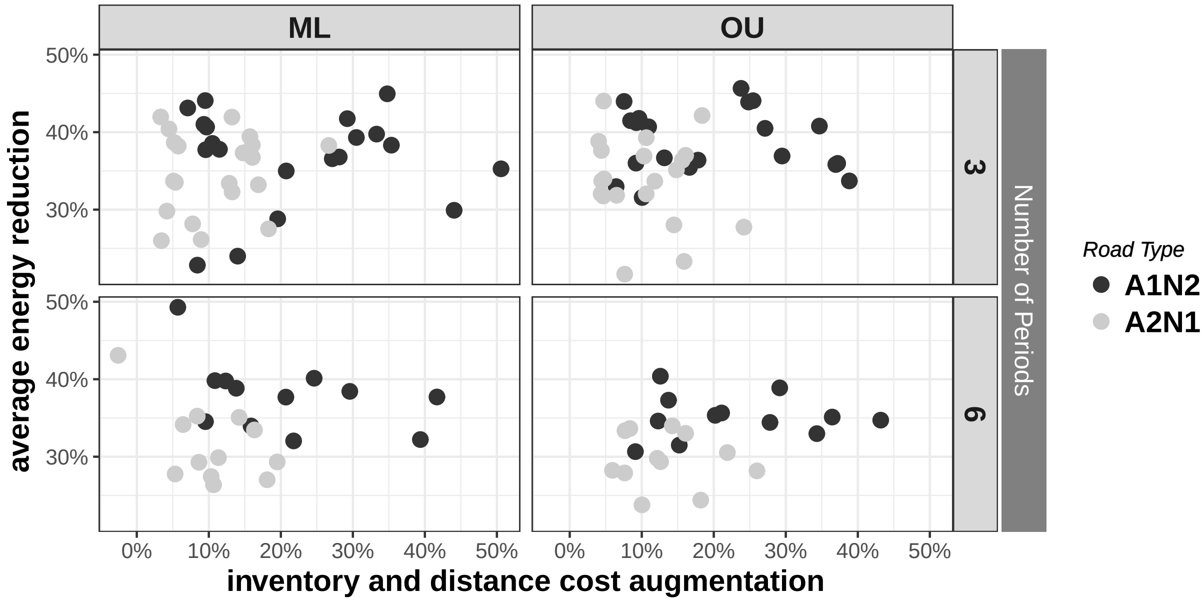
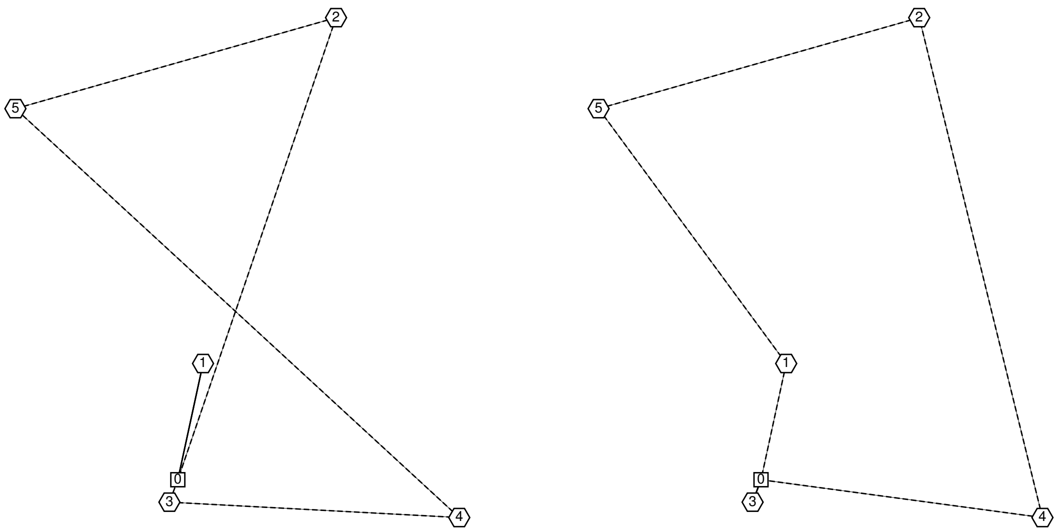


Figure 4.4: Distance or inventory cost and energy cost under different configurations

Third, different from the results in Section 3.5.3 for the EEAVSP under the context of the assembly lines, energy reduction is not equivalent to distance minimization. The shortest route is not necessarily the one that minimizes the energy. This result is not in contradiction with that in Section 3.5.3, because in EEAVSP, the route of the vehicle is fixed in each tour, and the variation of the energy consumption is only dependent on the variation of the loads and the number of stops. In addition, distance or the number of tours are highly weighted in the objective of EEAVSP due to the heavy weight of the vehicle in comparison with its loads. The results for the IRP-EC shows that both the distance and the vehicle weight and loads account for the energy consumption. The vehicle with a high load tends to start his journey with the least unit energy cost arc and put to the end the visit to a customer in an area with high unit energy cost. In this way, the most heavy loads are transported on roads with the smallest energy cost per unit of mass.

Figure 4.5 shows an example of the route of the vehicle under different objectives with 3 periods and OU policy. Figure4.5a is the route obtained with energy minimization. The vehicle serves Customer 1 with 65 kg products in the first period, then makes a tour by visiting customers 3(1230) $\rightarrow$ 4(766) $\rightarrow$ 5(478) $\rightarrow$ 2(280) in the second period (the number in the parentheses is the mass flow on the corresponding arc), and no delivery is done in the third period. With distance and inventory minimization (Figure 4.5b), the vehicle serves Customer 3 with 232 kg and 464 kg products in the first and last period respectively, and in the second period, it visits 4(896) $\rightarrow$ 2(608) $\rightarrow$ 5(328) $\rightarrow$ 1(130). In the route given by energy minimization, only one national route is used (the arc $(3,4)$) and the maximal mass flow is distributed on arc $(0,3)$ which corresponds to the minimum cost per unit of mass in this instance. In the route given by inventory and distance minimization, however, only one highway is used (the arc $(1,0)$) and one delivery is planned in addition in period 3, which induces a lot of energy use because the vehicle weight (4000 kg) is important in relation to the payload (464 kg).



(a) Energy Minimization. (b) Inventory and Distance Minimization.

Figure 4.5: Vehicle routes under different objectives

The relation between the distance and the energy is influenced by the road type or more precisely, the vehicle dynamics such as the number of stops and vehicle speeds. As shown in Figure 4.6, in a world with more highways (A2N1), we can achieve 40% energy reduction with at most 30% augmentation of distance, whereas in A1N2 configuration, the augmentation of distance can be as high as 60% to have an energy reduction of 35%. This confirms the fact that a free-flow configuration is better for energy use.

Last but not least, inventory replenishment strategy can also impact the energy reduction potential of an inventory routing system (Figure 4.7). Under ML policy, inventory change to save energy is higher than under OU policy, since ML policy is more flexible than OU policy.

4.5 Discussion and Conclusion

Energy consumption is an important aspect from both economical and ecological point of view. It becomes more and more important with the sustainable requirements of the supply chains. Traditional supply chain management strategies concentrates on economical costs only, which can result in energy inefficiency and in return, high economical costs on energy.

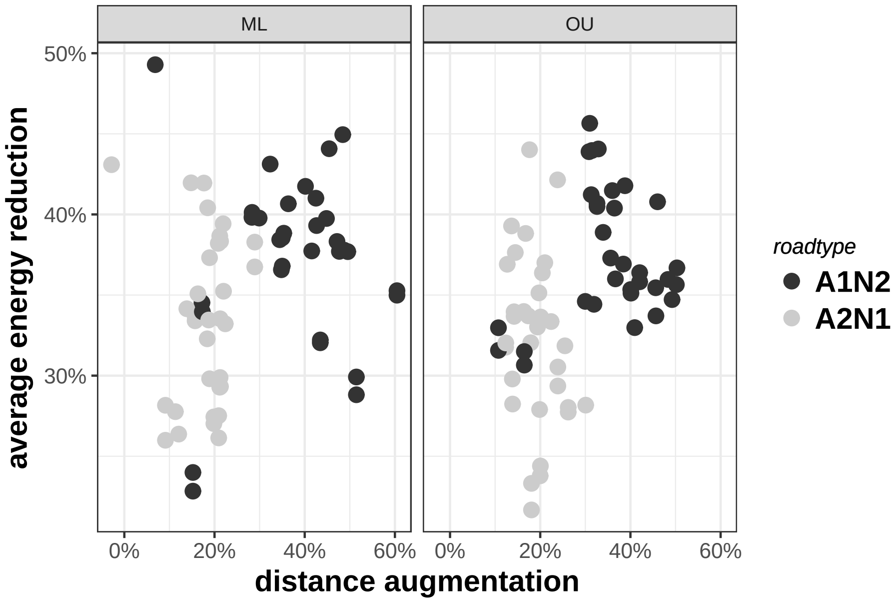


Figure 4.6: Distance and energy cost under different road types

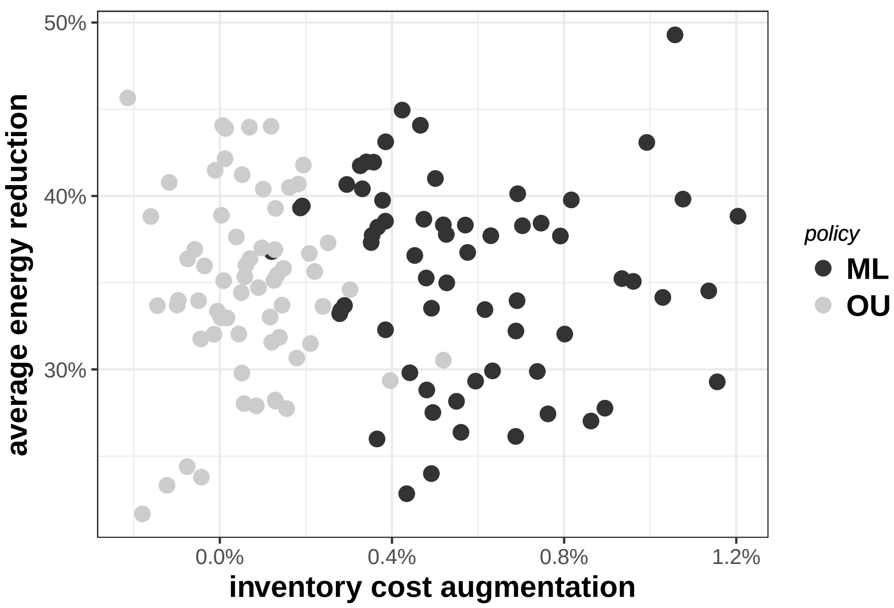


Figure 4.7: Inventory and energy cost under different inventory policies

In this chapter, the energy awareness in the distribution process is discussed. First, the EEAVSP considers the feeding system inside an assembly line. The energy is analysed for the tow train between a pair of workstations. By considering the component mass and integrating the stop decision into the graph representation, a powerful network flow model is proposed. By comparing the energy consumption with different objectives of the same model (i.e. distance minimization, energy minimization, minimization of number of stops etc.), we show that the energy-efficiency is not in conflict with the economical benefits but complementary. The system can be more energy efficient by controlling the stops and the mass of components delivered to each workstations, without additional economical costs.

In the second part, we addressed the combination of inventory management, vehicle routing and energy minimization. An energy estimation methods takes consideration on vehicle dynamics (speed) and road characteristics (stop rate per kilometre and road slope). A mass-flow based formulation of the IRP is proposed with explicit energy consumption. This estimation gives us an energy cost function that is linear to the total mass. In this formulation, the mass is taken as a decision variable and the energy cost function is considered as an objective. Our first experimentation shows that there is a great potential in improving the energy efficiency in the inventory routing systems.

Various factors can have an impact on the energy consumption of such an inventory routing system. From the transportation aspect, vehicle speed and the number of stops are important. On a road network with congestion, energy consumption can be reduced more than on a network without congestion. From the inventory management aspect, inventory strategy influences the energy consumption. Under the condition that no customer is in stock-out, the ML policy provides much more flexibility for higher energy savings than the OU policy.

Both problem instances are solved with the default B&C algorithm of commercial solvers. Since both problems are NP-hard, they become difficult to solve to optimality when the size of the instances increases to be realistic. An interesting thing to note is that the energy reduction tends to be higher with more workstations or customers. It can be expected that in real situations, the energy efficiency could be further improved. More efficient solution methods need to be developed to see the energy reduction potential in real cases. Heuristics and decompositions methods are two main research area to solve these problems.

Further works include also the modelling of traffic networks, so that different traffic conditions as well as vehicle speed levels could be considered in the decision process. The inventory routing model needs to be improved to better control the time and quantity of each delivery. In fact, the classical IRP is very aggregated in terms of inventory monitoring and delivery scheduling since all the inventory levels are summarized in periods and in each period, one vehicle route is made. In real situations, however, finer time granularity is important for both routing and inventory monitoring. For the routing, traffic conditions can change in a day and as a result, timing of delivery could be crucial to the energy consumption. Moreover, with a finer time granularity, inventory variation can happen at the same time as the vehicle makes the delivery. For example, in a situation where a large amount of consumption arrives at the end of a day, it may be more energy-efficient to plan a delivery at the end of the day by using a route with no congestion. So the delivery quantity could change according to the deliver times and consequently influence the energy consumption.

In the next chapter, we present an industrial IRP, where the combined decision of scheduling and routing becomes a real challenge.

Part III

Large Scale Problems and Decomposition Methods

In this part, a real-life Inventory Routing Problem (IRP) is first studied. This problem includes many industrial constraints and decisions, notably, the scheduling of the drivers, the timing of visits, the continuous monitoring of the customer inventory levels, and so on. To solve this real-life IRP, a decomposition method with several components is proposed, which embeds randomized greedy heuristics, a fixed-sequence mixed integer fractional program and a column generation-based approach. Theoretical insights are provided on relevant sub-problems.

Then, a simplified version of the problem is presented with consideration on energy consumption. This version keeps the decisions on the timing of visits and inventory monitoring in continuous time while integrating the energy consideration. Since travel duration can influence both the energy consumption of the vehicle on a road segment and the inventory variation of a customer, multi-graph representation of the road network is proposed. In the multi-graph, each arc is characterised by the costs of energy as well as the travel duration. The solution method for this Multi-Graph Inventory Routing Problem with Energy Consideration (MG-IRP-EC) is a decomposition method based on Lagrangian relaxation. Finally, some preliminary results are presented.

Chapter 5

A Real Life Inventory Routing Problem

In this chapter, a real-life inventory routing problem is presented. This problem is studied under the general background of the ROADEF/EURO Challenge (abbreviated “Challenge” in the following). After a presentation of the Challenge problem, a solution method with several components is proposed. This chapter is finished by a discussion on the solution method and its pros and cons when applied to real-life inventory problems. For a comprehensive description of the Challenge please refer to the web page [6].

5.1 Background

The Challenge was organized in 2016 by the French Operational Research and Decision Support Society (Société française de Recherche Opérationnelle et Aide à la Décision) (ROADEF) jointly with the European Operational Research Society (EURO). It is dedicated to the Inventory Routing Problem (IRP) with the industrial partner Air Liquide, a world leader in gases, technologies and services for industry and health. In this Challenge, example contexts from the healthcare business of Air Liquide are studied, where large (bulk) volumes of liquid oxygen are delivered to over 7500 hospitals worldwide.

For bulk delivery customers, Air Liquide installs on-site storage vessels (“tanks”), which are regularly refilled by trucks driven by drivers (“vehicles”) transporting liquid gases from the air separation unit to the customers. Through its remote telemetry system, Air Liquide monitors customer tank levels and consumption rates and forecasts the future consumption of each customer over the coming hours and days. Through this process, Air Liquide takes the responsibility to guarantee that sufficient inventory of product will be maintained on-site at the customers to meet their demands , which corresponds to the Vendor Managed Inventory (VMI) policy. While VMI customers are in the majority, a smaller but still significant set of customers (called “call-in” customers) are supplied on an “on demand” policy, as they directly place orders with Air Liquide when they need product.

In this VMI context, Air Liquide must efficiently organize the safe and reliable round trips of its truck fleet. On a daily basis, the dispatchers review the forecast of each customer’s consumption and then adapt the schedule of the transportation accordingly. Their goal is to reduce the cost per delivered unit (in €/kg) over a long term, while avoiding product shortage, satisfying the orders of the call-in customers and respecting safety and regulatory constraints (e.g., limits on continuous driving time of each driver).

The Challenge IRP has many new features compared to classic IRPs:

- the average size of the problem is bigger than those generally studied in literature;
- the objective is rational: the goal is to minimize a ratio (total cost per unit delivered in €/kg);
- the decisions to make are a combination of scheduling, assignment and routing, which makes the problem very complicated; in addition, each short term decision impacts the future cost: indeed, a decision that can reduce the cost today may not be good over the long term;
- the solution must satisfy specific business-related constraints that are generally not taken into consideration in the literature:
  - Air Liquide considers an effectively continuous time (accurate to the minute) for the working time of the drivers and the timing of each operations, while for inventory control, it is a discrete time (in hours) that is used; in the literature, time horizon is often divided into periods and each period corresponds to a much longer time unit such as a day or a week;
  - special constraints exist for the working time of the drivers and the opening time of the customers; for example, there are special working time windows and driving time limit for each driver and opening time windows for each customer; these parameters will be detailed in Sections 5.2.1.2 and 5.2.1.3 and the related constraints wille be explained in Section 5.2.4.2.
  - a driver cannot drive every type of trucks and a customer cannot accept certain type of trucks, neither; so a solution should satisfy the compatibility between each driver (or customer) and each trailer;
  - trucks are not refilled automatically in the starting/ending point of each trip; there are some special location called “sources” where the truck can be refilled;
  - there are some special customers such as call-in customers that have to be paid attention to;
  - loadings and deliveries can alternate in the whole working time of a driver and apart from the driver’s working time limits, there are no special restrictions to the length of a trip.

The following section presents the Air Liquide IRP in details and gives a sketch of the solution method.

5.2 Problem Presentation

The problem is to plan bulk distribution in order to minimize total distribution cost over the long term. The goal is to build delivery shifts to match the demand requirements subject to given resources and technical constraints in order to minimize the logistic ratio (defined later in this section). For a detailed formal description, please refer to the subject of the 2016 Challenge [5].

In this section, the terminology is first introduced with the notations used. After that, decision variables are defined and business-related constraints are explained. The objective function is then discussed. Finally, the principal components of the solution method are presented.

5.2.1 Parameters

The terminology used in the following of this chapter is defined below.

Trailer: the truck with a tank that is used for delivering the product; a trailer is by convention denoted by tl and the set of trailers is denoted by $\mathcal{TL}$.

Driver: the worker who drives a trailer; a driver is denoted by d and the set of drivers is referred to as $\mathcal{DR}$.

Vehicle: the combination of a driver and a trailer;

Location or Site: a stopping point of a vehicle, generally denoted by i or j;

Base: the starting and the ending location of a vehicle; there is only one such location called "the Base" and it is usually denoted by 0;

Source: a refilling location of the trailers; a source is referred to as so and the set of sources is denoted by $\mathcal{SO}$;

Customer: a location where the product is consumed gradually; a customer is usually denoted by i or j and the set of customers is denoted by $\mathcal{Z}$;

Delivery: a stop of a vehicle at a customer site: the delivery hose of the customer is hooked up to the vessel of the tank of the trailer and the product is delivered;

Loading: a stop of a vehicle at a source site, where the vehicle is filled with products;

Shift: a chronological list of activities made by a driver during his working period (precisely defined in Section 5.2.2); it starts and ends at Base; a shift is often denoted by s and the set of shifts is referred to as $\mathcal{SH}$;

Layover: a fixed idle time interval in a shift when one or more "layover customers" (to be explained later) are delivered; it allows the driver to travel for an additional duration, covering a larger area;

Time horizon: the total scheduling horizon, denoted by $\mathcal{H}$. The total number of hours in the horizon is denoted by H.

5.2.1.1 Trailer Specific Parameters

Each trailer $tl \in \mathcal{TL}$ is characterized by:

Distance cost C_{tl}^{dis} the cost per distance unit for the trailer (in €/km);

Capacity B_{tl} the capacity in mass (kg) of the trailer, or the maximum quantity of product that can be loaded in the trailer and delivered to customers;

Initial quantity J_{tl}^{0} the initial mass (kg) of usable product in the trailer at the start of the time horizon 0.

5.2.1.2 Driver Specific Parameters

Each driver $d \in \mathcal{DR}$ has the following characteristics:

Time windows $\mathcal{TW}_d$ the set of availability intervals of the driver; each time window $tw \in \mathcal{TW}_d$ is defined by the starting time a_d^{tw} and the ending time b_d^{tw} (in minutes), each included in the horizon $\mathcal{H}$;

Time cost C_d^{time} the cost per minute of working of the driver (in €/minute). Working time means that this cost doesn't apply during the layover pause when the driver is on a rest;

Maximum driving duration MDD_d the maximum driving duration for the driver, before ending the shift at the Base or doing a layover (in minutes);

Minimum inter-shift duration MIS_d the minimum time interval for the driver d between two consecutive shifts (in minutes);

Layover cost C_d^{lo} the cost of a layover pause of the driver (in €);

Layover duration LOD_d a fixed duration of a layover pause of the driver;

Compatible trailer $\mathcal{TL}_d$ the set of trailers that can be driven by this driver.

5.2.1.3 Location Specific Parameters

A location may be Base, a source $so \in \mathcal{SO}$, or a customer $i \in \mathcal{Z}$. All locations (Base, sources and customers) have the following common characteristics:

Distance $D_{i,j}$ distance between two locations i and j (in km);

Time $T_{i,j}$ travel time from location i to location j (in minutes).

Each source or customer $i \in \mathcal{SO} \bigcup \mathcal{Z}$ have the following characteristics in common:

Setup time ST_i the fixed loading or delivery time for a location (delivery for a customer or loading for a source) (in minutes);

Allowed trailers $\mathcal{TL}_i$ the set of trailers that is compatible with the customer or the source;

Remember that a customer is called "*VMI*" if his tank level is monitored by Air Liquide and "*call-in*" otherwise. A customer is called "*layover*" if a direct delivery is not possible from the depot. The set of call-in (VMI and layover) customers is denoted by $\mathcal{Z}_{ci}$ ($\mathcal{Z}_{vmi}$ and $\mathcal{Z}_{lo}$, respectively). Each customer $i \in \mathcal{Z}$ has a set of time windows defined as follows:

Time windows $\mathcal{TW}_i$ the set of opening time intervals of the customer i, each time window $tw \in \mathcal{TW}_i$ is defined by the starting time a_i^{tw} and the ending time b_i^{tw} (in minutes) within the time horizon $\mathcal{H}$. Deliveries should be completed within these intervals.

A call-in customer only place orders. Their inventory levels are not monitored. Therefore, each call-in customer corresponds to a set of orders (denoted by $\mathcal{OD}_i$) that have to be fulfilled, and each $od \in \mathcal{OD}_i$ has the following characteristics:

Quantity R_i^{od} the ordered quantity;

Flexibility f_i^{od} a value between 0 and 100 indicating the minimum percentage of the ordered quantity to deliver to the customer so that the order can be considered as satisfied;

Earliest time a_i^{od} the earliest starting time (in minutes) for all the delivery operations related to the order *od*;

Latest time b_i^{od} the latest ending time (in minutes) for all the delivery operations related to the order *od*.

The tank of all the VMI customers is monitored by Air Liquide. The tank level is forecasted according to historical data and a delivery is made whenever it is necessary and cost efficient. The following characteristics correspond to a VMI customer $i \in \mathcal{Z}_{vmi}$:

Safety level $\underline{I}_i$ the minimum level of the customer's tank to avoid product shortage;

Capacity $\overline{I}_i$ the maximum amount of product that can be stored in the tank of the customer;

Initial tank quantity I_i^0 the amount of product available in the customer's tank at the beginning of the horizon;

Minimum operation quantity R_i^{min} the minimum amount of product that should be delivered once the customer is visited;

Forecasts R_i^t the amount of product that is used by the customer in each hour $t \in \mathcal{H}$.

5.2.1.4 Units of measure for quantity and time

For the amount of product in the deliveries and the tank level monitoring, the mass in kilograms (kg) is used. Hours and minutes are both used as discrete time breakdowns of the horizon. Note that the inventory forecasts are given hourly, while driver working time, travel time between locations and customer opening time are counted in minutes.

5.2.2 Decision Variables

The problem is to decide a set of shifts, each with a sequence of operations, so that all the customers are never out of product while a certain number of constraints defined later in Section 5.2.4 are satisfied.

A solution of the problem is a set of shifts (denoted by $\mathcal{SH}$). The following variables should be decided for each shift $s \in \mathcal{SH}$:

- the driver performing this shift;
- the trailer used in this shift;
- the starting time of the shift (within the time horizon in minutes)
- a list of operations (loadings, deliveries) performed during the shift denoted by $\mathcal{N}_s$

For each operation $o \in \mathcal{N}_s$ of the shift s, the following variables should be decided:

- the arrival time (within the time horizon in minutes);
- the location (a source or a customer) where the operation takes place;
- the amount to be delivered or loaded during the operation. It is negative for loading operations at sources, positive for delivery operations at customers.

5.2.3 Notations

The notations for each parameter presented above are summarized in Table 5.1 in alphabetic order.

Table 5.1: Notations

symbol	meaning
a	the starting time of a time window or the earliest time of an order in minute
b	the ending time of a time window or the latest time of an order in minute
B_{tl}	the capacity of trailer tl in kg
C_d^{time}	the cost per minute of working of a driver d in €/minute
C_d^{lo}	the cost of a layover pause of a driver d in €
C_{tl}^{dis}	the cost per distance unit for a trailer tl in €/km
$\mathcal{DR}$	the set of drivers
d	a driver
$D_{i,j}$	the distance between locations i and j in km
f_i^{od}	the flexibility of an order od of a call-in customer i
$\mathcal{H}$	the planning horizon in hours
H	the number of hours in the horizon
$\overline{I}$	the capacity in kg
$\underline{I}$	the safety level in kg
I_i^0	the initial level of a customer i in kg
i or j	a location or site

LOD_d	the layover duration of a driver d in minutes
J^0_{tl}	the initial inventory in the trailer tl in kg
MDD_d	the maximum driving duration of a driver d in minutes
MIS_d	the minimum time interval between two consecutive shifts of a driver d in minutes
o	an operation
od	an order
R^{od}_i	the quantity of an order od of a call-in customer i int kg
R^{min}_i	the minimum operation quantity of a VMI customer i in kg
$\mathcal{N}_s$	the sequence of operations in a shift s
R^t_i	the forecast of customer i for the hour t
s	a shift
$\mathcal{SH}$	the set of shifts
so	a source site
$\mathcal{SO}$	the set of sources
ST	the setup time in minutes
t	a time step in hour
$T_{i,j}$	travel time between locations i and j in minutes
$\mathcal{TL}$	the set of trailers
tl	a trailer
$\mathcal{TW}_d$	the set of time windows of driver d
$\mathcal{TW}_i$	the set of time windows of customer i
tw	a time window
$\mathcal{Z}$	the set of customer sites
$\mathcal{Z}_{ci}$	the set of call-in customers
$\mathcal{Z}_{lo}$	the set of layover customers
$\mathcal{Z}_{vmi}$	the set of VMI customers

5.2.4 Business-Related Constraints

There are several constraints related to specific business considerations in the problem that should be satisfied.

