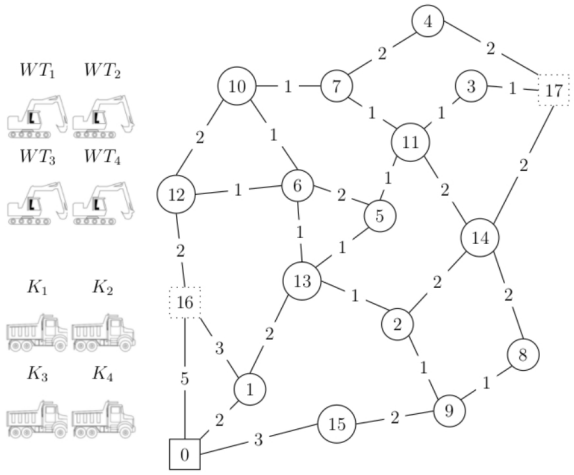
Models and Methods for the Integrated Scheduling Routing Problem Applied to Major Disasters

Guilherme de Castro Pena

Thèse
de doctorat
de l'UTT

Guilherme DE CASTRO PENA

Models and Methods for the Integrated Scheduling Routing Problem Applied to Major Disasters



Champ disciplinaire :
Sciences pour l'Ingénieur

2021TROY0033

Année 2021

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

THESE

pour l'obtention du grade de

DOCTEUR
de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

en SCIENCES POUR L'INGENIEUR

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Guilherme DE CASTRO PENA

Le 25 octobre 2021

Models and Methods for the Integrated Scheduling Routing Problem Applied to Major Disasters

JURY

M. A. YASSINE	PROFESSEUR DES UNIVERSITES	Président
M. O. PÉTON	PROFESSEUR DES UNIVERSITES	Rapporteur
M. M. SEVAUX	PROFESSEUR DES UNIVERSITES	Rapporteur
M. C. SATOSHI SAKURABA	PROFESSOR ADJUNTO	Examinateur
M. C. PRINS	PROFESSEUR DES UNIVERSITES	Directeur de thèse
Mme A. C. SANTOS	PROFESSEURE DES UNIVERSITES	Directrice de thèse

Acknowledgements

This PhD project has been a very important milestone in my life, an experience that has strengthened me professionally and personally.

I deeply thank my supervisors Andréa Cynthia Santos Duhamel and Christian Prins who shared with me all their knowledge and experience within every advice and suggestion. I thank them for for the great support they gave me both in the internal and external environments of the university. Furthermore, I thank them for their kindness, great professionalism and for the opportunity to be part of this project, because it was been an immeasurable honor to have worked with you. I can assure that I keep every piece of advice with me and I will gladly take and replicate them in my lifetime.

I also thank the members of the jury, the professors Marc Sevaux, Olivier Péton, Adnan Yassine and Celso Satoshi Sakuraba, who accepted to participate and evaluate this work, thus providing me with valuable contributions and suggestions for the improvement of our research.

I thank my parents, my brother, all my family and friends in Brazil, my girlfriend Thais and her family, who despite the distance, keep giving me precious support. In special to Thais for the great support every day, for having lived part of this story by my side, your presence is always very important to me, I love you very much. I would also like to thank all my friends from *Université de Technologie de Troyes* and the city of Troyes, who have become my family in France during this period, helping my adaptation and providing excellent moments of indispensable leisure during these years, especially the Team Chouffe.

Thanks to all the professors and the administrative team at *Université de Technologie de Troyes*, Pascale Denis, Isabelle Leclercq and Thérèse Kazarian and especially to Véronique Banse and Bernadette André from LIST3N, for all precious help during this period.

Finally, I would like to thank the *Université de Technologie de Troyes*, the *Universidade Federal de Viçosa*, the *Université Le Havre Normandie* and especially

the financial support of the Allocation *RÉGION GRAND EST* and *CONSEIL DEPARTEMENTAL DE L'AUBE* under the project TRIDE, where this research would not have been possible without their support.

Contents

List of figures

List of tables

List of Algorithms

Acronyms

BI Best Improvement

CARP Capacitated Arc Routing Problem

CSRP Crew Scheduling and Routing Problem

DP Dynamic Programming

FI First Improvement

1-FTPDP one-commodity Full Truckload Pickup and Delivery Problem

FTPDP-mVS Full Truckload Pickup and Delivery Problem with a multiple vehicle synchronization

FTPDP-RS Full Truckload Pickup and Delivery Problem with Resource Synchronization

FTVRP Full Truckload Vehicle Routing Problem

GCH Greedy Constructive Heuristic

GRASP Greedy Randomized Adaptive Search Procedure

ICSMD International Charter on Space and Major Disasters

ILS Iterated Local Search

LNS Large Neighborhood Search

L-CVRP Location-Capacitated Vehicle Routing Problem

LP Linear Programming

LTSP Log-Truck Scheduling Problem

MILP Mixed Integer Linear Programming

MDVRPI Multi-Depot Vehicle Routing Problem with Inter-Depot Routes.

NRCSRP Network Repair Crew Scheduling and Routing Problem

OPM Open Pit Mining

PVRP-IF Periodic Vehicle Routing Problem with Intermediate Facilities

PDP Pickup and Delivery Problems

SRP-CD Scheduling vehicle Routing Problem to clean debris

RCH Random Constructive Heuristic

RNPD Road Network Problem with Disruptions

SA Simulated Annealing

SDVRP Split Delivery Vehicle Routing Problem

SDCP Stochastic Debris Clearance Problem

RCPSP Resource-Constrained Project Scheduling Problem

RCPSPR Resource-Constrained Project Scheduling Problem with Routing

RNAP Road Network Accessibility Problem

VRP Vehicle Routing Problem

VRPMSs Vehicle Routing Problems with Multiple Synchronization constraints

WT Work-Troop

WSP Work-troops Scheduling Problem

Chapter 1

Introduction

1.1 General context

The accumulation of debris in urban regions is a concern, especially after big public engineering works or major disasters. It requires a great effort to perform cleaning operations when there is a big amount of debris. The operations to remove debris performed by big public engineering works can be planned in advance, while in the case of major disasters, the amount of debris and the removal planning can become very complex since an unexpected and large urban area can be affected.

In recent years, the world has been stricken by several natural major disasters or industrial accidents, such as the India floods, 2020; rupture of dams in Brazil, 2019; hurricanes in USA, 2018; ocean earthquake and tsunami in Indonesia, 2018; earthquakes in Nepal, 2015; tsunamis and floods in Japan, 2011; nuclear explosion in Japan, 2011; earthquakes in Haiti, 2010, among others. According to recent surveys, 55% of the world population lives in urban areas, around 4.2 billion people, with a projection of 68% by the year of 2050 [United Nations, 2018]. When such disasters hit inhabited regions, they strongly impact population, causing displacement, homelessness and fatalities; environment by modifying ecosystems, occasioning damages such as desertification, etc; and resulting in interruptions on urban infrastructures and the network services like urban transportation, water and electricity services. As a consequence, such impacts imply a huge cost of humanitarian aid and reconstruction [EM-DAT, 2014]. In this context, an effort is required to

improve post disasters operations. Therefore, the cleaning operations are an important phase in the reconstruction process for inhabitants to recover from their effects.

In general, a disaster can be divided in four phases as defined by the authors Altay and Green III [2006]:

- **Mitigation** is related to a set of preventive actions that could decrease the impacts of a disaster or prevent its happening;
- **Preparedness** details actions, taught before a catastrophic event, that the population should follow in the situation of a disaster;
- **Response** concerns the use of resources and the activation of emergency plans as an immediate response to the disaster to avoid worse consequences for the population, environment, infrastructures and economy;
- **Recovery** involves the planning of long-term actions related to the period after the immediate impact of a disaster, in order to help the population to restore to a normal life.

The operations of cleaning debris, focused in this thesis, are present in the recovery phase, where they can take months or years after a catastrophic event. For instance, Figures 1.1 depict Port-au-Prince city, Haiti, before (Figure 1.1-(a)) and after (Figure 1.1-(b)) the earthquake in 2010. After the event, approximately 400,000 buildings were affected, where 80,000 had suffered serious damage [UNDP, 2013]. In total, an amount of 10 million cubic meters of debris has been estimated by the authorities. In Port-au-Prince, the operations for removal and transportation of debris moved them to discharge areas, called landfills, or recycling stations. Figure 1.2 illustrates the evolution of an area used as a landfill in Port-au-Prince.

This thesis investigates the cleaning operations, removal and transportation of debris, after major disasters in the recovery phase by means of optimization. The operations are considered as an integrated multi-period Scheduling vehicle Routing Problem to clean debris (SRP-CD) in an urban area, where strategical and operational levels are addressed. More precisely, the problem of optimizing the cleaning

(a) (b)

Figure 1.1: A same region in Port-au-Prince, Haiti, (a) before and (b) after the earthquake. Source: Google Earth [Gorelick et al., 2017].

(a) (b) (c)

Figure 1.2: Progress of a landfill area in Port-au-Prince, Haiti. Source: Google Earth [Gorelick et al., 2017].

operations in an urban area as fast as possible is investigated, considering limited resources such as Work-Troops (WTs) and a fleet of homogeneous dump trucks responsible to transport debris to landfills. Each Work-Troop (WT) is considered as a single entity composed by heavy machinery such as excavators, bulldozers and the human resources required to manage such equipment. The overall cleaning area can be modeled by means of a graph containing a node to represent a depot with the vehicles (WTs and trucks), debris nodes and landfill areas; and arcs representing the transportation network which connect the nodes. The operations are performed in debris nodes by assigning WTs, by assigning and loading dump trucks, synchronizing WTs and dump trucks, and between dump trucks, and operations in landfill nodes to unload dump trucks. In addition, these operations are indexed in the two scales of time: a limited working day and a global time horizon in days.

From the strategical point of view, the major decision concerns the assignation of WTs to debris nodes in the time horizon. In the operational context, the dump trucks will perform multi-trips each day to move debris from their origin to landfills. Therefore, the goal is twofold: minimizing the total time, in working days, to perform the events for cleaning the overall area, in the strategical level; and minimizing the total costs of vehicle routes in the operational level.

The integration of different levels of decision in optimization problems is a growing trend in the scientific literature. An interesting entry point on the integration of Vehicle Routing Problems (VRP) is the study of Coelho et al. [2016]. Examples of successful applications for integrated optimization problems are Coelho et al. [2014] and Prodhon and Prins [2014]. In the former, the management inventory was integrated to the distribution of products to customers in a time horizon. In the latter, the location of additional depots are integrated to the distribution for customers. Integrated problems often deal with different optimization levels: strategical, tactical and operational; hence, they are a challenging field since the global complexity can increase according to the complexity of each single problem. As showed in Prodhon and Prins [2014], the gains of addressing a problem in an integrated way are about of 15%. In practical terms, integrated optimization problems allow working with more realistic scenarios by including relevant constraints involved in the real application. As it is the case in SRP-CD, two levels of synchronization between the assigned WTs and dump trucks and between dump trucks are considered, along with the limited working days and service times for operations. For sake of clarity, WT and dump trucks are synchronized since dump trucks may not visit nodes without WTs. The synchronization between two dump trucks occurs if a dump truck is being loaded at a debris node. Then, the other dump trucks that visit the same node, at the same moment, have to wait the current operation to finish. These synchronizations are stated based on the study of Drexl [2012].

The SRP-CD is related to three classical optimization problems: the Resource-Constrained Project Scheduling Problem (RCPSP) in the strategical level, where, within a project, tasks with resources requirements and processing times need to be completed in such a way that the overall project completion date is minimal

[Brucker et al., 1999; Hartmann and Briskorn, 2010; Pritsker et al., 1969]; and two vehicle routing problems according to the amount of debris can be addressed in the SRP-CD operational level: the Full Truckload Vehicle Routing Problem (FTVRP) [Desrosiers et al., 1988] and the Split Delivery Vehicle Routing Problem (SDVRP) [Archetti and Speranza, 2008; Golden et al., 2008]. The former consists in designing routes for trucks in order to make a number of fully load trips between locations. The latter aims at designing routes for vehicles in order to visit customers, where the demand of each customer is lower than or equal to the capacity of a truck. Each customer can be visited more than once, and each demand can be divided in different vehicles.

The amount of debris to be removed plays a key role on the operational level and define which optimization problem should be used to model the situation. Three cases can be distinguished:

(1) the first case happens when the amount of debris is very important, and the problem aims to assign resources, close to a RCPSP integrated to FTVRP. One may note that in this case, the possible gains of using the SDVRP is not enough to have an impact on the number of days to clean the overall area, neither on the total costs. This was the case of Port-au-Prince earthquake in 2010, where the cleaning of the city have taken years [UNDP, 2013].

(2) the second case appears when the number of full truck trips are not high and the use of SDVRP can bring gains in terms of total costs. To our knowledge, the impact of this case considering the minimization of days to clean the affected area is still an open question.

(3) the last case occurs when the amount of debris is smaller than or equal to the vehicle capacity. Thus, the problem applies the RCPSP integrated to SDVRP. An example of disaster with this configuration is the Texas hurricane in 2017, where several set of sites with a small-medium amount of debris can be identified in the overall damaged area [Sayarshad et al., 2020].

The cases specified above rely on $\mathcal{NP}$-Hard problems as showed by Archetti et al. [2011]; Blazewicz et al. [1983]. Characteristics of the RCPSP appear in

the strategical level of the SRP-CD, where the project could represent the overall target area to be cleaned and the tasks could be seen as the cleaning operations in each debris node. The periods could be considered as the working days and the main objective in the SRP-CD is to find the minimal makespan, similar to the RCPSP. Without taking the operational level into consideration and supposing that the number of dump trucks is unlimited, the assignment of WTs in debris nodes and the transportation of debris from their origin to landfills transforms the SRP-CD into a problem close to the RCPSP, which is $\mathcal{NP}$-Hard [Blazewicz et al., 1983]. Nevertheless, the SRP-CD is more complex because the number of dump trucks is limited and routing decisions are considered in a second optimization level. From these assumptions, it is considered that the SRP-CD is $\mathcal{NP}$-Hard. Optimal solutions can be obtained for small-size instances within an acceptable running time. However, when the number of WTs, dump trucks and debris nodes increases, then the combinatorial level of the SRP-CD also raises. This way, larger instances with more debris nodes and more vehicles need to be executed in methods such as heuristics or metaheuristics. A schema of the three cases (1), (2) and (3) in terms of debris is given in Figure 1.3.

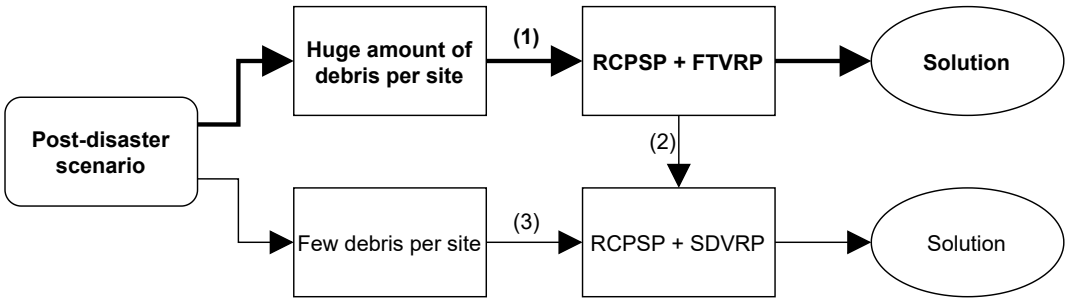


Figure 1.3: Classification of SRP-CD according to the amount of debris.

This thesis is focused on the first case (1) since the volume of debris from catastrophes is important, which can increase the time and costs to perform the cleaning operations. Thus, the effort of the set of vehicles is high and a good strategy to assign WTs and dump trucks is required to find solutions that lead to a faster reconstruction process of the affected area. In practical terms, this enables inhabitants to recover faster from their effects; and, in technical terms, there was a lack

of studies dedicated to this particular optimization problem. The main contributions include the integration of the strategical (scheduling) and operational levels (vehicles routing) of the SRP-CD optimization problem, motivated by removing debris after major disasters, and several approaches for the SRP-CD through a new mathematical model, heuristics and metaheuristics. The model is based on a multi-flow formulation [Ahuja et al., 1989], and making use of a dynamic flow [Aronson, 1989; Wang, 2018]. Thus, it is indexed over two scales of time, corresponding to the time horizon and each working day; and including two levels of synchronizations for work-troops and dump trucks, and between dump trucks. The proposed algorithms are constructive heuristics and metaheuristics based on Large Neighborhood Search (LNS) and a hybrid metaheuristic based on LNS and Simulated Annealing (SA). In addition, a data benchmark of interesting instances for SRP-CD is described along with an instance generation mechanism. Moreover, a case study for the earthquake that struck Port-au-Prince, Haiti in 2010 is also provided. To the best of our knowledge, these are the first approaches, focusing on this specific integrated problem with all the constraints mentioned, in the context of disasters.

1.2 Thesis structure

This manuscript is organized in four major chapters. The first one is dedicated to a general review of the literature. In the second chapter, the formal formulation of the SRP-CD is defined. In the sequel, the formal definitions of the approaches are detailed. The last chapter is devoted to the presentation of the numerical results. The structure ordered by chapters is as follows:

- Chapter 2 is dedicated to a general bibliography review on the classical problems related to the SRP-CD in Section 2.1. A review of works concerning applications related to SRP-CD are presented in Section 2.2. Additionally, the position of this thesis is presented in Section 2.3.
- In Chapter 3 is described the formal definition of the SRP-CD in Section 3.1. In Section 3.2 is presented the mathematical formulation.

- The Chapter 4 is devoted to the algorithms proposed in this thesis. In Section 4.1 is introduced the constructive heuristics. The LNS-based metaheuristics are defined in Section 4.2. The hybrid metaheuristic based on LNS and SA is presented in Section 4.3.

- In Chapter 5 is described the results through computational experiments. The data generation mechanism is showed in Section 5.1. Numerical and comparison results are described in Section 5.2. The case study of Haiti earthquake results are reported in Section 5.2.6.

- Finally, in Chapter 6, the main contributions and perspectives for future research are given.

Chapter 2

Literature review

In this chapter, SRP-CD related problems, involving classical optimization problems and variants of *Vehicle Routing* and *Scheduling* are reviewed. Moreover, a review on different kind of applications applied to similar problems in crisis logistics in urban areas and *Open Pit Mining* are done. Then, an overview provides similarities and differences concerning the SRP-CD from the existing studies from the literature.