5.2.4.1 Constraints Related to Assignments

A shift is defined for a pair of driver/trailer. The driver and trailer in a shift must be compatible, that is to say, the driver should be able to drive the trailer in this shift. Moreover, only one trailer and one driver can be assigned to each shift. For each operation in each shift, the location must also accept the trailer assigned to the shift.

5.2.4.2 Constraints Related to Time

For each driver $d \in \mathcal{DR}$, his working time is defined in his time windows by a set of shifts. Each shift performed by d must be totally included in one of his time windows. The duration between two consecutive shifts assigned to d must be at least the minimum inter-shift duration.

In each shift performed by each driver d, the *cumulated driving time* is the total travel time starting from the Base to make a tour around the locations of operations and returning back to the Base, without idle or setup time. This cumulated driving time cannot be longer than the maximum allowed driving duration of the driver. The shift can be extended further by making a layover pause without returning to depot. After such a pause, the cumulated driving time of the driver is reset to 0 and another cycle lasting at most the maximum allowed driving duration can be made. Note that the layover pause is only allowed once in a shift with at least a layover customer.

In each shift, the vehicle starts from the Base, makes deliveries or loadings at some sites (customers or sources) and returns to the Base at the end. The arrival at a site is the sum of the departure time from the previous site plus travelling time between the two sites, plus eventually the layover duration or idle time. An arbitrary long idle time (where the driver rests at the door of a customer or a source) can precede any operation in a shift, as long as all of the constraints are respected. In particular, if the idle time is longer than the layover duration of the driver assigned to this shift, then a layover is triggered.

Loading and delivery take a constant time for each source or customer. It is assumed that the entire quantity delivered (or loaded) in an operation is immediately available in the customer tank (or the trailer tank) as soon as the truck arrives at the location of operation, without considering the fixed set-up time needed to complete the delivery (or the loading).

Delivery operations are performed during opening hours of customers. For all operations, the interval between arrival and departure of the trailer must be fully included in one of the opening time windows of the location of operation.

Each order of each call-in customer should be satisfied by one or more operations. Those operations should begin after the earliest time and before the latest time of the order. Each operation on a call-in customer should be related to an order, meaning no operations are possible if there is no related order.

5.2.4.3 Constraints Related to Quantities

The quantity in each trailer cannot be negative or exceed its capacity. Additionally, trailer quantity is preserved from one shift to another. In other words, initial quantity of a trailer for each shift is equal to the quantity left in the trailer at the end of the previous shift.

For each VMI customer, the tank level at each time step t is equal to the level at the previous time step $t-1$, minus the forecast consumption at t, plus all the deliveries performed at t. The tank level must be maintained greater than or equal to the safety level and must not exceed its tank capacity. Moreover, the delivered quantity must be at least the minimum operation quantity.

To satisfy an order of a call-in customer, the total quantity delivered between the earliest and latest time of the order should be at least the minimum percentage of the ordered quantity. And the delivered quantity should not be larger than the ordered quantity.

5.2.5 Objective Function

The goal is to minimize the distribution costs required to meet customer demands for product over a long term. To achieve this goal, the logistic ratio $\mathscr{LR}$ is defined as the total cost of all the shifts divided by the total quantity *delivered* in all shifts:

$$\mathscr{LR} = \frac{\mathscr{T} + \mathscr{D} + \mathscr{L}}{\mathscr{Q}} \tag{5.1}$$

In fact, the cost of a shift represents the distribution costs related to this shift, including:

- the distance cost, applied to the total length of the shift, induced by the usage of trailers (covering fuel consumption and maintenance);
- the time cost applied to the total duration of the shift, induced by the working of drivers (covering the driver salary and charges);
- the layover cost, if the shift contains a layover.

Equation (5.1) defines the logistic ratio mathematically, with $\mathscr{T}$ ($\mathscr{D}$ and $\mathscr{L}$) denoting the total time (distance and layover) costs of all the shifts in a solution (respectively). The total quantity $\mathscr{Q}$ delivered over all shifts is the sum of the quantities in each *delivery* operation of each shift.

Note that the objective value could be influenced by the choice of the length of the time horizon, since the objective is to minimize the total distribution costs over a long period of time covering one or more replenishment cycles for all the customers. If the time horizon is too short, the optimization can be short-sighted because of the end-of-period side effect—customers that do not strictly require delivery within this short horizon are excluded from the planning. This might increase the risk of shortage just beyond the horizon. Considering a longer time horizon could make the relative impact of this side effect negligible, but it complicates the problem and requires longer forecast on the customers, which could not be realistic. As a consequence, a good value for time horizon depends on many factors, particularly on the degree of certainty of the customer forecasts within the horizon.

5.2.6 Principal Components of the Solution Method

The solution method is divided into three steps as shown in Figure 5.1. In the following Section 5.3–Section 5.5, we are going to present the solution method designed to solve this Challenge. The principal components are summarized here.

The first step includes randomized greedy heuristics with a Fixed-Sequence Mixed Integer Linear Fractional Programming (FS-MILFP) and a post-processing procedure. In this step, the greedy algorithms are first used to generate initial solutions (feasible or infeasible). When a feasible solution is found, it is given to

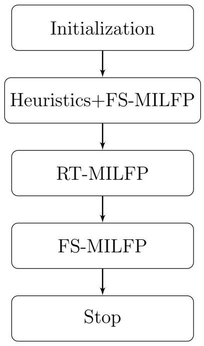


Figure 5.1: Solution method in three steps

the FS-MILFP for improvement. With the sequence of visits fixed by the heuristics, the FS-MILFP tries to optimize the shift start/end times and the arrival time and the quantity of each operation. After the FS-MILFP optimization, a post-processing procedure is applied to remove the visits to a customer (or a source) without delivery (or loading) quantity. The FS-MILFP and the post-processing are applied to several starting solutions obtained through the randomization of the greedy heuristics.

The second step is a column generation algorithm and a Route-Based Mixed Integer Linear Fractional Programming with Time Aggregation (RT-MILFP). A route-based formulation is proposed that aggregate time unit to the hour. This formulation contains one variable for each possible driver route. More precisely, a *route* is defined as the complete sequence of visits of a driver during the entire time horizon. The routes are obtained using a column generation scheme by solving a pricing sub-problem, which aims at selecting the most profitable routes of a driver among a set of exponential routes.

The third step is for re-optimization. Using the FS-MILFP in the first step, the feasibility of the solution obtained in the second step is checked with the correct time precision (in minutes and hours). It is re-optimized if possible. In the following, Section 5.3 presents two types of greedy heuristics. Section 5.4 gives the FS-MILFP model used to determine the optimal timing and quantities with fixed sequence of operations. Section 5.5 describes the RT-MILFP model, the column generation method and a dynamic programming method for the pricing sub-problem.

5.3 Greedy Heuristics

In order to obtain starting solutions for the problem, two greedy heuristics were designed. The first greedy is based on the state of the system. The second considers the urgency of the customers.

5.3.1 State-Based Greedy Heuristic

For each triplet consisting of a driver $d \in \mathcal{DR}$, a trailer $tl \in \mathcal{TL}$ and a customer $i \in \mathcal{Z}$, we compute a time $\theta_{d,tl,i}$ in minutes. This time represents the slack for driver/trailer pair (d, tl) from its current location to arrive at the customer i to avoid a stock-out or fulfil an order (see Figure 5.2 for the definition of $\theta_{d,tl,i}$).

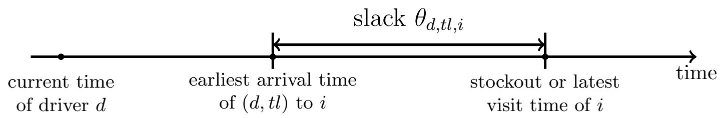


Figure 5.2: the definition of slack $\theta_{d,tl,i}$ for driver/trailer pair (d, tl) and customer i

To explain the computation of $\theta_{d,tl,i}$, we introduce the notion of *state*. A *state* of the system is defined by

- the position of each trailer and each driver;
- the driving duration spent by each driver;
- the quantity left in each trailer and each customer;
- the time of the last visit to each customer.

According to the current state of the system, $\theta_{d,tl,i}$ is computed with respect to time windows of the driver d and the customer i by checking the maximum driving duration of d and the possible need to load at the source.

Given a state of the system, for a pair of compatible driver/trailer (d, tl), there are 5 possible actions to take before visiting a customer:

- starting a new shift (without changing time window of the driver),
- changing to a new driver time window,
- making a layover pause (without changing shift),
- visiting a source
- waiting (for the customer to open).

The enumeration of all the possible combinations of these 5 actions gives us a set of valid arrival times of a driver d and a trailer tl to visit a customer i with the corresponding maximum quantity that can be delivered. The value $\theta_{d,tl,i}$ for the triplet (d, tl, i) is chosen from the valid arrival times according to a criterion randomly chosen among a set of criteria such as earliest arrival time, latest arrival time, most close to the stock-out arrival time etc.

The state-based greedy first construct a list of customers to be served in the horizon. It then randomly chooses a customer i with respect to the latest visit time lvt_i defined as follows. For a VMI customer i, the value of lvt_i is the minimum between the customer stock-out time and the latest time window end time before the stock-out. For a call-in customer i, it is the latest visiting time of the next order. The value of lvt_i is computed according to the state of the customers and their time windows, which are independent of drivers or trailers. It is the latest time to visit the customer i to avoid a stock-out or fulfil an order.

Then, a driver/trailer pair (d, tl) is chosen for this customer by a randomized criterion, according to the time $\theta_{d,tl,i}$ for each compatible driver/trailer/customer triplet (d, tl, i). Since the arrival time is known in advance by the combination of the 5 actions, the operation is added to the solution following the 5 actions with randomized delivery quantity between the minimum delivery quantity and the maximum deliverable quantity. If nothing can be done for visiting this customer, then the customer is considered lost and he is removed from the list of customers to be served.

The state of the system is then updated. The process is iterated until no one would be in stock-out or no order exists by the end of the time horizon. Several solutions can be obtained with this greedy heuristic by randomization. The complete algorithm is summarized in Algorithm 3.

Algorithm 3 State-based greedy heuristic

```
1: Initialize the state of the system
2: Construct a list of customers to be served denoted by L
3: Compute the slack θ_{d,tl,i} for each triplet of driver/trailer/customer (d, tl, i) and the
   actions to take to achieve this slack
4: while L ≠ ∅ do
5:    Randomly choose a customer i from L
6:    Choose a driver/trailer pair (d, tl) according to a randomized criteria in relation to
      θ_{d,tl,i}
7:    if Pair (d, tl) exists then
8:       add operation to i by (d, tl) following the actions computed in Step 3
9:    else
10:      remove i from L
11:   Update the state of the system and recompute the slack
12:   if i will not have stock out in the planning horizon then
13:      remove i from L
```

5.3.2 Urgency-Based Greedy Heuristic

The second greedy heuristic is based on the urgency of the customers (see Algorithm 4). It is inspired by the heuristic proposed by Benoist et al[28]. In this heuristic, the *urgency* of a customer is measured by his next stock-out time. A list of customers to be served is also maintained as in the state-based greedy heuristic, but this time the customers in the list are classified according to the departure time from the Base to be served before the stock-out, which is defined as the "*deadline*". In the case of orders, it is the departure time from the Base to arrive before the end of the time window of an order. First, an attempt is made to visit the customer with the earliest deadline in the list. If the customer can be served before a stock-out or

Algorithm 4 Urgency-based greedy heuristic

```
Initialize a list of customers to be served L
Choose the customer i with the earliest deadline from L
Make an attempt to visit i
for all Driver/trailer pair (d, tl) do
    compute the earliest arrival time and the minimum quantity for (d, tl) to deliver to
    i with consideration on driving duration and trailer inventory level
Choose (d, tl) corresponding to the earliest arrival time
if The deadline of i is NOT violated then
    insert i into a shift of (d, tl) by checking whether a layover is triggered
    compute the next deadline of i
    reinsert i into the list L
else
    abort the insertion, remove i from the list
```

the end of an order, the next deadline is computed and the customer is reinserted in the list. Otherwise, the customer is removed from the list and is not considered any more. It means that the solution can be infeasible at the end of the greedy method. Now, it is explained how to look for possible insertion of a visit in the general case and in some special cases.

In general, a visit is always inserted at the end of the last shift in the tours of the drivers and trailers, or into a new shift created at the end of the sequence of shifts of a driver/trailer pair. In both cases, the earliest possible time is computed to deliver to the customer the minimum quantity demanded. If the deadline is violated, the inclusion is aborted. Otherwise, we check if a visit to the nearest source is needed before the visit to the customer and we update the arrival time. In the case of the creation of a new shift, if a visit to a source is necessary, it is tested whether the visit to the source can be added at the end of the last shift of the trailer. If not, the visit to the source is added to the beginning of the new shift.

If a visit is possible, the quantity to deliver is computed as the minimum value among the remaining quantity in the trailer, the difference between the capacity of the customer and its current quantity, and the quantity needed such that the customer does not need to be refilled again until the end of the time horizon.

If the last customer visited in the shift is the same as the one to insert, instead of creating a new visit, we try to postpone the already scheduled visit. For this, in addition to the time windows of the drivers and customers, it should also be made sure that no layover is created. Concerning the layover, it is always placed before the visit that will exceed the maximum driving duration of the driver.

Among all the possible inclusions in existing shifts and newly-created shifts, we select the one which allows the earliest visit time. This will allow more visits inside the same shift and creating the smaller idle time for trailers and drivers.

5.4 Fixed-Sequence MILFP

When the heuristics find a feasible solution, a Fixed-Sequence Mixed Integer Linear Fractional Programming (FS-MILFP) is launched to further optimize the operation arrival time and quantity. With the sequence of operations in each shift fixed by the solution of the heuristics, the role of the FS-MILFP is to fix the timing of the shifts and the operations as well as the delivery quantity for each operation in each shift.

More precisely, the problem is to decide, for each compatible vehicle, the starting and ending time of each shift, the quantity of product that should be delivered to each customer in this shift (or the quantity to get from the source if included) and the time (in minutes) of visit to each site. To perform the stock balance by hour, we also have to know at which hour each delivery operation is performed. For layovers, we need to decide whether a layover pause is needed before each operation and the time (in minutes) of the layover pause. The following parts of this section describe the FS-MILFP proposed for solving this problem.

5.4.1 Parameters and Notations

The parameters fixed by the heuristics is described below and the notations are summarized in Table 5.2. First, each shift is assigned to a compatible pair of driver/trailer (a vehicle) by the heuristics. Let d_s be the driver performing a given shift $s \in \mathcal{SH}$. Let $\mathcal{SH}^d$ denote the sequence of all shifts performed by driver $d \in \mathcal{DR}$ and $\mathcal{SH}^{tl}$ be the sequence of all shifts using trailer $tl \in \mathcal{TL}$.

Second, a feasible sequence of shifts $\mathcal{SH}^{tw}$ is considered fixed by the greedy heuristics for each working time window $tw \in \mathcal{TW}_d$ of a driver $d \in \mathcal{DR}$. Let $s \in \mathcal{SH}^{tw}$ denote the s-th shift in the sequence $\mathcal{SH}^{tw}$.

Third, the order of the operations in a shift is fixed. Recall that $\mathcal{N}_s$ denotes the sequence of operations in a shift s. In the following, it also denotes the number of operations inside this shift. In addition, 0 and $\mathcal{N}_s + 1$ denote the Base at the beginning and the end of the shift, respectively. Let i_k^s denote the index of the k-th site visited in shift s with $k \in \mathcal{N}_s$ and $i_0^s = i_{\mathcal{N}_s+1}^s = 0$.

Forth, the greedy algorithms assign each VMI customer $i \in \mathcal{I}_{vmi}$ to one of his time windows denoted by $tw \in \mathcal{TW}_i$.

Furthermore, a shift can contain layover customers according to the solution of the heuristics, and the set of all the shifts with at least one layover customer is denoted $\mathcal{SH}^l$.

Table 5.2: Parameters fixed by the greedy heuristics

symbol	meaning
d_s	the driver performing the shift s
i^s_k	the index of the ok-th site visited in shift s
$\mathcal{SH}^d$	the sequence of shifts performed by driver d
$\mathcal{SH}^l$	the sequence of shifts with at least one layover customer
$\mathcal{SH}^{tl}$	the sequence of shifts performed by trailer tl
$\mathcal{SH}^{tw}$	the sequence of shifts inside timewindow tw

5.4.2 Variables

The variables are listed in Table 5.3:

Table 5.3: Variables in the FS-MILFP model

symbol	meaning
art^s_k	the arrival time (in minutes) of the k-th operation in each shift s. It is a set of continuous variables varying in the corresponding time window of the driver d in this shift. In particular, the arrival time at the Base (art^s_0) and the duplicate of the Base ($art^s_{\mathcal{N}_s+1}$) represent the starting and ending time of the shift s.
$z^{s,t}_k$	a set of binary variables, equal to 1 if the k-th operation in the sequence $\mathcal{N}_s$ in shift s is visited in the hour t in the corresponding time window of the driver and 0 otherwise.
ρ^s_k	a set of continuous variables representing the remainder time (in minutes) of the k-th operation in shift s.
$q^{s,t}_k$	a set of continuous variables representing the quantity delivered in operation k to customer $i^s_k \in \mathcal{Z}$ (or picked-up from the source $i^s_k \in \mathcal{SO}$) in shift s in hour $t \in \mathcal{H}$.
I^t_i	a set of continuous variables representing the inventory level of each VMI customer $i \in \mathcal{Z}_{vmi}$ at each hour $t \in \mathcal{H}$.
$J^{s,k}_{tl}$	a set of continuous variables denoting the tank level in the trailer tl after operation k in shift s.
μ^s_k	a set of binary variables that indicate whether the layover pause is planed before the k-th operation of shift s ($\mu^s_k = 1$) or not ($\mu^s_k = 0$).
λ^s_k	a set of continuous variables in $[0, 1[$ representing ratio of the time elapsed (in minutes) of the layover pause compared to the total travel duration between the $(k-1)$-th and the k-th operation in shift s.

5.4.3 Constraints

The constraints are stated by categories. For a complete model please refer to Annexe A.

Timing of shifts

$$a_d^{tw} \leq art_0^s \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw} \qquad (5.2)$$

$$art_{\mathcal{N}_s+1}^s \leq b_d^{tw} \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw} \qquad (5.3)$$

Constraints (5.2)–(5.3) ensure that the starting and ending time of the shift are included in the time window containing this shift.

Precedence of operations

$$ST_{i_k^s} + T_{i_k^s, i_{k+1}^s} + \mu_{k+1}^s \ LOD_d \leq art_{k+1}^s - art_k^s \quad \forall d \in \mathcal{DR}, \forall s \in \mathcal{SH}^d, \forall k \in \mathcal{N}_s \quad (5.4)$$

$$MIS_d \leq art_0^{s+1} - art_{\mathcal{N}_s+1}^s \qquad \forall d \in \mathcal{DR}, \forall s \in \mathcal{SH}^d \quad (5.5)$$

Constraints (5.4)–(5.5) ensure that the operations in the shift are timed according to the sequence. Constraints (5.4) make sure that if there is no layover, then the duration between two consecutive operations should be at least the service time of the previous operation and the travelling duration between the two sites. In particular, if there is a layover before the $(k+1)$-th operation in the shift, the layover duration should be added to the duration between operations k and $k+1$. Constraints (5.5) says that the duration between two consecutive shifts of the same driver should be no less than the minimum inter-shift duration.

Timing of operations in minutes

$$60t \ z_k^{s,t} \leq art_k^s \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s, \forall t \in \mathcal{H} \qquad (5.6)$$

$$art_k^s \leq 60 \ H + (60 \ (t+1) - 60 \ H) z_k^{s,t} - 1 \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s, \forall t \in \mathcal{H} \qquad (5.7)$$

$$art_k^s = 60 \sum_{t \in \mathcal{H}} t z_k^{s,t} + \rho_k^s \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \qquad (5.8)$$

$$0 \leq \rho_k^s \leq 59 \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \qquad (5.9)$$

$$\sum_{t \in \mathcal{H}} z_k^{s,t} = 1, \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \qquad (5.10)$$

Constraints (5.6)–(5.9) ensure the coherence between the binary variables $z_k^{s,t}$ and the continuous variables art_i^s. If art_i^s is inside an hour t, then $z_k^{s,t}$ is set to 1. Constraints (5.8) and (5.9) are redundant with Constraints (5.6) and (5.7), but they avoid using Big M constraints. Constraints (5.10) ensure that each operation in a shift is scheduled at one specific hour.

VMI customer time windows

$$a_{i_k^s}^{tw_{i_k^s}} \leq art_k^s \leq b_{i_k^s}^{tw_{i_k^s}} - ST_{i_k^s} \qquad \forall s \in \mathcal{SH}, \forall k \in \{k \in \mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\} \qquad (5.11)$$

Constraints (5.11) make sure that all the visits to a VMI customer are in his time window fixed by the greedy heuristic.

Inventory in trailers

$$J_{tl}^{s,k+1} - J_{tl}^{s,k} = -\sum_{t \in \mathcal{H}} q_k^{s,t} \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \qquad (5.12)$$

$$J_{tl}^{s+1,0} = J_{tl}^{s,\mathcal{N}_s+1} \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl} \qquad (5.13)$$

$$-B_{tl} \leq q_k^{s,t} \leq 0 \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{SO}\} \qquad (5.14)$$

$$0 \leq J_{tl}^{s,k} \leq B_{tl} \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \qquad (5.15)$$

Constraints (5.12) are for the coherence of trailer quantity between consecutive operations inside each shift. Constraints (5.13) are for the coherence of trailer quantity among consecutive shifts with the same trailer. Constraints (5.14) ensure that the quantity obtained from the sources never exceeds the vehicle capacity. Constraints (5.15) state that the quantity in each vehicle is positive and never exceeds the capacity of the vehicle.

Inventory in customers

$$I_i^{t+1} - I_i^t = \sum_{s \in \mathcal{SH}} \sum_{k \in \{\mathcal{N}_s | i_k^s = i\}} q_k^{s,t} - R_i^t \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \setminus H \qquad (5.16)$$

$$R_{i_k^s} z_k^{s,t} \leq q_k^{s,t} \leq (\overline{I}_{i_k^s} - \underline{I}_{i_k^s}) z_k^{s,t} \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\}, \forall t \in \mathcal{H} \qquad (5.17)$$

$$0 \leq q_k^{s,t} \leq \overline{I}_{i_k^s} \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\}, \forall t \in \mathcal{H} \qquad (5.18)$$

$$\underline{I}_i \leq I_i^t \leq \overline{I}_i \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \qquad (5.19)$$

$$R_i^{od} f_i^{od} \leq \sum_{s \in \mathcal{SH}} \sum_{k \in \{\mathcal{N}_s | i_k^s = i\}} \sum_{t \in \left[a_i^{od}, b_i^{od}\right]} q_k^{s,t} \leq R_i^{od} \qquad \forall i \in \mathcal{Z}_{ci}, \forall od \in \mathcal{OD}_i \qquad (5.20)$$

Constraints (5.16) are for the time coherence of inventory levels of each VMI customer. Constraints (5.17) and (5.18) make sure that the quantity delivered is inside the allowed limits of VMI customers. Constraints (5.19) ensure that the quantity in a VMI customer never exceeds his capacity and is always above his safety level. Constraints (5.20) are for the fulfilment of call-in orders. They impose the delivered quantity be at lease the minimum percentage needed to satisfy an order and not exceed the maximum deliverable amount.

Layover pauses

$$\sum_{k \in \mathcal{N}_s} \mu_k^s = 1 \qquad \forall s \in \mathcal{SH}^l \tag{5.21}$$

$$\sum_{k \in \mathcal{N}_s} \mu_k^s = 0 \qquad \forall s \in \mathcal{SH} \backslash \mathcal{SH}^l \tag{5.22}$$

$$\lambda_k^s \leq \mu_k^s \qquad \forall k \in \mathcal{N}_s, \forall s \in \mathcal{SH} \tag{5.23}$$

$$\sum_{j=k}^{\mathcal{N}_s} \sum_{l=j}^{\mathcal{N}_s} T_{i^s_{j-2}, i^s_{j-1}} \ \mu_l^s + T_{i^s_{k-1}, i^s_k} \ \lambda_k^s \leq MDD_{d_s} \qquad \forall k \in \{2, \ldots, \mathcal{N}_s + 1\}, \forall s \in \mathcal{SH} \tag{5.24}$$

$$\sum_{j=1}^{\mathcal{N}_s} T_{i^s_{j-1}, i^s_j} - (\sum_{j=k}^{\mathcal{N}_s} \sum_{l=j}^{\mathcal{N}_s} T_{i^s_{j-2}, i^s_{j-1}} \ \mu_l^s + T_{i^s_{k-1}, i^s_k} \ \lambda_i^s) \leq MDD_{d_s} \quad \forall k \in \{2, \ldots, \mathcal{N}_s + 1\}, \forall s \in \mathcal{SH} \tag{5.25}$$

Constraints (5.21)–(5.25) are for layover pauses. Constraints (5.21) make sure that a shift with layover customer has one and only one layover pause. Constraints (5.22) make sure that there is no layover pause inside a shift without a layover customer. Constraints (5.23) ensure that if there is no layover before operation k of a shift s then the time λ_k^s is set to 0. Constraints (5.24)–(5.25) make sure that the driving duration inside each shift never exceeds the maximum driving duration of the driver performing this shift. If the layover pause exists, Constraints (5.24) are for the total driving time before the layover pause and constraints (5.25) are for the total driving time after the layover pause; otherwise, constraints (5.24) becomes $0 \leq MDD_{d_s}$ and constraints (5.25) says that the total driving time is bounded by the maximum driving duration, which is always true given a feasible initial sequence of operations.