2.1 Related classical optimization problems

The SRP-CD considers aspects of different known classical optimization problems both in the strategical and operational levels. For the former, it can be mentioned the RCPSP, while for the latter, vehicle routing problems are closely related such as FTVRP, SDVRP, Pickup and Delivery Problems (PDP), and Multi-Depot Vehicle Routing Problem with Inter-Depot Routes (MDVRPI).

As described in Brucker et al. [1999]; Hartmann and Briskorn [2010]; Pritsker et al. [1969], the Resource-Constrained Project Scheduling Problem (RCPSP) can be defined as a project containing a set of tasks, which must be concluded without preemption. Each task has resource requirements to be performed, a duration, and some of them may have precedence relations. Objectives like minimizing the completion time or the makespan are usually used. The models look for finishing all tasks of the project such that resources and precedence constraints are satisfied.

Compared to SRP-CD, the RCPSP appears in the strategical level, where the project represents the overall target area to be cleaned and the tasks correspond to the cleaning operations at debris nodes. This way, the WTs could be the resources that are assigned to process tasks (clean the debris nodes). The periods could be considered as the working days and there is no due date for each debris node. However, once the nodes must be cleaned within a limited working day, the number of days to complete the process can vary according to the dump trucks' routes. The idea of precedence appears in variations of the problem when there are blockage on the road network. This way, each blocked node would have a set of predecessors. Nevertheless, this variant of the SRP-CD is not focused on this thesis. The main objective in the SRP-CD is to find the minimal makespan, similar to the RCPSP. Some differences exist between RCPSP and SRP-CD such as the integration with VRP in SRP-CD, and the fact that SRP-CD is modeled on a graph based on the road network.

The Full Truckload Vehicle Routing Problem (FTVRP) [Desrosiers et al., 1988] related to the operational level of SRP-CD was defined as follows: given a set of vehicles, the problem consists in defining routes containing trips where, on each trip, identical vehicles travel and transport loads from their origin to their destination, routes start and end at the depot and they do not exceed a prespecified time limit. On each trip, the vehicles are fully loaded. The objective is to minimize the total distance travelled, such that the vehicle capacity constraint is satisfied. Some variants of the FTVRP may impose constraints in maximum route length, time windows and vehicle capacities. The differences between the routing in SRP-CD and the FTVRP are: (1) vehicles can only visit debris nodes with WTs assigned, (2) the pairs of load points and unload points are not fixed in SRP-CD, which arises the problem complexity, (3) the loading and unloading operations have fixed service time each to be performed, which affect the routes and (4) the objective in SRP-CD aims to perform all the cleaning operations as fast as possible in number of days, and minimizing the total travel time for the trucks.

In the Split Delivery Vehicle Routing Problem (SDVRP) [Archetti and Speranza, 2008], a fleet of vehicles serves a set of customers by starting and ending their

route at the depot, but the goods can be delivered (loaded) by different vehicles. The total goods might be greater than the vehicle capacity. The problem aims to serve all customers, without exceeding the vehicle capacity and minimizing the total distance of routes. Both the FTVRP multi-trips and SDVRP allow a customer to be served by several vehicles, the difference relies on the fact that in the SDVRP, the demands can be decomposed in small parts and distributed in different vehicles. Despite the fact that SDVRP can also includes full truckloads, the main similarity to the SRP-CD lies in the cases where the amount of debris per site are smaller than the capacity of a single truck. In these cases, the truck should visit different sites to complete its capacity before moving to a landfill node. Concerning the dissimilarities, we can mention the fact that in the SRP-CD, a debris node can be visited by dump trucks if and only if a WT is assigned to the corresponding debris node. Moreover, the SRP-CD has landfills, and a limitation of hours on the working day.

The Pickup and Delivery Problems (PDP) was referred in Parragh et al. [2008b,a] as a special variant of VRP, where a vehicle performs pickups and deliveries within a time period, and the vehicle starts and ends its route at a depot. The authors showed others variants including multiple vehicles, multiple depots, paired or unpaired pickup and delivery nodes, time windows, etc. The one-commodity Full Truckload Pickup and Delivery Problem (1-FTPDP) described in Gendreau et al. [2015] considers trucks that travel with full load once at each pair pickup-delivery. In comparison to the SRP-CD, in the operational level, debris and landfills could be seen as pickup and delivery nodes. However the SRP-CD is more complex than the PDP variants since the former includes simultaneously synchronization between WTs and dump trucks, synchronization between dump trucks, loading and unloading times in a limited working day and multiple trips. The SRP-CD could be classified as several static pickup and delivery problems along the working days, but the transformation from SRP-CD to PDP could imply additional costs or loss of information in the original description of the integrated problem.

The Multi-Depot Vehicle Routing Problem with Inter-Depot Routes (MDVRPI) introduced by Crevier et al. [2007] extends the VRP by including the idea of multi-

depots and intermediate depots. In the problem, a fleet of vehicles serves a set of customers, where each customer was visited once to fulfill the corresponding demand with a service time. The vehicles start and end their routes in the same depot and the duration of each route could not exceed a predefined working day. Nevertheless, the vehicles were allowed to visit intermediate depots along their routes in order to replenish their capacity. The goal was to minimize the total duration of the routes. In Angelelli and Speranza [2002], a similar problem was described, and called the Periodic Vehicle Routing Problem with Intermediate Facilities (PVRP-IF). In addition to the constraints of the MDVRPI, the PVRP-IF includes a time horizon in working days and a list of visiting schedules for each customer. Each customer could be visited at most once per working day and the objective is to minimize the total length of the routes. For further details on intermediate depots, we refer the interested reader for the study of Guastaroba et al. [2016]. Compared to the SRP-CD, the similarities comprise the customers as debris nodes, the intermediate depots/facilities as landfills, the service time, the working day and the objective for the operational level. In spite of the similarities, the SRP-CD differs from those problems since the dump trucks can only visit debris nodes (customers) with assigned WTs and the nodes can be visited more than once every working day. The PVRP-IF schedules the customers' visits using pre-defined dates, while the SRP-CD does not consider this aspect.

The study of Lacomme et al. [2019] addressed the RCPSP with routing (RCPSPR), where in the RCPSP each activity of a project requires an amount of resources to be performed. Trucks performed pickup and delivery operations of resources between activities, and the goal was to minimize the makespan. The activities could be served by different vehicles, subject to precedence constraints. The authors dealt with only one resource. A Mixed Integer Linear Programming (MILP) model, a constraint programming formulation, and a hybrid heuristic, which couples a Greedy Randomized Adaptive Search Procedure (GRASP) and an evolutionary local search, were proposed. The experiments include a set of small-size instances with less than 8 activities and a set of medium-size instances with about 30 activities. As results, the hybrid heuristic was able to find solutions

with an average gap of 3.3% compared to optimal solutions for small instances. In relation to SRP-CD, the RCPSPR is different due to the following dissimilarities. The RCPSPR does not describe WTs-like entities for the activities, the minimization of travel time for trucks is not a goal, the pickup and delivery operations are performed between activities without loading or unloading times, there are no discharge points like landfills and there is no limited working days.

An application in a biomass supply chain was found in Soares et al. [2019]. The problem coupled a FTPDP with a multiple vehicle synchronization (FTPDP-mVS). Three types of vehicles were used (trucks, loaders and lorries) to perform the material transportation from pickup nodes to delivery ones within a planning horizon of a single working day. The problem had multi-depots and two types of synchronization were done: the first happens between lorries to pick-up and drop-off loaders at pickup nodes, and the second occurs between trucks and pickup nodes with loaders. According to the demand, multiple pickups and deliveries are allowed at each node and idle times might exist for vehicles at pickup nodes. The objective was to minimize the overall cost that couple transportation, fixed costs for vehicles, distance and time costs. A multi-flow based mathematical model was provided, where the set of nodes was expanded into artificial sets to design multi-trips and the synchronizations were represented by the sum of arrival times. A matheuristic using a fix-and-optimize algorithm coupled with a variable neighbourhood decomposition search was proposed. The methods were tested over 19 instances varying pickup/delivery requests and number of vehicles. The mathematical model obtained gaps from 4.4% to 33%. Although, this application contains several close related issues with the operational decisions of SRP-CD, one may note that it does not consider the strategical decisions, integrated to the operational ones. Another difference relies on the objective function since the SRP-CD addresses first the minimization of working days to perform cleaning operations. Hence, the mathematical model differs from the proposed model for SRP-CD in this thesis, mainly in the time representation.

In Drexl [2012], Vehicle Routing Problems with Multiple Synchronization constraints (VRPMSs) are introduced, where a fleet of vehicles ought to be synchro-

nized at nodes to fulfill common tasks, with goals of minimizing transportation costs, overall distance traveled, number of vehicles used, maximal route duration or complete route plan duration. The author reports the interdependence problem in VRPMSs where changes in some routes may affect other routes, contrary to standard VRPs where the routes are independent. In addition, a categorization with five types of synchronization is described: task, operation, resource, movement and load synchronizations. Then, several works are reviewed and classified in these categories, but none of the referenced studies resemble SRP-CD, which contains limited working days, service times and the minimization of working days as main goal. Overall, the SRP-CD could be classified as a VRPMSs since task, operation, resource and loading synchronizations can be identified in our problem, as well as the interdependence problem.

2.2 Applications related to SRP-CD

2.2.1 Disaster relief applications

Several applications in disaster relief, cleaning and repairing urban networks address aspects related to the SRP-CD. The closest related applications are reviewed below.

The first group of applications listed here are crew scheduling problems including or not routing. The study of Duque et al. [2016] addressed the Network Repair Crew Scheduling and Routing Problem (NRCSRP) in post-disaster relief, focusing on a single repair crew. The idea was to start from a single depot to establish network connectivity and provide humanitarian relief. The goal was to minimize the time in which the blocked nodes become accessible. A Dynamic Programming (DP) approach was applied to solve small instances and an iterated greedy-randomized constructive heuristic was proposed to solve medium- to large-size instances. The instances varied from 21 to 41 nodes (small), 61 to 401 nodes (others) and both with 5% to 50% of damaged nodes. According to the authors, the DP approach was able to find the optimal solution in 225 out of 300 small instances, limiting the maximum computing time to 24 hours for each instance. The heuristic was able

to find the optimal solution for 92,8% out of 225 small instances. Comparing to SRP-CD, these studies do not consider an integration of the two levels of decision that include a synchronization between WTs and dump trucks. The repair crew is not divided into assigning WTs and removing debris to landfills using trucks. This means that the operational decisions to transport debris for landfills are not considered. Moreover, their problem is not multi-period. Then, working days are not considered, as well as service times for loading and unloading operations, nor landfills to unload debris.

The Crew Scheduling and Routing Problem (CSRP) investigated in Moreno et al. [2019] takes into account a single crew, with the goal of minimizing the time affected areas remained inaccessible. A mathematical model for the problem was presented based on the model of Duque et al. [2016], by using Miller, Tucker and Zemlin (MTZ) constraints. A branch-and-Benders-cut algorithm was proposed, where the problem was divided into a master problem for determinating the crew schedule, a subproblem to check the feasibility of the schedule, and a subproblem to calculate the total cost of the schedule. A simple greedy constructive heuristic and two local search heuristics were introduced. Another work focusing on the CSRP was found in Shin et al. [2019], where a single crew, integrated with a single vehicle distributes relief goods after the repair of damaged places. A MILP model was used, it was based on the model of Duque et al. [2016] and the objective was to minimize the time which the last demand node was served. Since the model becomes intractable for larger instances, an ant colony optimization algorithm was applied to medium to large instances taking into account both repair and distribution. As a consequence, a route for the repair crew and a route for the relief vehicle were obtained. Compared to the SRP-CD, as for the previous study, these works on CSRP do not consider the synchronization of WTs and dump trucks and the complex integration of two levels of decision as it is done in this thesis. They lack of operational decisions for debris transportation, multi-period and service times.

The Stochastic Debris Clearance Problem (SDCP) was investigated in Çelik et al. [2015], in which the main goal is to unblock roads in the response phase. The debris are moved to the sides of the roads to reach other demand nodes, in order

to maximize the benefit accrued by satisfying relief demand. A partially observable Markov decision process model was applied to optimally define the solution for small and large instances. The approaches were tested on randomly generated instances and instances based on a real-world earthquake scenario of Boston simulated by the Federal Emergency Management Agency (FEMA) application.

The problem to unblock roads in order to reconnect the network affected in the response phase while minimizing the operation time, was addressed by Akbari and Salman [2017]. The WTs cleaned the road by removing and pushing debris to sides. A synchronization between vehicles was taken into account when two or more vehicles were assigned to the same road by using a waiting time. A MILP formulation, a matheuristic based on a MILP-relaxation, and a local search algorithm were proposed. Tests were done over data based on Istanbul road network and randomly generated data with Euclidean distances with 40 or 50 nodes. According to the results, the matheuristic found optimal or near-optimal solutions.

In Li and Teo [2019], the road network repair work scheduling and relief logistics problem focuses on the response phase. The problem was divided in two stages: a first one where a multi-period scheduling problem with limited resources was solved; and a second stage where routes for relief materials delivery were defined. A mathematical model with two levels was proposed, which aimed to maximize the network accessibility in the upper level and to minimize the time in which materials were delivered in the lower level. The authors proposed a heuristic for the lower level problem which was introduced and incorporated in a steady-state parallel genetic algorithm from literature. The hybrid approach was tested on generated instances varying from 25 to 1000 nodes, with 5% to 50% of damaged nodes and the number of repair crews up to 34. The approach was also tested in a case study derived from the Wenchuan earthquake in China, but the authors do not provide further information about the real data.

The road emergency rehabilitation problem in Sakuraba et al. [2016] focuses on two problems: the Road Network Accessibility Problem (RNAP) where the goal is to find paths for relief teams to reach the population as fast as possible; and the Work-troops Scheduling Problem (WSP), where the goal is to create a

repairing schedule to improve access to refugee areas and improving the distribution of humanitarian aid for such areas. The authors proposed a mathematical model and three heuristics (a savings, a ranking and a lexicographical one) to solve large-scale realistic instance of Haiti earthquake in 2010 containing 16,660 vertices and 19,866 routes for the urban network, and more than 500 blocked roads. A group of simulated instances was also used in the experiments and optimal solutions were found in 72 out of 80 of these instances. Moreover, the lexicographical heuristic was able to find an average relative deviation of less than 3% from optimal solutions obtained. An extension of this work was found in Barbalho et al. [2020], where the authors proposed a dedicated local search for the WSP and two metaheuristics, a GRASP and an Iterated Local Search (ILS). The methods were applied to solve theoretical instances and the large-scale realistic instance of Haiti earthquake in 2010 from Sakuraba et al. [2016].

Compared to the SRP-CD, in the studies aforementioned [Çelik et al., 2015; Akbari and Salman, 2017; Li and Teo, 2019; Sakuraba et al., 2016; Barbalho et al., 2020], the integration of scheduling to routing decisions for debris transportation is not addressed in these studies. The main goals were to improve accessibility or the access for relief teams in the response phase. In addition, discharge areas such as landfills to accommodate debris or service times for operations are not considered.

Another group of related applications describes the problem of debris collection, including only the routing part. The debris collection problem was treated in Pramudita and Taniguchi [2014]; Pramudita et al. [2014] as a Capacitated Arc Routing Problem (CARP), including blocked access to some points. The CARP was transformed into a Location-Capacitated Vehicle Routing Problem (L-CVRP), establishing, from candidate locations, intermediate depots for trucks, so that the trucks can unload (the truck described by the authors could load itself). In short, the problem described by the authors consists of an integrated problem of location-routing with the objective of minimizing costs. A mathematical model was proposed, where the time to complete the cleaning operations was continuous. In addition, a tabu search was described to solve the augmented CARP formulation for the debris collection operation. A case study of Tokyo Metropolitan Area was

presented, together with an estimation on the amount of debris in the collection points. This study mainly differs from SRP-CD because a different integration considering the synchronization between WTs and dump trucks and different goals are taken into account in the latter.

The debris clearance was described by Lorca et al. [2017] as a set of operations such as collection, transportation, reduction, recycling and disposal, with the goal of minimizing cost management. A mathematical model including all the mentioned decisions was presented, and applied to a case study of the Hurricane Andrew. The authors developed their model in the Microsoft Office Excel software using the open source optimization package GLPK. Hence, in the case study, they analysed the impact in the management costs of varying parameters like number of work teams, allocating more landfills or more incinerator sites. In addition, some insights on debris estimation errors were provided. The post-disaster clearance problem as a waste supply chain management model was proposed by Boonmee et al. [2018], where the goal is to minimize the total costs. The possibility of separating and recycling the waste collection was described in order to minimize transportation and the operational costs by selling the appropriate waste in markets. Moreover, the environmental and human impacts of the clearance process were also optimized. A mathematical model was proposed to handle these issues and two metaheuristics were developed: a particle swarm optimization and a differential evolution. The authors tested 20 instances from small to very-large sizes using LINGO. The metaheuristic results reported maximum gaps of 2.36% and in some cases LINGO was not able to run in 12 hours. These works of debris clearance differ from SRP-CD since they do not handle the strategical decisions (scheduling) integrated to the operational ones (routing), they do not consider service times nor limited working days.

2.2.2 Logistics applications

The studies Huang et al. [2020a,b] focused on recovering urban networks in normal time when disruptions happen, either predictable (road works, maintenance, social events, etc.) or not (accidents, bad weather conditions, disasters, etc.) referred to

as Road Network Problem with Disruptions (RNPD) and connecting requirements. The goal was to define alternative paths, while ensuring the strong connectivity of the road network. The authors had proposed mono and bi-objective methods, and solved simulated and realistic instances. An extension of these works was found in Coco et al. [2020], where the scheduling of planned disruptions was coupled to RNPD using simple and multi-graphs. An ϵ-constraint method was proposed and applied to instances of Troyes in France. Although RNPD problems can occur after major disasters, compared to the SRP-CD, these works do not describe strategies for debris removal in an affected network.

The study of Grimault et al. [2017] focused on the FTPDP with resource synchronization (FTPDP-RS), with application in the context of public work and road construction. The problem was described in a time horizon of one working day, where a set of heterogeneous trucks served requests for material transportation (asphalt, gravel or waste). The requests were composed of pairs of pickups and deliveries. Trucks had to be compatible with a request in order to be scheduled to demands. Pickups and deliveries operations were associated with a service time and a resource. Moreover, trucks had fixed costs, travel costs and hourly costs. The objective was to minimize the overall cost for the routes to satisfy all requests. A mathematical model and an adaptive large neighborhood search were proposed. The experiments included two sets (set-1 and set-2) of 20 instances from the literature and a group of 7 instances from a real case study. The proposed heuristic obtained an average gap of 0.00% for the set-1 and -0.14% for set-2 compared to results presented in the literature. In the SRP-CD, the time horizon is composed of several working days and the fleet of trucks is homogeneous. Another difference lies in the objective function, since SRP-CD has the minimization of working days as primary goal and the minimization of trucks' travel time as secondary goal.