Variable domains

$$z_k^{s,t} \in \{0,1\} \qquad \forall s \in \mathcal{SH}, \forall k \in \mathcal{N}_s, \forall t \in \mathcal{H} \tag{5.26}$$

$$\mu_k^s \in \{0,1\} \qquad \forall s \in \mathcal{SH}, \forall k \in \{1, \ldots, \mathcal{N}_s, \mathcal{N}_s + 1\} \tag{5.27}$$

$$\rho_k^s \in [0, 59[\qquad \forall s \in \mathcal{SH}, \forall k \in \{0, 1, \ldots, \mathcal{N}_s\} \tag{5.28}$$
$$\lambda_k^s \in [0, 1[\qquad \forall s \in \mathcal{SH}, \forall k \in \{1, \ldots, \mathcal{N}_s, \mathcal{N}_s + 1\} \tag{5.29}$$
$$I_i^t \in \left[\underline{I}_i, \overline{I}_i\right] \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \tag{5.30}$$
$$J_{tl}^{s,k} \in [0, B_{tl}] \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \tag{5.31}$$
$$q_k^{s,t} \in \left[R_{i_k^s}, \overline{I}_{i_k^s}\right] \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\} \forall t \in \mathcal{H} \tag{5.32}$$
$$q_k^{s,t} \geq 0 \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{ci}\} \forall t \in \mathcal{H} \tag{5.33}$$
$$art_k^s \in \left[a_d^{tw}, b_d^{tw}\right] \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw}, \forall k \in \{0\} \bigcup \mathcal{N}_s \tag{5.34}$$

Constraints (5.26)–(5.34) define the variable domains.

5.4.4 Linearisation of Fractional Objective

The objective is to optimize the logistic ratio $\mathscr{LR} = \frac{\mathscr{T}+\mathscr{D}+\mathscr{L}}{\mathscr{Q}}$. It can be linearised according to [55]. In general, let S the searching space defined on $\mathbb{R}^n$, let $N(x)$ and $D(x)$ be two real-valued continuous functions with respect to variable $x \in S$ and D a function positive. The minimization of a fractional function $\eta(x) = \frac{N(x)}{D(x)}$ is closely related to the minimization of $f(x) = N(x) - \eta D(x)$ with a coefficient $\eta \in \mathbb{R}$. More precisely, we have the following theorem extracted from [55].

Theorem 1. *The optimum of the fractional function* $\eta^* = \frac{N(x^*)}{D(x^*)} = \min\{\frac{N(x)}{D(x)} | x \in S\}$ *if and only if the optimum of the linearised function* $F(\eta^*) = F(\eta^*, x^*) = \min\{N(x) - \eta^* D(x) | x \in S\} = 0$

In [55], an iterative method is proposed for solving the non-linear fractional programming problem with concave $N(x)$ and convex $D(x)$. It starts with any feasible x with $\eta \geq 0$. In each iteration, it is the linearised problem $\min\{N(x) - \eta^* D(x) | x \in S\} = 0$ that is solved and the coefficient η is updated. Finally, the value of η would converge to the minimization of $\eta(x)$. The algorithm for fractional programming applied to our problem is presented in Algorithm 5.

In our case, the objective is to minimize $\mathscr{LR} = \frac{\mathscr{T}+\mathscr{D}+\mathscr{L}}{\mathscr{Q}}$, which is equivalent to the minimization of $(\mathscr{T} + \mathscr{D} + \mathscr{L}) - \eta\mathscr{Q}$ iteratively.

In the FS-MILFP, the objective can be more precisely written as:

$$\min(\mathscr{T} + \mathscr{D} + \mathscr{L}) - \eta\mathscr{Q} \tag{5.35}$$

with

$$\mathscr{Q} = \sum_{s \in \mathcal{SH}} \sum_{k \in \{\mathcal{N}_s | i_o^s \in \mathcal{Z}\}} \sum_{t \in \mathcal{H}} q_k^{s,t} \tag{5.36}$$
$$\mathscr{T} = \sum_{d \in \mathcal{DR}} \sum_{tw \in \mathcal{TW}_d} \sum_{s \in \mathcal{SH}^{tw}} C_d^{time}(art_{\mathcal{N}_s+1}^s - art_0^s - \sum_{k \in \mathcal{N}_s} \mu_k^s LOD_d) \tag{5.37}$$
$$\mathscr{D} = \sum_{tl \in \mathcal{TL}} \sum_{s \in \mathcal{SH}^{tl}} \sum_{k \in \{0\} \bigcup \mathcal{N}_s} C_{tl}^{dis} D_{k,k+1} \tag{5.38}$$

$$\mathscr{L} = \sum_{d\in\mathcal{DR}} \sum_{tw\in\mathcal{TW}_d} \sum_{s\in\mathcal{SH}^{tw}} \sum_{k\in\mathcal{N}_s} \mu_k^s C_d^{lo} \tag{5.39}$$

With the variables defined above, the objective function of the FS-MILFP can be developped as:

$$\begin{aligned}
\max & \sum_{d\in\mathcal{DR}} \sum_{tw\in\mathcal{TW}_d} \sum_{s\in\mathcal{SH}^{tw}} C_d^{time}(art^s_{\mathcal{N}_s+1} - art^s_0 - \sum_{k\in\mathcal{N}_s} \mu_k^s LOD_d) \\
& + \sum_{tl\in\mathcal{TL}} \sum_{s\in\mathcal{SH}^{tl}} \sum_{k\in\{0\}\bigcup\mathcal{N}_s} C_{tl}^{dis} D_{k,k+1} \\
& + \sum_{d\in\mathcal{DR}} \sum_{tw\in\mathcal{TW}_d} \sum_{s\in\mathcal{SH}^{tw}} \sum_{k\in\mathcal{N}_s} C_d^{lo} \mu_k^s \\
& -\eta \sum_{s\in\mathcal{SH}} \sum_{k\in\{\mathcal{N}_s | i_o^s \in \mathcal{Z}\}} \sum_{t\in\mathcal{H}} q
\end{aligned} \tag{5.40}$$

Algorithm 5 Algorithm for fractional programming

1: Initialize η with a feasible logistic ratio $\mathscr{LR}_0$
2: Solve the FS-MILFP model with the linearised objective function 5.40
3: Recompute the coefficient $\eta' = \frac{\mathscr{T}+\mathscr{D}+\mathscr{L}}{\mathscr{Q}}$ with the current solution
4: **while** $\eta' - \eta < 0$ **do**
5: $\quad \eta \leftarrow \eta'$
6: $\quad$ Repeat step 2 and 3

The FS-MILFP can be solved with a Mixed Integer Linear Programming (MILP) solver. The convergence of this algorithm is ensured but can be slow. In reality, after solving an iteration of the FS-MILFP model, some of the customers in the sequence decided by the greedy heuristics are visited without any delivery quantity. A post-processing is then applied to remove these operations from the solution. The solution after post-processing is given back to the FS-MILFP solver to re-optimize the arrival time and quantity of each operation.

This model can be easily adapted to the case of an infeasible initial solution by adding variables for missed order or lack of quantity for stock-outs and by changing the objective to minimize the total number of missed order and the total lack of quantity of stock-outs. However, the optimization is not effective for infeasible initial solutions.

5.5 Route-Based MILFP with Time Aggregation

To better optimize the sequence of visits of the initial solution given by the greedy heuristics, a formulation based on routes or combinations of shifts are proposed. In this formulation, all the time units are aggregated to hours.

In this Route-Based Mixed Integer Linear Fractional Programming with Time Aggregation (RT-MILFP), a *route* is defined as a sequence of shifts with partially decided operations inside the whole planning horizon. One route is to be selected for each driver. The selection of routes is thus the principal decision variable in this problem. Since the total number of routes is exponential, each route selection variable is considered as a column in the master problem and is generated by a pricing sub-problem.

The master problem contains the decision of the route selection and the inventory management of the customers and the trailers. The sub-problem takes the dual values of the route selection variables computed in the master problem. Its role is to generate beneficial routes (in terms of reduced cost) while satisfying the constraints concerning working time of the drivers, as well as respecting time windows of the drivers and the customers. Since a route is constructed by shifts, which are themselves composed by operations, the sub-problem is decomposed into two steps: the first is to find the promising shifts and the second is to combine these shifts to form a complete route. This section presents the RT-MILFP under the column generation scheme and the two-step solution of the pricing sub-problem.

5.5.1 Route-Based Formulation with Time Aggregation

Let $\mathcal{RO}_d$ denote the set of all the possible routes for driver $d \in \mathcal{DR}$, with parameters

- $u_{tl}^{r,t} \in \{0,1\}$ that equals 1 if route $r \in \mathcal{RO}_d$ uses trailer $tl \in \mathcal{TL}$ at hour $t \in \mathcal{H}$,
- $z_i^{r,t} \in \{0,1\}$ that equals 1 if route $r \in \mathcal{RO}_d$ visits site $i \in \mathcal{Z} \bigcup \mathcal{SO}$ at hour $t \in \mathcal{H}$.

The RT-MILFP contains the following variables:

- For each route $r \in \mathcal{RO}_d$ and each driver $d \in \mathcal{DR}$, p_d^r is a binary variable which is equal to 1 if route r is selected for driver d.
- For each customer (or source site) $i \in \mathcal{Z}$ (or $i \in \mathcal{SO}$), for each trailer $tl \in \mathcal{TL}$, and for each hour $t \in \mathcal{H}$, $q_{i,tl}^t$ is the quantity delivered (or loaded) at hour t at the site i by the trailer tl.

We make the additional assumption that all the time windows of the drivers and customers, which are initially given in minutes, are now rounded to hours. For the objective 5.41, the same linearisation method is used. For reason of simplicity, the coefficient η is set to a fixed value. Again, the constraints are presented according to their functionality. For parameter notations, please refer to Table 5.1 on Page 96.

$$\min \quad \sum_{d \in \mathcal{DR}} \sum_{r \in \mathcal{RO}_d} C_{r,d}^{route} p_d^r - \eta \sum_{i \in \mathcal{Z}} \sum_{tl \in \mathcal{TL}} \sum_{t \in \mathcal{H}} q_{i,tl}^t \tag{5.41}$$

Assignment of driver/trailer to route

$$-\sum_{r \in \mathcal{RO}_d} p_d^r \geq -1 \qquad \forall d \in \mathcal{DR} \tag{5.42}$$

$$-\sum_{d\in\mathcal{DR}}\sum_{r\in\mathcal{RO}_d} p_d^r u_{tl}^{r,t} \geq -1 \qquad \forall t \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.43)$$

Constraints (5.42) ensure that at most one hourly-timed route is assigned to each driver. Constraints (5.43) make sure that at most one trailer is used in each hour of a route of a driver.

Quantity limits

$$\sum_{d\in\mathcal{DR}}\sum_{r\in\mathcal{RO}_d} p_d^r u_{tl}^{r,t} z_i^{r,t} B_{tl} \geq -q_{i,tl}^t \qquad \forall i \in \mathcal{SO}, \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.44)$$

$$\sum_{d\in\mathcal{DR}}\sum_{r\in\mathcal{RO}_d} p_d^r u_{tl}^{r,t} z_i^{r,t} B_{tl} \geq q_{i,tl}^t \qquad \forall i \in \mathcal{Z}, \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.45)$$

$$-\sum_{d\in\mathcal{DR}}\sum_{r\in\mathcal{RO}_d} p_d^r u_{tl}^{r,t} z_i^{r,t} R_i^{min} \geq -q_{i,tl}^t \qquad \forall i \in \mathcal{Z}, \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.46)$$

Constraints (5.44) ensure that the quantity loaded at a source is always with a negative sign and never exceeds the trailer capacity. Constraints (5.45) limit the quantity delivered to a customer to the trailer capacity. Constraints (5.46) make sure that the quantity delivered to a customer is at least the minimum delivery quantity required by the customer.

Inventory of trailers

$$J_{tl}^t = J_{tl}^{t-1} - \sum_{i\in\mathcal{Z}\bigcup\mathcal{SO}} q_{i,tl}^t \qquad \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.47)$$

$$0 \leq J_{tl}^t \leq B_{tl}, \qquad \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.48)$$

Constraints (5.47) are for the inventory balance of each trailer from one time step to another. Constraints (5.48) limit the tank level in the trailer to be positive and smaller than or equal to the trailer capacity.

Satisfaction of customer inventory levels or demands

$$I_i^t = I_i^{t-1} + \sum_{tl\in\mathcal{TL}_i} q_{i,tl}^t - R_i^t, \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \qquad (5.49)$$

$$\sum_{t=a_i^{od}}^{b_i^{od}}\sum_{tl\in\mathcal{TL}} q_{i,tl}^t \geq f_i^{od} R_i^{od} \qquad \forall i \in \mathcal{Z}_{ci}, \forall od \in \mathcal{OD}_i \qquad (5.50)$$

$$\sum_{t=a_i^{od}}^{b_i^{od}}\sum_{tl\in\mathcal{TL}} q_{i,tl}^t \leq R_i^{od} \qquad \forall i \in \mathcal{Z}_{ci}, \forall od \in \mathcal{OD}_i \qquad (5.51)$$

Constraints (5.49) are for the inventory balance of each VMI customer from one time step to another. Constraints (5.50) ensure that the quantity delivered inside the time limits of an order is enough to satisfy the corresponding order. Constraints (5.51) set limits on the quantity delivered to call-in customers.

Variable domains

$$\overline{I}_i \le I_i^t \le \overline{I}_i \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \qquad (5.52)$$

$$0 \le J_{tl}^t \le B_{tl} \qquad \forall tl \in \mathcal{TL}, \forall t \in \mathcal{H} \qquad (5.53)$$

$$B_{tl} \le q_{i,tl}^t \le 0 \qquad \forall i \in \mathcal{SO}, \forall tl \in \mathcal{TL}_i, \forall t \in \mathcal{H} \qquad (5.54)$$

$$0 \le q_{i,tl}^t \le \overline{I}_i \qquad \forall i \in \mathcal{Z}, \forall tl \in \mathcal{TL}_i, \forall t \in \mathcal{H} \qquad (5.55)$$

$$p_d^r \in \{0,1\} \qquad \forall d \in \mathcal{DR}, \forall r \in \mathcal{RO}_d \qquad (5.56)$$

Constraints (5.52)— (5.56) define the domain of each variable.

Since the number of routes for each driver could be exponential, a column generation approach is used to tackle this problem. In the following, the Master Problem (MP) is defined as the linear relaxation of the RT-MILFP formulation with the complete route set $\mathcal{RO}$. The Restricted Master Problem (RMP) is the restriction of the MP to a subset of $\mathcal{RO}_1 \subset \mathcal{RO}$.

The general idea of the column generation is based on the principals presented in Section 1.3.3.1 of Chapter 1. The column generation begins with a RMP defined on a set $\mathcal{RO}_1$ containing only a few columns. Then, we check whether there exists a beneficial column with respect to the reduced cost by solving a pricing sub-problem. If such a column exists, it is added to the RMP and the RMP is solved again. Otherwise, the optimal solution for the RMP is also the optimal solution for the MP and the RMP is finally solved for an integer solution.

5.5.2 Pricing Sub-problem

Let us first see what happens when we dualize constraints related to route variables in the RT-MILFP. By dualizing constraints (5.42)–(5.46) with relevance to the route variables p, the following dual variables are introduced:

- $\omega_d \in \mathbb{R}_+$ for each driver $d \in \mathcal{DR}$ are associated to constraints (5.42). In the MP, these constraints ensure that at most one shift pattern can be selected for each driver. In the dual problem, the value of each ω_d introduces an additional cost for each driver d.

- $\psi_{tl}^t \in \mathbb{R}_+$ for each trailer $tl \in \mathcal{TL}$ and for each hour $t \in \mathcal{H}$ denote the dual variables associated with constraints (5.43), which state that only one trailer can be used in one route for each hour in the MP. In the dual, the values of ψ_{tl}^t can be interpreted as the cost for the usage of each trailer tl during the hour t.

- $\chi^t_{i,tl} \in \mathbb{R}_+$ for each customer or source site $i \in \mathcal{Z} \bigcup \mathcal{SO}$, for each trailer $tl \in \mathcal{TL}$ and for each hour $t \in \mathcal{H}$ denote the dual variables connected with constraints (5.44) and (5.45). These constraints set the maximum amount of product of a trailer tl at a given hour t after loading at a source $i \in \mathcal{SO}$ or delivering to a customer $i \in \mathcal{Z}$. In the dual problem, the values of $\chi^t_{i,tl}$ can be understood as a benefit of the visit by trailer tl to the site i in hour t.

- $\phi^t_{i,tl} \in \mathbb{R}_+$ for each customer $i \in \mathcal{Z}$, for each trailer $tl \in \mathcal{TL}$ and each hour $t \in \mathcal{H}$ are related to constraints (5.46). In the master, the minimum delivery quantity by trailer tl to customer i at a given hour t is set by these constraints. In the dual, the values of $\phi^t_{i,tl}$ can also be seen as a benefit brought by the visit of tl to i in t.

The constraints corresponding to variables p in the dual problem can then be written as:

$$-\omega_d - \sum_{tl \in \mathcal{TL}} \sum_{t \in \mathcal{H}} u^{r,t}_{tl} \psi^t_{tl} \quad + \sum_{tl \in \mathcal{TL}} \sum_{t \in \mathcal{H}} \sum_{i \in \mathcal{Z} \bigcup \mathcal{SO}} u^{r,t}_{tl} z^{r,t}_i (B_{tl} \chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}) - C^{route}_{r,d} \leq 0$$

$$\forall r \in \mathcal{RO}_d, \forall d \in \mathcal{DR}$$

with $C^{route}_{r,d}$ the total cost of distance, time and layover of all the shifts in the route r for the driver d. Supposing $\phi^t_{i,tl} = 0 \ \forall i \in \mathcal{SO}$, the combined term $B_{tl}\chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}$ represents the total benefit to visit each $i \in \mathcal{Z} \bigcup \mathcal{SO}$ by each trailer $tl \in \mathcal{TL}$ in each hour $t \in \mathcal{H}$.

Let

$$f(r) = - \sum_{tl \in \mathcal{TL}} \sum_{t \in \mathcal{H}} u^{r,t}_{tl} \psi^t_{tl} + \sum_{tl \in \mathcal{TL}} \sum_{t \in \mathcal{H}} \sum_{i \in \mathcal{Z} \bigcup \mathcal{SO}} u^{r,t}_{tl} z^{r,t}_i (B_{tl} \chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}) - C^{route}_{r,d}. \tag{5.57}$$

Finding a new column to be added to the restricted master problem amounts to finding a route r for a driver d such that $f(r)$ is strictly larger than ω_d. Or in other words, for a route r of a driver d to enter the master, it has to be profitable enough to compensate the total cost ω_d. Mathematically, this is equivalent to finding r^* such that

$$\begin{aligned} r^* &\in \arg\max_{r \in \mathcal{RO}} f(r) \\ f(r^*) &> \omega_d. \end{aligned}$$

5.5.2.1 Problem Statement

For each driver d, the sub-problem amounts to the search of a route with maximum profit. As already defined, a route corresponds to a planning of shifts of visits to customers with a trailer by a driver in each hour. The parameters $u^{r,t}_{tl}$ and $u^{r,t}_{tl}$ defined in the MP now become decision variables for each route r:

- The assignment of trailers in each hour is decided by binary variables $u^{r,t}_{tl}$.

For each $tl \in \mathcal{TL},\ t \in \mathcal{H}$, u_{tl}^{t} equals 1 if trailer tl is used in the route in hour t.

- The location of the driver in each hour is decided by binary variables $z_i^{r,t}$. For each $i \in \mathcal{Z} \bigcup \mathcal{SO},\ t \in \mathcal{H}$. z_i^t equals 1 if site i is visited in the route at hour t.

The following constraints need to be satisfied in a route r:

- the trailers in the route can all be driven by the driver d;
- the driving duration of d are respected;
- the driver d only works in his time windows;
- each visit to a customer i happens in one of the customer's time windows;
- each customer i is visited by a trailer tl that he can allow.

In general, these constraints are those related to assignments and time as presented in Sections 5.2.4.1–5.2.4.2. The objective is to maximize the profit of the route defined by $f(r)$ in Equation (5.57). Note that the objective is non-linear with respect to variables $u_{tl}^{r,t}$ and $z_i^{r,t}$.

5.5.2.2 Decomposition of the Sub-Problem

The sub-problem of finding the most profitable route can be broken down by driver time windows into shifts. The shifts are further divided into operations to sites. The most profitable shifts are composed by most profitable operations in each hour. This property of the sub-problem leads to a dynamic programming solution method in two steps.

Sub-problem 1 (SP1) finds for each time window of driver d, the set of all the shifts s with strictly positive objective $f(r_s) > 0$ with r_s the route containing only one shift s. This problem is modelled on a time-space graph of sites for each compatible trailer of the driver d. The time-space graph is explained in details in Section 5.5.2.3. The labelling algorithm for solving this problem is given in Section 5.5.2.4.

Note that the shifts with negative costs are not interesting, since if we had found a final valid route r with a shift s with negative cost, then the route without this negative cost shift would be more advantageous than r.

Sub-problem 2 (SP2) chooses shifts from each time window of driver d and connects them to form a complete route r so that the total profit is strictly greater than $\omega_d + C_{r,d}^{route}$. The algorithm is similar to the one for solving SP1 except that it is based on the graph of compatible shifts. It is explained in general in Section 5.5.2.5.

5.5.2.3 Time-Space Graph of Sites with Compatible Trailers

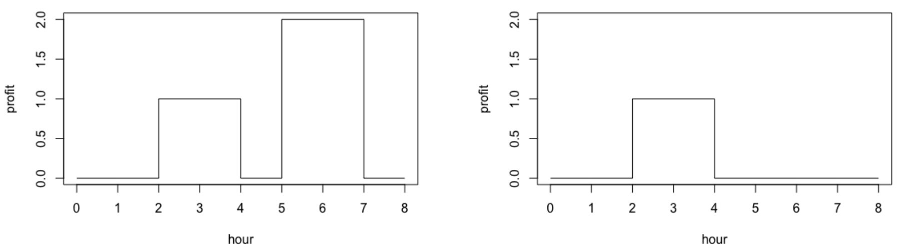
Since the profit $B_{tl}\chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}$ of visiting a site i by a trailer tl is piecewise constant with relation to the hour t over the whole horizon, for each site i with each trailer tl, the hours in the horizon can be further aggregated into a set of time slots with the same visiting profit. In addition, the time windows of the customers can be easily integrated into the time slots by setting the profit to zero for hours not belonging to a time window of the corresponding customer. Algorithm 6 is for constructing the time slots for each type of sites.

Algorithm 6 Time slots construction

for all Trailer $tl \in \mathcal{TL}$ **do**
  for all Source $i \in \mathcal{SO}$ **do**
    Initialize the set of slots $TW_s \leftarrow \emptyset$, $t \leftarrow 0$, $a \leftarrow t$
    while $t < H - 1$ **do**
      if $B_{tl}\chi^{t+1}_{i,tl} - R^{min}_i \phi^{t+1}_{i,tl} = B_{tl}\chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}$ **then**
        $t \leftarrow t + 1$
      else
        $b \leftarrow t$
        Add $[a, b]$ to TW_s, $a \leftarrow t + 1$
  for all VMI customer $i \in \mathcal{Z}_{vmi}$ **do**
    Initialize the set of slots $TW_s \leftarrow \emptyset$
    for all Time window $tw \in \mathcal{TW}_i$ of the customer **do**
      $t \leftarrow a^{tw}_i$ (the starting time of the first time window *(in hours)*), $a \leftarrow t$
      while $t < b^{tw}_i - 1$ **do**
        if $B_{tl}\chi^{t+1}_{i,tl} - R^{min}_i \phi^{t+1}_{i,tl} = B_{tl}\chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}$ **then**
          $t \leftarrow t + 1$
        else
          $b \leftarrow t$
          Add $[a, b]$ to TW_s, $a \leftarrow t + 1$
  for all call-in customer $i \in \mathcal{Z}_{ci}$ **do**
    Initialize the set of slots $TW_s \leftarrow \emptyset$
    for all Order $od \in \mathcal{OD}_i$ **do**
      $t \leftarrow a^{od}_i$ (the earliest time of the first order *(in hours)*), $a \leftarrow t$
      while $t < b^{od}_i - 1$ **do**
        if $B_{tl}\chi^{t+1}_{i,tl} - R^{min}_i \phi^{t+1}_{i,tl} = B_{tl}\chi^t_{i,tl} - R^{min}_i \phi^t_{i,tl}$ **then**
          $t \leftarrow t + 1$
        else
          $b \leftarrow t$
          Add $[a, b]$ to TW_s, $a \leftarrow t + 1$

Then for each trailer tl, the time-space graph of compatible sites denoted by $G = (V, A)$ can be defined as follows. Each vertex $v \in V$ corresponds to a site i compatible with trailer tl in a time slot $[a_v, b_v]$. The profit of visiting the vertex is the profit of the corresponding time slot. There exists an arc $(v_1, v_2) \in A$ from vertex v_1 to vertex v_2 if and only if the duration from the end of the time slot of v_1 to the start of the time slot of v_2 is smaller than the layover duration of the driver considered. In this way, it is ensured that no layover will be generated. The layover can be integrated by adding some special constraints when generating the graph but it is too specific to be treated in this thesis. The labelling algorithm proposed later for SP1 would not generate shifts with layovers.

Example Figure 5.3 shows the profits given by the dual values of a problem with two customers and one compatible trailer on a horizon defined by $[0, 8[$ in hours (including 0 not including 8). Suppose that the worker works during the whole horizon; Customer 1 in Figure 5.3a is open in hours $[3, 7[$; Customer 2 is open during the entire horizon. The set of time slots for this customer associated with this trailer is thus the intervals $[0, 3[\,, [3, 4[\,, [4, 5[\,, [5, 7[$ and $[7, 8[$ with profit $0, 1, 0, 2, 0$ respectively. Note that the profit value zero of the time slots $[0, 3[$ and $[7, 8[$ is because of the opening time window of the customer, while the value zero of the time slot $[4, 5[$ is due to the dual values. It is obvious that the time slots of Customer 2 is $[0, 2[\,, [2, 4[$ and $[4, 8[$ with profit $0, 1, 0$, respectively. The complete graph is constructed for the trailer by setting a node for each non-zero slot for each customer. Figure 5.4 shows the time-space graph of the trailer in this example. Nodes v_1^1 and v_1^2 corresponds to the two slots with non-zero profits of Customer 1, and node v_2^1 is for the only non-zero profit time slot of Customer 2. The Base node 0 is duplicated by node $0'$ for the return to the Base at the end of the horizon. The intervals above the nodes are the corresponding time slots. There is an arc when the vehicle can visit two nodes consecutively according to the working time of the driver. Note that between v_2^1 and v_1^1, the arc is bi-directed because it is possible to visit any of them first and then visit the other one as long as the driving time and the service time allow.