The Log-Truck Scheduling Problem (LTSP) is reported in El Hachemi et al. [2011], where the problem considers scheduling the pre-defined transportation requests of logs from forest areas to woodmills and routing the fleet of vehicles to fulfill all requests in a time interval (similar to a single working day). The objective is to minimize the total transportation cost, reducing waiting times and empty

trucks trips. Each request considers full truckloads and, at each node, there is a single log-loader that ensures the loading and unloading of all trucks. A hybrid constraint programming approach was proposed. LTSP differs from SRP-CD in the goals and because characteristics like scheduling of loaders in different locations integrated to routing and several limited working days are not included.

2.2.3 Open pit mining applications

The Open Pit Mining (OPM) [Alarie and Gamache, 2002] consists in defining routes for material transportation from loading to dumping nodes in mining fields, with the goals of improving productivity. The term productivity was used to indicate the amount of material transported by vehicles and as a consequence the reduction of total operational costs. In spite of the context and the fact that OPM was most solved in the literature by means of simulation, several similarities exist between OPM and SRP-CD. For instance, OPM has loading and dumping points, and strategical and operational decisions. The differences rely on the following issues: there is a fixed time to plan the operations, and the objective function looks for minimizing resource costs. Moreover, in general, OPM is addressed in a known and medium size area, differing from a disaster which can hit everywhere and cover unexpected and large area.

The authors in Tan and Takakuwa [2016] treated the OPM as a two stages process as follows. At the first stage, given an excavation plan for the mine, the optimal number of shovels and trucks and the production amount for every shovel were computed. In the second stage, a simulation algorithm was applied in order to dispatch each truck for the shovels based on the current and remaining data to achieve the plan. The results improved the excavation time and the total travel distances, respectively in about 4 000 minutes and 3 500 km. Study Subtil et al. [2011] also handled OPM in a two stage process. In the first stage, a Linear Programming (LP) model was applied to maximize the productivity and solved using a Branch-and-Bound. In the second stage, a simulation was done in order to assign trucks to loading nodes in an exhaustive strategy. Results indicated an improvement on the production and shovels idle times, while loading waiting times

were slightly improved.

Although OPM was mainly solved by means of simulation [Alarie and Gamache, 2002], a few studies investigated the problem using an optimization approach. In Fioroni et al. [2008], the authors focused on reducing mining costs, coupling an optimization mathematical model to a simulation one. A scenario for a mining plan was simulated and, while the simulation ran, the reallocation of the equipment (loader and transport) was optimized with the optimization model whenever an interference event, such as maintenance of equipment or mining area exhausted appeared. Results indicated that the approach reduced the operation costs by about 4.97%. In Ercelebi and Bascetin [2009], a queuing network strategy was proposed to optimize the number of truck assignments to shovels, which analysed all possible feasible paths between load and dump nodes. Then, a LP formulation was proposed for dispatching trucks to shovels, minimizing the number of trucks. A coal mine case study was described, and results indicated a production of 10.1 million m^3 using 4 shovels and 18 trucks over a year. Another study using optimization was found in Souza et al. [2010], where a MILP optimized the deviations of the production, quality goals and the number of vehicles required for operations. A hybrid heuristic was also proposed to solve the problem. The experimental results were performed using instances from a set of real-data problems and the results showed that the heuristic was able to find good solutions quickly with low variability.

2.3 Position of this thesis

Tables 2.1 and 2.2 present a summary of the most related works to the SRP-CD regarding the characteristics of problems found in the literature and the approaches proposed. The works describing at least 4 features similar to SRP-CD were included. The works are sorted by year and by classical problem or by application. The dots in the table means that the corresponding work includes the characteristic and the red dot represents the SRP-CD.

For sake of clarity, the characteristic *WT/Crew* for the post-disaster studies also represents the loader for OPM studies. The characteristic *Material collection* represents the material which is transported according to each study, being *hidden*

when the study does not provide further details about the material.

As can be observed, no studies in the literature cover all characteristics of the SRP-CD. But, the SRP-CD is a crucial problem for medium and long terms operations. Its goal is to allow a faster recovery process of the affected area. From the government point of view, it allows to rebuild the area and infrastructures, while to the population, it permits somehow to overcome the disaster.

SRP-CD deals with a high amount of debris, where the time and the costs required to removal operations are very relevant. An optimized planning can improve such operations. For such purposes, SRP-CD objectives are to minimize the number of days necessary to clean the overall area, and to optimize the operational costs for the set of dump trucks. In summary, the models and methods focus here look for answering the following main questions. What are the best assignments of WTs at debris nodes for each day ? Accordingly, what are the routes planning for dump trucks that minimizes costs ? These decisions require an organization between WTs and dump trucks, and thus constraints to ensure dump trucks will visit nodes with WTs, to coordinate dump trucks whenever more than one works in a specific node, to guarantee that the number of resources and their corresponding capacity are satisfied (WT and dump trucks), and regulation constraints on the number of hours to work in a day.

In the next chapters, the SRP-CD is formally defined in Chapter 3, heuristics and metaheuristics are described in Chapter 4, and computational experiments are reported in Chapter 5.

		Characteristics																				Approaches			
		Depot		WT/Crew		Fleet		Material collection				Sync.				Working day		Goal							
Classical	Works	Single	Multi	Single	Several	Homogeneous	Heterogeneous	Hidden	Explicit	Unload area	Service times	WT-WT	WT-Truck	Truck-Truck	Multi-trip	One	Several	Min. time	Min. cost	Max. satisf.	Mathematical model	Exact method	Heuristics	Simulation
RCPSPR	Lacomme et al. [2019]	•				•			•						•			•			•		•	
FTVRP	Desrosiers et al. [1988]		•			•		•						•		•			•		•	•		
SDVRP	Archetti and Speranza [2008]	•				•		•						•	•				•		•	•	•	
PDP	Parragh et al. [2008b,a]	•				•		•		•	•			•		•			•		•	•	•	
1-FTPDP	Gendreau et al. [2015]	•				•		•		•				•					•		•	•		
FTPDP-mVS	Soares et al. [2019]		•		•	•			•	•	•		•	•	•	•			•		•		•	
MDVRPI	Crevier et al. [2007]		•			•		•		•	•			•		•			•		•		•	
PVRP-IF	Angelelli and Speranza [2002]	•				•			•	•	•			•			•		•				•	
	SRP-CD	•			•	•			•	•	•		•	•	•		•	•	•		•		•	

Table 2.1: Summary of classical optimization problems and works compared to SRP-CD.

Applications	Works	Depot: Single	Depot: Multi	WT/Crew: Single	WT/Crew: Several	Fleet: Homogeneous	Fleet: Heterogeneous	Material collection: Hidden	Material collection: Explicit	Unload area	Service times	Sync.: WT-WT	Sync.: WT-Truck	Sync.: Truck-Truck	Multi-trip	Working day: One	Working day: Several	Goal: Min. time	Goal: Min. cost	Goal: Max. satisf.	Approaches: Mathematical model	Approaches: Exact method	Approaches: Heuristics	Approaches: Simulation
Disaster relief	Barbalho et al. [2020]				•			•										•			•		•	
	Moreno et al. [2019]	•		•				•										•			•	•	•	
	Shin et al. [2019]	•		•				•										•			•		•	
	Li and Teo [2019]		•		•			•									•			•	•		•	
	Boonmee et al. [2018]								•	•									•		•		•	
	Akbari and Salman [2017]		•		•			•				•						•			•		•	
	Lorca et al. [2017]				•				•	•							•		•		•			•
	Sakuraba et al. [2016]				•			•										•			•		•	
	Duque et al. [2016]	•		•				•										•			•	•	•	
	Çelik et al. [2015]				•			•									•			•	•		•	
	Pramudita et al. [2014a, 2014b]	•				•			•	•									•		•		•	
Logistics	Grimault et al. [2017]		•		•		•		•	•	•		•			•			•		•		•	
	El Hachemi et al. [2011]		•			•		•			•			•		•			•		•	•		
Open pit mining	Tan and Takakuwa [2016]				•	•			•	•			•	•			•		•			•		•
	Subtil et al. [2011]				•		•		•	•			•			•				•	•	•		•
	Souza et al. [2010]				•		•		•	•			•		•				•		•		•	
	Ercelebi and Bascetin [2009]				•		•		•	•			•		•				•		•	•		
	Fioroni et al. [2008]				•	•			•	•			•		•	•			•		•			•
	SRP-CD	•			•	•			•	•	•		•	•	•		•	•	•		•		•	

Table 2.2: Summary of the application works related to the SRP-CD.

Chapter 3

Formal definition for SRP-CD

This chapter is dedicated to the formal definition for SRP-CD. For this purpose, in Section 3.1 definitions, hypothesis and notation are given, followed by some examples. In the sequel, the mathematical formulation is detailed in Section 3.2.

3.1 Definitions and notation

As mentioned before, SRP-CD problem aims at assigning WTs to debris nodes, loading, transporting and unloading debris during working days by synchronizing work troops and dump trucks, and by synchronizing dump trucks, in a time horizon. WT and dump trucks are synchronized since dump trucks may not visit nodes without WTs. The synchronization between two dump trucks occurs if a dump truck is being loaded at a debris node. Then, the other dump trucks that visit the same node, at the same moment, have to wait the current operation to finish. These synchronizations are stated based on the study of Drexl [2012]. Let us consider the following parameters: a single depot, a limited number of WTs W, and a fleet $K = \{1, ..., k\}, k \in \mathbb{N}^+$ of homogeneous dump trucks available at the depot, with a capacity Q each in tons.

The transportation network is modeled on a connected undirected and simple graph $G = (V, E)$, where V is the set of n vertices and E is the set of m edges. Let $O \subset V$ be the subset with the single depot, where all the WTs and dump trucks are available, $O = \{0\}$. Let $D \subset V$ be the subset of vertices corresponding to debris

nodes to be cleaned, where WTs can be assigned, and where the dump trucks will be loaded. Moreover, let $L \subset V$ be the subset of vertices corresponding to landfills, where dump trucks will unload debris. Thus, $V = D \cup L \cup O$ and $D \cap L \cap O = \emptyset$. Each vertex $i \in D$ has a volume $v_i \in \mathbb{R}_+^*$ of debris to be removed. In the edge-set E, edge $[i, j]$ models a shortest path linking nodes i and j in the real network, with pre-computed travel times $c_{ij} \in \mathbb{R}_+^*$, that are assumed to be symmetric and the triangular inequality holds. Moreover, $A = \{1...H\}$ represents a time horizon (with a step of one working day) considered in the model, and every working day has a time limit $B = \{1...T\}$ to represent a time discretization for a working day (with a step of one time unit).

SRP-CD consists of two main levels of decision:

(1) the strategical level focuses on assigning a set of WTs to debris nodes over the time horizon; and

(2) the operational level aims at defining synchronized multiple-trips for the fleet of dump trucks during every working day, to perform the collection debris process, in such a way that the dump trucks capacity is satisfied.

Each WT is assigned to a vertex $i \in D$ over a certain number of time periods if and only if there is no WT at node i, and a WT remains at node i until all debris are removed. Dump trucks visit node $i \in D$ if and only if there is a WT at node i. In addition, loading and unloading debris has a fixed service time, which is represented respectively by $t_l, t_u \in B$. Dump trucks start and finish a working day at the depot. SRP-CD addresses the two objectives in lexicographic way: the primary objective is to minimize the total number of working days to clean all sites containing debris, and the second one aims at minimizing the dump trucks' total travel time.

Some additional hypothesis are considered:

- Following the real context, WTs travel to debris nodes before a working day. In some cases, they are even transported by using a support vehicle (lorries) as

in [Soares et al., 2019]. This is done since WTs move quite slow, and traveling before starting a working day may increase the operations performance.

- Each dump truck is able to unload itself, without the support of a WT or other additional machine. There is no synchronization between dump trucks in landfill nodes. Hence, two or more different trucks can unload at the same time at the same landfill.

- Each debris node has at most one WT assigned. Moreover, a same WT cannot load dump trucks in two different locations in the current working day.

- In some cases, a debris node can be considered as an agglomeration of different debris sources (roads, buildings, crops). And all debris are considered homogeneous (only one type). Multiple types of debris would be an extension of this research. Especially if there are some precedence constraints between types.

- Landfill nodes are considered with unlimited capacity.

3.1.1 Examples for the SRP-CD

For the sake of clarity, we illustrate in this section by means of examples how SRP-CD works. The example in Figure 3.1 presents a graph for SRP-CD, where $V = \{0,1,2,3,4,5,6\}$, $O = \{0\}$, $D = \{1, 3, 4, 6\}$ and $L = \{2, 5\}$. In this figure, the depot, each debris node and each landfill are identified respectively by a solid square, a solid circle and a dotted square. For each edge the corresponding travel time, in time units, is detailed. In each debris node $i \in D$, the amount of debris v_i, in tons, is detailed next to it. The depot is represented by a figure containing a set of excavators (WTs) and a fleet of dump trucks. A synchronization between a WT and a dump truck is represented in the figure of the debris node *1*, where a loading operation is performed. An unloading operation is represented in the figure for the

landfill node *2*. Finally, the debris node *4* symbolizes a debris site non-visited by any WT.

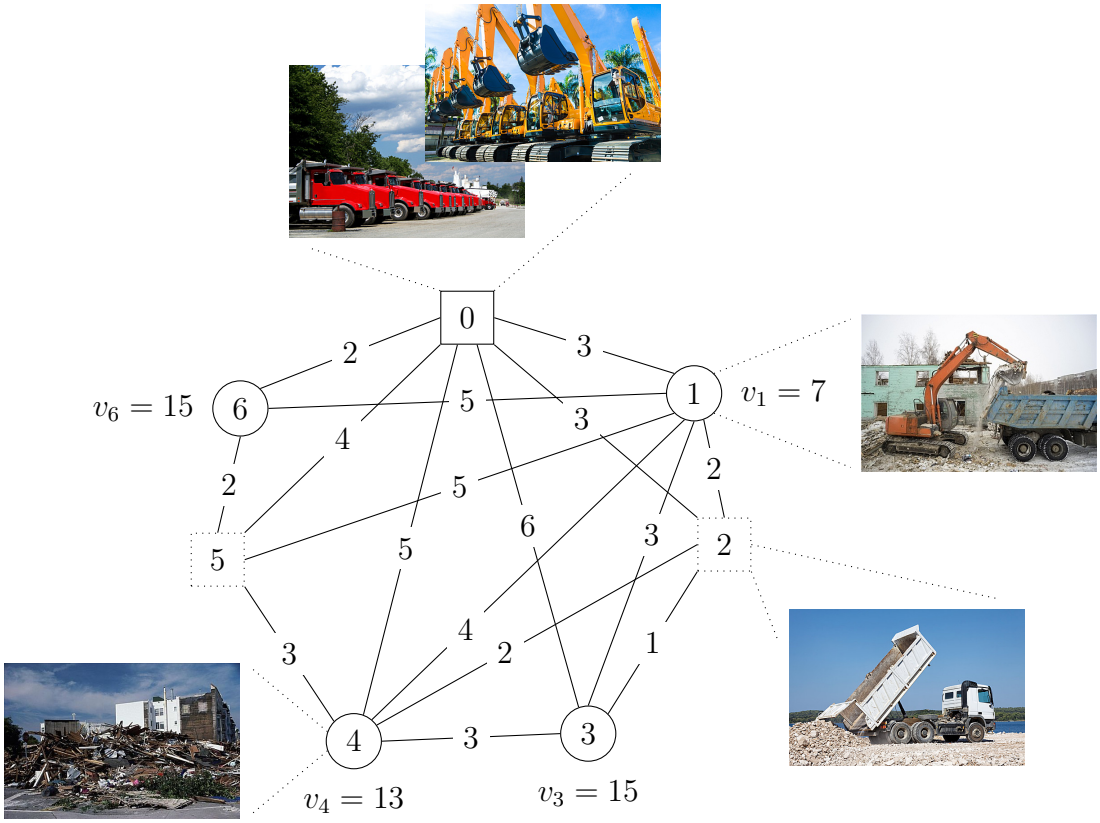


Figure 3.1: A graph to represent the SRP-CD. Images source: Quanjing [Quanjing, 2021]

A smaller example is depicted in Figure 3.2, where a graph contains $O = \{0\}$, $D = \{1, 2\}$, $L = \{3\}$, $E = \{(0,1), (0,2), (0,3), (1,2), (1,3), (2,3)\}$, the amount of debris in sites *1* and *2* are $v_1 = 12$t and $v_2 = 8$t respectively and the travel times in time units are designated in each edge of the graph. In addition, $Q = 2t$, $t_l = 1$, $t_u = 1$, $T = 20$.