(a) Customer 1 (b) Customer 2

Figure 5.3: Example profits of visit

Problem SP1 is to construct all the shifts with positive profits which start from the depot and return back to depot inside each time window $tw \in \mathcal{TW}_d$ for each driver $d \in \mathcal{DR}$. The objective is to find the path with the maximum profit in the time-space graph $G = (V, A)$ defined above. Each time a vertex is visited, the arrival time should fall inside its time slot. The total driving duration in a path should not exceed the maximum driving duration of the driver.

This problem can be seen as an Orienteering Problem with Time Windows (OPTW), under the constraints that (i) each node in the time-space graph is included at most once in the generated shift; (ii) a driving duration of the driver in

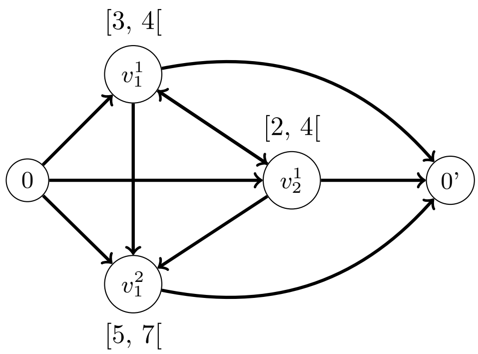


Figure 5.4: Example of a time-space graph of a trailer with two customers

a shift is limited; (iii) each node has a time slot. The OPTW is NP-hard as proved by [92]. One can also discard the first set of constraints and allow several visits to the same customer in the same time slot. In that case, this problem can be seen as a Shortest Path Problem with Resource Constraints (SPPRC) with the aim to maximize the total profit while respecting the driving time limit. In the following, several visits to the same customer in the same time slot are allowed since one time slot for a customer could last very long in practice. A labelling algorithm is proposed to solve this problem.

5.5.2.4 Labelling Algorithm to Find Shifts

In SP1, a shift corresponds to a path from the origin Base node to the destination Base node on the time-space graph inside one of the driver d's time windows. A partial path in the time-space graph is defined by $P_k = (0, v_1, v_2, \ldots, v_k)$ with v_k the k-th vertex visited in this path. Given a partial path P_k, its label is defined by $L_k = (art_k, \Pi_k, f(P_k))$ with art_k the arriving hour to the vertex v_k, Π_k the accumulated driving time in hours and $f(P_k)$ the total profit of this partial path defined by (5.57). The total time, distance and layover cost is also kept in the label in $f(P_k)$. We now present the labelling algorithm to find the path with the maximum profit in the time-space graph of the sites with the compatible trailers.

With a fixed driver d in a time window tw, the algorithm starts with the path (0) corresponding to the Base at the first hour a_d^{tw} in tw. For each compatible trailer tl of the driver d, it tries to construct a profitable shift. For each site i, it searches for the next opening slot and tests whether the path can be extended to visit this site in this slot. In the end, there will be a set of profitable shifts for each compatible trailer of the driver.

The labelling algorithm relies on the manipulation of two sets: the set of unprocessed paths denoted by $\mathcal{U}$, which are to be extended in future iterations; and the set of processed paths denoted by $\mathcal{P}$, which are Pareto-optimal or can be extended to Pareto-optimal paths. The labelling algorithm is given in Algorithm 7, which is adapted from the one presented in [90].

Each iteration takes a path P_k from the set of unprocessed path $\mathcal{U}$ and try to

Algorithm 7 Labelling algorithm with a driver d in time window tw

$\mathcal{U} \leftarrow \{(0)\}$, $\mathcal{P} \leftarrow \emptyset$
while $\mathcal{U} \neq \emptyset$ **do**
  Select one path $P_k \in \mathcal{U}$ labelled by $L_k = (art_k, \Pi_k, f(P_k))$ *(*path selection*)*
  Remove P_k from $\mathcal{U}$
  for all adjacent vertex v of v_k **do**
    Extend path P_k to (P_k, v)*(*label extension*)*
    if (P_k, v) is feasible **then** *(feasibility check)*
      Add (P_k, v) to the set $\mathcal{U}$
    else
      Reject the label (P_k, v)
    Add P_k to the processed set $\mathcal{P}$
    if $P \in \mathcal{U} \bigcup \mathcal{P}$ is dominated **then** *(application of dominance rule)*
      Remove P from the corresponding set

extend it with another visit to vertex v. If the extended path (P_k, v) is feasible, it is added to $\mathcal{U}$. Otherwise it is rejected. Then P_k is added to the set of processed path $\mathcal{P}$, Finally, we check if there exist paths that can be dominated by other paths. If that is the case, such a path will be deleted from the sets of paths. The process continues until the set $\mathcal{U}$ is empty.

Label extension Given a partial path $P_k = (0, v_1, v_2, \ldots, v_k)$ labelled by $L_k = (art_k, \Pi_k, f(P_k))$ and another vertex v, there are two cases for label extension:

If $v_k = v$, then the trailer stays at the same site in the same slot at the hour art_{k+1}, P_k stays the same but the label is updated to $(art_{k+1}, \Pi_{k+1}, f(P_k) - \sum_{t=art_k}^{art_{k+1}} \psi_{tl}^t - 60C_d^{time}(art_{k+1} - art_k + 1))$ (this case is interesting only if ψ_{tl}^{t+1} is strictly negative and it will generate one same path with two different labels);

Otherwise, let i and j denote the sites of vertex v_k and v respectively. If site j can accept the trailer tl and $\Pi_k + T_{i,j} < MDD_d$, then $P_{k+1} = (0, v_1, v_2, \ldots, v_k, v_{k+1})$ with $v_{k+1} = v$ and $L_{k+1} = (art_{k+1}, \Pi_k + T_{i,j}, f(P_{k+1}))$, where

$$\begin{aligned} f(P_{k+1}) &= f(P_k) - \sum_{t=art_k}^{art_{k+1}} \psi_{tl}^t + B_{tl}\chi_{i,tl}^{art_{k+1}} - R_i^{min}\phi_{i,tl}^{art_{k+1}} \\ &- 60C_d^{time}(art_{k+1} - art_{k+1}) - C_{tl}^{dis}D_{i,j} \end{aligned}$$

with $art_{k+1} \geq art_k + ST_i + T_{i,j}$.

Feasibility check The aim of the feasibility check is to make sure that all the extended labels added to $\mathcal{U}$ are feasible in terms of driving duration or working time of the driver. Given an extended label $L_k = (art_k, \Pi_k, f(P_k))$ corresponding to a partial path P_k with i the site for v_k, we check that (i) the maximum driving duration is not exceeded with return to the Base ($\Pi_k + T_{i,0} < MDD_d$) and (ii) the return time to the Base is before the end of the time window tw of the driver d ($art_k + ST + T_{i,0} <= b_d^{tw}$). For each label of a complete path, it is also checked that the total driving duration Π does not exceed the maximum driving duration MDD_d of the driver d.

Dominance rule The number of candidate paths grows exponentially with a factor of $Z + |\mathcal{SO}|$ (the number of sites) as the algorithm continues in the time-space graph. The dominance rule helps to remove useless partial paths to speed up the search. Given two partial paths P_{k_1} with label $art_{k_1}, \Pi_{k_1}, f(P_{k_1}))$ and P_{k_2} labelled by $(art_{k_2}, \Pi_{k_2}, f(P_{k_2}))$, P_{k_1} is said to dominate P_{k_2} if and only if one of the following conditions is satisfied:

- P_{k_1} belongs to the extension of path P_{k_2} and $art_{k_1} < art_{k_2}$ (Path P_{k_1} contains more visits than path P_{k_2} but the arrival time in P_{k_1} is earlier).
- $P_{k_1} = P_{k_2}$ and $f(P_{k_1}) > f(P_{k_2})$ (The two paths are the same, but the first one is more profitable).

This labelling algorithm generates a certain number of profitable shifts for each time window of the driver. In SP2, these shifts are combined to construct complete routes.

5.5.2.5 Construction of Routes with Shifts

Problem SP2 constructs routes using profitable shifts in the compatible shift graph. In this graph, a node is related to a shift and the transition from one shift to another is possible if and only if:

- the shifts do not overlap in time;
- the sites are compatible with both trailers in the two shifts;
- the minimum inter-shift duration of the driver is respected.

Since the timing of each shift is fixed in SP1 and once a shift decided, we cannot visit other shifts that begin before this shift, the shift graph is a Directed Acyclic Graph (DAG).

Each shift node brings a profit given by the solution of SP1. The goal is to find the path with maximal profit in the shift graph. If the total maximal profit can compensate the combined cost of the driver and the route $\omega_d + C^{route}_{r,d}$, this route is added to the master; otherwise, the column generation finishes.

The SP2 can be solved in polynomial time. It starts with the origin node of a shift in the first time window of the driver. Then at each iteration, it searches among the shifts that starts after the previous shift and appends the shift with the maximum of profit to the route. The algorithm runs in $O(|\mathcal{TW}_d|S)$ time, with $|\mathcal{TW}_d|$ the number of time windows of the driver d and S the number of most profitable shifts generated in SP2 in each time window.

5.6 Experimentation and Discussion

The initial tests are performed with the instances B of the Challenge (see the webpage [6]). Table 5.4 summarize the results obtained for the final phase of the challenge. Column "LR" is the results after the first step of the complete method (the heuristics with the FS-MILFP). A symbol "–" is marked if no feasible solution is found. For reference, the best results obtained by the winner of the challenge are reported in column "Best". In the following of this section, the Challenge instances sets B and X are analyzed. Small instances have been generated to study the effect of each component of the solution algorithm. Different components are compared to give insights on the property of the complete method.

Table 5.4: Results for the Final Instances Set B

Instance	LR	Best
2.12	0.019173	0.010266
2.13	0.054794	0.030768
2.14	0.072882	0.037582
2.15	0.061999	0.026608
2.16	0.024876	0.012420
2.17	–	0.031538
2.18	–	0.033018
2.19	0.079422	0.012992
2.20	0.024292	0.013311
2.21	0.025571	0.013033
2.22	–	0.012411
2.23	–	0.012866
2.24	0.017486	0.010234
2.25	0.037039	0.012410
2.26	0.055342	0.012866

5.6.1 Instance Analysis

Let us first look at the size of the instances (Table 5.5). The number of customers Z varies from 12 to 324. The number of call-in customers and layover customers is given by $|\mathcal{Z}_{ci}|$ and $|\mathcal{Z}_{lo}|$, respectively. The horizon H is between 10 to 35 days (240 to 840 hours). The number of drivers $|\mathcal{DR}|$ is from 4 to 13 and the number of trailers $|\mathcal{TL}|$ is from 3 to 15. There can be at most 2 sources. According to the number of customers, the instances can be categorized into 5 types (called "map" in the following part). Each map corresponds to a set of sites with identical (or nearly identical for instances X) distance and time matrices. Note that there are inconsistencies in the data of the initial instances with regard to layover customers. There exist customers that are declared layover but are directly reachable from the Base and others that are not declared layover but not directly reachable from the Base. It is also observed that there is often a layover customer near the second

Table 5.5: Characteristics of the Instances Set B and X

Map	Z	$\lvert\mathcal{Z}_{ci}\rvert$	$\lvert\mathcal{Z}_{lo}\rvert$	$\lvert\mathcal{SO}\rvert$	$\lvert\mathcal{DR}\rvert$	$\lvert\mathcal{TL}\rvert$	max H	Instances
25	32	9	0	2	5	6	840	V2.24, V2.25, V2.26
14	53	0	14	1	5	5	840	V2.13, V2.14, V2.19
17	134	3	16	1	4	3	840	V2.15, V2.17, V2.18, X3
20	184	1	5	1	7	4	840	V2.16, V2.20, V2.21, X2
12	324	23	12	1	13	15	504	V2.12, V2.22, V2.23, X1, X4, X5

type of customers, so that a feasible solution is always possible, but this creates additional difficulty when looking for a feasible solution. We are not sure about how Air Liquide declares the layover customers, so in the generated data, all such customers were removed.

We have generated additional smaller instances in the following way. First, for each type of map, instances are generated for a horizon from 1 day to 10 days with a step of 1 day, and from 10 days to 35 days with a step of 5 days, keeping the number of customers the same as in the initial instances (from 32 to 324 according to the corresponding map). Then, to have more variety in the number of customers, instances with different horizons are reduced to a set of instances with the number of customers varying from 5 to 50 with a step of 5. The customers are randomly chosen from the initial instances.

In total, 73 instances with various horizon lengths were generated. These instances are denoted "instances H" in the next sections. 372 instances with various horizon lengths and various numbers of customers were also generated, denoted by "instances C" in the following. Different components are tested separately to see their effect to the solution of the problem. We have also remarked that the instances can be categorized into one set with some meaningful minimum operation quantity for each customer and another set with minimum operation quantity set to 1. Tests are performed on all the instances H with only 1 unit of minimum operation quantity, too. The influence of the minimum operation quantity to the solution of the problem is not consistently observable with our method.

In the next section, we first compare the two heuristics with instances H. The number of feasible solutions and the corresponding logistic ratios are compared. Then, with 30 minutes solution time limit, the heuristics are compared with the combination of heuristics and the FS-MILFP using instances H. Finally, the combination of the heuristics and the FS-MILFP is compared with the column generation approach using instances H and C.

The solver used is CPLEX 12.6.1 with one thread. All tests were run in the computation platform of the LAAS-CNRS with maximum 16GB memory limit with 1 core of Xeon E5-2695 v3 2.30GHz CPU.

Table 5.6: The number of feasible solutions obtained by the two greedy heuristics on instances H

H	$3min$		$30min$		$3h$	
	G1	G2	G1	G2	G1	G2
24	29773.3	41624.7	50000	49999.7	50000	49999.7
48	35342.8	41489	42237.6	49999.6	49210.8	49999.6
72	22451.4	40371.4	40425.8	47838	50000	50000
96	12613.4	34582.2	40176.2	45598.8	41132.2	49997.6
120	5870.6	26515.6	30954	43960.4	37065.8	49794.2
144	3800.2	21840	22638.2	42398	43362.25	49841.6
168	1989	16326.2	17458.2	41610	32660.6	49853
192	1270	11138.6	12046	31351.4	22426.4	37427.25
216	781.8	8806.4	7327	30926.8	15715.4	37174
240	453.4	6666.4	4314.4	29494	12659.8	37080
360	1.5	2938.8	17.5	27453	87.2	36548.2
480	0.8	1907.75	10.5	19065.25	59.8	33727.2
600	0	1201.5	6.2	12147.8	57	27198.8
720	0	766	4.8	7572.2	34.8	22764.5
840	0	504.5	4.5	5171	39	18425

Table 5.7: The average difference of logistic ratio given by the two greedy heuristics on instances H

H	3 min	30 min	3 h
24	-0.2501	-0.252	-0.252
48	0.1647	0.1560	0.154
72	-0.0738	-0.103	-0.103
96	-0.0210	-0.0449	-0.0560
120	-0.0398	-0.0495	-0.0604
144	-0.0236	-0.0385	-0.0553
168	0.0114	-0.000691	-0.00362
192	0.0861	-0.00709	-0.0198
216	0.121	0.00583	-0.0109
240	0.141	0.0112	-0.00473
360	0.0747	0.0437	0.0312
480	0.0462	-0.00192	-0.0116
600	–	-0.243	-0.280
720	–	-0.207	-0.317
840	–	-0.246	-0.306

Table 5.8: The heuristics alone vs the heuristics with FS-MILFP on instances H

H	OBJ	LR	TSC	TDQ
24	-0.176	-0.0778	-0.0541	0.0302
48	-0.111	-0.0956	-0.00868	0.102
72	-0.513	-0.1229	-0.0822	0.0462
96	-17.550	-0.1105	-0.0399	0.0815
120	-0.454	-0.1190	-0.0517	0.0771
144	-0.845	-0.1165	-0.0402	0.0890
168	-0.378	-0.0924	-0.0452	0.0509
192	-0.353	-0.1062	-0.0391	0.0793
216	-0.442	-0.0954	-0.0451	0.0576
240	-0.428	-0.0884	-0.0225	0.0711
360	-0.105	-0.0519	-0.00794	0.0464
480	-0.182	-0.0598	-0.0654	-0.00598
600	-0.171	-0.0503	-0.0451	0.00553
720	-0.128	-0.0449	-0.0456	-0.000669
840	-0.153	-0.0533	-0.0564	-0.00322

5.6.2 Results and Discussion

Firstly, the two greedy heuristics are compared using instances H with a solution time limit of 3 minutes, 30 minutes and 3 hours, respectively. The maximum number of iterations is set to 50 000. Table 5.6 shows the number of feasible solutions obtained within the time limits. The first column of this table is the planning horizon in hours. In the following, "G1" is the state-based heuristic and "G2" is the urgency-based one. As expected, more feasible solutions are found by the heuristics, as the solution time gets longer. From Table 5.6, we can see that G2 can generate more feasible solutions in general, especially when the horizon becomes longer than 360 hours. Let us now look at the difference of the logistic ratio of the two heuristics defined by $\frac{\mathscr{LR}_{G1}-\mathscr{LR}_{G2}}{\mathscr{LR}_{G2}}$. In Table 5.7, the average difference is given for each of the solution time limits. The table only contains instances where both of the two heuristics can find a feasible solution. Within 3 miniutes of time limit, G2 finds a solution strictly better than G1 for 32 out of 49 instances and the average difference is 0.0199. Within 30 minutes of time limit, G2 is better than G1 in 30 out of 55 instances, but the average difference is -0.0313. Within 3 hours of time limit, the solutions given by G2 are better those given by G1 for 29 out of 55 instances. The average difference is then -0.0439. It can be concluded that G2 can find more feasible solutions in less time than G1, but the solutions found by G1 are usually better than that of G2 given a longer solution time, especially when the horizon is longer than 600 hours and when both have difficulty in finding a feasible solution.

Secondly, solutions obtained by the heuristics alone are compared with that obtained by the first step of the complete method. Recall that this step is a combination of the heuristics with FS-MILFP as explained in Section 5.2.6.

Table 5.9: The improvement by the RT-MILFP with column generation compared to that by the heuristics with FS-MILFP on instances H

H	heuristics+FS-MILFP				RT-MILFP			
	OBJ	LR	TSC	TDQ	OBJ	LR	TSC	TDQ
24	0	0	–	–	-2.32	-0.316	0.635	1.83
48	-0.0108	-0.00169	-0.00752	-0.00588	-3.79	-0.185	3.83	6.69
72	-0.0130	-0.00531	-0.00523	0.0000765	-13.4	-0.134	0.640	1.10
96	-0.00378	-0.000253	-0.00345	-0.00320	-0.601	-0.0171	0.385	0.426
120	-0.00600	-0.00302	0.0247	0.0281	-0.708	-0.0522	0.0667	0.153
144	-0.039886	-0.002778	0.000673	0.00344	-0.0563	-0.0102	-0.0102	0.0000214
168	-0.521	-0.00602	-0.000355	0.00576	-0.00956	-0.00336	-0.00332	0.0000384
192	0	0	–	–	-0.0396	-0.00410	-0.00411	-0.0000105
216	-0.0468	-0.0141	-0.00237	0.0121	-0.0774	-0.00377	-0.00372	0.0000492
240	-12.7	-0.0283	-0.0259	0.00187	-2.25	-0.00347	-0.00345	0.0000172
336	0	0	–	–	0	0	–	–
360	-0.0124	-0.00291	-0.00447	-0.00158	0	0	–	–
408	0	0	–	–	0	0	–	–
480	0	-0.00172	0.0112	0.0130	0	0	–	–
504	0	0	–	–	0	0	–	–
600	-0.00400	-0.00232	-0.000122	0.00223	0	0	–	–
720	-0.000713	-0.000340	0.000301	0.000646	0	0	–	–
840	-0.00677	-0.00566	-0.00256	0.00314	0	0	–	–

More precisely, 50 iterations of each of the randomized heuristics are run first. Once a better feasible solution is obtained, it is passed to the FS-MILFP for re-optimization. The total time limit is set to 30 minutes in both cases. It is observed that in nearly 80% of all the cases, the combination of heuristics with FS-MILFP generates better solutions than the two heuristics alone. Table 5.8 reports the differences of the linearized objective (column "OBJ"), the logistic ratio (column "LR"), the total shift costs (column "TSC") and the total delivery quantity (column "TDQ") for each length of horizon in hours. The difference is defined by $\frac{\text{Value obtained with FS-MILFP component}-\text{Value obtained with heuristics only}}{\text{Value obtained with heuristics only}}$. Only the solutions which have been improved by the FS-MILFP in comparison to those obtained by the heuristics are included in this table. For the linearised objective, it is computed by (5.40) with a fixed coefficient η set to the best solution obtained by the winner of the Challenge. It is shown that the method with FS-MILFP can improve the results of the heuristics greatly. The improvement comes from a decrease in the total shift costs and an increase of total delivery quantity. In average, the improvement with the FS-MILFP for the linearised objective can be as high as 31.7% and the improvement for the logistic ratio is 8.6%. This FS-MILFP is very effective and it can take the layover into consideration. That is why we reused it in the third phase for post-optimisation.

Since the RT-MILFP could possibly generate infeasible solutions, we do not compare the solution directly after the column generation. Instead, solutions obtained after 1 hour of the first step are compared with that obtained after the second and the third step, wihch is the RT-MILFP with FS-MILFP re-optimisation. The column generation approach works as follows. First, the initial solution is set to the best solution obtained by 30 minutes combination of random heuristics and the FS-MILFP. Then, new columns are generated with 1000 iteration limit and 15 minutes time limit. The column generation stops if no columns can be found or if the time limit or iteration limit is reached. After that, the master problem is solved as a MILP with the linearised objective with fixed coefficient η. The integer solution obtained is finally passed to the FS-MILFP for re-optimization. The final FS-MILFP is limited to 15 minutes so that the entire solution time (together with the 30 minutes for obtaining the initial solution) is 1 hour. Table 5.9 is the average difference of solutions for instances H of the above procedure compared with 1 hour combination of random heuristics and the FS-MILFP. The values reported are the average difference defined by $\frac{\text{Value given by the method}-\text{Value given by the initial 30 minutes solution}}{\text{Value given by the initial 30 minutes solution}}$. Again, LR (OBJ, TSC and TDQ) is for logistic ratio (the linearised objective with fixed coefficient η, the total costs and the total delivery quantity, respectively). If there is no difference, then a zero is marked and the values for TSC and TDQ are not reported. In general, 45.1% instances can be improved better with RT-MILFP in terms of linearised objective and 49.3% in terms of logistic ratio. The average improvement rate for the linearised objective is 359% with RT-MILFP component compared to 351% without this component. For the logistic ratio, the average improvement rate is only 1.6% with 1 hour of combination of random heuristics and the FS-MILFP while with the RT-MILFP component, the improvement can be as high as 12.7%. One can see from table 5.9 that the column generation works well with the small instances provided that the horizon is not longer than ten days. In addition to the general idea that more delivery quantities with fewer costs can improve the logistic ratio, it should be noted that sometimes a small increase of total shifts costs to deliver large quantities can be worthwhile. This is what we observed in Table 5.9 with the RT-MILFP component for instances with a horizon shorter than five days.

To study the influence of the number of customers, the same tests are performed on instances C. Once more, about 60% of the instances can be better improved with the RT-MILFP component. Table 5.10 shows the improvements of the methods with and without the RT-MILFP component. It can be seen that for instances with less than 35 customers, the improvement with the RT-MILFP can be tremendous compared to the random heuristics. As the number of customers becomes larger, the method with column generation become less powerful than the random heuristics.

Finally, the complete method is tested on the instances B of the challenge with no layover customers. Taking an infeasible initial solution obtained by 6 hours of combined random heuristics and FS-MILFP, with 3 hours of column generation and re-optimization, a feasible solution has been found for instances V2.25 (32 customers, 840 hours) with a logistic ratio of 0.024026.

Table 5.10: The RT-MILFP with column generation and the heuristics with FS-MILFP on instances C

H	heuristics+FS-MILFP		RT-MILFP	
	OBJ	LR	OBJ	LR
5	-3.9969	-0.0060	-21.9978	-0.1399
10	-0.0308	-0.0117	-0.5546	-0.0861
15	-0.0753	-0.0127	-0.3051	-0.0539
20	-0.0822	-0.0169	-0.1450	-0.0395
25	-0.0288	-0.0146	-4.7030	-0.0305
30	-0.0298	-0.0166	-0.0423	-0.0216
35	-0.0334	-0.0127	-0.0827	-0.0198
40	-0.0903	-0.0635	-0.0383	-0.0222
45	-0.0306	-0.0143	-0.0252	-0.0105
50	-0.0737	-0.0502	-0.0306	-0.0128

5.7 Conclusion and Perspectives

In conclusion, the first step of our method is able to solve a large part of the challenge instances and the RT-MILFP with column generation can notably improve the solution. The method is more adapted to small instances (with a horizon shorter than ten days and smaller than 35 customers). The two mixed integer programming components are quite powerful for further optimization of the complete problem. The real challenge lies in combined decision of inventory management and vehicle routing. It is shown that once the vehicle routes are fixed, the resulting problem becomes much easier to solve. Some polynomial sub-cases have been identified.

This work can be a start of many future research activities. First, layover generation should be included in the dynamic programming for solving the column generation subproblem so as to confirm the improvement on instances with layover constraints. Second, a decomposition method with aggregation could be interesting. Instance reduction techniques such as reducing the long horizon to smaller ones or clustering the customers into delivery regions might be necessary before applying a method of column generation or mixed integer programming. Third, the relation between the linearised objective and the logistic ratio could be further studied. In column generation scheme, the coefficient η can have an influence on the dual values of the route selection variables and then influence the construction of new routes. An algorithm that combines the column generation and non-linear fractional programming could thus be helpful. Fourth, in the sub-problem, the shifts are constructed for each trailer and the routes are combined for each driver independently. This could create incompatible routes for the master, deteriorating the quality of the generated columns. If this compatibility could be checked in the sub-problem, the algorithm should be able to generate more feasible routes. Moreover, to solve the integer master problem, the column generation scheme can be integrated into a Branch–and–Price methods. Valid inequalities similar to the ones proposed in [52] might also be applied.

From a management point of view, the logistic ratio is an interesting criterion for the performance of the inventory routing system. Decreasing the total costs of the shifts while increasing the delivery quantity is helpful, but sometimes a small increase in total shift costs can make it possible to delivery much more quantities, resulting in a better logistic ratio. It might be useful to study the maximum quantities one can expect to deliver without increasing the costs. One can also find an efficient way to deliver, so that the minimum increase in shift costs brings the maximum increase in delivery quantity. Therefore, the problem can be studied under multiple objectives.

Chapter 6

Multi-graph Inventory Routing with Energy Consumption

The experience with the real life inventory routing challenge reveals three major drawbacks of classic academic models for the deterministic Inventory Routing Problem (IRP):

1. these models aggregate time into periods assuming that the supplying tours can be performed in one period;
2. customer demands are also aggregated into periods whereas the demand periods for each customer can be irregular in essence;
3. the classic models do not allow counting the travel time and costs with an adequate precision.

In particular, these drawbacks become more serious from the energy point of view. The consumption could be different between the same pair of sites if different paths are taken with different speed or travel duration. Different travel time (or arrival time to the customers) can also result in various combinations of delivery quantity or mass transported on road, which, in turn, would affect the energy consumption.

In this chapter, we present the Multi-Graph Inventory Routing Problem with Energy Consideration (MG-IRP-EC), which is derived from the mass-flow formulation presented in Chapter 4. In this problem, the real road network is represented by a multi-graph. This allows the choice of different paths (with different energetic costs and travel durations) between the same pair of stations. At the same time, the inventory level is monitored in finer time granularity, which is expressed in any relevant possible time unit (hours, minutes or seconds) instead of periods. In this way, the combined decision of inventory and routing—who to visit, how much to deliver, which route to take—is completely integrated with the energy criteria by the timing of visits and by the decisions on the mass flows.

This chapter is organised as follows: in Section 6.1, the multi-graph representation of real road networks is explained. In Section 6.2, the problem on the multi-graph is defined. Then, a simple numerical example is illustrated in Section 6.3. In Section 6.4, the mathematical model is introduced with a decomposition method based on Lagrangian relaxation. The solution method is explained in details in Section 6.5. And finally, some preliminary results are given in Section 6.6.

6.1 Multi-graph Representation of Real Road Networks

A multi-graph representation of the road network is proposed in [75]. It allows alternative routes with a compromise between different attributes of the arcs. Inspired by this representation, we propose the following multi-graph. We first introduce a general *simple* graph of road networks and then explain how a multi-graph can be derived from that.

The real road network is represented by graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$. In this graph, $\mathcal{V} = V \bigcup V'$ is the set of vertices. The set $V = \mathcal{Z} \bigcup \{0\}$ contains all *meaningful nodes*—the set of customers (denoted by $\mathcal{Z}$) and the depot (denoted by 0). The set V' is for the intermediate points—the beginning or end points of a road, the intersection of roads, the traffic lights, etc. Each arc $(i, j) \in \mathcal{A}$ represents a link between two points $i,\ j \in \mathcal{V}$ on the road network and is characterized by a set $\mathcal{C}$ of attributes (the number of attributes $|\mathcal{C}| \geq 1$). For each attribute $a \in \mathcal{C}$, $C^a_{i,j}$ is the value of this attribute on arc (i, j). The graph $G = (V, A)$ can be incomplete and it is simple.

For each arc $(i, j) \in V \times V$ between meaningful nodes, $\mathcal{P}_{i,j}$ denotes the set of Pareto-optimal paths from node $i \in V$ to node $j \in V$ considering the $\mathcal{C}$ attributes in $\mathcal{G}$. Based on the original graph of road network $\mathcal{G}$, the multi-graph $G' = (V, A)$ is introduced. The set of vertices $V = \mathcal{Z} \bigcup \{0\}$ contains the depot and the customers. The set of arcs A is defined as follows. For each pair of nodes $(i, j) \in V \times V$ and road path $l \in \mathcal{P}_{i,j}$ linking i and j, there is an arc $(i, j)^l \in A$. The value of each attribute $a \in \mathcal{C}$ of the arc $(i, j)^l$ is given by $C^a_{i,j,l}$.

In the following, two attributes are considered: the travelling time and the energy consumption. The travelling time is a fixed value per arc. It is denoted by $T^l_{i,j}$ for the path $l \in \mathcal{P}_{i,j}$ from node $i \in V$ to node $j \in V$. The energy consumption is a fixed value per arc per unit of product. It is denoted by $C^e_{i,j,l}$ for the arc $l \in \mathcal{P}_{i,j}$ from node $i \in V$ to node $j \in V$. The same energy estimation model as presented in the Section 3.2 of Chapter 4 is applied to compute the energy consumption of a vehicle on a road segment.

To construct the multigraph, a Multi-Objective Shortest Path Problem (MOSP) needs to be solved between each pair of meaningful nodes $i,\ j \in V$. The problem is restricted to bi-objective case in this thesis. Even though, the work [83] implies that any algorithm solving a bi-objective shortest path problem is, at least, exponential in the worst case since the number of Pareto-optimal paths can be exponential in the number of nodes. The MOSP has not been discussed into details in this thesis. In the following, only two paths are considered for each pair of meaningful nodes, one minimizing the travelling time and the other minimizing the energy consumption. These two paths are obtained by solving a mono-objective all-pair Shortest Path Problem (SPP) twice, each time with a different objective using the classic dynamic programming algorithm of Floyd-Warshall [70].

6.2 Problem Presentation

The general setting is the same as in the classic IRPs presented in Section 2.1.1 in Chapter 2. The vendor (or transporter) manages the inventory of a set of customers denoted by $\mathcal{Z}$. Each customer $i \in \mathcal{Z}$ has a minimum and maximum inventory level, denoted by $\overline{I}_i$ and $\underline{I}_i$ respectively. By making deliveries, he ensures that the inventory never goes below (or beyond) the minimum (or maximum) inventory level for each customer i (respectively). The vendor has a fleet $\mathcal{K}$ of *homogeneous* vehicles at disposal for the deliveries. Each vehicle $k \in \mathcal{K}$ has the same capacity B and curb weight W (both in kg). The inventory is monitored for a *finite* horizon denoted by $\mathcal{H}$. At the beginning of the time horizon, the inventory level of customer $i \in \mathcal{Z}$ is given by I_i^0, and the vehicles are all at the depot 0 with an initial quantity not higher than its capacity B. It is assumed that each vehicle makes one tour during the whole horizon.

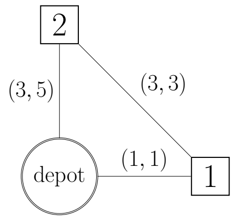
The MG-IRP-EC possesses some new features. During the time horizon with totally H hours, the customers consume a *single* product gradually. Each customer $i \in \mathcal{Z}$ has a series of consumption events, denoted by the set $\mathcal{R}_i$. Each consumption $r \in \mathcal{R}_i$ is defined by a time θ_i^r and an amount of product R_i^r. Without loss of generality, it is assumed that the consumption events are ordered with respect to the non-decreasing order of θ_i^r in the series $\mathcal{R}_i$. They are enumerated from 1 to $|\mathcal{R}_i|$, with $|\mathcal{R}_i|$ the number of consumption events. In the following, we do not distinguish the set $\mathcal{R}_i$ and the number $|\mathcal{R}_i|$.

The problem is defined on the multi-graph G'. The decisions are the same as in classic IRPs adding decisions on the exact timing of each delivery to each customer. This problem contains the following categories of constraints, which do not vary in essence from the classic ones, even though their formulation will be rather different because of the timing of the deliveries:

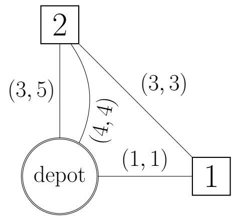
- inventory capacity: the inventory level of each customer should never go beyond the minimum and the maximum level;
- vehicle capacity: the quantity of product transported by a vehicle should never exceed its capacity;
- vehicle flow: the vehicle should depart from the depot, make a tour among the customers to be visited and return back to the depot;
- routing: each customer is visited at most once in the tour of each vehicle;
- timing: the arrival time between two consecutive locations $i, j \in V$ visited in the same tour should be at least the travel time from i to j;

The objective is to minimize the total energy consumed in the routing while respecting the above constraints.

The differences between this problem and the classic IRP are as follows.



(a) Graph



(b) Multi-graph

Figure 6.1: Example with 1 depot and 2 customers

- The inventory monitoring and the routing are both performed in continuous time. There is no notion of periods as in the classic inventory routing. Consequently, the problem combinatorics in relation to time get higher.

- The decision of how much to deliver and which route to take is combined by the decision of arrival time of each delivery. The customers consume product during the travelling time of the vehicles.

- The problem is based on a multi-graph. The choice of arcs on each tour introduces another dimension of combinatorics.

It should be noted that different arrival time of the vehicle to a customer might result in different delivered quantity. Moreover, the energy consumption of the vehicle on the route is dependent on the delivery quantity. These justify the use of a multi-graph from the modelling point of view.

In the following, we make the hypothesis that the number of vehicles in the fleet (denoted by K) is enough to cover all the customer consumption in the whole time horizon. The minimum inventory level of each customer is considered to be 0 for sake of simplicity. In addition, the delivery time by vehicle $k \in \mathcal{K}$ to customer $i \in \mathcal{Z}$ is assumed to be the same as the arrival time of k to i and the delivery is supposed to be instantaneous. By convention, each time a customer is visited before a certain consumption event, the sequence of the operations is the delivery first, followed by the consumption and finally the computation of the inventory level before the next consumption. The Hour 0 is an artificial start of the horizon without consumption or delivery. Moreover, the capacity of the depot is considered to be infinite and it contains as many products as needed.

6.3 Numerical Illustration of the Problem

In this section, a simple numerical example illustrates the problem. Let us consider a system with one central depot and two customers as shown in Figure 6.1a. In this example, each arc is labeled by two attributes: the travel time and the energy consumption per unit of mass. These attribute values are given in parentheses in Figure 6.1a, the former being the time and the latter the energy.

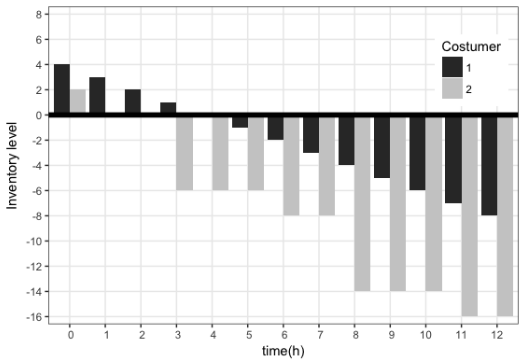
The capacity for each customer is $\overline{I}_1 = 4$ and $\overline{I}_2 = 8$. The capacity of a vehicle is $B = 6$ with a weight $W = 6$ and there are 5 vehicles in total. A total time horizon of 12 hours is considered. The customers are denoted by i_1 and i_2. The consumption of i_1 is assumed periodic with a rate of 1 unit per hour, while the consumption of i_2 varies sporadically over time. It is given by the vector $\mathcal{R}_2 = \{2, 6, 2, 6, 2\}$ for the quantity and the vector $\{1, 3, 6, 8, 11\}$ for the time. For instance, i_2 consumes 2 units of product at the beginning of the time unit 1, 6 units at the beginning of the time unit 3, and so on. Figure 6.2a shows the initial inventory variation of the two customers if nothing is delivered. If the inventory falls below zero, a stock-out is triggered. We can see that the next stock-out of i_2 will happen after the second consumption at the time unit 3, and that of i_1 at the time unit 5.

One can say that some "implicit time windows" lie in this problem due to the inventory limits of each customers. In this example, at the beginning of the horizon, i_1 can be delivered at the beginning of the time unit 2 at the earliest (because of the maximum inventory level and the initial level) and at the beginning of the time unit 5 at the latest (because of the minimum inventory level). Similarly, i_2 can be visited at the beginning of the time unit 1 at the earliest and at the beginning of the time unit 3 at the latest.

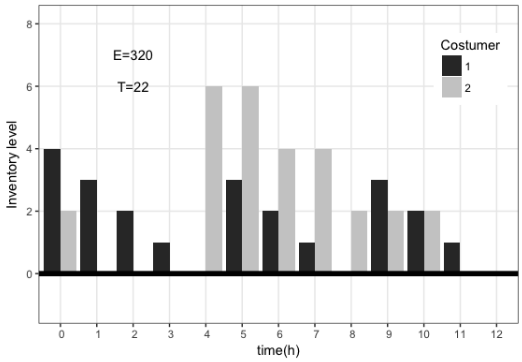
Now we give an empirical solution to this problem. First of all, the delivery to i_2 is imminent because there is no road to i_2 faster than 3 time units of travelling. The amount to be delivered to i_2 cannot exceed 6 units due to the capacity of the vehicle, and it should give 6 units at least because otherwise, the quantity in i_2 would still not be enough to cover the consumption (6 units). Since the earliest arrival time of this vehicle from i_2 to i_1 is at the time unit 6 but the next stock-out of i_1 is at the time unit 5, another vehicle is needed to cover the consumption of i_1, but the exact arrival time needs to be decided. If the vehicle arrives at the time unit 1, it could only deliver 1 unit to i_1, or else a later delivery with higher quantity can be planned, but is that efficient in terms of energy? That is where the combinatorial aspect becomes complicated.

Figure 6.2b and 6.2c show the inventory level variations obtained when minimizing the total travel time and the energy, respectively. Routing details of the solutions are presented in Table 6.1. The starting time is given in the column titled "Start" and the details of the routes are presented in the column titled "Route details". In the "Route details" column, the numbers on the left and right of an arrow are the indices of the stations visited in the route, the number above an arrow is the mass transported on the arc between the two stations linked by this arrow, and the number below an arrow is the travelling time spent on the arc.

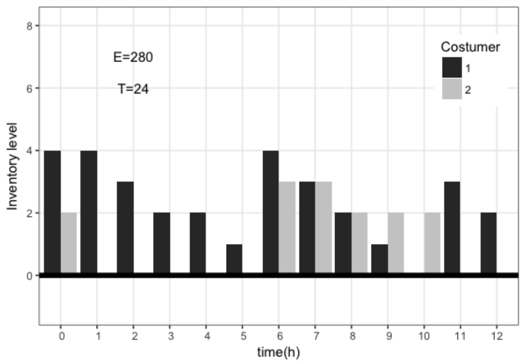
Table 6.1a and Table 6.1b correspond to the routing details in the time minimizing and the energy minimizing solutions, respectively. With time minimization, only direct shippings are found in this example. There are 5 routes in total. The customer i_1 is visited at the time unit 5 and 9, each with an amount of 4. The customer i_2 is visited at the time unit 3 with an amount of 6, at the time unit 4 with an amount of 6 and at the time unit 8 with an amount of 4. With energy minimization, it is more common to have several deliveries to different customers



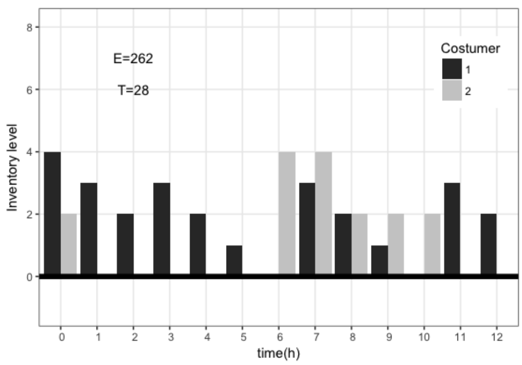
(a) Inventory level at the end of each hour



(b) Time minimizing



(c) Energy minimizing



(d) Energy minimizing on multigraph

Figure 6.2: Inventory variation with different objectives

combined in one route. The total time taken in the time minimizing solution is 22 hours with an energy consumption of 320, while in the energy minimizing solution, 24 hours is used with an energy consumption of only 280. It is shown that the time minimizing solution is not necessarily efficient in terms of energy, and vice versa.

Actually, a multi-graph can be constructed based on the initial graph as shown in Figure 6.1b. In this graph, we have another arc between the depot and the customer i_2 with a travel time of 4 hours and an energy consumption of 4 units per unit of mass. If we solve the problem again with consideration of this newly added arc with energy minimization as objective, we find a solution that improves the energy consumption to 262 with travelling time increased to 28.

In fact, the routes given in this solution take advantage of the newly added arc when the visit to i_2 is less urgent. Looking at the routing details in Table 6.1c, the newly added arc is used when visiting i_2 in Route 2 and when returning to the depot from i_2 in Routes 0,1 and 2. This arc represents a by-pass to i_1, which is forbidden in classic inventory routing since each customer is visited exactly once in a route.

Table 6.1: Routing details of the solutions

(a) simple graph time minimizing

	Start	Route details
1	0	$0 \xrightarrow[3]{12} 2 \xrightarrow[3]{6} 0$
2	1	$0 \xrightarrow[3]{12} 2 \xrightarrow[3]{6} 0$
3	4	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$
4	5	$0 \xrightarrow[3]{12} 2 \xrightarrow[3]{8} 0$
5	8	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$

(b) simple graph energy minimizing

	Start	Route details
1	0	$0 \xrightarrow[3]{12} 2 \xrightarrow[3]{6} 0$
2	2	$0 \xrightarrow[1]{12} 1 \xrightarrow[3]{11} 2 \xrightarrow[3]{6} 0$
3	2	$0 \xrightarrow[1]{12} 1 \xrightarrow[3]{11} 2 \xrightarrow[3]{6} 0$
4	6	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$
5	10	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$

(c) multi-graph energy minimizing

	Start	Route details
1	0	$0 \xrightarrow[3]{12} 2 \xrightarrow[4]{6} 0$
2	2	$0 \xrightarrow[4]{12} 2 \xrightarrow[4]{6} 0$
3	2	$0 \xrightarrow[1]{12} 1 \xrightarrow[3]{10} 2 \xrightarrow[4]{6} 0$
4	6	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$
5	10	$0 \xrightarrow[1]{12} 1 \xrightarrow[1]{8} 0$

6.4 Mathematical Model

A mathematical model for this problem is presented in this section. It is assumed that each vehicle makes at most one route.

6.4.1 Variable Definitions

The model contains the following decision variables:

For each arc $l \in \mathcal{P}_{i,j}$ linking each pair of locations $(i, j) \in V \times V$ and for each vehicle $k \in \mathcal{K}$, continuous variable $m^l_{i,j,k} \in [0, B]$ represents the mass transported by vehicle k on path l from node i to node j. It is bounded by the vehicle capacity B.

For each consumption $r \in \mathcal{R}_i$ of customer $i \in \mathcal{Z}$ and each vehicle $k \in \mathcal{K}$, variable $q^r_{i,k} \in \left[0, \overline{I}_i\right]$ is the amount of product given to customer i *before* the consumption event r by the vehicle k. It is limited by the customer capacity $\overline{I}_i$.

For each customer $i \in \mathcal{Z}$ and vehicle $k \in \mathcal{K}$, variable $art^k_i \in [0, H]$ is the delivery time to customer i by vehicle k.

For each arc $l \in \mathcal{P}_{i,j}$ linking each pair of locations $(i, j) \in V \times V$ and for each vehicle $k \in \mathcal{K}$, variable $x^l_{i,j,k} \in \{0, 1\}$ is a binary variable equal to 1 if the vehicle k travels from i to j using arc l, 0 otherwise.

For each consumption $r \in \mathcal{R}_i$ of customer $i \in \mathcal{Z}$ and each vehicle $k \in \mathcal{K}$, variable $z^r_{i,k} \in \{0, 1\}$ is a binary variable equal to 1 if the vehicle k makes a delivery to customer i *before* consumption r, 0 otherwise.

6.4.2 Complete Model

The complete model (denoted by **(P)**) is given below. The objective (6.1) is to minimize the total energy consumption. The first term is proportional to the flow transported on each arc. The second can be viewed as a fixed cost when a path l linking a pair of stations i and j are traversed by a vehicle.

P:

$$\text{minimize } \mathscr{E} = \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} m^l_{i,j,k} + W \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} x^l_{i,j,k} \tag{6.1}$$

s.t:

Transportation flow balance

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} m^l_{j,i,k} - \sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} m^l_{i,j,k} = \sum_{r\in\mathcal{R}_i} q^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.2}$$

Link routing and flow

$$m^l_{i,j,k} \leq B x^l_{i,j,k} \qquad \forall k \in \mathcal{K}, \forall (i,j) \in V \times V, \forall l \in \mathcal{P}_{i,j} \tag{6.3}$$

Routing

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{0,j}} x^l_{0,j,k} \leq 1 \qquad \forall k \in \mathcal{K} \tag{6.4}$$

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{j,i}} x^l_{j,i,k} - \sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} x^l_{i,j,k} = 0 \qquad \forall k \in \mathcal{K}, \forall i \in V \tag{6.5}$$

Arrival time

$$art^k_j - art^k_i \geq \sum_{l\in\mathcal{P}_{i,j}} T^l_{i,j} x^l_{i,j,k} - H(1 - \sum_{l\in\mathcal{P}_{i,j}} x^l_{i,j,k}) \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K} \tag{6.6}$$

$$art^k_i \geq \sum_{r\in\mathcal{R}_i\setminus\{1\}} (\theta^{r-1}_i + 1) z^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.7}$$

$$\sum_{r\in\mathcal{R}_i} \theta_i^r z_{i,k}^r \geq art_i^k \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K} \qquad (6.8)$$

Link visit and routing

$$\sum_{j\in V}\sum_{l\in\mathcal{P}_{j,i}} x_{j,i,k}^l = \sum_{r\in\mathcal{R}_i} z_{i,k}^r \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K} \qquad (6.9)$$

Link visit and amount of delivery

$$q_{i,k}^r \leq \overline{I}_i z_{i,k}^r \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K},\ \forall r \in \mathcal{R}_i \qquad (6.10)$$

Inventory monitoring

$$I_i^0 + \sum_{k\in\mathcal{K}}\sum_{u=1}^{r} q_{i,k}^u - \sum_{u=1}^{r} R_i^u \geq 0 \qquad \forall i \in \mathcal{Z},\ \forall r \in \mathcal{R}_i \qquad (6.11)$$

$$I_i^0 + \sum_{k\in\mathcal{K}}\sum_{u=1}^{r} q_{i,k}^u \leq \overline{I}_i \qquad \forall i \in \mathcal{Z},\ \forall r \in \mathcal{R}_i \qquad (6.12)$$

Additional constraints

$$\sum_{r\in\mathcal{R}_i} z_{i,k}^r \leq 1 \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \qquad (6.13)$$

$$\sum_{l\in\mathcal{P}_{i,j}} x_{i,j,k}^l \leq 1 \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K} \qquad (6.14)$$

$$\sum_{i\in\mathcal{Z}}\sum_{r\in\mathcal{R}_i} q_{i,k}^r \leq B \qquad \forall k \in \mathcal{K} \qquad (6.15)$$

Variable domains

$$x_{i,j,k}^l \in \{0,1\} \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K}, \forall l \in \mathcal{P}_{i,j} \qquad (6.16)$$

$$m_{i,j,k}^l \in [0,B] \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K}, \forall l \in \mathcal{P}_{i,j} \qquad (6.17)$$

$$z_{i,k}^r \in \{0,1\} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \qquad (6.18)$$

$$q_{i,k}^r \in \left[0, \overline{I}_i\right] \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \qquad (6.19)$$

$$art_i^k \in [0,H] \qquad \forall i \in V, \forall k \in \mathcal{K} \qquad (6.20)$$

Constraints (6.2) are for the flow balance of the mass transported in the network.

Constraints (6.3) link routing variables x and mass flow variables m. Constraints (6.4) and (6.5) are for the vehicle routing. Constraints (6.4) ensure that one vehicle performs at most one route. Constraints (6.5) ensure that the number of incoming arcs to a customer is the same with the number of outgoing arcs from this customer for each vehicle. Constraints (6.6)–(6.8) are for the timing of each visit. More precisely, Constraints (6.6) impose that the duration between two consecutive visits of each vehicle is larger than the travel time. Constraints (6.7) and (6.8) make certain that the delivery to a customer is between two consecutive consumption events of this customer.

Constraints (6.9) say that if the vehicle visits a customer, then there is at least one delivery before one of the customer's consumptions. This set of constraints links the routing variables x with the delivery decision variables z. Constraints (6.10) link the delivery decision variables z and the delivery quantity variables q. They impose that if there is no delivery, then the delivered quantity is 0. Otherwise, the delivered quantity is no larger than the customer capacity.

Constraints (6.11) and (6.12) bound the inventory level in the acceptable range *before and after* each consumption or delivery.

Additionally, Constraints (6.13) ensure that there is at most one delivery to each customer per vehicle. Constraints (6.14) make sure that at most one path in the multi-graph is used in each route. Constraints (6.15) is for the capacity of the vehicles. Constraints (6.16)–(6.20) define the variable domains.

6.4.3 Flow Formulation for the Timing of Visits

According to [24], constraints (6.6)–(6.8) can be regarded as a type of the well-known Miller-Tucker-Zemlin (MTZ) constraints for connectivity and route duration. They can be advantageously replaced by the following constraints (6.21)–(6.23) with (6.24) defining the domain of variables tf. In these constraints, variables art are replaced by a set of flow variables $tf_{i,j,k} \in [0, H]$ indicating the arrival time of vehicle k to node j if the previous node visited is node i.

$$\sum_{j \in V \setminus \{i\}} tf_{i,j,k} - \sum_{j \in V \setminus \{i\}} tf_{j,i,k} \geq \sum_{j \in V \setminus \{i\}} \sum_{l \in \mathcal{P}_{i,j}} T^l_{i,j} x^l_{i,j,k} \qquad \forall i \in V, \forall k \in \mathcal{K} \quad (6.21)$$

$$\sum_{j \in V \setminus \{i\}} tf_{j,i,k} \geq \sum_{r \in \mathcal{R}_i \setminus \{1\}} (\theta^{r-1}_i + 1) z^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \quad (6.22)$$

$$\sum_{r \in \mathcal{R}_i} \theta^r_i z^r_{i,k} \geq \sum_{j \in V \setminus \{i\}} tf_{j,i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \quad (6.23)$$

$$tf_{i,j,k} \in [0, H] \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K} \quad (6.24)$$

Constraints (6.21) are for the precedence of visits. Note that they are now related to vertex, instead of arcs as in (6.6). In Constraints (6.21), the term $\sum_{j \in V \setminus \{i\}} tf_{i,j,k}$ sets the arrival time to the next location visited after the location i and the term $\sum_{j \in V \setminus \{i\}} tf_{j,i,k}$ gives the arrival time to i. Constraints (6.22) and (6.23) bound the arrival time to each customer i between two consecutive consumption events of this customer.

These flow variables for the timing of visits (called "time flow variables") reduces the number of precedence constraints, to the detriment of the number of variables. Their efficiency will be discussed in the Section 6.6.

6.5 Lagrangian Relaxation and Decomposition Method

A decomposition method based on Lagrangian relaxation is presented in this section. From now on, we consider the complete model with time flow variables defined by:

$$\begin{aligned} \text{minimize} \quad & (6.1) \\ \text{s.t.} \quad & (6.2)\text{–}(6.5),\ (6.9)\text{–}(6.19) \text{ and } (6.21)\text{–}(6.24). \end{aligned}$$

Some modifications are applied to this model so that it can be decomposed to two independent sub-problems after the relaxation. In the following of this section, the modified model is first presented. Then, the modified model is relaxed by dualizing some linking constraints. In this way, two sub-problems are obtained and their complexities are analysed. The solution algorithm is described in the end.