For this graph in Figure 3.2, let us consider an instance to solve the SRP-CD using 2 WTs (i.e. $W = 2$) and 2 dump trucks (i.e. $K = \{K_1, K_2\}$), then a solution is illustrated in Figure 3.3. In Figure 3.3-(a), the scheduling of the WTs is represented by rectangles containing the identifier of the corresponding debris node, in the range of working days. In Figure 3.3-(b), the routing of each dump truck is illustrated and the range is denoted in time units, where empty truck trips and full truck trips are shown, respectively, by dotted arrows and solid arrows. The solid

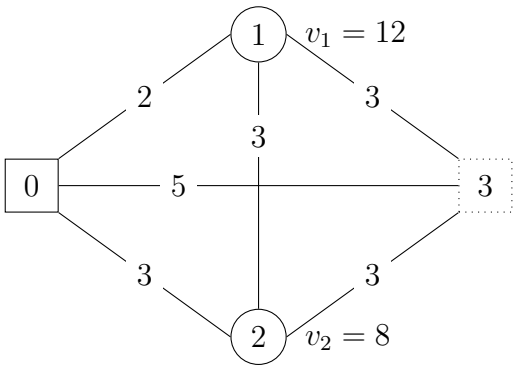
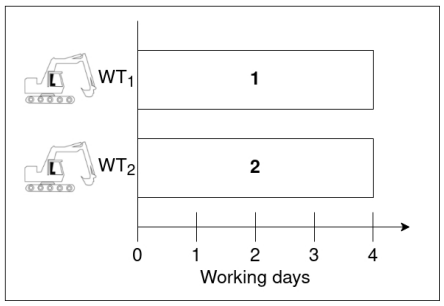


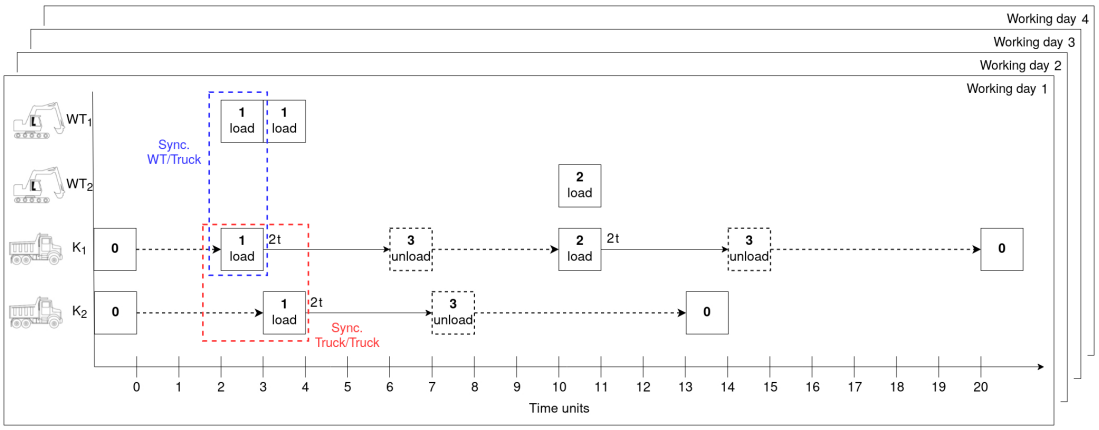
Figure 3.2: Small graph to show the different aspects of the SRP-CD.

square represents the depot or the loading operation in the corresponding debris node (synchronization WT-dump truck, highlighted by the blue dotted square), the black dotted square indicates the unloading operation in the landfill, and the amount of debris loaded in each trip is shown close to the load square in trucks' routes.

In this solution, dump trucks are allowed to visit both debris sites in the same working day, because each site has a WT assigned. However, there is a synchronization between dump trucks, since two dump trucks cannot be loaded at the same time unit. This detail is highlighted by the red dotted square. The solution shows the scheduling of the two WTs in different debris nodes (Figure 3.3-(a)). Since $v_1 = 12$t, $v_2 = 8$t, $Q = 2t$, $T = 20$ time units, and $t_l = 1$ and $t_u = 1$ time unit each, dump trucks are routed as shown in the Figure 3.3-(b), where K_1 is able to transport 4t of debris visiting the debris nodes 1 and 2 with a travel time of 16 time units. Due to the synchronization, K_2 waits 1 time unit to be loaded at debris node 1 and it is able to transport 2t of debris with a travel time of 10 time units. For the overall cleaning, the route of K_1 is repeated similarly for 4 working days and the route of K_2 is repeated similarly for 2 working days, thus completing the cleaning of 20t. Despite of the synchronization, the route of each dump truck is independent so that both can work on the same day. This way, the final solution presents a total of 4 working days and 84 ($= 4 \times 16 + 2 \times 10$) time units of total travel time, considering the trucks K_1 and K_2.



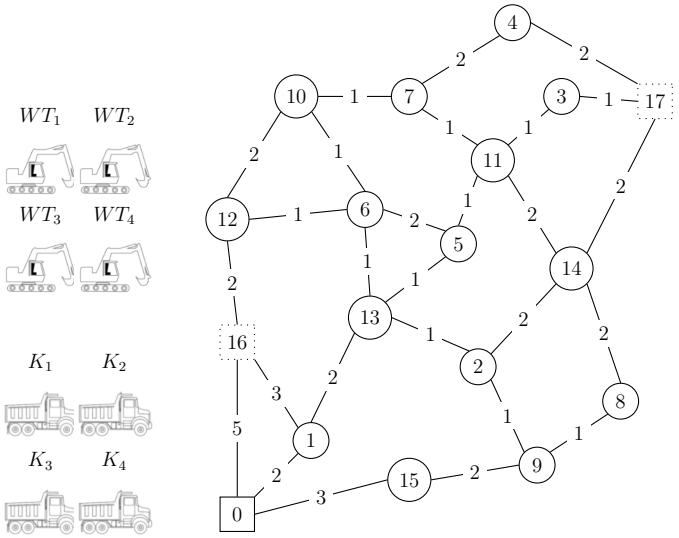
(a) Scheduling for WT_1 and WT_2.



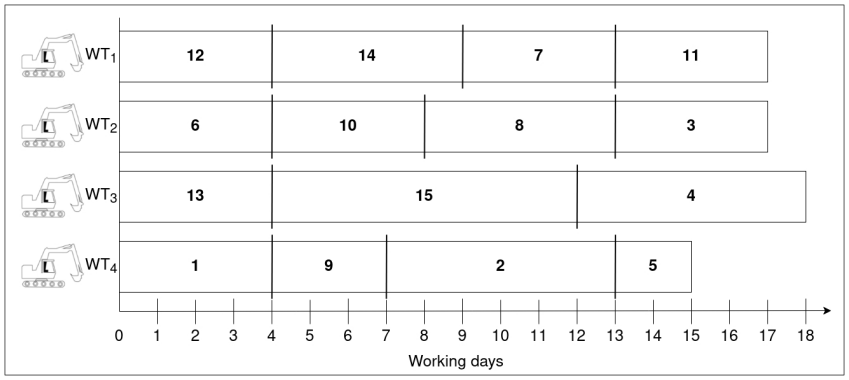
(b) Route K_1: travel time of 16 time units; 4t of debris removed in each working day 1, 2, 3 and 4.
Route K_2: travel time of 10 time units; 2t of debris removed in each working day 1 and 2.

Figure 3.3: Solution for the instance with 2 WTs and 2 dump trucks.

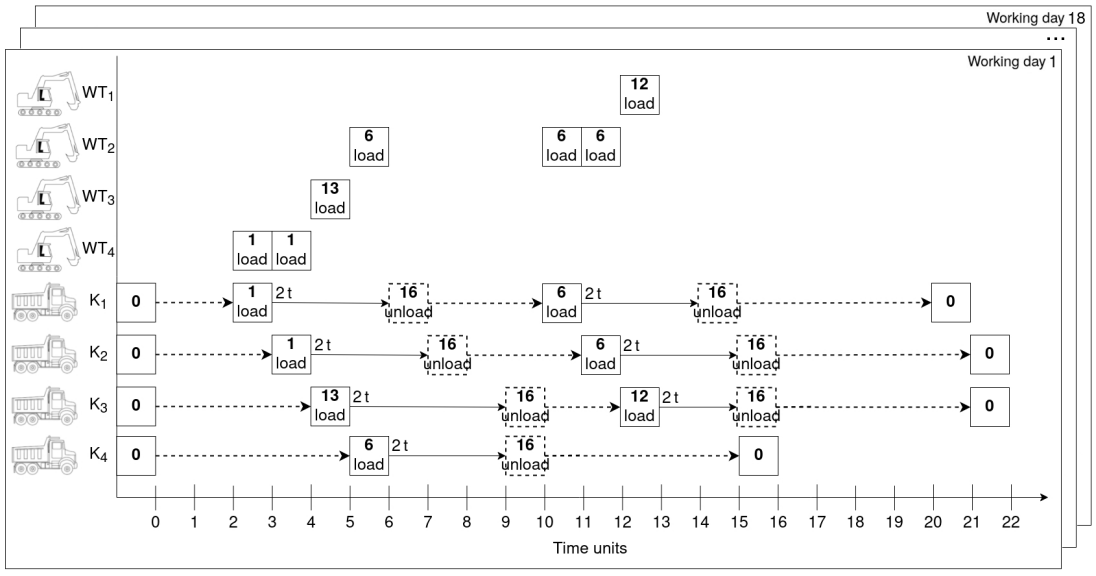
Figure 3.4 depicts a large example with the following characteristics: $O = \{0\}$, $D = \{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\}$, $L = \{16,\ 17\}$, $WT = 4$, $K = \{K_1, K_2, K_3, K_4\}$, $Q = 2t$, $t_l = 1$, $t_u = 1$ and $T = 22$. The graph is presented in Figure 3.4-(a). The scheduling is shown in the Figure 3.4-(b), where the solution has a total of 18 working days. Figure 3.4-(c) shows routes of the dump trucks in "the first working day". Dump trucks K_1 and K_2 were able to visit nodes 1 and 6 with synchronized routes. K_3 visited nodes 13 and 12, and the dump truck K_4 visited the debris node 6. The times of each route can be noticed, where K_1 has a route of 21 time units of which 16 time units for travel. K_2 has a route similar to K_1 but there is a waiting time to be loaded. K_3 has a route of 22 time units in which 17 time units correspond to trips. Finally, K_4 has a route of 16 time units in which 13 time units correspond to the trips.



(a)



(b)



(c)

Figure 3.4: Example for the SRP-CD with (a) the graph and the fleet, (b) the scheduling of the work-troops and (c) the routing of the dump trucks in the first working day.

3.2 Mathematical formulation

SRP-CD presents two major complex issues: two integrated time scales, one for each working day and another for the overall time horizon; and the synchronizations between WTs and dump trucks and between dump trucks. In addition, it also considers service times for loading and unloading operations, the capacity for each dump truck and the multi-trip for each dump truck. A maximum time horizon limit is given in each instance in order to strengthen the model.

The proposed mathematical formulation is given in Equations (3.1) to (3.36). It is based on a multi-flow formulation [Ahuja et al., 1989], and makes use of a dynamic flow. As shown by Aronson [1989] and Wang [2018], dynamic flows can be interpreted in several ways. A most frequent representation is a discretization of the time. This is the strategy applied here, where the working days and the time horizon are indexed over the time, according to a discretization considering respectively time units and working days. Although this strategy presents a large number of variables and constraints, it allows to further address the two levels of synchronization. To our knowledge, the proposed dynamic multi-flow formulation is the first one found in the literature for SRP-CD.

There is a flow for WTs in the time horizon and a flow for the travels of dump trucks, which is indexed over the time horizon and working days. The design for the flow of WTs is adapted from the model of Sakuraba et al. [2016]. This flow allows the formulation to ensure that a WT remains at a debris node until it is totally cleaned. Without such constraints, the model can assign each WT in different debris nodes every day before finishing to clean it completely. This is not a suitable situation in the context of disaster recovery. The design for the flow of dump trucks is found in several models of VRPs [Golden et al., 2008]. Such ideas have been used for SRP-CD. In addition, the synchronization is done using the assignment of the WTs in the working days to allow visiting dump trucks and the date a dump truck arrives, each working day, to control the incoming flow of others dump trucks.

The proposed mathematical model is referred to as a *multi-flow multi-period lexicographic bi-objective model* and it is divided in two MILP formulations. The

first formulation corresponds to the strategic level in SRP-CD, where the objective function $\mathcal{F}_1$ represents a number of days for the overall cleaning. The second formulation corresponds to the operational level in SRP-CD, where the objective function $\mathcal{F}_2$ gives the optimized value for the total travel time for the fleet of dump trucks, respecting the first optimization. More details of each level are described in the following, where the first formulation is depicted in Equations (3.1) to (3.19), and the second one is shown in Equations (3.20) to (3.36).

The mathematical model makes use of the following binary variables:

- x_i^h establishes if a WT is assigned to the node i in a working day h ($x_i^h = 1$), otherwise ($x_i^h = 0$).
- y_{ij}^h determines if the WT visits the node j after i in the beginning of a working day h ($y_{ij}^h = 1$), otherwise ($y_{ij}^h = 0$).
- f_{ijk}^{th} states if the dump truck k arrives at node j from i in time period t of a working day h ($f_{ijk}^{th} = 1$), otherwise ($f_{ijk}^{th} = 0$).

In addition, the following variables are used:

- l_{ik}^{th} determines the accumulated load of each dump truck k at node i in time period t in a working day h, $0 \leq l_{ik}^{th} \leq Q$.
- u_{ik}^{th} states the amount of debris removed and loaded in the dump truck k at i in time period t in a working day h, $0 \leq u_{ik}^{th} \leq Q$.
- δ represents a number of working days used in the time horizon, $\delta \geq 0$.

A summary of the notations and variables used in the formulation is presented in the following Table 3.1.

Symbol/Set	Description
$G = (V, E)$	Graph for the transportation network
V	Set of vertices (nodes)
O	Set containing a single depot, $(O \subset V)$
D	Set containing debris nodes, $(D \subset V)$
L	Set containing landfill nodes, $(L \subset V)$
E	Set containing edges $[i, j]$, $(i, j \in V)$
W	Number of WTs available
$K = \{1, ..., k\}$	Fleet of homogeneous dump trucks
Q	Capacity of a dump truck
v_i	Volume of debris for a node $i \in D$, $(v_i \in \mathbb{R}_+^*)$
c_{ij}	Pre-computed travel time for an edge $[i, j] \in E$, $(c_{ij} \in \mathbb{R}_+^*)$
$A = \{1, ..., H\}$	Time horizon containing H working days
$B = \{1, ..., T\}$	Time discretization for a working day containing T time periods
t_l	Service time for a load operation, $(t_l \in B)$
t_u	Service time for an unload operation, $(t_u \in B)$
x_i^h	Binary variables for the WT assignment
y_{ij}^h	Binary variables for the WT flow
f_{ijk}^{th}	Binary variables for the dump truck flow
l_{ik}^{th}	Cumulative variables on the load of a dump truck
u_{jk}^{th}	Variables stating the amount of debris loaded in a dump truck
δ	Integer variable for the final number of working days found by the model
$\mathcal{F}_1$	First objective function for the model (Strategic level)
$\mathcal{F}_2$	Second objective function for the model (Operational level)

Table 3.1: Notations used in the SRP-CD formulation.

$$\mathcal{F}_1 = min \;\; \delta \tag{3.1}$$

subject to:

$$\delta \geq \lceil h x_i^h \rceil \qquad \begin{matrix} \forall i \in D, \\ h = \{1...H\} \end{matrix} \tag{3.2}$$

$$\sum_{h=1}^{H} \sum_{j \in D} y_{0j}^h \leq W \tag{3.3}$$

$$x_i^h = \sum_{j \in \{0\} \cup D} y_{ji}^h \qquad \begin{matrix} \forall i \in D, \\ h = \{1...H\} \end{matrix} \tag{3.4}$$

$$\sum_{j \in \{0\} \cup D} y_{ji}^h = \sum_{j \in \{0\} \cup D} y_{ij}^{h+1} \qquad \begin{matrix} \forall i \in D, \\ h = \{1...(H-1)\} \end{matrix} \tag{3.5}$$

$$\sum_{h=1}^{H} \sum_{\substack{j \in \{0\} \cup D, \\ j \neq i}} y_{ji}^h = 1 \qquad \forall i \in D \tag{3.6}$$

$$\sum_{i \in D} \sum_{t=1}^{T - c_{0i}} f_{0ik}^{(t+c_{0i})h} \leq 1 \qquad \begin{matrix} \forall k \in K, \\ h = \{1...H\} \end{matrix} \tag{3.7}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{th} = \sum_{j:(i,j) \in E} f_{ijk}^{(t+s)h}, \begin{cases} s = c_{ij} + t_l, \text{if } i \in D \\ s = c_{ij} + t_u, \text{if } i \in L \end{cases} \qquad \begin{matrix} \forall k \in K, \\ t = \{1...(T-s)\}, \\ h = \{1...H\} \end{matrix} \tag{3.8}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{(t+c_{ji})h} \leq x_i^h \qquad \begin{matrix} \forall i \in D, \forall k \in K, \\ t = \{1...(T-c_{ji})\}, \\ h = \{1...H\} \end{matrix} \tag{3.9}$$

$$f_{jik}^{th} + \sum_{j' \in \{0\} \cup L} \sum_{\substack{k' \in K, \\ k' \neq k}} \sum_{t'=t}^{t+t_l-1} f_{j'ik'}^{t'h} \leq 1 \qquad \begin{matrix} \forall i \in D, \forall j \in \{0\} \cup L, \\ \forall k \in K, \\ t = \{1...(T-t_l+1)\}, \\ h = \{1...H\} \end{matrix} \tag{3.10}$$

$$l_{jk}^{(t+s)h} \geq l_{ik}^{th} + u_{jk}^{(t+s)h} - Q(1 - f_{ijk}^{(t+c_{ij})h}), \begin{cases} s = c_{ij} + t_l \end{cases} \qquad \begin{matrix} \forall i \in V, \forall j \in D, \\ \forall k \in K, \\ t = \{1...(T-s)\}, \\ h = \{1...H\} \end{matrix} \tag{3.11}$$

$$u_{jk}^{th} \leq Q \sum_{\substack{i:(j,i)\in E,\\ i\in L}} f_{jik}^{(t+c_{ji})h} \qquad \begin{array}{r} \forall j \in D, \forall k \in K, \\ t = \{1...(T - c_{ji})\}, \\ h = \{1...H\} \end{array} \tag{3.12}$$

$$\sum_{h=1}^{H} \sum_{t=1}^{T} \sum_{k\in K} u_{jk}^{th} = v_j \qquad \forall j \in D \tag{3.13}$$

$$y_{ij}^{h} \in \{0,1\} \qquad \begin{array}{r} \forall i, j \in D \cup \{0\}, \\ h = \{1...H\} \end{array} \tag{3.14}$$

$$x_{i}^{h} \in \{0,1\} \qquad \forall i \in D \cup \{0\}, h = \{1...H\} \tag{3.15}$$

$$f_{ijk}^{th} \in \{0,1\} \qquad \begin{array}{r} \forall i, j \in V, \forall k \in K, \\ t = \{1...T\}, h = \{1...H\} \end{array} \tag{3.16}$$

$$0 \leq l_{ik}^{th} \leq Q \qquad \begin{array}{r} \forall i \in V, \forall k \in K, \\ t = \{1...T\}, h = \{1...H\} \end{array} \tag{3.17}$$

$$0 \leq u_{jk}^{th} \leq Q \qquad \begin{array}{r} \forall j \in D, \forall k \in K, \\ t = \{1...T\}, h = \{1...H\} \end{array} \tag{3.18}$$

$$\delta \geq 0 \tag{3.19}$$

The modeling for the strategic level includes the objective function $\mathcal{F}_1$ (Equation (3.1)), which minimizes the number of days to clean all sites containing debris. $\mathcal{F}_1$ is related to the constraints (3.2), where the number of working days necessary to clean all debris is bounded by a variable δ. It is connected to the assignment of the WTs by means of the variables x_i^h. Then, the objective function (3.1) seeks the minimal value to δ that satisfies the constraints (3.2). This technique was successfully employed in the model described by Solano-Charris et al. [2015].

Constraints (3.3) to (3.6) are the flow of WTs. Inequalities (3.3) guarantees that W WTs leaves the depot only once in all the time horizon. Equalities (3.4) link variables x to y and ensure the assignment of one WT to a debris node i if there is an incoming flow at i. Such WT can come from the depot, from another debris node or from the same debris node. In the latter, one can note that the permanence of a WT in a same node i until the end of its cleaning, along different working days is also ensured by variables y. This occurs when $\{y_{ii}^h = 1 \mid i \in D\}$. Constraints (3.5) are the flow conservation for the WTs. The general idea remains

similar to a classical flow conservation constraints, where all incoming flows to debris node have to leave the corresponding node. The main difference relies on the indexation over the time horizon, where the outgoing flows are indexed in the next working day. Equalities (3.6) determine that exactly one WT is assigned to each debris node at some working day.

Constraints (3.7) to (3.10) address the flow related to the routes for each dump truck in every working day. Inequalities (3.7) establish that each dump truck is either used or not, where the corresponding flow leaving the depot is up to one. Constraints (3.8) state the flow conservation for the dump trucks. The value s corresponds to the sum of the traveling time and the service time (either t_l or t_u according to the node). This allows to control the date of incoming and outgoing flows accordingly, if the corresponding dump trucks arrives either at a debris node or at a landfill. Inequalities (3.9) certify the synchronization between WTs and dump trucks since there is a link between variables x_i^h and f_{ijk}^{th}. They ensure that, for each dump truck k, a flow enters a debris node i at time period t and working day h, if and only if there is a WT assigned to i in working day h. Constraints (3.10) guarantee the synchronization between two dump trucks. If a truck k is loading at a debris node i, the others trucks k' have to wait the operation of loading k to be completed. This is done by the flow variables using the indexation over the time to control the date of the incoming flows.

Constraints (3.11) to (3.13) are related to the volume of debris, the amount of debris loaded at the dump trucks in each node and the capacity of each truck. Inequalities (3.11) determine the accumulated amount of debris, loaded at a dump truck. Such constraints are inspired on the well known Miller, Tucker and Zemlin (MTZ) constraints for the travelling salesman problem [Miller et al., 1960]. Constraints (3.12) along with the domains (3.17) and (3.18) establish that the dump trucks capacity is satisfied. Equations (3.13) guarantee that all debris nodes have to be cleaned until the last working day in the time horizon. Finally, the domain of the variables presented in the model are given from (3.14) to (3.19).

The modeling of the operational level contains the objective function $\mathcal{F}_2$ (Equation (3.20)) and it optimizes the total travel time used by dump trucks. The math-

ematical model works in a lexicographic order, by considering the time horizon found ($\mathcal{F}_1$) as following: the constraints of the second formulation are similar to the constraints of the strategic level model, but a time horizon (H') is set by the objective value obtained in the first model, such as $H' = \mathcal{F}_1$. In addition, constraints (3.2) and (3.19) are not required and used in the operational level.

$$\mathcal{F}_2 = min \sum_{h=1}^{H'} \sum_{t=1}^{T} \sum_{k \in K} \sum_{(i,j) \in E} c_{ij} f_{ijk}^{th} \tag{3.20}$$

subject to:

$$\sum_{h=1}^{H'} \sum_{j \in D} y_{0j}^{h} \leq W \tag{3.21}$$

$$x_i^h = \sum_{j \in \{0\} \cup D} y_{ji}^h \qquad \begin{array}{r} \forall i \in D, \\ h = \{1...H'\} \end{array} \tag{3.22}$$

$$\sum_{j \in \{0\} \cup D} y_{ji}^h = \sum_{j \in \{0\} \cup D} y_{ij}^{h+1} \qquad \begin{array}{r} \forall i \in D, \\ h = \{1...(H'-1)\} \end{array} \tag{3.23}$$

$$\sum_{h=1}^{H'} \sum_{\substack{j \in \{0\} \cup D, \\ j \neq i}} y_{ji}^h = 1 \qquad \forall i \in D \tag{3.24}$$

$$\sum_{i \in D} \sum_{t=1}^{T - c_{0i}} f_{0ik}^{(t+c_{0i})h} \leq 1 \qquad \begin{array}{r} \forall k \in K, \\ h = \{1...H'\} \end{array} \tag{3.25}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{th} = \sum_{j:(i,j) \in E} f_{ijk}^{(t+s)h}, \begin{cases} s = c_{ij} + t_l, \text{if } i \in D \\ s = c_{ij} + t_u, \text{if } i \in L \end{cases} \qquad \begin{array}{r} \forall k \in K, \\ t = \{1...(T-s)\}, \\ h = \{1...H'\} \end{array} \tag{3.26}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{(t+c_{ji})h} \leq x_i^h \qquad \begin{array}{r} \forall i \in D, \forall k \in K, \\ t = \{1...(T-c_{ji})\}, \\ h = \{1...H'\} \end{array} \tag{3.27}$$

$$f_{jik}^{th} + \sum_{j' \in \{0\} \cup L} \sum_{\substack{k' \in K, \\ k' \neq k}} \sum_{t'=t}^{t+t_l-1} f_{j'ik'}^{t'h} \leq 1 \qquad \begin{array}{r} \forall i \in D, \forall j \in \{0\} \cup L, \\ \forall k \in K, \\ t = \{1...(T - t_l + 1)\}, \\ h = \{1...H'\} \end{array} \tag{3.28}$$

$$l_{jk}^{(t+s)h} \geq l_{ik}^{th} + u_{jk}^{(t+s)h} - Q(1 - f_{ijk}^{(t+c_{ij})h}), \begin{cases} s = c_{ij} + t_l \end{cases} \qquad \begin{array}{r} \forall i \in V, \forall j \in D, \\ \forall k \in K, \\ t = \{1...(T - s)\}, \\ h = \{1...H'\} \end{array} \tag{3.29}$$

$$u_{jk}^{th} \leq Q \sum_{\substack{i:(j,i) \in E, \\ i \in L}} f_{jik}^{(t+c_{ji})h} \qquad \begin{array}{r} \forall j \in D, \forall k \in K, \\ t = \{1...(T - c_{ji})\}, \\ h = \{1...H'\} \end{array} \tag{3.30}$$

$$\sum_{h=1}^{H'} \sum_{t=1}^{T} \sum_{k \in K} u_{jk}^{th} = v_j \qquad \forall j \in D \tag{3.31}$$

$$y_{ij}^{h} \in \{0, 1\} \qquad \begin{array}{r} \forall i, j \in D \cup \{0\}, \\ h = \{1...H'\} \end{array} \tag{3.32}$$

$$x_{i}^{h} \in \{0, 1\} \qquad \forall i \in D \cup \{0\}, h = \{1...H'\} \tag{3.33}$$

$$f_{ijk}^{th} \in \{0, 1\} \qquad \begin{array}{r} \forall i, j \in V, \forall k \in K, \\ t = \{1...T\}, h = \{1...H'\} \end{array} \tag{3.34}$$

$$0 \leq l_{ik}^{th} \leq Q \qquad \begin{array}{r} \forall i \in V, \forall k \in K, \\ t = \{1...T\}, h = \{1...H'\} \end{array} \tag{3.35}$$

$$0 \leq u_{jk}^{th} \leq Q \qquad \begin{array}{r} \forall j \in D, \forall k \in K, \\ t = \{1...T\}, h = \{1...H'\} \end{array} \tag{3.36}$$

The objective function is given in (3.20). The WTs flow is set on constraints (3.21) to (3.24), while the dump trucks flow is given from (3.25) to (3.26). The synchronizations are respectively addressed in inequalities (3.27) for WTs and dump trucks, and in constraints (3.28) for dump trucks. MTZ constraints for the loading of dump trucks is provided in constraints (3.29), while dump trucks capacity is ensured by inequalities (3.30). Equalities (3.31) determine the overall cleaning of each debris node. The variables definition is given from (3.32) to (3.36).

Chapter 4

Heuristics and metaheuristics for the SRP-CD

In this chapter, the proposed heuristics and metaheuristics are presented. In Section 4.1, the constructive heuristics designed for SRP-CD based on greedy and random strategies are described. The LNS-based metaheuristics are introduced in Section 4.2. In the sequel, the hybrid metaheuristics are detailed in Section 4.3.

4.1 Constructive heuristics

As described in Dorigo and Stützle [2003], constructive heuristics often are able to generate feasible solutions of reasonable quality at low computational costs. In construction methods, a solution is built from scratch and iteratively, the methods add components to the solution without backtracking. The construction stops whenever the solution is complete.

A general scheme of a constructive heuristic is proposed here for SRP-CD, where the idea is to assign the WTs in debris nodes and define routes for dump trucks by synchronizing the vehicles with each other.

The constructive heuristics use a criterion for each decision level: the strategical and the operational. Six criteria are proposed for the strategic level of decision and seven for the operational, on a total of $6 \times 7 = 42$ constructive heuristics that produce a complete feasible solution for SRP-CD. In addition, a totally ran-

dom criterion is also used for the two levels. Sections 4.1.1 and 4.1.2 detail the deterministic heuristics and the random one, respectively.

The constructive heuristics generate initial feasible solutions for SRP-CD by performing the following four main steps:

(a) the assignment of WTs to debris nodes;

(b) routing dump trucks to debris nodes already with WTs;

(c) loading dump trucks at the debris nodes; and

(d) unloading dump trucks at the landfills.

Step (a) focuses on the strategical decisions of the WT scheduling, while steps (b), (c) and (d) are related to operational decisions of dump trucks' trips. The criteria for WTs are referred to as cWT, and for the dump trucks are referred to as cK. According to the chosen criteria, an assignment order is applied in the construction during steps (a) and (b). Step (c) performs the loading operation at each dump truck in the corresponding debris node. In step (d), full-loaded dump trucks are routed to the nearest landfill and the unloading operation is performed.

The general structure of the method to build a solution respecting the problem constrains is given in Algorithm 1. The inputs are: the graph $G = (V, E)$, the criteria (cWT), and (cK). The solution s is initialized in line 1. In line 2, an auxiliary set of debris nodes D$'$ receives a copy of D to preserve the original set. The loop in lines 4-9 is repeated until D$'$ becomes empty. At the end, in line 10, the constructed solution s is returned. The procedures in the algorithm are detailed in the following:

- $initialSet()$ - This procedure sets the current state of each vehicle (WTs and dump trucks) as *idle*. The route of each dump truck is set as *empty*, and a counter variable for the working days is initialized as one, standing for the first working day.

- *assignmentWT*() - This procedure assigns each *idle* WT in the current working day to a non-visited debris node in D', when applicable, using the criterion cWT. Then, the state of each assigned WT is set to *busy* and the assignments are registered in s.

- *routingDumpTrucks*() - In this procedure, new trips for *idle* dump trucks are defined in order to visit debris nodes in the current working day. Using the criterion cK, if the new trip does not surpass the working day (value T) and there is debris in the target node, the trip is set. Otherwise, the truck returns to the depot. The partial solution s is updated.

- *load*() - The load procedure performs the loading operation of each pair WT-dump truck and the truck is set to the *busy* state. When the debris node becomes empty, it is removed from D' and the corresponding WT is set back to *idle* for the next working day. If there is no more available debris nodes, the WT returns to the depot. The partial solution s is updated.

- *unload*() - The unload procedure sends each *busy* dump truck to the nearest landfill and performs the unloading operation. The partial solution s is updated. Then, the dump truck is set back to *idle*.

- *resetDay*() - In this procedure, the working day is considered finished when all the dump trucks have returned to the depot. Thus, the working day counter is incremented in one unit.

Algorithm 1: Building solution

Data: $G = (V, E)$, cWT, cK

Result: s

```
1  s ← ∅
2  D′ ← D
3  initialSet()
4  while D′ ≠ ∅ do
5      assignmentWT(s, cWT, D′)
6      routingDumpTrucks(s, cK, D′, G)
7      load(s, D′)
8      unload(s, L)
9      resetDay()
10 return s
```

The 6 greedy criteria and the random one cWT (detailed in Sections 4.1.1 and 4.1.2) and the 7 criteria and the random one cK (detailed in Section 4.1.1 and 4.1.2) define the way a solution is built. According to Dorigo and Stützle [2003], the simplest way is the choice of a random criterion to construct the solution. Another option is the choice of a greedy criterion to define the order which the components of a solution are included. They state that greedy criteria produce better quality solutions than a random criterion. However, greedy criteria are able to generate a limited number of solutions and during the greedy construction, the early decisions strongly affect later decisions leading to very poor moves.

4.1.1 Greedy constructive heuristics

A greedy criterion represents the choice of a benefit associated with some characteristic of the problem. At each iteration, the component which leads to the better benefit is chosen to be included in the partial solution [Dorigo and Stützle, 2003].

The goal of the proposed greedy criteria is to allow the constructive heuristic to quickly generate feasible solutions as a start to the metaheuristics. In general, the most difficult task is to satisfy the set of constraints and the solutions'

improvements are left for the metaheuristic.

Six greedy criteria are described for the assignment of WT and seven criteria are described for the routing of dump trucks, resulting in $6 \times 7 = 42$ variations of the constructive heuristic able to produce a complete feasible solution for SRP-CD. The six criteria for WTs are as follows:

- Less Debris First (LDF): In this criterion, the WTs are assigned to debris nodes with less amount of debris to collect. This is done in increasing order until all WT are assigned.

- More Debris First (MDF): In this criterion, the WTs are assigned in non increasing order of the biggest amount of debris.

- Smaller Travel Time First (STTF): For this criterion, the WTs are set to the nearest nodes from the depot, in terms of travel time, according to an ascending order.

- Greater Travel Time First (GTTF): In this criterion, the WTs are assigned to most distant debris nodes from the depot, in terms of a non increasing order of travel time.

- Smaller Ratio (Debris/Travel Time) First (SDTTF): For this criterion, a ratio between the amount of debris and the travel time from the depot is done. Then, WTs are assigned in increasing order of debris nodes with the smaller ratio.

- Greater Ratio (Debris/Travel Time) First (GDTTF): In this criterion, a similar ratio between the amount of debris and the travel time from the depot is done. However, the WTs are assigned in non increasing order of debris nodes with the greatest ratio.

The greedy criteria LDF, MDF, STTF, GTTF, SDTTF and GDTTF are also adapted for the operational level of decision and assigning dump trucks accordingly. For the criteria using travel time (STTF, GTTF, SDTTF and GDTTF), the following difference exists for dump trucks. Each criterion takes into account the

nearest or the farthest debris node from the current node where the truck is located (depot or landfill). Since dump trucks do not visit sites without WTs, dump trucks are set to debris nodes by considering the current WT scheduling. Other than the aforementioned criteria, Less Trucks First (LTF) is the seventh greedy criterion defined for routing dump trucks to debris nodes. The idea is to assign idle dump trucks to nodes with WTs, but without dump trucks, or either with less dump trucks. This strategy allows to spread the assignment of dump trucks. This is done in increasing order of the number of dump trucks already assigned at each debris node with WT.

Algorithm 2 builds the corresponding deterministic solution for every combination of criteria. In this method, the best solution s, the incumbent solution s', and the criteria sets $C1$ (for WT) and $C2$ (for dump trucks) are initialized in lines 1-4. The loops of lines 5-9 and 6-9 vary the sets of criteria, where in each iteration, the procedure *buildingSolution*() produces a new feasible solution, according to Algorithm 1. In lines 8-9, with the objective functions $\mathcal{F}_1$ and $\mathcal{F}_2$, the best solution found so far is updated. The algorithm returns the best solution in line 10. This heuristic is referred to as Greedy Constructive Heuristic (GCH).

Algorithm 2: Greedy constructive heuristic

Data: $G = (V, E)$
Result: s

1 $s \leftarrow \infty$
2 $s' \leftarrow \emptyset$
3 $C1 \leftarrow \{LDF, MDF, LDF, GDF, SDTTF, GDTTF\}$
4 $C2 \leftarrow \{LDF, MDF, LDF, GDF, SDTTF, GDTTF, LTF\}$
5 **forall** $cWT \in C1$ **do**
6 **forall** $cK \in C2$ **do**
7 $s' \leftarrow buildingSolution(G, cWT, cK)$
8 **if** ($(\mathcal{F}_1(s') < \mathcal{F}_1(s))$ or ($(\mathcal{F}_1(s') = \mathcal{F}_1(s))$ and $(\mathcal{F}_2(s') < \mathcal{F}_2(s))$)) **then**
9 $s \leftarrow s'$
10 return s

Figure 4.1 illustrates the partial solutions using the greedy criteria based on the transportation network detailed in Figure 4.1-(a) with $W = 2$ and $K = K_1, K_2$:

- Figure 4.1-(b) illustrates the LDF, where the allocation order through this

criterion is {*1,4,3,6*}. The arrows represent the assignment. Thus, the WT_1 is assigned to the debris node *1* and the WT_2 is assigned to *4*. The next *idle* WTs are assigned to *3* and *6*. For a tie-breaking rule, the debris node with the smaller *ID* is chosen.

- Figure 4.1-(c) depicts the MDF that produces the order {*3,6,4,1*}, where first the WT_1 is assigned to *3* and the WT_2 is assigned to *6*.
- Figure 4.1-(d) represents the STTF and it produces the allocation order {*6,1,4,3*}, in which WT_1 and WT_2 are assigned to *6* and *1* respectively.
- The criterion GTTF is showed in Figure 4.1-(e), the order is {*3,4,1,6*}. Thus, WT_1 is assigned to *3* and WT_2 is assigned to *4*.
- The SDTTF is presented in Figure 4.1-(f) and the order of the ratio is {*1,3,4,6*}. This way, WT_1 and WT_2 are respectively assigned to *1* and *3*.
- And, the GDTTF is shown in Figure 4.1-(g), the assignment order is {*6,4,3,1*}. Thus, the WT_1 and WT_2 are respectively assigned to *6* and *4*.