6.5.1 Modified Model

The idea is to relax the linking constraints (6.2) and (6.9) between inventory management and routing, while keeping the transportation flow balance and the exact timing of visits in the routing part. Additional variables $w_{i,k}$ are introduced for the mass delivered by each vehicle $k \in \mathcal{K}$ to each customer $i \in \mathcal{Z}$. Besides, constraints (6.2) are replaced by (6.25). Constraints (6.26) are also added into the model to link variables w and q. Constraints (6.27) is redondant with constraints (6.15) but they will be used in the decomposed problem later. Constraints (6.28) define the domain of variables w.

$$\sum_{j \in V} \sum_{l \in \mathcal{P}_{i,j}} m^l_{j,i,k} - \sum_{j \in V} \sum_{l \in \mathcal{P}_{i,j}} m^l_{i,j,k} = w_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \qquad (6.25)$$

$$w_{i,k} = \sum_{r \in \mathcal{R}_i} q^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \qquad (6.26)$$

$$\sum_{i \in V} w_{i,k} \leq B \qquad \forall k \in \mathcal{K} \qquad (6.27)$$

$$w_{i,k} \in [0, B] \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \qquad (6.28)$$

This modified model is denoted **(P′)** (see Annexe B for the complete model before the Lagrangian relaxation).

P′:

$$\begin{aligned} \text{minimize} \quad & (6.1) \\ \text{s.t.} \quad & (6.25),\ (6.26),\ (6.3)\text{–}(6.5),\ (6.10)\text{–}(6.19),\ (6.21)\text{–}(6.24) \text{ and } (6.27)\text{–}(6.28). \end{aligned}$$

6.5.2 Lagrangian Relaxation of the Modified Model

Let us take the Lagrangian relaxation of **(P′)** that dualizes constraints (6.26), (6.9), (6.22) and (6.23) respectively with the multipliers $\alpha^k_i \in \mathbb{R}$, $\beta^k_i \in \mathbb{R}$, $\gamma^k_i \in \mathbb{R}^+$, $\delta^k_i \in \mathbb{R}^+$ for each $i \in \mathcal{Z}$ and $k \in \mathcal{K}$. The relaxed problem can then be written:

L($\alpha, \beta, \gamma, \delta$):

$$\text{minimize } \mathscr{E}_L(\alpha,\beta,\gamma,\delta) = \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} m^l_{i,j,k} + \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} \tilde{c}_{i,j,k,l} x^l_{i,j,k}$$
$$+ \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{Z}} \alpha^k_i w_{i,k} + \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{Z}} (\delta^k_i - \gamma^k_i) \sum_{j\in V\setminus\{i\}} tf_{j,i,k}$$
$$- \sum_{k\in\mathcal{K}} \sum_{i\in\mathcal{Z}} \alpha^k_i \sum_{r\in\mathcal{R}_i} q^r_{i,k} - \sum_{k\in\mathcal{K}} \sum_{i\in V} \beta^k_i \sum_{r\in\mathcal{R}_i} z^r_{i,k}$$
$$+ \sum_{k\in\mathcal{K}} \sum_{i\in V} \gamma^k_i \sum_{r\in\mathcal{R}_i\setminus\{1\}} (\theta^{r-1}_i + 1) z^r_{i,k} - \sum_{k\in\mathcal{K}} \sum_{i\in V} \delta^k_i \sum_{r\in\mathcal{R}_i} \theta^r_i z^r_{i,k} \tag{6.29}$$

with constraints (6.25), (6.3)–(6.5), (6.21), (6.10)–(6.15), (6.27), (6.16)–(6.19), (6.24) and (6.28), where

$$\tilde{c}_{i,j,k,l} = WC^e_{i,j,l} + \beta^k_j \qquad \forall k \in \mathcal{K},\ (i,j) \in V \times V,\ l \in \mathcal{P}_{i,j} \tag{6.30}$$

Since the sets of variables $\{q, z\}$ and $\{m, w, x\}$ are now independent of each other, the relaxed problem $(\mathbf{L}(\alpha, \beta, \gamma, \delta))$ can be decomposed into

- an Inventory Allocation Problem (IA) with variables q and z with inventory monitoring;
- for each vehicle k, a Capacitated Cycle Problem (CCP) with fixed charges and precedence constraints on a multi-graph.

6.5.3 Inventory Allocation Sub-problem

The problem IA consists in allocating vehicle deliveries to customer consumption events. In this problem, each customer is visited at most once by each vehicle in the whole horizon. The total delivery quantity for each vehicle should not exceed the capacity. The inventory level of each customer must never exceed the capacity or fall below zero.

A fixed cost $\nu^r_{i,k}$ exists for visiting a customer $i \in \mathcal{Z}$ before each consumption $r \in \mathcal{R}_i$ by each vehicle $k \in \mathcal{K}$. It is defined by a combination of β^k_i, γ^k_i and δ^k_i:

$$\nu^r_{i,k} = -\beta^k_i + \gamma^k_i(\theta^{r-1}_i + 1) - \delta^k_i \theta^r_i \qquad \forall i \in \mathcal{Z}, \forall r \in \mathcal{R}_i, \forall k \in \mathcal{K} \tag{6.31}$$

under the additional assumption that $\theta^0_i = -1$.

The values of α^k_i define a cost proportional to the quantity delivered to each customer $i \in \mathcal{Z}$ by each vehicle $k \in \mathcal{K}$. The objective is a combination of these costs, resulting from the Lagrangian multipliers.

The problem is to find the best delivery quantity and time to each customer by each vehicle. Since the exact time of each delivery makes no difference in this problem, we only need to decide the consumption event before which to make each visit. The objective is to minimize the total costs induced by the multipliers, while making sure that the inventory level of each customer always stays inside the limits and that the vehicle never carries more than its capacity.

The decision variables are:

- $q^r_{i,k} \in \left[0, \overline{I}_i\right]$ for the amount of product delivered to each customer $i \in \mathcal{Z}$ after each consumption $r \in \mathcal{R}_i$ by each vehicle $k \in \mathcal{K}$;
- $z^r_{i,k} \in \{0,1\}$ for the decision of whether vehicle $k \in \mathcal{K}$ visit customer $i \in \mathcal{Z}$ before his consumption $r \in \mathcal{R}_i$ (equal 1) or not (0);

The model is composed of constraints (6.10)–(6.13), (6.15), and variables domains (6.18) and (6.19) derived from the relaxed problem.

IA:

$$\min -\sum_{k\in\mathcal{K}}\sum_{i\in\mathcal{Z}}\alpha_i^k\sum_{r\in\mathcal{R}_i} q^r_{i,k} \quad +\sum_{k\in\mathcal{K}}\sum_{i\in V}\sum_{r\in\mathcal{R}_i}\nu^r_{i,k} z^r_{i,k} \tag{6.32}$$

s.t

$$q^r_{i,k} \le \overline{I}_i z^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.10}$$

$$I_i^0 + \sum_{k\in\mathcal{K}}\sum_{u=1}^{r} q^u_{i,k} - \sum_{u=1}^{r} R_i^u \ge 0 \qquad \forall i \in \mathcal{Z}, \forall r \in \mathcal{R}_i \tag{6.11}$$

$$I_i^0 + \sum_{k\in\mathcal{K}}\sum_{u=1}^{r} q^u_{i,k} \le \overline{I}_i \qquad \forall i \in \mathcal{Z}, \forall r \in \mathcal{R}_i \tag{6.12}$$

$$\sum_{r\in\mathcal{R}_i} z^r_{i,k} \le 1 \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.13}$$

$$\sum_{i\in\mathcal{Z}}\sum_{r\in\mathcal{R}_i} q^r_{i,k} \le B \qquad \forall k \in \mathcal{K} \tag{6.15}$$

$$z^r_{i,k} \in \{0,1\} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.18}$$

$$q^r_{i,k} \in \left[0, \overline{I}_i\right] \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.19}$$

In the following, the problem *with heterogeneous fleet* is demonstrated to be NP-hard even with one single consumption per customer and with non-negative fixed visiting costs ν. The fixed visiting costs can be restricted to be non-negative because if there is a negative fixed visiting cost $\nu^r_{i,k}$ before a certain consumption r of a customer i by a vehicle k, it will always be profitable to deliver to i before the consumption event r by vehicle k. By redefining the fixed costs for these visits to zero, we obtain a problem instance with non-negative fixed visiting costs.

Proposition 2. *IA with heterogeneous fleet is NP-hard.*

Proof. This proof shows that IA can be transformed from the MinCostFixedFlow problem.

First of all, an instance of the MinCostFixedFlow is defined on a directed graph $G = (V, A)$. The vertex set V contains two special vertex, namely, a source and a sink. Each arc $(i, j) \in A$ is associated with a non-negative cost $c_{i,j}$ and a non-negative capacity $u_{i,j}$. The problem is to find a minimum cost subset A' of A such that in (V, A'), the flow from the source s to the sink t is at least F.

It is demonstrated to be strongly NP-hard even on bipartite graphs in [97].

From an instance of the MinCostFixedFlow problem on a bipartite graph, we construct an instance of IA problem and demonstrate that the solution of the MinCostFixedFlow problem exists if and only if IA gets an optimal solution.

We apply the reduction from the problem with one source and multiple sinks, which is obviously reducible to the basic version of the problem with single source and single sink.

Consider now an instance of the MinCostFixedFlow problem with one source and multiple sinks on a bipartite graph $G = (V, A)$. The set of vertices $V = \{s\} \bigcup T \bigcup \mathcal{K} \bigcup \mathcal{Z}$ contains a special source vertex s, a set T of sink vertices and two disjoint sets $\mathcal{K}$ and $\mathcal{Z}$. There is an arc from s to each vertex $k \in \mathcal{K}$, and from each vertex $i \in \mathcal{Z}$ to a sink $t_i \in T$. Each vertex $i \in \mathcal{Z}$ is linked to each vertex $k \in \mathcal{K}$. The problem is to find a minimum cost subset A' of A such that in (V, A') the flow from the source s to each sink $t_i \in T$ is at least F_i.

Note that the bipartite graph $(\mathcal{K} \bigcup \mathcal{Z}, A)$ can be incomplete, but this does not influence the analysis of NP-hardness, since the graph $\mathcal{G}$ can be transformed to a complete one by associating a very large cost with the arcs in the complete graph but not in A. In the following, without loss of generality, $\mathcal{G}$ is considered to be complete.

We now construct an instance of IA:

- $\mathcal{K}$ defines the set of vehicles
- $\mathcal{Z}$ defines the set of customers
- T defines the set of consumption events of customers
- the capacity of each vehicle is equal to the capacity of each arc (s, k) with $k \in \mathcal{K}$
- the capacity of each customer $i \in \mathcal{Z}$ is equal to F_i
- each customer has one single consumption with an amount F_i
- the initial level of each customer is 0
- the fixed cost $\nu^1_{i,k}$ of visiting a customer $i \in \mathcal{Z}$ by vehicle $k \in \mathcal{K}$ before his only consumption 1 is defined by the sum of fixed costs of the arcs (s, k), (k, i), (i, t) : $c_{s,k} + c_{k,i} + c_{i,t}$

If one obtains a minimum cost subset $A' \subset A$ such that the flow in (V, A') from source s to each sink $t_i \in T$ is at least F_i, then one can define a solution of IA by setting $z^1_{i,k}$ to 1 if arc $(k, i) \in A'$, to 0, otherwise, with $q^r_{i,k}$ set to the corresponding flow amount on arc (k, i). If one obtains a solution of the above instance of IA, one can identify a minimum cost subset A' with arcs $(k, i) \in A'$ if vehicle k is set to visit customer i before his consumption. This transformation can obviously be done in polynomial time. □

Note that the complexity of the IA problem *with homogeneous fleet* is still open. Actually, the IA problem in general can be seen as a special case of the more general Minimum Concave-cost Network Flow problem, which is NP-hard in the strong sense [80]. It is also closely related to the Lot Sizing problem. In fact, the delivery of each vehicle to each customer before one of his consumption can be seen as a production activity. The fixed cost of each visit can then be seen as a fixed set-up cost under the context of production planning. If we consider only one customer with a set of consumption events with one single capacitated vehicle and allow several visits to the same customer by this vehicle, the problem transforms into one of Economic Lot Sizing, which can be solved in polynomial time [132].

If we consider the fleet as a set of sources, fix the total quantity to deliver by each vehicle and remove the capacity of vehicles and customers, this problem can also be seen as a resource allocation problem with separable convex objective and network constraints [95]. The time complexity is thus proportional to the running time of the maximum flow algorithm inside the network of vehicles and customer consumptions.

6.5.4 Capacitated Cycle Sub-problem

The exact sequencing of each visit to each customer by each vehicle is decided in sub-problems CCP. Since the vehicles are independent from each other, there is one CCP per vehicle. For each vehicle k, a cycle problem with fixed costs on a multi-graph has to be solved under precedence and capacity constraints. In this problem, each vehicle has to make a tour that starts and ends at the depot and travels among profitable customers to make deliveries.

Each time the vehicle k travels from site i to site j using the arc $l \in \mathcal{P}_{i,j}$, there is a modified cost $\tilde{c}_{i,j,k,l}$ related to the Lagrangian multiplier β_j^k as given in Equation (6.30). There is also a cost proportional to the amount of commodity flow on each arc $l \in \mathcal{P}_{i,j}$ from site i to site j defined by the cost $C^e_{i,j,l}$ of the original problem **(P)**. Each time vehicle k delivers any product to customer i, there is a cost proportional to the amount of delivery defined by α_i^k. Additional costs proportional to the arrival time of vehicle k to customer i are given by $\delta_i^k - \gamma_i^k$.

The objective is to minimize the total cost. The following mathematical model for a fixed vehicle k is derived from the relaxed model $(\mathbf{L}(\alpha, \beta, \gamma, \delta))$, with constraints (6.25), (6.3)–(6.5), (6.21), (6.14), (6.27), (6.16) (6.17), (6.24) and (6.28).

CCP:

$$\min \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} m^l_{i,j,k} + \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} \tilde{c}_{i,j,k,l} x^l_{i,j,k} + \sum_{i\in\mathcal{Z}} \alpha_i^k w_{i,k} + \sum_{i\in\mathcal{Z}} (\delta_i^k - \gamma_i^k) \sum_{j\in V\setminus\{i\}} tf^k_{j,i} \tag{6.33}$$

s.t.

$$\sum_{j\in V}\sum_{l\in\mathcal{P}_{i,j}} m^l_{j,i,k} - \sum_{j\in V}\sum_{l\in\mathcal{P}_{i,j}} m^l_{i,j,k} = w_{i,k} \qquad \forall i\in\mathcal{Z} \quad (6.25)$$

$$m^l_{i,j,k} \le Bx^l_{i,j,k} \qquad \forall (i,j)\in V\times V, \forall l\in\mathcal{P}_{i,j} \quad (6.3)$$

$$\sum_{j\in V}\sum_{l\in\mathcal{P}_{0,j}} x^l_{0,j,k} \le 1 \qquad (6.4)$$

$$\sum_{j\in V}\sum_{l\in\mathcal{P}_{j,i}} x^l_{j,i,k} - \sum_{j\in V}\sum_{l\in\mathcal{P}_{i,j}} x^l_{i,j,k} = 0 \qquad \forall i\in V \quad (6.5)$$

$$\sum_{j\in V\setminus\{i\}} tf_{i,j,k} - \sum_{j\in V\setminus\{i\}} tf_{j,i,k} \ge \sum_{j\in V\setminus\{i\}}\sum_{l\in\mathcal{P}_{i,j}} T^l_{i,j}x^l_{i,j,k} \qquad \forall i\in V \quad (6.21)$$

$$\sum_{l\in\mathcal{P}_{i,j}} x^l_{i,j,k} \le 1 \qquad \forall (i,j)\in V\times V \quad (6.14)$$

$$\sum_{i\in V} w_{i,k} \le B \qquad (6.27)$$

$$x^l_{i,j,k} \in \{0,1\} \qquad \forall (i,j)\in V\times V, \forall l\in\mathcal{P}_{i,j} \quad (6.16)$$

$$m^l_{i,j,k} \in [0,B] \qquad \forall (i,j)\in V\times V, \forall l\in\mathcal{P}_{i,j} \quad (6.17)$$

$$tf_{i,j,k} \in [0,H] \qquad \forall (i,j)\in V\times V \quad (6.24)$$

$$w_{i,k} \in [0,B] \qquad \forall i\in\mathcal{Z} \quad (6.28)$$

The problem is NP-hard as it contains as a special case the problem of finding a Hamiltonian tour. Indeed, given simple graph $G=(V,A')$, by setting $C^e_{i,j,l}=0$, $\tilde{c}_{i,j,k,l}=0$ for all $(i,j)\in A'$, $C^e_{i,j,l}=1$, $\tilde{c}_{i,j,k,l}=1$ for all $(i,j)\in A\setminus A'$ and $\alpha^k_i=-1$, $\delta^k_i-\gamma^k_i=0$ for all $i\in V$, the Hamiltonian tour on the graph G is also the min-cost cycle with cost equal to $-n$ with n the number of vertices in V.

6.5.5 Solution Algorithm Based on Lagrangian Relaxation

The Lagrangian relaxation can give a lower bound to the original problem as discussed in Section 1.3.3.2 of Chapter 1. Algorithm 8 proposes the Lagrangian relaxation solution scheme. The algorithm goes on iteratively, keeping and updating an upper bound UB and a lower bound LB to the original MG-IRP-EC.

At the beginning, a heuristic method based on the urgency of customers is applied to solve the original problem. If the heuristic can find a feasible solution, the UB is set to the objective of this initial solution. Otherwise, the heuristic gives a set of tours and a list of lost customers. Fictive vehicles are added to make direct deliveries to these lost customers by taking the path with the maximum energy. The value of the UB is set to the sum of total energy costs of the tours given by the heuristic and all the fictive direct shippings.

Initial values of Lagrangian multipliers can have a strong influence on the quality of the lower bound obtained with the Lagrangian relaxation. In our algorithm, all the multipliers are set to zero initially except for the values of α, which are computed by Equation (6.34). In this way, the visits to customers are always profitable in the

Algorithm 8 Lagrange relaxation based decomposition method

```
Solve the initial problem by a heuristic method
Initialisation of the multipliers
LB ← −∞, UB ← +∞
Number of iteration n ← 0
while stop condition not satisfied do
    Solve IA
    for all vehicle k ∈ K do
        Solve CCP
    Get the current solution S^(n) with the combined objectives of IA and CCP E_L^(n)(S^(n))
    Update the multipliers with the current solution S^(n)
    if E_L^(n) > LB then
        LB ← E_L^(n)
    Apply a repairing algorithm to get a feasible solution from the solution of IA and
    CCP and get its objective Ē^(n)
    if Ē^(n) < UB then
        UB ← Ē^(n)
    n ← n + 1
```

relaxed problem at the beginning of the algorithms.

$$\alpha_i^{k(0)} = -\sum_{j \in V \setminus \{i\}} \sum_{l \in \mathcal{P}_{i,j}} C_{i,j}^{e} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.34}$$

In each iteration, problem $(\mathbf{L}(\alpha, \beta, \gamma, \delta))$ is decomposed into sub-problems IA and CCP. By solving IA, a delivery assignment of each customer to each vehicle is obtained. By solving CCP for each vehicle, a routing plan for deliveries to customers is obtained. Then, a sub-gradient method is used to update the multipliers. These values of multipliers will define new coefficients in objective functions for IA and CCP and the algorithm passes to new iteration.

The sum $\mathscr{E}_L^{(n)}$ of objective values of IA and each CCP in each iteration n gives a lower bound to the initial problem. The LB is updated if the current lower bound is better. The solutions obtained by the relaxation can be repaired to get a feasible solution to the original MG-IRP-EC. To repair a solution, the values of the variables x are set to 1 if their values in the current solution of the relaxed problem is 1, otherwise, they are undetermined. A Mixed Integer Linear Programming (MILP) is then solved to get a complete solution to MG-IRP-EC. If a better feasible solution can be found within a certain time limit, the upper bound UB is updated. The algorithm continues in this way until a certain stop condition. In our case, the condition is when $UB = LB$ (which is impossible in practice) or when a total time limit is reached.

According to the Lagrangian duality, inside each iteration, the objective $\mathscr{E}_L(\alpha, \beta, \gamma, \delta)$ as defined by (6.29) is to be minimized in relation to continuous variables $\alpha \in \mathbb{R}$, $\beta \in \mathbb{R}$, $\gamma \in \mathbb{R}^+$ and $\delta \in \mathbb{R}^+$.

The sub-gradient method is a common approach for the solution of such a problem [108]. In our case, the following formulae are applied to update the multipliers:

$$\alpha_i^{k(n+1)} = \alpha_i^{k(n)} + \frac{w_{i,k} - \sum_{r\in\mathcal{R}_i} q_{i,k}^r}{N} \cdot \zeta \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}$$

$$\beta_i^{k(n+1)} = \beta_i^{k(n)} + \frac{\sum_{j\in V}\sum_{l\in\mathcal{P}_{i,j}} x_{i,j,k}^l - \sum_{r\in\mathcal{R}_i} z_{i,k}^r}{N} \cdot \zeta \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}$$

$$\gamma_i^{k(n+1)} = \gamma_i^{k(n)} + \frac{\sum_{r\in\mathcal{R}_i\setminus\{1\}} (\theta_i^{r-1}+1) z_{i,k}^r - \sum_{j\in V\setminus\{i\}} tf_{j,i,k}}{N} \cdot \zeta \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}$$

$$\delta_i^{k(n+1)} = \delta_i^{k(n)} + \frac{\sum_{j\in V\setminus\{i\}} tf_{j,i,k} - \sum_{r\in\mathcal{R}_i} \theta_i^r z_{i,k}^r}{N} \cdot \zeta \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}$$

The value N is the sum of squares of the difference between the left-hand-side and the right-hand-side of the relaxed constraints (6.26), (6.9), (6.22) and (6.23) with the solution in the current iteration. The coefficient ζ is defined by $\sigma \cdot (UB - \mathscr{E}_L^{(n)})$ with UB the best upper bound and $\mathscr{E}_L^{(n)}$ the objective value of the solution in the current iteration. The value σ is set to a value between 0 and 2 and it is halved when the lower bound is not improved during a certain number of iterations.

6.6 Experimentation and Preliminary Results

Some preliminary experimentations are performed to test these solution methods. In this section, we first describe how to generate the test instances from the IRP benchmarks. Then, some results with regard to these methods are presented to show its efficiency and difficulty. In particular, the effectiveness of the "time flow variables" are discussed. Finally, future work for improvements on the solution methods is stated and some interesting problems are introduced for further study.

6.6.1 Instance Preparation

The instances are generated based on the IRP benchmarks with single vehicle on a horizon of 3 hours [16]. First, the energy consumption per unit of mass on each arc is generated according to the same rule as defined in the Section 4.4.1 of Chapter 4 with two types of roads (national and highway).

Then, a randomized waiting time per stop is set on each arc according to the road type of this arc. If the arc is national, the waiting time is set to 1 second per stop. If the arc is a highway, the waiting time is 1 hour on the whole arc. The travel duration on each arc is then computed with the vehicle dynamics, the distance of the arc and the waiting time, assuming a fixed acceleration. A matrix of travel duration as well as a matrix of energy costs per unit of mass are obtained.

In addition, one period in a benchmark instance is considered to be one day with 24 hours. To generate the customer consumption, the customer demands per period in the initial instances are broken down into a random number of consumption events. Each consumption event is associated with an arrival time between 0 and 24 and a randomized quantity. The sum of the quantity of all the consumption events in one period is equal to the demand rate per period in the initial instances. The fleet in the generated instances contains maximum 3 vehicles, which corresponds to one single vehicle for each period in the initial instances.

The multi-graph is obtained from the initial network of customers. As explained in the end of Section 6.1, a Shortest Path Problem (SPP) is solved twice with different objectives. One is to minimize total travel time and the other is to minimize the energy per unit of mass. The solutions with these two objectives give the two extremities of the Pareto front. The problem is finally solved on this multi-graph.

6.6.2 Preliminary Results

The complete model is coded in C++. It is solved using IBM ILOG CPLEX 12.6.1. In 2 hours time limit, the solver is only able to find optimal solutions to instances with 5 customers. Table 6.2 shows the results of CPLEX with 2 hours time limit on a model with time flow constraints and with the MTZ type constraints. The upper and lower bound obtained by the decomposition method with and without time flow variables are reported in Tables 6.3.

The first column in these tables shows the instance number under the format $(a)n(b)$, with (a) the instance set and (b) the number of customers in the instance. The column "LB" indicates the best lower bound obtained after 2 hours of solution. The column "UB" shows the best upper bound after 2 hours of solution with the number of vehicles K actually used in the solution given in parentheses. The column "GAP" presents the optimality gap defined by Equation (6.35). In Table 6.3, the number of iterations after 2 hours of solution is also reported in the column "# iter". In addition, if the initial heuristic can not find a feasible solution, the upper bound in Table 6.3 is a fictive one with a question mark in parentheses.

$$\mathrm{GAP} = \frac{\mathrm{UB} - \mathrm{LB}}{\mathrm{UB}} \tag{6.35}$$

From the results with 5 customers in Table 6.2, one can see that for small instances, model with the MTZ type constraints is more effective. However, for larger instances, the flow formulation seems better. This could be explained by the working mechanism of CPLEX. With flow formulation, CPLEX might recognise and exploit the special network flow structure and is able to apply preprocessing and automatic cuts to reduce the size of the model. It is also observed that the memory used for solving a model with MTZ type constraints is huge compared to the one with time flow constraints.