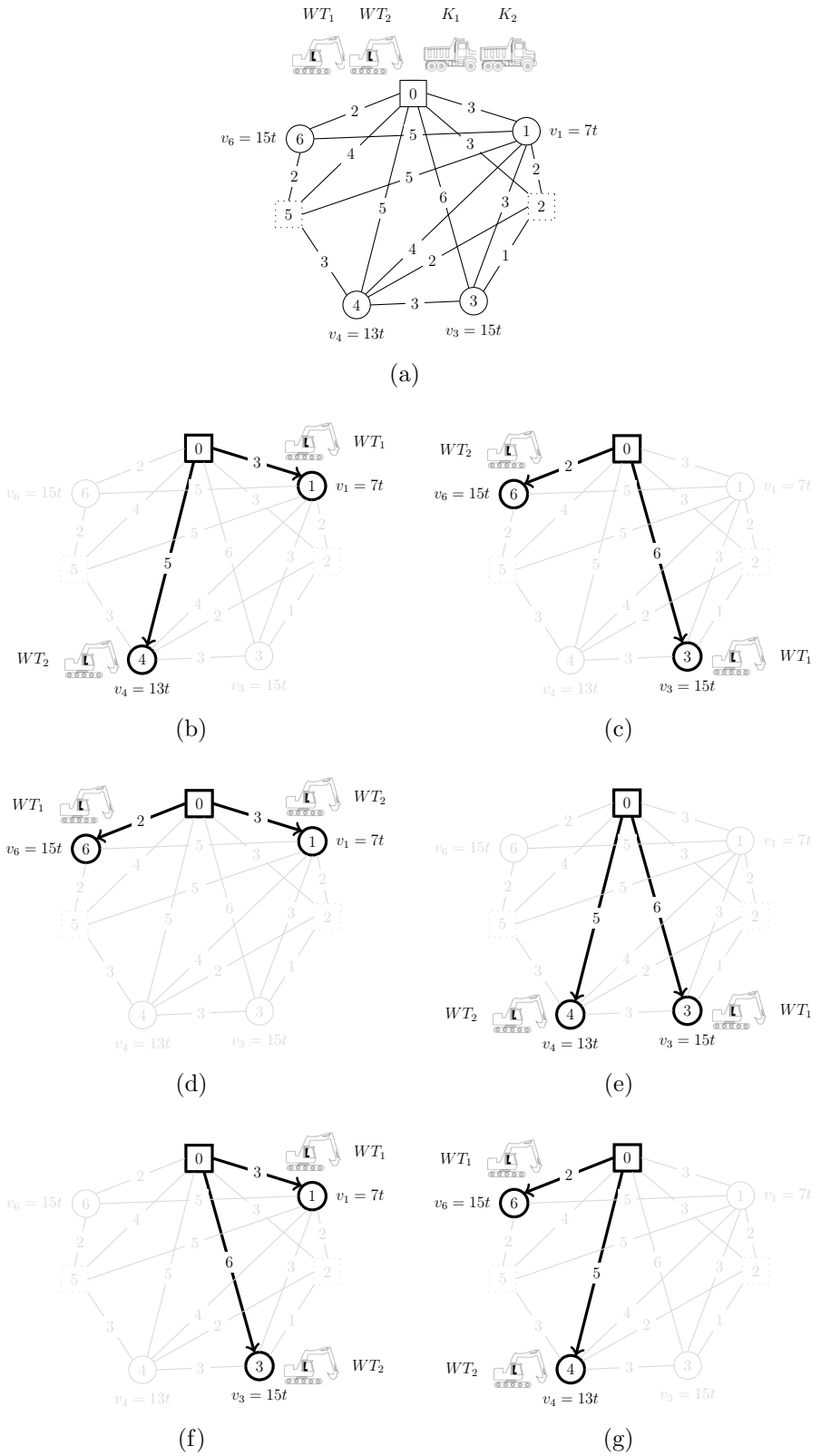


Figure 4.1: First assignments applying greedy criteria on WTs. A transportation network in (a) and the first assignments according to the criteria (b) LDF; (c) MDF; (d) STTF; (e) GTTF; (f) SDTTF; and (g) GDTTF.

Figure 4.2 illustrates the use of criteria for routing dump trucks based on the scheduling defined by Figure 4.1-(d), in which only the debris nodes {*1, 6*} can be visited by dump trucks. For this example in particular, Figure 4.2-(a) depicts the assignment for the dump trucks K_1 and K_2 by the criteria LDF, GTTF and SDTTF, where both trucks visit *1*. Figure 4.2-(b) shows the assignment for the dump trucks K_1 and K_2 by the criteria MDF, STTF and GDTTF, where both trucks visit *6*. Finally, Figure 4.2-(c) presents the assignment using the criterion LTF, where the dump trucks K_1 and K_2 are routed for the debris nodes *1* and *6* respectively.

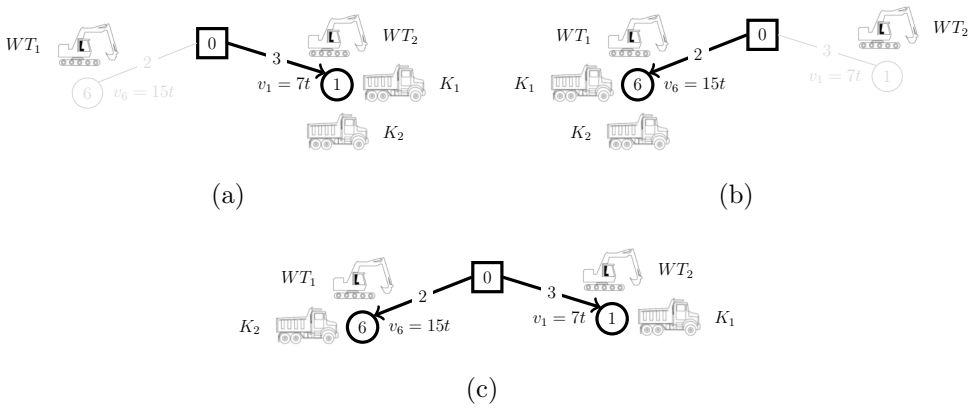


Figure 4.2: First assignments using the greedy criteria on routing dump trucks. The first assignments according to the criteria (a) LDF, GTTF and SDTTF; (b) MDF, STTF and GDTTF; and (c) LTF.

4.1.2 Random constructive heuristics

The Random Constructive Heuristic (RCH) builds a feasible solution using a random choice to assign WT to debris nodes, and to allocate dump trucks to debris nodes. RCH given in Algorithm 3 slightly differs from GCH, depicted in Algorithm 2, due to a random criterion (*RDM*) used for both strategical and operational levels. Furthermore, the RCH includes a stopping-criterion for the loop in lines 2-5. While the stopping-criterion is not met, at each iteration, the procedure *buildingSolution*() builds a new feasible solution, according to Algorithm 1 in line 3. The best solution found so far is assigned to *s* in lines 4-5. The algorithm returns the best solution found in line 6. Different variants of the RCH can be considered

according to the stopping-criterion. For instance, a fixed number of iterations, a fixed execution time or a fixed number of iterations without any improvements.

Algorithm 3: Random constructive heuristic

Data: $G = (V, E)$

Result: s

```
1 s ← ∞, s' ← ∅
2 while stopping-criterion not met do
3     s' ← buildingSolution(G, RDM, RDM)
4     if ( (F1(s') < F1(s)) or ( (F1(s') = F1(s)) and (F2(s') < F2(s))) ) then
5         s ← s'
6 return s
```

For sake of clarity, using the transportation network of Figure 4.1-(a), if a random choice for scheduling is considered, the first *idle* WT is randomly assigned to a debris node from {*1,3,4,6*}. Assuming it was assigned to *4*, the next *idle* WT is randomly assigned to a debris node from {*1,3,6*}. The same idea applies to dump trucks, but the possibilities of choice are limited by the debris nodes with assigned WTs.

4.2 Metaheuristics based on Large Neighborhood Search

The Large Neighborhood Search (LNS) is a metaheuristic that performs iterative improvements based in three main procedures: removal, insertion and acceptance. The LNS was first proposed by Shaw [1998]. The use of LNS for SRP-CD is motivated by its simplicity, and according to the literature, the high quality of solutions obtained in solving complex problems and in addressing difficult constraints.

The LNS removal procedure is a strategy employed to remove components from a current solution, which are re-inserted into a partial solution by means of an insertion procedure. This way, a new neighbor solution is generated. Then, the acceptance procedure decides how the neighbor solution found is accepted to be

the new current one. The iterations end when a stopping-criterion is reached.

In later studies such as Pisinger and Ropke [2010], the method has been successful applied for difficult $\mathcal{NP}$-problems. An additional description of the LNS was proposed, where the removal and insertion procedures were referred to as destroy and repair respectively. Different acceptance methods as well as some destroy and repair methods were discussed in Pisinger and Ropke [2010]; Ribeiro and Laporte [2012]; Ropke and Pisinger [2006]. For the acceptance, Shaw [1998] employed a procedure of accepting only the improving neighbor solutions, while Ribeiro and Laporte [2012] and Ropke and Pisinger [2006] adopted a strategy of accepting improving neighbor solutions and accept degrading solutions with a probability based on the principles of the Simulated Annealing, as shown in Section 4.3. In Pisinger and Ropke [2010], some destroy methods were described such as *random destroy, worst destroy, related/shaw destroy* proposed by Shaw [1998]. For the repair method, approximation algorithms, exact algorithms or greedy ones were also reported by the authors.

4.2.1 LNS for the SRP-CD

In the context of SRP-CD, LNS is applied since it seems to fit well the treatment of integrated difficult optimization problems with several constraints due to its performance and simplicity. One may note that in SRP-CD, a decision in the strategical level directly impacts the operational level, resulting in relevant modifications in the solution.

The proposed metaheuristic uses the removal and insertion procedures in the WTs scheduling as follows:

- Given a feasible solution, which is referred to as the incumbent solution, the removal procedure removes a random debris node from the solution (*Random-removal*). It is worth mentioning that removing a debris node from the solution implies that the vehicles' routes are also removed and need to be redesigned again after the insertion.

- In the sequel, two strategies were designed to insert the removed node into the partial solution, *Best Improvement* (BI) or *First Improvement* (FI). At each

insertion, the routing is reconstructed according to a greedy criterion (Section 4.1.1). For the BI, all possible insertions in the new solution are checked and the one that improves most the objective values in lexicographic order, compared to the incumbent solution, is held. For the FI, the first insertion in the new solution that improves the objective values in lexicographic order, in relation to the incumbent solution, is kept.

- The acceptance procedure follows the strategy adopted by [Shaw, 1998], in which after the remove-insertion operations, a neighbor solution is accepted to be the new incumbent solution if and only if it improves the incumbent one. For this reason, there is a cut procedure to optimize the construction of solution after an insertion. During the construction, the value of the first objective (number of days) is evaluated and compared to the corresponding value of the incumbent solution. Thus, the procedure rejects the new partial solution if this value is greater than the one in the incumbent solution.

The removal, insertion and acceptance procedures are applied iteratively until a stopping-criterion is met, either a fixed number of iterations, a fixed execution time or a fixed number of iterations without any improvements. Variants of LNS-based metaheuristic are obtained according to the initial solution (greedy or random), and the local search strategy (BI or FI).

A generic view of the LNS-based metaheuristic for the SRP-CD is given in Algorithm 4, where the following parameters are considered: the graph $G = (V, E)$, the criteria (cWT and cK), a removal method (cR), an insertion method (cI) and a criterion for routing (cK') to be applied after the insertion. The best solution (s), the incumbent solution (s') and an auxiliary one (s'') are initialized in line 1. Then, a feasible initial solution is generated in line 2 and the best solution is set in line 3. The loop in lines 4-11 is repeated until the predefined stopping-criterion is met. The auxiliary solution is initialized in line 5. The removal procedure is done in line 6, where s'' becomes a partial solution without the removed node stored in $\mathcal{R}$. Then, in line 7, the insertion procedure re-insert the node belonging to $\mathcal{R}$ into the partial solution s''. The condition in lines 8-9 updates the best solution found so far, if it applies. In the following, the acceptance method is used to update the

incumbent solution in lines 10-11. The best solution found is returned in line 12.

Algorithm 4: LNS-based metaheuristic

```
   Data: G = (V, E), cWT, cK, cR, cI, cK'
   Result: s
 1 s ← ∅, s' ← ∅, s'' ← ∅
 2 s' ← buildingSolution(G, cWT, cK)
 3 s ← s'
 4 while stopping-criterion not met do
 5 |   s'' ← s'
 6 |   R ← remove(s'', cR)
 7 |   insert(s'', R, cI, cK')
 8 |   if ( (F1(s'') < F1(s)) or ( (F1(s'') = F1(s)) and (F2(s'') < F2(s))) ) then
 9 |   |   s ← s''
10 |   if accept(s'', s') then
11 |   |   s' ← s''
12 return s
```

The criteria cWT, cK and cK' can be greedy or random. According to this choice, *LNS-GG* refers to cWT, cK and cK' set as greedy, *LNS-RG* indicates cWT and cK are random and cK' is greedy. Several variants of the proposed LNS can be obtained according to the initial solution construction and the local search type (BI or FI). Concerning these parameters, numerical experiments have been performed (see Section 5.2.3) to decide the better ones.

4.3 Hybrid method based on LNS and Simulated Annealing

A hybrid metaheuristic based on LNS (Section 4.2) and Simulated Annealing (SA) is proposed in this section, henceforth referred to as *LNS-SA*, similar to the strategy adopted by Ribeiro and Laporte [2012]; Ropke and Pisinger [2006].

The SA is a metaheuristic inspired by the process of physical annealing for crystalline solids, where after heating, the solid is slowly cooled to reduce their effects over the material, achieving a solid with a superior structural integrity [Henderson et al., 2003]. The SA was initially proposed by Kirkpatrick et al.

[1983], where the physical concept of annealing was adapted for some combinatorial optimization problems into iterative improvement. The solid is analogous to a solution of the corresponding problem, and the cooling process is analogous to the search for a better quality solution during the iterations of the method.

As described by the authors in Henderson et al. [2003], the SA considers an incumbent solution, a temperature ($\mathcal{T}$), a *cooling rate* (θ) and a stopping-criterion. While the stopping-criterion is not met, the method generates new neighbor solutions from the incumbent one at each iteration per temperature, in order to explore the search space. Then, $\mathcal{T}$ is cooled by the rate θ. Whenever the incumbent solution is improved, it is updated. The Boltzmann criterion is used to accept solutions that does improve the current one.

4.3.1 LNS-SA for the SRP-CD

The *LNS-SA* proposed for the SRP-CD couples the idea of removal-insertion operations from the LNS and the cooling scheme from the SA with the Boltzmann criterion as follows: the method considers the temperature ($\mathcal{T}$), the *cooling rate* (θ), a number of iterations per temperature (I_{max}), an initial feasible solution referred to as incumbent solution, and the stopping-criterion. While the stopping-criterion is not met, at each iteration, the neighbor solution is generated by applying the operations removal and insertion from LNS. The removal operation (cR) is the same aforementioned in Section 4.2, where a debris node is randomly removed from the scheduling of the solution, which becomes a partial solution. The insertion operation (cI) does not follow the former strategies BI or FI. Removed debris nodes are randomly inserted in the scheduling of the partial solution (*Random-insertion*). This was done in order to draw a random move in the method. Then, the routing is reconstructed by a greedy criterion (Section 4.1.1). Then, the best solution found so far is preserved. In relation to the incumbent solution, the neighbor solution is accepted to be the new incumbent solution when it is better or when the aforementioned Boltzmann criterion takes place. Then, the current temperature $\mathcal{T}$ is cooled by the rate θ at each I_{max} iterations.

Additional removal-insertion procedures are also incorporated in the *LNS-SA* as

follows: each $|D|$ iterations without improvements, a removal operation (cR') considers a number between 1 and $|D|/2$ of debris nodes which are randomly removed from the incumbent solution. Then, an insertion operation (cI') re-insert the nodes using the local searches (BI or FI). These procedures aim to create a very distant neighbor solution. However, they are time-consuming, thus $|D|$ iterations force the method to call them less often in large instances.

The general scheme of the *LNS-SA-based* is given in Algorithm 5, where $I_{max} = 1$. Some components are initialized in lines 1 and 2: the best solution (s), the incumbent solution (s') and the auxiliary solution (s'') and the current temperature ($\mathcal{T}$). In line 3, the initial feasible solution is built and assigned to s', and to s in line 4. An iteration counter (i) and a counter for iterations without improvements (j) are started in line 5. The cooling process occurs in the loop 6-17. According to the number of iterations without improvements, removal-insertion operations generate the neighbor solution (s'') in lines 9-10 or in lines 12-13. $\mathcal{F}$ represents the objective value of a solution, thus, $\Delta\mathcal{F}$ is calculated in line 14 and the best solution found is updated in line 16. In the lines 20-21, an evaluation of improving solutions or the Boltzmann criterion is done to accept the neighbor solution (s''). The temperature is cooled in lines 22-23 each I_{max} iterations. Finally, the best solution found is returned in the line 25.

Regarding the stopping-criterion and the parameters θ and $\mathcal{T}_0$, some strategies are discussed in the literature. According to Júnior et al. [2005]; Kirkpatrick et al. [1983], the SA is more effective when the method cools the temperature very slowly, for this reason, as used in Júnior et al. [2005], it is assumed a value $\theta = 0.998$. For the stop-criterion, Júnior et al. [2005] defined a threshold (ϵ) for the temperature close to zero to interrupt the method. Following this strategy, it is proposed to use a value $\epsilon = 10^{-4}$ as limit for the temperature $\mathcal{T}$, as in Júnior et al. [2005].

Concerning $\mathcal{T}_0$, Kirkpatrick et al. [1983] report that higher initial temperatures suggest large changes in the objective function. In addition, Ben-Ameur [2004] describes the choice of a temperature, which ensures a high acceptance of neighbor solutions at the beginning of the method, for example, 80%. For this reason, Júnior et al. [2005] defined a strategy to calculate $\mathcal{T}_0$ shown in Algorithm 6. The method

Algorithm 5: LNS-SA-based metaheuristic

Data: $G = (V, E)$, cWT, cK, $\mathcal{T}_0$, θ, cR, cI, cK', I_{max}, cR', cI'
Result: s

```
1  s ← ∅, s' ← ∅, s'' ← ∅
2  T ← T_0
3  s' ← buildingSolution(G, cWT, cK)
4  s ← s'
5  i ← 1, j ← 0
6  while stopping-criterion not met do
7      s'' ← s'
8      if j < |D| then
9          R ← remove(s'', cR)
10         insert(s'', R, cI, cK')
11     else
12         R ← remove(s'', cR')
13         insert(s'', R, cI', cK')
14         j ← 0
15     ΔF ← (F(s'') − F(s'))
16     if (F(s'') < F(s)) then
17         s ← s''
18         j ← 0
19     else
20         j ← j + 1
21     if ((ΔF < 0) or (rnd(0,1) < e^{−ΔF/T})) then
22         s' ← s''
23     if (i mod I_max = 0) then
24         T ← θ.T
25     i ← i + 1
26 return s
```

simulates the first iteration of the SA and the idea is to find an initial temperature that guarantees a high acceptance. Thus, the algorithm considers an initial solution (s'), a number of iterations per temperature (I_{max}), an acceptance probability (γ) and a temperature increment rate (λ), where $\lambda > 1$. In line 1, $\mathcal{T}_0$ starts with a low value. The loop in lines 3-12 stops when the desired acceptance is met. $\mathcal{N}$ represents the generation of a neighbor solution. The number of accepted solutions is set as zero in line 4. Thus, for each temperature, the loop in lines 5-9 counts the number of accepted solutions. In case the desired acceptance is not met in lines 10-11, the temperature is increased by λ. In line 13, the resulting temperature is

returned.