Compared to the results given by CPLEX in Table 6.2, the lower and upper bounds are still to be improved in the decomposition method (Table 6.3). For instances with 5 customers, both CPLEX and the decomposition method can find the optimal solution. However, the Branch-and-Cut (B&C) by CPLEX can prove the optimality but the gap in the decomposition method is still about 30%. For instances with 10 customers, the gap is about 50% in the decomposition method

Table 6.2: Solutions given by CPLEX for models with and without time flow variables

Inst.	With Time Flow			With MTZ type constraints		
	LB	UB (K)	GAP	LB	UB	GAP
1n5	218.874	218.874 (2)	0	218.874	218.874 (2)	0
2n5	164.015	164.015 (1)	0	164.015	164.015 (1)	0
3n5	281.374	281.374 (2)	0	281.374	281.374 (2)	0
4n5	226.184	226.184 (2)	0	226.184	226.184 (2)	0
5n5	158.914	158.914 (1)	0	158.914	158.914 (1)	0
1n10	530.4486	676.0565 (3)	21.54%	526.0691	644.7315 (2)	18.4%
2n10	425.9035	533.8335 (2)	20.22%	494.1432	506.8906 (2)	2.51%
3n10	322.3254	374.9160 (3)	14.03%	323.2392	370.4111 (3)	12.74%
4n10	419.3095	551.1373 (3)	23.92%	421.6551	544.9040 (3)	22.62%
5n10	508.9303	627.4120 (3)	18.88%	575.0575	618.7176 (3)	7.06%
1n15	651.7868	–	–	662.1170	–	–
2n15	683.1724	–	–	689.8412	–	–
3n15	716.6038	–	–	708.8311	–	–
4n15	608.6757	–	–	613.2173	–	–
5n15	694.8486	–	–	761.1755	–	–

Table 6.3: Bounds obtained by the decomposition method with and without time flow variables

Inst.	With Time Flow				With MTZ type constraints			
	LB	UB (K)	GAP	# iter	LB	UB (K)	GAP	# iter
$1n5$	170.291	218.874 (2)	22.20%	344	160.141	218.874 (2)	26.83%	284
$2n5$	119.663	164.015 (1)	27.04%	300	105.747	164.015 (1)	35.52%	275
$3n5$	199.783	281.374 (2)	29.00%	205	200.391	281.374 (2)	28.78%	220
$4n5$	199.978	226.184 (2)	11.59%	186	202.395	226.184 (2)	10.52%	175
$5n5$	110.820	158.914 (1)	30.26%	288	110.778	158.914 (1)	30.29%	342
$1n10$	396.379	651.748 (2)	39.18%	120	399.328	651.748 (2)	38.73%	99
$2n10$	305.136	583.867 (2)	47.74%	105	303.468	552.450 (2)	45.07%	96
$3n10$	203.765	405.838 (2)	49.79%	146	201.208	382.027 (2)	47.36%	128
$4n10$	362.325	847.365 (3)	57.24%	101	361.483	512.012 (3)	29.40%	97
$5n10$	397.507	667.904 (3)	40.48%	112	394.632	667.904 (3)	40.91%	104
$1n15$	508.565	5833.55 (?)	91.28%	84	511.919	5833.55 (?)	91.22%	89
$2n15$	543.413	1085.60 (3)	49.94%	92	545.759	937.527 (3)	41.79%	124
$3n15$	491.502	5998.66 (?)	91.81%	96	488.931	1300.11 (3)	64.07%	93
$4n15$	484.708	8091.55 (?)	94.01%	79	495.716	8091.55 (?)	93.87%	92
$5n15$	568.785	6007.45 (?)	90.53%	85	581.446	1102.63 (3)	47.27%	105

compared to about 20% by CPLEX. For instances with 15 customers, CPLEX cannot find feasible solution for any of the instances. However, the decomposition method is able to find feasible solutions for 1 out of 5 instances with time flow variables; and for 3 out of 5 instances with the MTZ type constraints. But the lower bounds given by the decomposition method are still not as good as those given by CPLEX.

6.7 Conclusion and Future Works

In this chapter, the MG-IRP-EC is presented. It is closer to the reality with a continuous-time monitoring of inventory levels and continuou-time travel durations on a multi-graph. The link between inventory management and routing is tightened by the fact that the consumption events of the customers happens at the same time as the vehicles travel in the road network and that the selection of routes is dependent both on the travel time and the delivery quantity.

The work presented in this chapter is still on-going and it can be improved in many ways. From the modelling point of view, the multi-graph can be constructed upon a real road network. More intermediate points can be added and a multi-objective shortest path problem can be solved to compute Pareto-optimal paths between each pair of location nodes. For the solution method, the application of Lagrangian relaxation can be improved. The sub-problems are to be studied more in details, especially the solution of the network flow problem with fixed charges. The complexity of the IA sub-problem is still open and it might be solvable by a method based on dynamic programming. The information given by the solution of IA can help the solution of CCP. In addition, this decomposition method can be integrated with a fast heuristic to derive feasible solutions from the solutions of the relaxed problem. The constraints to be relaxed might also be adjusted to facilitate the solution of the sub-problems.

Conclusion

Routing problems with energy consideration have been studied in this thesis. The energy consumption of vehicles in the road network has been estimated according to vehicle dynamics under a predefined speed profile. Special attention has been paid to the number of stops and the travel duration on each road segment. The energy aspect has then been integrated into several routing problems with inventory management, notably, Energy-Efficient Assembly-line Vehicle Supplying Problem (EEAVSP), Inventory Routing Problem with Energy Consideration (IRP-EC) and Multi-Graph Inventory Routing Problem with Energy Consideration (MG-IRP-EC). A real-life Inventory Routing Problem (IRP) has also been studied in details. Both energy and economic cost minimizing objectives have been considered, in order to determine the influence of the former to the later.

First of all, the transported mass variations of vehicles are important decision variables that affect both the energy consumption and the inventory level of each customer. For the integration of this set of variables into the problem, a mass-flow model was first proposed for the EEAVSP. This problem where each delivery component has different mass, was proved to be NP-hard, even with fixed routes. The results have shown that in the most favourable case, as much as half of the total energy in the transportation activities of the raw-material feeding system studied in the EEAVSP can be reduced. This can be achieved by rearranging the delivery activities in time and by coordinating the quantities delivered to each workstation. In general, the consideration of energy does not reduce the economic benefits. In contrary, these two aspects can be mutually improved.

The mass-flow model was then adapted to the classic IRP. This problem is called the IRP-EC. Integrated decisions of routing and inventory management were specifically studied to see the possible influence of the energy consideration on the inventory routing process. It has been shown that a compromise between economic and energy costs is needed. Traffic conditions can have an impact on the energy consumption and influence the distance travelled. In addition, it has been shown that a flexible inventory policy as the Maximum Level (ML) can favour an energy efficient inventory routing.

A real-life IRP was also studied under various business-related constraints. In particular, the inventory management in this IRP is nearly continuous (in hours). It also includes the assignment problem of driver/trailer to a set of shifts (with time accurate to minutes). The customer set is made up of both VMI customers and call-in customers. The existence of time windows on both the driver side and the customer side complicates the problem even more.

Inspired by this real-life IRP, the time aspect was integrated to the IRP-EC by considering a set of consumption events for each customer. A multi-graph representation of the road networks has been defined, which integrates the travel time to the classic IRP, yielding the MG-IRP-EC. In this problem, routing and inventory decisions are both related to the timing of each delivery, and the timing might also induce different energy consumption.

Concerning the solution method, this thesis has mainly focused on exact methods with mathematical programming. The EEAVSP and IRP-EC were solved by commercial solvers with the default Branch-and-Cut (B&C) algorithms.

For the real-life IRP, heuristics have first been proposed to get an initial solution. Then, the Fixed-Sequence Mixed Integer Linear Fractional Programming (FS-MILFP) has been proposed for the optimization of timing and quantity. The real-life problem was also decomposed based on the column generation scheme. The pricing sub-problem is further decomposed into two parts due to the column structure. One is strongly NP-hard and solved by a labelling algorithm; the other is identified as polynomial.

A Lagrangian relaxation based decomposition method has been proposed for the solution of the MG-IRP-EC. The relaxed problem has been decomposed into two sub-problems with fixed visiting costs. One is the Inventory Allocation Problem (IA). It is proved to be NP-hard with a heterogeneous fleet by a reduction from the MinCostFixedFlow problem. The other is the Capacitated Cycle Problem (CCP) for each vehicle. It is NP-hard because it contains the Travelling Salesman Problem (TSP) as a sub-case. Lower bounds have been obtained by solving each sub-problem of the relaxed problem iteratively. Upper bounds can also be obtained by repairing the solution of the relaxed problem.

This thesis gives rise to several future research directions. First, the Lagrangian relaxation based decomposition method for the solution of MG-IRP-EC is still ongoing by the end of the thesis. In this method, the sub-problem solution could be accelerated with a dynamic programming scheme for IA and a B&C scheme for CCP. Other constraints might be relaxed to see whether simpler sub-problems could be obtained while guaranteeing a good lower bound.

In general, more attentions should be paid to the solution approach of the IRP with energy consideration, which makes the IRP much more difficult to solve. One reason for this could be the large combinatorics introduced by energy related variables as the mass-flow variables in the case of IRP-EC. Methods like column generation or Branch–and–Cut–and–Price (B&C&P) could be applied to solve the EEAVSP and the IRP-EC. Meta-heuristics or hybrid methods such as matheuristics might be developed to find a solution of good quality in less computation time. For the timing of delivery activities in MG-IRP-EC, aggregation or disaggregation of the time horizon could be helpful to make the problem tractable while retaining the optimal solution in the searching space.

The methods proposed in this thesis could apply to real instances, by adding demand forecasts of customers as given in the real-life IRP or using geographical locations given by datasets such as OpenStreetMap. On the other hand, the methods to solve real-life IRP need improvements. Dominance rules, valid inequalities or applications of filtering algorithms are to be discovered in the future. In addition, compromise between logistic costs, inventory management costs and energetic costs could be studied. The Multi-Objective Shortest Path Problem (MOSP) with energy minimizing as one of the objectives could be investigated more in details to generate the Pareto front. The influence of a multi-graph representation of the road network to the solution method might also be analysed. Moreover, in order to take the traffic congestion factor into account, the multi-graph can be extended to a time-dependent case, where the time and energy costs of an arc could both depend on the starting time of traversing the arc.

As the information systems are becoming more and more powerful these days, together with the development of the Internet of Things paradigm, the integration of different types of operational research problems could be more and more popular in the future. We believe that the problems studied in this thesis are only a beginning. The integration of problems like vehicle routing, inventory management, production scheduling, time-tabling and lot-sizing etc. could contribute to more agile and energy-efficient logistic chains in the near future.

Appendix A

Fixed-Sequence Mixed Integer Linear Fractional Programming Model for the Real-Life IRP

$$\max \sum_{s\in\mathcal{SH}} \sum_{k\in\{\mathcal{N}_s | i_k^s \in \mathcal{Z}\}} \sum_{t\in\mathcal{H}} q$$
$$-\eta\Big(\sum_{d\in\mathcal{DR}} \sum_{tw\in\mathcal{TW}_d} \sum_{s\in\mathcal{SH}^{tw}} C_d^{time}(art^s_{\mathcal{N}_s+1} - art^s_0 - \sum_{k\in\mathcal{N}_s} \mu_k^s LOD_d)$$
$$+ \sum_{tl\in\mathcal{TL}} \sum_{s\in\mathcal{SH}^{tl}} \sum_{k\in\mathcal{N}_s} C_{tl}^{dis} D_{k,k+1}$$
$$+ \sum_{d\in\mathcal{DR}} \sum_{tw\in\mathcal{TW}_d} \sum_{s\in\mathcal{SH}^{tw}} \sum_{k\in\mathcal{N}_s} C_d^{lo} \mu_k^s\Big) \tag{5.40}$$

Timing of shifts

$$a_d^{tw} \leq art_0^s \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw} \tag{5.2}$$
$$art^s_{\mathcal{N}_s+1} \leq b_d^{tw} \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw} \tag{5.3}$$

Precedence of operations

$$ST_{i_k^s} + T_{i_k^s, i_{k+1}^s} + \mu_{k+1}^s \; LOD_d \leq art^s_{k+1} - art^s_k \quad \forall d \in \mathcal{DR}, \forall s \in \mathcal{SH}^d, \forall k \in \mathcal{N}_s \tag{5.4}$$
$$MIS_d \leq art_0^{s+1} - art^s_{\mathcal{N}_s+1} \qquad \forall d \in \mathcal{DR}, \forall s \in \mathcal{SH}^d \tag{5.5}$$

Timing of operations in minutes

$$60t \; z_k^{s,t} \leq art_k^s \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s, \forall t \in \mathcal{H} \tag{5.6}$$
$$art_k^s \leq 60H + (60(t+1) - 60H) z_k^{s,t} - 1 \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s, \forall t \in \mathcal{H} \tag{5.7}$$
$$art_k^s = 60 \sum_{t\in\mathcal{H}} t \; z_k^{s,t} + \rho_k^s \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \tag{5.8}$$
$$0 \leq \rho_k^s \leq 59 \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \tag{5.9}$$
$$\sum_{t\in\mathcal{H}} z_k^{s,t} = 1, \qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s \tag{5.10}$$

VMI customer time windows

$$a_{i_k^s}^{tw_{i_k^s}} \leq art_k^s \leq b_{i_k^s}^{tw_{i_k^s}} - ST_{i_k^s} \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\} \tag{5.11}$$

Inventory in trailers

$$J_{tl}^{s,k+1} - J_{tl}^{s,k} = -\sum_{t \in \mathcal{H}} q_k^{s,t} \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \tag{5.12}$$

$$J_{tl}^{s+1,0} = J_t^{s,\mathcal{N}_s+1}, \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl} \tag{5.13}$$

$$-B_{tl} \leq q_k^{s,t} \leq 0 \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}_t, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{SO}\} \tag{5.14}$$

$$0 \leq J_{tl}^{s,k} \leq B_{tl} \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \tag{5.15}$$

Inventory in customers

$$I_i^{t+1} - I_i^t = \sum_{s \in \mathcal{SH}} \sum_{k \in \{\mathcal{N}_s | i_k^s = i\}} q_k^{s,t} - R_i^t \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \setminus H \tag{5.16}$$

$$R_{i_k^s} z_k^{s,t} \leq q_k^{s,t} \leq (\overline{I}_{i_k^s} - \underline{I}_{i_k^s}) z_k^{s,t} \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\}, \forall t \in \mathcal{H} \tag{5.17}$$

$$0 \leq q_k^{s,t} \leq \overline{I}_{i_k^s} \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i_k^s \in \mathcal{Z}_{vmi}\}, \forall t \in \mathcal{H} \tag{5.18}$$

$$\underline{I}_i \leq I_i^t \leq \overline{I}_i \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \tag{5.19}$$

$$R_i^{od} f_i^{od} \leq \sum_{s \in \mathcal{SH}} \sum_{k \in \{\mathcal{N}_s | i_k^s = i\}} \sum_{t = a_i^{od}}^{b_i^{od}} q_k^{s,t} \leq R_i^{od} \qquad \forall i \in \mathcal{Z}_{ci}, \forall od \in \mathcal{OD}_i \tag{5.20}$$

Layover pauses

$$\sum_{k \in \mathcal{N}_s} \mu_k^s = 1 \qquad \forall s \in \mathcal{SH}^l \tag{5.21}$$

$$\sum_{k \in \mathcal{N}_s} \mu_k^s = 0 \qquad \forall s \in \mathcal{SH} \backslash \mathcal{SH}^l \tag{5.22}$$

$$\lambda_k^s \leq \mu_k^s \qquad \forall k \in \mathcal{N}_s, \forall s \in \mathcal{SH} \tag{5.23}$$

$$\sum_{j=k}^{\mathcal{N}_s} \sum_{l=j}^{\mathcal{N}_s} T_{i_{j-2}^s, i_{j-1}^s} \mu_l^s + T_{i_{k-1}^s, i_k^s} \lambda_k^s \leq MDD_{d_s} \qquad \forall k \in \{2, \ldots, \mathcal{N}_s + 1\}, \forall s \in \mathcal{SH} \tag{5.24}$$

$$\sum_{j=1}^{\mathcal{N}_s} T_{i^s_{j-1},i^s_j} - (\sum_{j=k}^{\mathcal{N}_s}\sum_{l=j}^{\mathcal{N}_s} T_{i^s_{j-2},i^s_{j-1}}\mu^s_l + T_{i^s_{k-1},i^s_k}\lambda^s_i) \leq MDD_{d_s} \quad \forall k \in \{2,\ldots,\mathcal{N}_s+1\}, \forall s \in \mathcal{SH} \tag{5.25}$$

Variable domains

$$z^{s,t}_k \in \{0,1\} \qquad \forall s \in \mathcal{SH}, \forall k \in \mathcal{N}_s, \forall t \in \mathcal{H} \tag{5.26}$$

$$\mu^s_k \in \{0,1\} \qquad \forall s \in \mathcal{SH}, \forall k \in \mathcal{N}_s \bigcup \{\mathcal{N}_s+1\} \tag{5.27}$$

$$\rho^s_k \in [0,59[\qquad \forall s \in \mathcal{SH}, \forall k \in \{0\} \bigcup \mathcal{N}_s, \tag{5.28}$$

$$\lambda^s_k \in [0,1[\qquad \forall s \in \mathcal{SH}, \forall k \in \mathcal{N}_s \bigcup \{\mathcal{N}_s+1\}, \tag{5.29}$$

$$I^t_i \in \left[\underline{I}_i, \overline{I}_i\right] \qquad \forall i \in \mathcal{Z}_{vmi}, \forall t \in \mathcal{H} \tag{5.30}$$

$$J^{s,k}_{tl} \in [0, B_{tl}] \qquad \forall tl \in \mathcal{TL}, \forall s \in \mathcal{SH}^{tl}, \forall k \in \mathcal{N}_s \tag{5.31}$$

$$q^{s,t}_k \in \left[R_{i^s_k}, \overline{I}_{i^s_o}\right] \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i^s_k \in \mathcal{Z}_{vmi}\} \tag{5.32}$$

$$q^{s,t}_k \geq 0 \qquad \forall s \in \mathcal{SH}, \forall k \in \{\mathcal{N}_s | i^s_k \in \mathcal{Z}_{ci}\} \tag{5.33}$$

$$art^s_k \in \left[a^{tw}_d, b^{tw}_d\right] \qquad \forall d \in \mathcal{DR}, \forall tw \in \mathcal{TW}_d, \forall s \in \mathcal{SH}^{tw}, \forall k \in \mathcal{N}_s \tag{5.34}$$

Appendix B

Complete Model before Lagrangian Relaxation of the Multi-graph Inventory Routing with Energy Consideration

---

$$\text{minimize} \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} m^l_{i,j,k} + W \sum_{k\in\mathcal{K}} \sum_{(i,j)\in V\times V} \sum_{l\in\mathcal{P}_{i,j}} C^e_{i,j,l} x^l_{i,j,k} \tag{6.1}$$

s.t:

Transportation flow balance

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} m^l_{j,i,k} - \sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} m^l_{i,j,k} = w_{i,k} \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K} \tag{6.25}$$

$$w_{i,k} = \sum_{r\in\mathcal{R}_i} q^r_{i,k} \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K} \tag{6.26}$$

Link routing and flow

$$m^l_{i,j,k} \le B x^l_{i,j,k} \qquad \forall k \in \mathcal{K},\ \forall (i,j) \in V \times V,\ \forall l \in \mathcal{P}_{i,j} \tag{6.3}$$

Routing

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{0,j}} x^l_{0,j,k} \le 1 \qquad \forall k \in \mathcal{K} \tag{6.4}$$

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{j,i}} x^l_{j,i,k} - \sum_{j\in V} \sum_{l\in\mathcal{P}_{i,j}} x^l_{i,j,k} = 0 \qquad \forall k \in \mathcal{K},\ \forall i \in V \tag{6.5}$$

Arrival time

$$\sum_{j\in V\setminus\{i\}} tf_{i,j,k} - \sum_{j\in V\setminus\{i\}} tf_{j,i,k} \ge \sum_{j\in V\setminus\{i\}} \sum_{l\in\mathcal{P}_{i,j}} T^l_{i,j} x^l_{i,j,k} \qquad \forall i \in V, \forall k \in \mathcal{K} \tag{6.21}$$

$$\sum_{j\in V\setminus\{i\}} tf_{j,i,k} \ge \sum_{r\in\mathcal{R}_i\setminus\{1\}} (\theta^{r-1}_i + 1) z^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.22}$$

$$\sum_{r\in\mathcal{R}_i} \theta^r_i z^r_{i,k} \ge \sum_{j\in V\setminus\{i\}} tf_{j,i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.23}$$

Link visit and routing

$$\sum_{j\in V} \sum_{l\in\mathcal{P}_{j,i}} x^l_{j,i,k} = \sum_{r\in\mathcal{R}_i} z^r_{i,k} \qquad \forall i \in \mathcal{Z},\ \forall k \in \mathcal{K} \tag{6.9}$$

Link visit and amount of delivery

$$q^r_{i,k} \leq \overline{I}_i z^r_{i,k} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.10}$$

Inventory monitoring

$$I^0_i + \sum_{k \in \mathcal{K}} \sum_{u=1}^{r} q^u_{i,k} - \sum_{u=1}^{r} R^u_i \geq 0 \qquad \forall i \in \mathcal{Z}, \forall r \in \mathcal{R}_i \tag{6.11}$$

$$I^0_i + \sum_{k \in \mathcal{K}} \sum_{u=1}^{r} q^u_{i,k} \leq \overline{I}_i \qquad \forall i \in \mathcal{Z}, \forall r \in \mathcal{R}_i \tag{6.12}$$

Additional constraints

$$\sum_{r \in \mathcal{R}_i} z^r_{i,k} \leq 1 \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.13}$$

$$\sum_{l \in \mathcal{P}_{i,j}} x^l_{i,j,k} \leq 1 \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K} \tag{6.14}$$

$$\sum_{i \in \mathcal{Z}} \sum_{r \in \mathcal{R}_i} q^r_{i,k} \leq B \qquad \forall k \in \mathcal{K} \tag{6.15}$$

$$\sum_{i \in \mathcal{Z}} w_{i,k} \leq B \qquad \forall k \in \mathcal{K} \tag{6.27}$$

Variable domains

$$x^l_{i,j,k} \in \{0,1\} \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K}, \forall l \in \mathcal{P}_{i,j} \tag{6.16}$$

$$m^l_{i,j,k} \in [0,B] \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K}, \forall l \in \mathcal{P}_{i,j} \tag{6.17}$$

$$z^r_{i,k} \in \{0,1\} \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.18}$$

$$q^r_{i,k} \in \left[0, \overline{I}_i\right] \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K}, \forall r \in \mathcal{R}_i \tag{6.19}$$

$$tf_{i,j,k} \in [0,H] \qquad \forall (i,j) \in V \times V, \forall k \in \mathcal{K} \tag{6.24}$$

$$w_{i,k} \in [0,B] \qquad \forall i \in \mathcal{Z}, \forall k \in \mathcal{K} \tag{6.28}$$

Bibliography

[1] Abdelmaguid, T. F., Dessouky, M. M., and Ordóñez, F. Heuristic approaches for the inventory-routing problem with backlogging. *Computers & Industrial Engineering 56*, 4 (2009), 1519–1534. (Cited on pages 28 and 30.)

[2] Adulyasak, Y., Cordeau, J.-F., and Jans, R. Formulations and Branch-and-Cut Algorithms for Multivehicle Production and Inventory Routing Problems. *INFORMS Journal on Computing 26*, 1 (feb 2014), 103–120. (Cited on pages 30, 31, and 32.)

[3] Adulyasak, Y., Cordeau, J.-F., and Jans, R. Benders Decomposition for Production Routing Under Demand Uncertainty. *Operations Research 63*, 4 (2015), 851–867. (Cited on page 28.)

[4] Aghezzaf, E. H., Raa, B., and Van Landeghem, H. Modeling inventory routing problems in supply chains of high consumption products. *European Journal of Operational Research 169*, 3 (2006), 1048–1063. (Cited on pages 27, 29, and 30.)

[5] Air Liquide, EURO, and ROADEF. Inventory Routing Problem description for ROADEF/EURO 2016 Challenge. `http://challenge.roadef.org/2016/files/IRP_AL_Model%20Description%20for%20EURO-ROADEF%20Challenge%20Version%202.2.pdf`, 2016. (Cited on page 93.)

[6] Air Liquide, EURO, and ROADEF. ROADEF/EURO Challenge 2016: Inventory Routing Problem. `http://challenge.roadef.org/2016/en/index.php`, 2016. (Cited on pages 3, 91, and 121.)

[7] Aksen, D., Kaya, O., Salman, F. S., and Tüncel, Ö. An adaptive large neighborhood search algorithm for a selective and periodic inventory routing problem. *European Journal of Operational Research 239*, 2 (2014), 413–426. (Cited on page 29.)

[8] Alkawaleet, N., Hsieh, Y.-F., and Wang, Y. Inventory Routing Problem with CO_2 Emissions Consideration. In *Logistics Operations, Supply Chain Management and Sustainability*. Springer International Publishing, Cham, 2014, pp. 611–619. (Cited on pages 26 and 36.)

[9] Andersson, H., Christiansen, M., and Desaulniers, G. A new decomposition algorithm for a liquefied natural gas inventory routing problem. *International Journal of Production Research 54*, 2 (2016), 564–578. (Cited on pages 26, 27, and 30.)

[10] Andersson, H., Hoff, A., Christiansen, M., Hasle, G., and Løkketangen, A. Industrial aspects and literature survey: Combined inventory management and routing. *Computers and Operations Research 37*, 9 (2010), 1515–1536. (Cited on pages 25, 26, and 27.)

[11] André, M., Hassel, D., and Weber, F.-j. Development of short driving cycles–Short driving cycles for the inspection of in-use cars–Representative European driving cycles for the assessment of the I/M schemes. Tech. Rep. May, INRETS - LEN, Laboratoire Énergie Nuisances, 1998. (Cited on page 68.)

[12] Anily, S., and Federgruen, A. One Warehouse Multiple Retailer Systems with Vehicle Routing Costs. *Management Science 36*, 1 (1990), 92–114. (Cited on pages 26, 27, 29, and 31.)

[13] APOLLO. Advanced planning of linked logistics and operations (apollo). `http://www.apollo.management.soton.ac.uk/prplib.htm`. (Cited on page 36.)

[14] Aras, N., Aksen, D., and Tekin, M. T. Selective multi-depot vehicle routing problem with pricing. *Transportation Research Part C: Emerging Technologies 19*, 5 (2011), 866–884. Freight Transportation and Logistics (selected papers from {ODYSSEUS} 2009 - the 4th International Workshop on Freight Transportation and Logistics). (Cited on pages 15 and 33.)

[15] Archetti, C., Bertazzi, L., Hertz, A., and Speranza, M. G. A Hybrid Heuristic for an Inventory Routing Problem. *INFORMS Journal on Computing 24*, 1 (2012), 101–116. (Cited on pages 27, 31, and 32.)

[16] Archetti, C., Bertazzi, L., Laporte, G., and Speranza, M. A Branch-and-Cut Algorithm for a Vendor-Managed Inventory-Routing Problem. *Transportation Science 41*, 3 (2007), 382–391. (Cited on pages 27, 28, 29, 30, 31, 32, 60, 61, 76, and 146.)

[17] Archetti, C., Bianchessi, N., Irnich, S., and Speranza, M. G. Formulations for an inventory routing problem. *International Transactions in Operational Research 21* (2014), 353–374. (Cited on pages 27, 30, 72, and 74.)

[18] Baita, F., Ukovich, W., Pesenti, R., and Favaretto, D. Dynamic routing-and-inventory problems: a review. *Transportation Research Part A: Policy and Practice 32*, 8 (1998), 585–598. (Cited on page 25.)