Algorithm 6: Initial temperature for SA

```
   Data: s', I_max, λ, γ
   Result: T_0
1  T_0 ← 2
2  accepted ← 0
3  repeat
4      accepted ← 0
5      for i ← 1 to I_max step 1 do
6          s'' ← N(s')
7          ΔF ← (F(s'') − F(s'))
8          if ((ΔF < 0) or (rnd(0,1) < e^{−ΔF/T_0})) then
9              accepted ← accepted + 1
10     if accepted < γ × I_max then
11         T_0 ← λ.T_0
12 until accepted ≥ γ × I_max;
13 return T_0
```

A preprocessing phase is discussed for the initial temperature in Section 5.2.3. In addition, numerical experiments are considered to decide the better parameters for the method. One can noticed that the same description regarding the parameters cWT, cK and cK' in Section 4.2 can be considered for the *LNS-SA-based* metaheuristic with the variants *LNS-SA-GG* and *LNS-SA-RG*.

Chapter 5

Computational experiments

This chapter is devoted to the computational experiments involving the proposed methods in order to test their performance in terms of the quality of the solutions and the running time. In Section 5.1, a data generation mechanism is introduced in Section 5.1.1. In Section 5.1.2, a set of realistic scenarios is proposed based on a case study of the Port-au-Prince, Haiti earthquake in 2010. The numerical results are presented and discussed in Section 5.2, where results for the mathematical model are discussed in Section 5.2.1. The greedy constructive heuristics are ranked in Section 5.2.2. Tuning experiments are described in Section 5.2.3, the proposed heuristic results using the benchmark of instances are detailed in Section 5.2.4 and analysis of robustness are described in Section 5.2.5. Results for the case study are discussed in Section 5.2.6.

The tests were performed on an Intel(R) Core(TM) i5-8350U CPU @1.70GHz with 16GB of RAM and 2GB of swap, using the operational system Ubuntu 18.04.1 LTS. All the cores and RAM were used. The proposed approaches were developed in C++. Moreover, *IBM ILOG CPLEX* 12.8, under default parameters, and *CPLEX Concert Technology* in C++ were used to run the mathematical formulation.

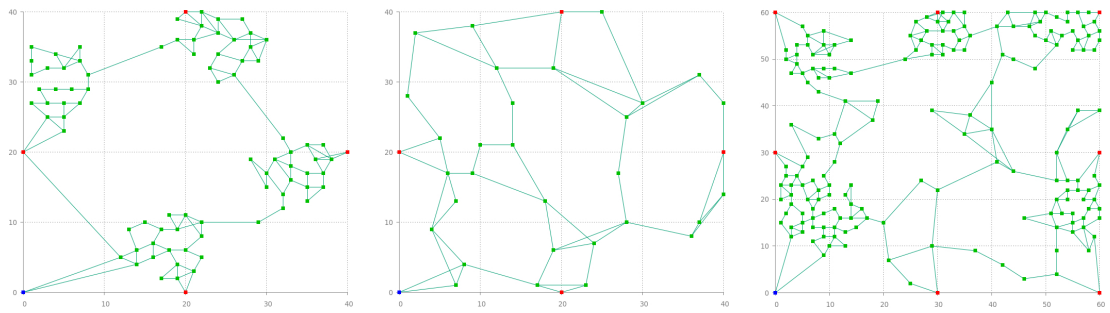
5.1 Data generation

5.1.1 Benchmark of instances

A data generation mechanism is introduced below. An instance format was developed containing information of all the relevant features of the SRP-CD. It is worth mentioning that the input graph is not necessarily complete, in this case, a distance matrix for each pair of nodes is created, during a preprocessing phase, by using Dijkstra algorithm [Dijkstra, 1959].

The generator was developed using C++ and Linux shell script, which makes a geographical distribution of debris nodes over a cartesian plane, inspired by urban districts' spacial organisation, and also Solomon's benchmark [Solomon, 1987]; Resulting in three network topologies, referred here as (C)luster, (R)andom, and (M)ix. As in Solomon [1987], the Euclidean distance was adopted for each edge to represent the travel time between two nodes. The single depot is fixed at the border of the generated data (*i.e.* outside of the affected area), the landfills areas vary from 1 up to 7 fixed areas, and the axis represent time units.

Figure 5.1 shows three examples generated by the algorithm with the depot in blue, the landfills in red and the debris nodes in green, 5.1-(a) depicts an instance with a cluster topology containing 80 debris nodes and 4 landfills, 5.1-(b) presents an instance with a random topology containing 30 debris nodes and 4 landfills, and 5.1(c) shows an instance with the mix topology containing 180 debris nodes and 7 landfills.



(a) (b) (c)

Figure 5.1: Example of generated instances with different network topologies.

In order to perform the tests, three sets of instances were created: S0 (small-

size), S1 (medium-size) and S2 (large-size). For the set S0, 36 instances were generated: 12 with a cluster topology (C), 12 for the random topology (R) and 12 with a mix topology (M). For each set S1 and S2, 90 instances were generated, 30 with a cluster topology (C), 30 for the random topology (R) and 30 with a mix topology (M). The characteristics of the instances' sets are presented in Table 5.1, where the columns represent respectively the number of instances (#Inst.), the number of debris nodes ($|D|$), the number of landfills ($|L|$), the number of WTs (W), the number of dump trucks ($|K|$), the capacity of dump trucks (Q), the time units for loading (t_l), unloading (t_u) and the working day (T). In S0, each instance has a number of dump trucks equals to or greater than the number of WTs. In S1 and S2, for each topology and each $|D|$, the instances have a number of dump trucks equals to or two times the number of WTs. The figures of the generated graphs for the instances in S1 and S2 are depicted in Appendix C.

	# Inst.	$\|D\|$	$\|L\|$	W	$\|K\|$	Q	t_l t_u	T
S0	36	5	1	2	2, 3	2	1	30, 35, 40
				3	3, 4			
S1	90	10, 20, 30, 40, 50	2	2	2, 4	2	1	40, 55, 60, 65
				3	3, 6			
				4	4, 8			
S2	90	100, 200, 300, 400, 500	3	5	5, 10	2	10	480, 720
				6	6, 12			
				7	7, 14			

Table 5.1: Characteristics of the instance sets.

5.1.2 A case study of Port-au-Prince earthquake, 2010

In January 12th, 2010, an earthquake of 7.0 magnitude struck Port-au-Prince, Haiti. It was followed by several aftershocks with magnitudes over 6.0 magnitude, causing major damages and casualties. After the event, approximately 400,000 buildings were affected, where 80,000 had suffered serious damage.

The case study for the earthquake of Port-au-Prince assembles information

from different sources: real geographic data were provided by the International Charter "Space and Major Disasters" [ICSMD, 2021] and transcribed into a graph by the Regional Image Processing and Remote Sensing Service [SERTIT, 2021] (a laboratory of rapid mapping); the number of dump trucks and work troops were obtained by satellite images from Google Earth; and the debris volume (10 million cubic meters) has been obtained by a United Nation report [UNDP, 2013].

The graph $G(V, E)$ of the aforementioned case study is depicted in Figure 5.2, where $|V| = 5,764$, $|E| = 7,301$, there is a single depot, a landfill and debris nodes, respectively given in blue, red and black dots. Using this graph, a set of realistic scenarios have been built by adding or computing the following additional information:

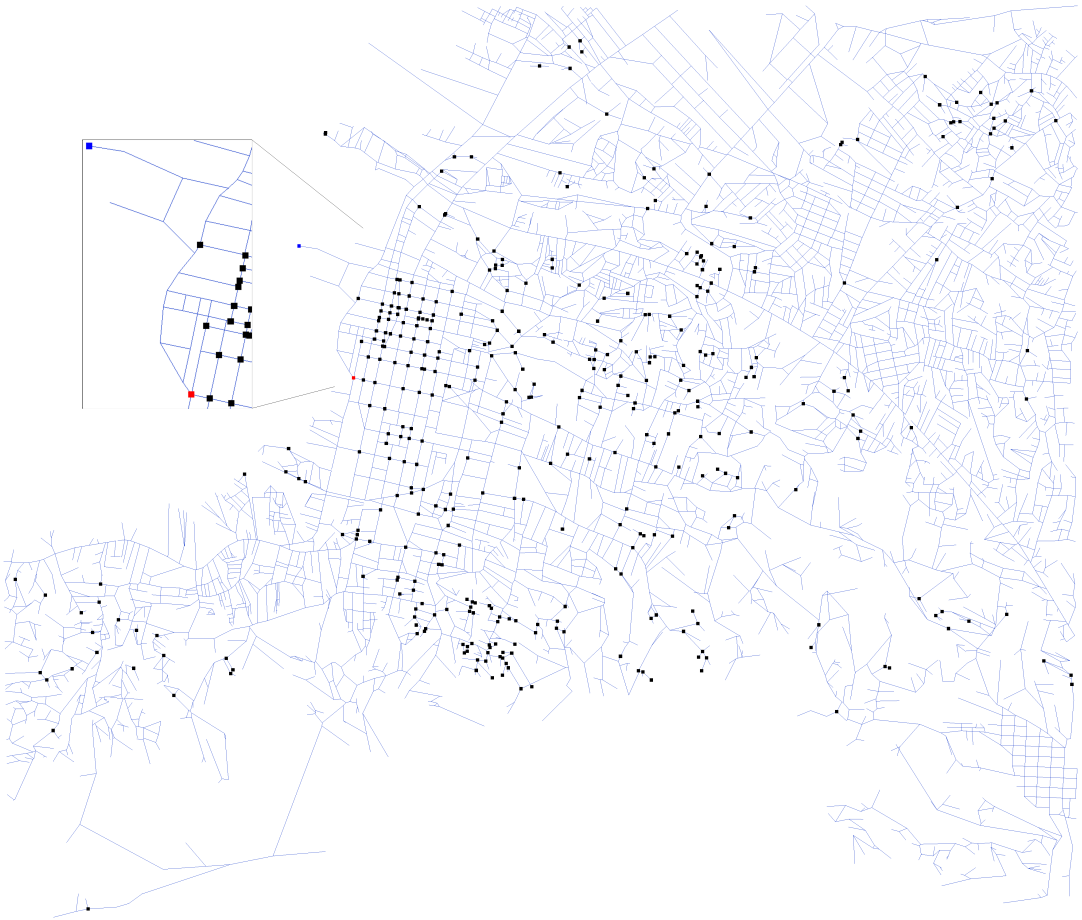


Figure 5.2: Graph of Port-au-Prince, Haiti in 2010

- Amount of debris: a distribution of debris was done by placing the amount of 10 million cubic meters. Such distribution is computed using an initial

amount of debris per route provided by the SERTIT. This amount is associated with the extremities nodes of a blocked route. In total, $|D| = 390$ debris nodes were established.

- Travel time c_{ij}: as the length of each edge was provided by SERTIT, a travel time c_{ij} for dump trucks, in seconds, was calculated based on an average speed of $30km/h$.

- Time periods and service times: three different working days in hours were considered, $T = 8$ hours, $T = 10$ hours and $T = 12$ hours. Loading and unloading were set to $t_l = 10$ and $t_u = 5$ minutes respectively.

- Number of WTs, dump trucks and capacity: as depicted in Figure 5.3, up to 21 WTs and 83 dump trucks were identified and highlighted respectively by yellow and red circles. The dump trucks' capacity was set to $Q = 20$ cubic meters, according to a technical description of similar dump trucks identified in the pictures [Sinotruk, 2021].

The time and the distances were converted, respectively in seconds and meters. The characteristics of each scenario instance are summarized in Table 5.2. The columns represent respectively the working day (T) in hours, the number of debris nodes ($|D|$), the number of landfills ($|L|$), the number of WTs (W), the number of dump trucks ($|K|$), the capacity of dump trucks (Q), the time units for loading (t_l) and for unloading (t_u).

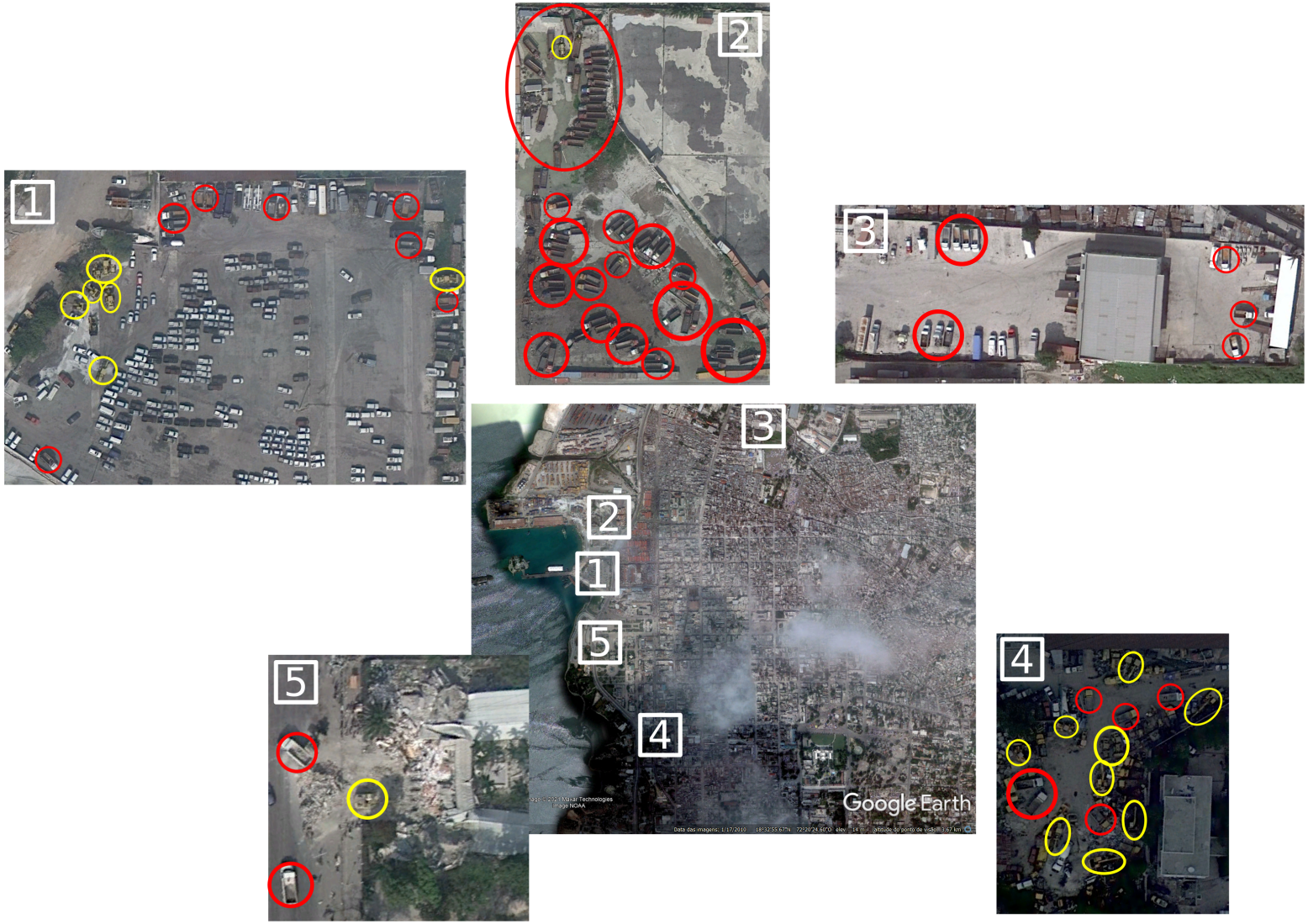


Figure 5.3: Port-au-Prince satellite image of January, 17th in 2010. Source: Google Earth [Gorelick et al., 2017]

T	$\|D\|$	$\|L\|$	W	$\|K\|$	Q	t_l	t_u
8h	390	1	21	21 42 83	$20m^3$	10 *min.*	5 *min.*
10h	390	1	21	21 42 83	$20m^3$	10 *min.*	5 *min.*
12h	390	1	21	21 42 83	$20m^3$	10 *min.*	5 *min.*

Table 5.2: Scenarios' characteristics.

5.2 Numerical results

The benchmarks of instances set S0 are used to test the mathematical model. The benchmarks of instances sets S1 and S2 are used to measure robustness and scalability of the proposed heuristics. Extensive numerical experiments have been done with the proposed methods and are detailed in this section.

5.2.1 Mathematical model results

For the sake of clarity, in order to strengthen the model, each instance has an upper bound value for the time horizon H calculated by Equation (5.1), where σ is pre-computed as follows: given all shortest paths in G, for each debris node $i \in D$, let ζ_i be the value of one complete route starting and ending at the depot, such as $\zeta_i = c_{0i} + t_l + c_{ij} + t_u + c_{j0}$, where $j \in L$ is the nearest landfill from i. Thus, σ is set to the value of the longest route ($\sigma = max(\zeta_i), i \in D$).

$$H = (\sum_{i \in D} v_i)/[(|K| \times Q \times T)/\sigma] \tag{5.1}$$

The results for the small-size instances in S0 are showed in Tables 5.3, 5.4 and 5.5, respectively for C, R and M topologies. The model was given CPU time limit of 2 hours for each objective function, which was set based on some studies from

literature [Souza et al., 2010; Soares et al., 2019]. The first three columns present the instances characteristics T, W and $|K|$. For each objective, the upper bound (bold in this column means the optimally has been proved), the time in seconds, the GAP, the number of nodes opened in the branch-and-cut tree (Nb_nodes), the number of variables (Nb_var.) and the number constraints (Nb_const.) are given.