[19] Baldacci, R., Battarra, M., and Vigo, D. Routing a Heterogeneous Fleet of Vehicles. In *The Vehicle Routing Problem: Latest Advances and New Challenges*, B. Golden, S. Raghavan, and E. Wasil, Eds. Springer US, Boston, MA, 2008, pp. 3–27. (Cited on page 15.)

[20] Baldacci, R., Mingozzi, A., and Roberti, R. Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. *European Journal of Operational Research 218*, 1 (2012), 1–6. (Cited on pages 17 and 18.)

[21] Balinski, M. L., and Quandt, R. E. On an Integer Program for a Delivery Problem. *Operations Research 12*, 2 (1964), 300–304. (Cited on page 18.)

[22] Bard, J. F., and Nananukul, N. A Branch-and-price Algorithm for an Integrated Production and Inventory Routing Problem. *Comput. Oper. Res. 37*, 12 (Dec. 2010), 2202–2217. (Cited on page 28.)

[23] Barth, M., Younglove, T., and Scora, G. Development of a Heavy-Duty Diesel Modal Emissions and Fuel Consumption Model. Tech. rep., California PATH, 2005. (Cited on page 35.)

[24] Bektaş, T., and Lysgaard, J. Optimal vehicle routing with lower and upper bounds on route durations. *Networks 65*, 2 (2015), 166–179. (Cited on page 138.)

[25] Bektaş, T., and Laporte, G. The Pollution-Routing Problem. *Transportation Research Part B: Methodological 45*, 8 (sep 2011), 1232–1250. (Cited on pages 2, 33, 34, 35, and 37.)

[26] Bell, W. J., Dalberton, L. M., Fisher, M. L., Greenfield, A. J., Jaikumar, R., Kedia, P., Mack, R. G., and Prutzman, P. J. Improving the Distribution of Industrial Gases with and On-Line Computerized Routing and Scheduling Optimizer. *Interfaces 13*, 6 (1983), 4–23. (Cited on pages 23, 26, and 27.)

[27] Benjaafar, S., Li, Y., and Daskin, M. Carbon footprint and the management of supply chains: Insights from simple models. *IEEE Transactions on Automation Science and Engineering 10*, 1 (2013), 99–116. (Cited on page 45.)

[28] Benoist, T., Gardi, F., Jeanjean, A., and Estellon, B. Randomized Local Search for Real-Life Inventory Routing. *Transportation Science 45*, 3 (aug 2011), 381–398. (Cited on page 102.)

[29] Bertazzi, L., and Speranza, M. G. Inventory routing problems: an introduction. *EURO Journal on Transportation and Logistics 1*, 4 (2012), 307–326. (Cited on page 25.)

[30] Bertazzi, L., and Speranza, M. G. Inventory routing problems with multiple customers. *EURO Journal on Transportation and Logistics 2*, 3 (2013), 255–275. (Cited on page 25.)

[31] Blumenfeld, D. E., Burns, L. D., Daganzo, C. F., Frick, M. C., and Hall, R. W. Reducing Logistics Costs At General Motors. *Interfaces 17*, 1 (1987), 26–47. (Cited on page 26.)

[32] Burke, E. K., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., and Qu, R. Hyper-heuristics: a survey of the state of the art. *Journal of the Operational Research Society 64*, 12 (Dec 2013), 1695–1724. (Cited on page 16.)

[33] Burns, L. D., Hall, R. W., Blumenfeld, D. E., and Daganzo, C. F. Distribution Strategies that Minimize Transportation and Inventory Costs. *Operations Research 33*, 3 (1985), 469–490. (Cited on pages 29, 30, and 31.)

[34] Campbell, A. M., Clarke, L. W., and Savelsbergh, M. W. P. Inventory Routing in Practice. In *The Vehicle Routing Problem*, P. Toth and D. Vigo, Eds. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, ch. 12, pp. 309–330. (Cited on pages 26 and 31.)

[35] Campbell, A. M., and Savelsbergh, M. W. P. A Decomposition Approach for the Inventory-Routing Problem. *Transportation Science 38*, 4 (2004), 488–502. (Cited on page 26.)

[36] Cheng, C., Yang, P., Qi, M., and Rousseau, L.-M. Modeling a green inventory routing problem with a heterogeneous fleet. *Transportation Research Part E: Logistics and Transportation Review 97* (2017), 9–112. (Cited on page 36.)

[37] Chien, T. W., Balakrishnan, A., and Wong, R. T. An Integrated Inventory Allocation and Vehicle Routing Problem. *Transportation Science 23*, December 2014 (may 1989), 67–76. (Cited on page 23.)

[38] Chitsaz, M., Divsalar, A., and Vansteenwegen, P. A two-phase algorithm for the cyclic inventory routing problem. *European Journal of Operational Research 254*, 2 (oct 2016), 410–426. (Cited on page 32.)

[39] Christiansen, M. Decomposition of a Combined Inventory and Time Constrained Ship Routing Problem. *Transportation Science 33*, 1 (feb 1999), 3–16. (Cited on page 27.)

[40] Christiansen, M., Fagerholt, K., and Ronen, D. Ship Routing and Scheduling: Status and Perspectives. *Transportation Science 38*, 1 (2004), 1–18. (Cited on page 26.)

[41] Coelho, L. C. Inventory Routing Instances. `http://www.leandro-coelho.com/instances/inventory-routing/`, 2014. (Cited on page 32.)

[42] Coelho, L. C., Cordeau, J.-F., and Laporte, G. Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies 24* (2012), 270–287. (Cited on pages 27, 29, 31, and 32.)

[43] Coelho, L. C., Cordeau, J.-F., and Laporte, G. Thirty Years of Inventory Routing. *Transportation Science 48*, 1 (2013), 1–19. (Cited on pages 26 and 27.)

[44] Coelho, L. C., and Laporte, G. The exact solution of several classes of inventory-routing problems. *Computers and Operations Research 40*, 2 (2013), 558–565. (Cited on pages 30, 31, and 32.)

[45] Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M. M., and Soumis, F. VRP with Time Windows. In *The Vehicle Routing Problem*, P. Toth and D. Vigo, Eds. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, ch. 7, pp. 157–193. (Cited on page 14.)

[46] Cordeau, J. F., Laganà, D., Musmanno, R., and Vocaturo, F. A decomposition-based heuristic for the multiple-product inventory-routing problem. *Computers and Operations Research 55* (2015), 153–166. (Cited on pages 23 and 31.)

[47] Dantzig, G. B., and Ramser, J. H. The Truck Dispatching Problem. *Management Science 6*, 1 (oct 1959), 80–91. (Cited on pages 1 and 14.)

[48] Demir, E., Bektaş, T., and Laporte, G. A review of recent research on green road freight transportation. *European Journal of Operational Research 237* (2014), 775–793. (Cited on pages 35 and 68.)

[49] Demir, E., Bektaş, T., and Laporte, G. The bi-objective pollution-routing problem. *European Journal of Operational Research 232*, 3 (2014), 464–478. (Cited on page 34.)

[50] Desaulniers, G., Desrosiers, J., Erdmann, A., Solomon, M. M., and Soumis, F. VRP with Pickup and Delivery. In *The Vehicle Routing Problem*, P. Toth and D. Vigo, Eds. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, ch. 9, pp. 225–242. (Cited on page 15.)

[51] Desaulniers, G., Desrosiers, J., and Solomon, M. M., Eds. *Column Generation*. Springer US, Boston, MA, 2005. (Cited on page 22.)

[52] Desaulniers, G., Rakke, J. G., and Coelho, L. C. A Branch-Price-and-Cut Algorithm for the Inventory-Routing Problem. *Transportation Science 50*, 3 (aug 2016), 1060–1076. (Cited on pages 32 and 128.)

[53] Desrosiers, J., and Lübbecke, M. E. A Primer in Column Generation. In *Column Generation*. Springer US, Boston, MA, 2005, pp. 1–32. (Cited on page 21.)

[54] Dethloff, J. Vehicle routing and reverse logistics: The vehicle routing problem with simultaneous delivery and pick-up. *OR-Spektrum 23*, 1 (2001), 79–96. (Cited on pages 15 and 33.)

[55] Dinkelbach, W. On Nonlinear Fractional Programming. *Management Science 13*, 7 (mar 1967), 492–498. (Cited on page 109.)

[56] Dong, Y., Pinto, J. M., Sundaramoorthy, A., and Maravelias, C. T. MIP Model for Inventory Routing in Industrial Gases Supply Chain. *Industrial & Engineering Chemistry Research 53*, 44 (2014), 17214–17225. (Cited on page 26.)

[57] Dror, M., and Ball, M. Inventory/routing: Reduction from an annual to a short-period problem. *Naval Research Logistics (NRL) 34* (1987), 891–905. (Cited on page 27.)

[58] Dror, M., Ball, M., and Golden, B. A computational comparison of algorithms for the inventory routing problem. *Annals of Operations Research 4*, 1 (dec 1985), 1–23. (Cited on page 31.)

[59] Dror, M., and Levy, L. A vehicle routing improvement algorithm comparison of a "greedy" and a matching implementation for inventory routing. *Computers & Operations Research 13*, 1 (1986), 33–45. (Cited on page 31.)

[60] Duan, Q., and Liao, T. W. Optimization of replenishment policies for decentralized and centralized capacitated supply chains under various demands. *International Journal of Production Economics 142*, 1 (2013), 194–204. Special Issue on Behavioral Operations Management/Production Economics. (Cited on page 28.)

[61] Dubedout, H., Cung, V.-D., Neagu, N., Dejax, P., and Yeung, T. G. Robust optimization of inventory routing for bulk gas distribution. In *IESM 2011, International Conference on Industrial Engineering and Systems Management* (Metz, France, 2011). (Cited on page 26.)

[62] Dubedout, H., Dejax, P., Neagu, N., and Yeung, T. A grasp for real life inventory routing problem: application to bulk gas distribution. In *9th International Conference of Modeling, Optimization and Simulation - MOSIM'12* (Bordeaux, France, 2012). (Cited on page 26.)

[63] EcoTransIT World Initiative (EWI). Ecological Transport Information Tool for Worldwide Transports Methodology and Data—Update . Tech. rep., IFEU Heidelberg and INFRAS Berne and IVE Hannover, 2014. (Cited on pages 50 and 76.)

[64] Ehmke, J. F., Campbell, A. M., and Thomas, B. W. Vehicle routing to minimize time-dependent emissions in urban areas. *European Journal of Operational Research 251* (2015), 478–494. (Cited on pages 2, 34, and 36.)

[65] Fagerholt, K., Laporte, G., and Norstad, I. Reducing fuel emissions by optimizing speed on shipping routes. *Journal of the Operational Research Society 61*, 3 (2010), 523–529. (Cited on page 34.)

[66] Fathi, M., Alvarez, M. J., Hassani Mehraban, F., and Rodríguez, V. A Multiobjective Optimization Algorithm to Solve the Part Feeding Problem in Mixed-Model Assembly Lines. *Mathematical Problems in Engineering 2014* (2014), 1–12. (Cited on pages ix and 47.)

[67] Federgruen, A., and Zipkin, P. A Combined Vehicle Routing and Inventory Allocation Problem. *Operations Research 32*, 5 (1984), 1019–1037. (Cited on page 30.)

[68] Fish, L. A. Managerial Best Practices to Promote Sustainable Supply Chain Management & New Product Development. In *Applications of Contemporary Management Approaches in Supply Chains*, H. Tozan and A. Ertürk, Eds. InTech, apr 2015, pp. 83–115. (Cited on pages 1 and 45.)

[69] Fisher, M. L. An Applications Oriented Guide to Lagrangian Relaxation. *Interfaces 15*, 2 (Apr. 1985), 10–21. (Cited on page 23.)

[70] Floyd, R. W. Algorithm 97: Shortest Path. *Commun. ACM 5*, 6 (June 1962), 345. (Cited on page 130.)

[71] Franceschetti, A., and Demir, E. A metaheuristic for the time-dependent pollution-routing problem. *European Journal of Operational Research 0* (2016), 1–20. (Cited on pages 15 and 36.)

[72] Franceschetti, A., Honhon, D., Van Woensel, T., Bektaş, T., and Laporte, G. The time-dependent pollution-routing problem. *Transportation Research Part B: Methodological 56* (oct 2013), 265–293. (Cited on pages 2, 33, 34, and 35.)

[73] Franco, C., López-Santana, E. R., and Méndez-Giraldo, G. *A Column Generation Approach for Solving a Green Bi-objective Inventory Routing Problem*. Springer International Publishing, Cham, 2016, pp. 101–112. (Cited on pages 34 and 36.)

[74] Fukasawa, R., He, Q., and Song, Y. A branch-cut-and-price algorithm for the energy minimization vehicle routing problem. *Transportation Science*, May (2015), 1–19. (Cited on page 33.)

[75] Garaix, T., Artigues, C., Feillet, D., and Josselin, D. Vehicle routing problems with alternative paths: An application to on-demand transportation. *European Journal of Operational Research 204*, 1 (2010), 62–75. (Cited on page 130.)

[76] Garcia, G., Rios, M. A., and Ramos, G. A power demand simulator of electric transportation systems for distribution utilities. In *2009 44th International Universities Power Engineering Conference (UPEC)* (Sept 2009), pp. 1–5. (Cited on page 47.)

[77] Garey, M. R., and Johnson, D. S. *Computers and intractability*, vol. 29. W.H. Freeman New York, 2002. (Cited on pages 2 and 51.)

[78] Geoffrion, A. M. Lagrangean relaxation for integer programming. In *Approaches to Integer Programming*, M. L. Balinski, Ed. Springer Berlin Heidelberg, Berlin, Heidelberg, 1974, pp. 82–114. (Cited on page 23.)

[79] Grønhaug, R., Christiansen, M., Desaulniers, G., and Desrosiers, J. A Branch-and-Price Method for a Liquefied Natural Gas Inventory Routing Problem. *Transportation Science 44*, 3 (2010), 400–415. (Cited on pages 26, 27, and 31.)

[80] Guisewite, G. M., and Pardalos, P. M. Minimum concave-cost network flow problems: Applications, complexity, and algorithms. *Annals of Operations Research 25*, 1 (Dec 1990), 75–99. (Cited on page 143.)

[81] Gürbüz, M. Ç., Moinzadeh, K., and Zhou, Y.-P. Coordinated Replenishment Strategies in Inventory/Distribution Systems. *Management Science 53*, 2 (2007), 293–307. (Cited on page 28.)

[82] Gutin, F., and Punnen, A. P., Eds. *The Traveling Salesman Problem and Its Variations (Combinatorial Optimization).* Springer, 2007. (Cited on page 13.)

[83] Hansen, P. Bicriterion Path Problems. In *Multiple Criteria Decision Making Theory and Application: Proceedings of the Third Conference Hagen/Königswinter, West Germany, August 20–24, 1979*, G. Fandel and T. Gal, Eds. Springer Berlin Heidelberg, Berlin, Heidelberg, 1980, pp. 109–127. (Cited on page 130.)

[84] He, Y., Briand, C., and Jozefowiez, N. Inventory routing with explicit energy consumption: A mass-flow formulation and first experimentation. In *Operations Research and Enterprise Systems: 5th International Conference, ICORES 2016, Rome, Italy, February 23-25, 2016, Revised Selected Papers*, B. Vitoriano and G. H. Parlier, Eds. Springer International Publishing, Cham, 2017, pp. 96–116. (Cited on page 53.)

[85] Hewitt, M., Nemhauser, G., Savelsbergh, M., and Song, J. H. A branch-and-price guided search approach to maritime inventory routing. *Computers and Operations Research 40*, 5 (2013), 1410–1419. (Cited on page 31.)

[86] Hillier, F. S., and Lieberman, G. J. *Introduction To Operations Research 7ed*, 7th ed. MCGRAW HILL INC *, 2001. (Cited on page 11.)

[87] IEA. Key Worlds Energy Trends (Excerpt from: World Energy balances). Tech. rep., International Energy Agency (IEA), 2016. (Cited on page 1.)

[88] IEA. World Energy Outlook 2016 (Executive Summary). Tech. rep., International Energy Agency (IEA), 2016. (Cited on page 1.)

[89] IPCC, 2014. *Climate Change 2014: Synthesis Report.* IPCC, Geneva, Switzerland, 2014. Contribution of Working Groups I, II and III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change [Core Writing Team, R.K. Pachauri and L.A. Meyer (eds.)]. (Cited on page 1.)

[90] Irnich, S., and Desaulniers, G. Shortest Path Problems with Resource Constraints. In *Column Generation*. Springer US, Boston, MA, 2005, pp. 33–65. (Cited on page 118.)

[91] Jabali, O., Van Woensel, T., and De Kok, a. G. Analysis of travel times and CO_2 emissions in time-dependent vehicle routing. *Production and Operations Management 21*, 6 (2012), 1060–1074. (Cited on pages 34 and 36.)

[92] Kantor, M. G., and Rosenwein, M. B. The Orienteering Problem with Time Windows. *The Journal of the Operational Research Society 43*, 6 (1992), 629–635. (Cited on page 118.)

[93] Kara, İ., Kara, B. Y., and Yetis, M. K. Energy Minimizing Vehicle Routing Problem. *Combinatorial Optimization and Applications: First International Conference, COCOA 4616* (2007), 62–71. (Cited on pages 33 and 34.)

[94] Kara, İ., Kara, B. Y., and Yetiş, M. K. Cumulative vehicle routing problems. In *Vehicle Routing Problem*. InTech, 2008. (Cited on page 33.)

[95] Katoh, N., and Ibaraki, T. Resource Allocation Problems. In *Handbook of Combinatorial Optimization*, D.-Z. Du and P. M. Pardalos, Eds., vol. 1–3. Springer US, Boston, MA, 1999, pp. 905–1006. (Cited on page 143.)

[96] Kopfer, H. W., Schönberger, J., and Kopfer, H. Reducing greenhouse gas emissions of a heterogeneous vehicle fleet. *Flexible Services and Manufacturing Journal 26*, 1 (2014), 221–248. (Cited on page 33.)

[97] Krumke, S. O., Noltemeier, H., Schwarz, S., Wirth, H.-C., and Ravi, R. Flow Improvement and Network Flows with Fixed Costs. In *Operations Research Proceedings 1998: Selected Papers of the International Conference on Operations Research Zurich, August 31 – September 3, 1998*, P. Kall and H.-J. Lüthi, Eds. Springer Berlin Heidelberg, Berlin, Heidelberg, 1999, pp. 158–167. (Cited on page 142.)

[98] Laporte, G. Fifty Years of Vehicle Routing. *Transportation Science 43*, 4 (2009), 408–416. (Cited on page 1.)

[99] Laporte, G., and Semet, F. Classical Heuristics for the Capacitated VRP. In *The Vehicle Routing Problem*, P. Toth and D. Vigo, Eds. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, ch. 5, pp. 109–128. (Cited on page 16.)

[100] Lawler, E. L., Lenstra, J. K., Rinnooy Kan, A. H. G., and Shmoys, D. B. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley, 1985. (Cited on page 16.)

[101] Le, T., Diabat, A., Richard, J. P., and Yih, Y. A column generation-based heuristic algorithm for an inventory routing problem with perishable goods. *Optimization Letters 7*, 7 (2013), 1481–1502. (Cited on page 26.)

[102] Lin, C., Choy, K., Ho, G., Chung, S., and Lam, H. Survey of Green Vehicle Routing Problem: Past and future trends. *Expert Systems with Applications 41*, 4 (mar 2014), 1118–1138. (Cited on pages 32 and 33.)

[103] Linde Material Handling. Electric Tow Tractor Capacity 6000 kg P 60 Z. `http://www.linde-world.de/mh-products/start.view?dealer=1&app=Transportieren&highlightIndex=5&range=P+60+Z&context=uk&rangeIndex=0&l=en&ippId=p_p60z&type=Schlepper-%5C%2FPlattformwagen`. (Cited on page 61.)

[104] Malandraki, C., and Daskin, M. S. Time Dependent Vehicle Routing Problems: Formulations, Properties and Heuristic Algorithms. *Transportation Science 26*, 3 (1992), 185–200. (Cited on page 15.)

[105] Muguerza, M., Briand, C., Jozefowiez, N., Ngueveu, S. U., Rodríguez, V., and Moris, M. U. A mass-flow MILP formulation for energy-efficient supplying in assembly lines. *ArXiv e-prints* (2015). (Cited on page 53.)

[106] Negawatt. The 2017-2050 négaWatt Scenario. Tech. rep., Association négaWatt, 2017. (Cited on page 1.)

[107] Nemhauser, G. L., and Wolsey, L. A. *Integer and Combinatorial Optimization*. Wiley-Interscience, New York, NY, USA, 1988. (Cited on page 12.)

[108] Pajouh, F. M., and Balasundaram, B. Gradient-Type Methods. In *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010. (Cited on page 146.)

[109] Park, Y.-B., Yoo, J.-S., and Park, H.-S. A genetic algorithm for the vendor-managed inventory routing problem with lost sales. *Expert Systems with Applications 53* (2016), 149–159. (Cited on page 28.)

[110] Prabhu, V. V., Trentesaux, D., and Taisch, M. Energy-aware manufacturing operations. *International Journal of Production Research 53*, 23 (2015), 6994–7004. (Cited on pages 1 and 45.)

[111] Qian, J., and Eglese, R. Fuel emissions optimization in vehicle routing problems with time-varying speeds. *European Journal of Operational Research 248*, 3 (feb 2016), 840–848. (Cited on page 36.)

[112] Raa, B. Fleet optimization for cyclic inventory routing problems. *International Journal of Production Economics 160* (2015), 172–181. (Cited on page 30.)

[113] Raa, B., and Aghezzaf, E.-H. A practical solution approach for the cyclic inventory routing problem. *European Journal of Operational Research 192*, 2 (2009), 429–441. (Cited on page 30.)

[114] Rahimi, M., Baboli, A., and Rekik, Y. Multi-objective inventory routing problem: A stochastic model to consider profit, service level and green criteria. *Transportation Research Part E: Logistics and Transportation Review 101* (2017), 59–83. (Cited on page 36.)

[115] Raidl, G. R., Puchinger, J., and Blum, C. Metaheuristic Hybrids. In *Handbook of Metaheuristics*. Springer US, Boston, MA, 2010, pp. 469–496. (Cited on page 16.)

[116] Rakke, J. G., Andersson, H., Christiansen, M., and Desaulniers, G. A New Formulation Based on Customer Delivery Patterns for a Maritime Inventory Routing Problem. *Transportation Science 49*, 2 (2015), 384–401. (Cited on page 28.)

[117] Savelsbergh, M., and Song, J. H. Inventory routing with continuous moves. *Computers and Operations Research 34* (2007), 1744–1763. (Cited on pages 28, 29, and 30.)

[118] Savelsbergh, M., and Song, J. H. An optimization algorithm for the inventory routing problem with continuous moves. *Computers and Operations Research 35* (2008), 2266–2282. (Cited on pages 26, 27, and 30.)

[119] Semet, F., Toth, P., and Vigo, D. Classical Exact Algorithms for the Capacitated Vehicle Routing Problem. In *Vehicle Routing*. Society for Industrial and Applied Mathematics, Philadelphia, PA, nov 2014, ch. 2, pp. 37–57. (Cited on pages 17 and 19.)

[120] Shao, Y., Furman, K. C., Goel, V., and Hoda, S. A hybrid heuristic strategy for liquefied natural gas inventory routing. *Transportation Research Part C: Emerging Technologies 53* (2015), 151–171. (Cited on pages 26 and 31.)

[121] Song, J.-H., and Savelsbergh, M. Performance Measurement for Inventory Routing. *Transportation Science 41*, 1 (2007), 44–54. (Cited on page 26.)

[122] Soysal, M., Bloemhof-Ruwaard, J. M., and Bektaş, T. The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. *International Journal of Production Economics 164*, August 2013 (jun 2015), 366–378. (Cited on page 36.)

[123] Soysal, M., Bloemhof-Ruwaard, J. M., Haijema, R., and van der Vorst, J. G. A. J. Modeling a green inventory routing problem for perishable products with horizontal collaboration. *Computers and Operations Research* (2016), 1–15. (Cited on pages 26, 33, 34, and 36.)

[124] Toth, P., and Vigo, D. An Overview of Vehicle Routing Problems. In *The Vehicle Routing Problem*, P. Toth and D. Vigo, Eds. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2001, ch. 1, pp. 1–26. (Cited on page 14.)

[125] Toth, P., and Vigo, D., Eds. *The Vehicle Routing Problem (Monographs on Discrete Mathematics and Applications)*. SIAM, 2001. (Cited on pages 14 and 16.)

[126] Toth, P., and Vigo, D. Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics 123*, 1–3 (2002), 487–512. (Cited on page 17.)

[127] Toth, P., and Vigo, D., Eds. *Vehicle Routing*. Society for Industrial and Applied Mathematics, Philadelphia, PA, nov 2014. (Cited on page 14.)

[128] Treitl, S., Nolz, P. C., and Jammernegg, W. Incorporating environmental aspects in an inventory routing problem. A case study from the petrochemical industry. *Flexible Services and Manufacturing Journal 26*, 1–2 (jun 2014), 143–169. (Cited on pages 34 and 36.)

[129] Vansteenwegen, P. The cyclic inventory routing problem. `http://www.mech.kuleuven.be/en/cib/op#section-33`, 2014. (Cited on page 32.)

[130] Vansteenwegen, P., and Mateo, M. An iterated local search algorithm for the single-vehicle cyclic inventory routing problem. *European Journal of Operational Research 237*, 3 (2014), 802–813. (Cited on page 32.)

[131] Vidal, T., Crainic, T. G., Gendreau, M., and Prins, C. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research 231*, 1 (2013), 1–21. (Cited on page 16.)

[132] Wagelmans, A., Hoesel, S. V., and Kolen, A. Economic Lot Sizing: An $O(nlogn)$ Algorithm That Runs in Linear Time in the Wagner-Whitin Case. *Operations Research 40* (1992), S145–S156. (Cited on page 143.)

[133] Walkowicz, K., Duran, A., and Burton, E. Fleet DNA Project Data Summary Report. `http://www.nrel.gov/transportation/fleettest_fleet_dna.html`, 2014. Accessed: 2015-3-20. (Cited on page 76.)

[134] Xiao, Y., and Konak, A. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. *Transportation Research Part E: Logistics and Transportation Review 88* (apr 2016), 146–166. (Cited on pages 33, 34, and 36.)

[135] Xiao, Y., Zhao, Q., Kaku, I., and Xu, Y. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. *Computers and Operations Research 39*, 7 (2012), 1419–1431. (Cited on page 33.)

[136] You, F., Pinto, J. M., Capón, E., Grossmann, I. E., Arora, N., and Megan, L. Optimal distribution-inventory planning of industrial gases. I. Fast computational strategies for large-scale problems. *Industrial and Engineering Chemistry Research 50*, 5 (2011), 2910–2927. (Cited on page 26.)