In Tables 5.3, 5.4 and 5.5, the number of variables and constraints increases proportionally as the T, W or $|K|$ raises. In Table 5.3, the model was able to prove optimally the values for $\mathcal{F}_1$ and for $\mathcal{F}_2$, except in the instances with $T = 40$, $W = 2$ and $|K| = 2$, $W = 3$ and $|K| = 3$, and $W = 3$ and $|K| = 4$, where the combinatorial raised, and the formulation spent the time limit, with a GAP up to 3.55% in $\mathcal{F}_2$. In Table 5.4, the model was able to optimally prove the values for $\mathcal{F}_1$ and for $\mathcal{F}_2$ in all instances. In Table 5.5, the model was able to prove optimally the values for $\mathcal{F}_1$ and for $\mathcal{F}_2$, except for the instances with $T = 35$, $W = 2$ and $|K| = 2$, $W = 3$ and $|K| = 3$, and $W = 3$ and $|K| = 4$, where the combinatorial raised, and the formulation spent the time limit, with a GAP up to 1.76% in $\mathcal{F}_2$.

Instances			$\mathcal{F}_1$						$\mathcal{F}_2$					
T	W	$\|K\|$	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.
30	2	2	**13**	1	0.00	0	22309	34136	**598**	0	0.00	0	41808	34071
	2	3	**9**	1	0.00	0	33203	51101	**598**	1	0.00	0	62452	51056
	3	3	**9**	2	0.00	0	33203	51101	**598**	0	0.00	0	62452	51056
	3	4	**7**	1	0.00	0	44097	68066	**598**	1	0.00	0	83096	68031
35	2	2	**13**	1	0.00	0	26079	39284	**598**	1	0.00	0	48828	39219
	2	3	**9**	4	0.00	0	38858	58823	**598**	1	0.00	0	72982	58778
	3	3	**9**	3	0.00	0	38858	58823	**598**	1	0.00	0	72982	58778
	3	4	**7**	3	0.00	0	51637	78362	**598**	1	0.00	0	97136	78327
40	2	2	**7**	7	0.00	0	29849	44354	432	7200	0.23	12591141	55848	44319
	2	3	**5**	7	0.00	0	44513	66428	**435**	609	0.00	1156068	83512	66403
	3	3	**5**	5	0.00	0	44513	66428	432	7200	3.55	9442795	83512	66403
	3	4	**4**	5	0.00	0	59177	88502	432	7202	2.49	11763212	111176	88482

Table 5.3: Results for the mathematical model in the instances with cluster topology and $|D| = 5$.

The mathematical formulation was able to prove optimally for the two objective functions in 9 small instances (resp. 3 instances for each topology, C, R and M) with $|D| = 5$, $W = 2$ and $|K| = 3$. In Table 5.3, the results for $T = 30$ and $T = 35$ are similar because increasing the range of a working day from 30 to 35 time units was not enough to allow more dump truck's travels for debris nodes in a single day. It can be noticed that for $T = 40$ time units, the solution is improved. This is due the fact that dump trucks remove more debris in a day. However, the most

Instances			$\mathcal{F}_1$						$\mathcal{F}_2$					
T	W	$\|K\|$	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.
30	2	2	**9**	4	0.00	0	25621	36236	**401**	2	0.00	11	48120	36191
	2	3	**6**	4	0.00	0	38131	54236	**404**	34	0.00	22279	71880	54206
	3	3	**6**	3	0.00	0	38131	54236	**401**	1	0.00	0	71880	54206
	3	4	**5**	1	0.00	0	50641	72236	**401**	1	0.00	0	95640	72211
35	2	2	**8**	12	0.00	0	29971	42086	**391**	1	0.00	0	56220	42046
	2	3	**6**	14	0.00	721	44656	63011	**391**	0	0.00	0	84030	62981
	3	3	**5**	8	0.00	0	44656	63011	**391**	0	0.00	0	84030	62986
	3	4	**4**	6	0.00	0	59341	83936	**391**	1	0.00	0	111840	83916
40	2	2	**7**	18	0.00	0	34321	47936	**381**	7	0.00	2846	64320	47901
	2	3	**5**	23	0.00	0	51181	71786	**381**	3	0.00	490	96180	71761
	3	3	**5**	16	0.00	0	51181	71786	**381**	3	0.00	1230	96180	71761
	3	4	**4**	14	0.00	0	68041	95636	**381**	4	0.00	3870	128040	95616

Table 5.4: Results for the mathematical model in the instances with random topology and $|D| = 5$.

Instances			$\mathcal{F}_1$						$\mathcal{F}_2$					
T	W	$\|K\|$	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.
30	2	2	**15**	0	0.00	0	26041	40016	**649**	0	0.00	0	48540	39941
	2	3	**10**	1	0.00	0	38761	59906	**649**	0	0.00	0	72510	59856
	3	3	**10**	1	0.00	0	38761	59906	**649**	1	0.00	0	72510	59856
	3	4	**8**	6	0.00	109	51481	79796	**649**	1	0.00	0	96480	79756
35	2	2	**8**	5	0.00	0	30391	45956	398	7380	1.51	5014235	56640	45916
	2	3	**5**	7	0.00	0	45286	68816	**398**	58	0.00	59943	84660	68791
	3	3	**5**	4	0.00	0	45286	68816	398	7202	1.76	18661674	84660	68791
	3	4	**4**	3	0.00	0	60181	91676	398	7237	1.19	18221662	112680	91656
40	2	2	**6**	11	0.00	63	34741	51806	**358**	452	0.00	281403	64740	51776
	2	3	**4**	13	0.00	0	51811	77591	**350**	65	0.00	115223	96810	77571
	3	3	**4**	9	0.00	0	51811	77591	**339**	526	0.00	666851	96810	77571
	3	4	**3**	10	0.00	0	68881	103376	**339**	11	0.00	5131	128880	103361

Table 5.5: Results for the mathematical model in the instances with mix topology and $|D| = 5$.

relevant is to show the limits of the model. For this purpose, we took the 18 smaller instances (resp. 6 for each topology) of S1 with $|D| = 10$ and provide the results in Table 5.6. The model was given CPU time limit of 2 hours for each objective function. The first three columns present the instances characteristics: topology type, number W of WTs and number $|K|$ of dump trucks. The other columns are similar to the previous tables.

The mathematical formulation was able to prove optimally for the first objective function for 7 out of 18 instances. For the cases where the model did not find an optimum value, it is worth trying to find an interesting solution for the routing. Obviously, whenever the W and $|K|$ increase, a large number of variables and constraints are required. When the number of W and $|K|$ are smaller as it is the case of the instances with $W = 2$ and $|K| = 2$, $W = 3$ and $|K| = 3$, $W = 2$ and $|K| = 4$, and $W = 3$ and $|K| = 6$, the combinatorial raised, and the formulation spent the

time limit, in some cases, without proving optimality. Using more resources such as with $W = 4$ and $|K| = 8$, the model was able to prove optimality for the first objective function, except for the corresponding instance with the mix topology. The GAPs for the scheduling are high, while for the routing, the GAPs are within 6%, except for one instance in mixed topology, with $W = 2$ and $|K| = 4$, where the GAP is about 11%. This aspect opens some avenues of research, such as the development of valid inequalities, cuts or reformulation. Once the scheduling are defined, the model is able to manage well the travels and the synchronization of dump trucks, except for the random instance with $W = 4$ and $|K| = 8$ which resulted in out of memory issues.

Instances			$\mathcal{F}_1$						$\mathcal{F}_2$					
	W	$\|K\|$	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.
C	2	2	19	7200	10.53	102621	160421	238441	1052	7200	4.28	117100	372580	238251
	3	3	12	7200	8.33	291829	238941	357261	1052	7201	4.28	258961	557180	357141
	4	4	**9**	468	0.00	4205	317461	476081	1052	7201	3.90	472127	741780	475991
	2	4	10	7200	10.00	196977	317461	476081	1072	7201	4.12	53730	741780	475981
	3	6	7	7200	14.29	1113730	474501	713721	1063	7201	5.04	160799	1110980	713651
	4	8	**5**	250	0.00	0	631541	951361	1060	7201	4.51	249279	1480180	951311
R	2	2	21	7200	9.52	203639	154501	236021	1184	7201	4.10	773956	358500	235811
	3	3	**13**	447	0.00	4283	230126	353646	1184	7201	4.10	1296215	536125	353516
	4	4	**10**	217	0.00	1954	305751	471271	1184	7201	4.10	900245	713750	471171
	2	4	11	7200	9.09	327932	305751	471271	1184	7202	4.10	395484	713750	471161
	3	6	**7**	447	0.00	1742	457001	706521	1184	7202	4.10	1333698	1069000	706451
	4	8	**5**	227	0.00	2018	608251	941771	-	-	-	-	-	-
M	2	2	18	7200	16.67	129199	130327	198597	964	7201	5.29	804928	301686	198417
	3	3	11	7201	9.09	476570	194125	297570	964	7202	5.29	1444497	451164	297460
	4	4	**8**	490	0.00	36538	257923	396543	964	7202	4.88	834157	600642	396463
	2	4	10	7200	20.00	137853	257923	396543	1028	7200	11.19	107320	600642	396443
	3	6	6	7200	16.67	181384	385519	594489	964	7204	5.29	352682	899598	594429
	4	8	5	7208	20.00	1264228	513115	792435	964	7201	3.79	1131294	1198554	792385

Table 5.6: Results for the mathematical model in the instances with $|D| = 10$.

5.2.2 Greedy constructive heuristics comparison

In this section, the 42 different combinations of criteria for the greedy constructive heuristics (see Section 4.1.1) are compared by ranking analysis, absolute rank (Abs-Rank) and the average rank (Avg-Rank):

- The Abs-Rank gives a reward value 1 for each heuristic that found, for a given instance, the best solution value of the scheduling ($\mathcal{F}_1$) and the routing ($\mathcal{F}_2$) among all the others heuristics. In case of two or more heuristics finding the same best solution value, both will be rewarded. The sum of all rewards

considering all the instances provide how many times a heuristic found the best solution value among all the others for a set of instances. The higher the final value is the better the heuristic by this classification is.

- The Avg-Rank defines a classification for the heuristics as follows. The heuristic that produces the best solution value among all the others receives the rank number 1; the heuristic which finds the second best solution value receives the rank number 2 and so on. If two or more heuristics find the similar solution value, then both will be set with the average of the corresponding individual ranks. For example, if three heuristics found the second best solution value, instead of receive the ranks 2, 3 and 4, they receive a rank value equals to $(2+3+4)/3 = 3$. The average of all Avg-Rank for every heuristic, considering all instances, indicates the average classification of a heuristic among all the others. The smaller the final value is the better the heuristic by this classification is.

The ranking analysis for each set of instances are presented in Table 5.7, where the best results are in bold. In the cell K\W, W indicates the criteria for worktroops (scheduling), and K indicates the criteria for dump trucks (routing).

		Abs-Rank						Avg-Rank					
	K \ W	LDF	MDF	STTF	GTTF	SDTTF	GDTTF	LDF	MDF	STTF	GTTF	SDTTF	GDTTF
S1	LDF	1	1	0	1	12	0	19.84	29.17	24.97	21.12	14.92	23.04
	MDF	0	6	0	0	0	0	29.29	19.24	27.99	20.66	26.96	29.40
	STTF	**17**	13	5	0	5	0	12.34	11.09	11.99	23.22	20.71	12.63
	GTTF	0	0	0	1	0	0	34.87	35.78	33.09	20.19	25.91	34.81
	SDTTF	0	0	0	1	0	0	34.52	35.53	32.06	19.33	16.52	33.30
	GDTTF	10	15	0	0	0	3	14.91	**10.49**	17.14	21.99	24.93	11.93
	LTF	9	6	3	2	3	2	11.04	11.07	11.29	11.54	11.06	11.09
S2	LDF	0	0	0	0	0	0	30.65	30.52	15.84	13.91	20.74	21.27
	MDF	0	0	0	0	0	0	31.37	30.10	16.77	15.61	27.14	27.69
	STTF	10	**21**	12	6	5	4	6.44	6.07	**5.78**	6.56	7.67	8.10
	GTTF	0	0	0	0	0	0	40.36	40.65	25.37	24.53	36.29	36.43
	SDTTF	0	0	0	0	0	0	40.35	40.48	22.38	22.17	33.52	32.98
	GDTTF	7	8	1	2	0	0	6.83	6.52	10.07	9.54	16.38	13.56
	LTF	0	0	6	6	0	0	27.63	27.00	12.02	11.81	21.50	22.42

Table 5.7: Ranking analysis for greedy criteria in both sets.

Results for S1 instances indicate that the best Abs-Ranks and Avg-Ranks are respectively LDF for the scheduling and STTF for the routing, and MDF for the scheduling and GDTTF for the routing. Concerning the set S2 of instances, the best

Abs-Rank and Avg-Rank are respectively MDF for the scheduling and STTF for the routing, and STTF for the scheduling and routing. Some greedy criteria overcome the others in terms of rankings in the two set of instances (STTF and MDF), although with different matches (LDF-STTF, MDF-GDTTF, MDF-STTF, STTF-STTF) for the two levels of optimization and according to the set of instances. This is the case of MDF for the scheduling level of decision, which has obtained the best absolute ranking for S2 and the best average ranking for S1; and STTF for the routing, which obtained the best absolute ranking for both S1 and S2.

The results pointed to four criteria from the highlighted combinations, LDF, MDF, STTF and GDTTF. Thus, these criteria are used to reduce the number of possibilities in the metaheuristics tuning in the following section.

5.2.3 Preprocessing and metaheuristics tuning

In this section, preprocessing and tuning experiments are described for the metaheuristics. The tuning was done using the Iterated racing for automatic algorithm configuration (IRACE) package [López-Ibánez et al., 2016] for both LNS and LNS-SA based methods, which works with a machine learning mechanism. The tuning experiments are very time-consuming using IRACE. Thus, S1 set of instances was used since it is a good compromise between the size of the instances and the time required to run the experiments. S1 contains medium-size instances and allowed to obtain a more reliable result in a reasonable running time.

Tuning the LNS-based method

In the LNS-based method described in Section 4.2.1, preprocessing tests were done to decide between the stopping criteria, and the results pointed to the criterion: number of iterations without improvements. Thus, the following parameters were tuned using IRACE with the set of instances S1:

(a) considering the initial solutions, the four criteria (LDF, MDF, STTF and GDTTF) mentioned in Section 5.2.2 were used for both scheduling and routing (criteria cWT, cK and cK');

(b) two types of local search for the insertion criterion (cI): BI and FI; and

(c) stopping criteria considering {50, 60, 70, 80, 90, 100} iterations without improving the best solution found by the corresponding heuristic.

The tuning was done considering the three topologies in order to find the parameters which raise the best results for each one. These results are presented in Table 5.8.

Topology	cWT	cK	cI	cK'	stopping criterion
C	STTF	STTF	FI	STTF	100
R	STTF	GDTTF	BI	STTF	90
M	STTF	LDF	FI	STTF	100

Table 5.8: LNS-based method parameters calibrated by the IRACE.

The results indicate the unique parameter that differs between the cluster and mix topologies is the cK criterion. It is noticeable that the random topology has a set of different parameters from the others ones such as cK, cI and the stopping criterion.

Preprocessing and tuning the LNS-SA-based method

For the LNS-SA-based metaheuristic (Algorithm 5), a preprocessing phase was done before the tuning experiments to calculate the initial temperature. Using Algorithm 6, $\mathcal{T}_0$ was calculated for each instance of S1 considering $I_{max} = |D|$ and each solution built by the methods *GCH* and *RCH* as initial. Thus the biggest $\mathcal{T}_0$ found was adopted to the LNS-SA-based metaheuristic.

Next, the following parameters were tuned using IRACE for the *LNS-SA-based* with the set of instances S1:

(a) the criteria cWT, cK and cK' considering LDF, MDF, STTF and GDTTF mentioned in Section 5.2.2; and

(b) the criterion cI' considering BI and FI.

The results of the tuning are presented by topology in Table 5.9. The results indicate the parameters for cluster and mix topologies are the same. The random

topology has the parameters cWT and cK different parameters from the others ones.

Topology	cWT	cK	cI'	cK'
C	LDF	GDTTF	BI	STTF
R	STTF	LDF	BI	STTF
M	LDF	GDTTF	BI	STTF

Table 5.9: LNS-SA-based method parameters calibrated by the IRACE.

5.2.4 Heuristics comparison

The results described in this section take into account the performance in terms of the quality of the solutions and the running time of each method in S1 and S2. For the S0 instances where CPLEX proved optimality, the proposed methods also proved optimality for $\mathcal{F}_1$ and for $\mathcal{F}_2$ in 26 out of 30 instances. The evaluated methods are the constructive heuristics *GCH* and *RCH* (using a fixed execution time of 20 seconds as stopping-criterion), and the metaheuristics using the IRACE calibration described in Section 5.2.3, *LNS-GG*, *LNS-RG*, *LNS-SA-GG* and *LNS-SA-RG*. For the *RCH* and the four metaheuristics, the results consider the best outcome of 10 different runs for each instance, using different seeds.

The results for S1 and S2 are reported in Tables 5.10, 5.11 and 5.12 respectively for instances with cluster, random and mix topologies, where the best results are highlighted in boldface. As mentioned in Section 5.1, S1 and S2 contain 90 instances each, resulting in a total of 180 instances. In Tables 5.10, 5.11 and 5.12, results are presented by topology, and each line represents 6 instances. The rows with *Avg. [Total]* indicate the average or whenever it applies, the total values in brackets. The tables with full results for these instances are showed in the Appendix B.

It worth mentioning that a solution is considered better when it has the smallest value of $\mathcal{F}_1$ (working days), or in case of a tie, when it has the smallest value of $\mathcal{F}_2$ (total travel time). Thus, a deviation and a relative deviation are computed between the obtained solution of each method ($\mathcal{F}(m)$) and the best known one ($\mathcal{F}(b)$) for the first and second objectives, as indicated in Equations (5.2) and

(5.3).

$$Dev = \mathcal{F}_1(m) - \mathcal{F}_1(b) \tag{5.2}$$

$$Dev(\%) = \frac{\mathcal{F}_2(m) - \mathcal{F}_2(b)}{\mathcal{F}_2(b)} \times 100 \tag{5.3}$$

The two first columns in Tables 5.10, 5.11 and 5.12 are the instances characteristics. Then, for each method, the following data are given. *Nb_best* depicts the number of best known (b.k.) solutions among the evaluated methods; α shows the *Dev* average for the 6 instances of each line; β provides the *Dev(%)* average of the 6 instances for each line, and *t(s)* gives the average running time in seconds. Therefore, a method is considered better according to a lexicographical order of *Nb_best*, α and β. For instance, if *Nb_best* has equal values, the α value is now considered, and so on.

Results in Table 5.10 indicate that *LNS-RG* performed globally better than the others methods finding about 48% (29 out of 60) of the b.k. solutions, with an average of 0.67 working days and 0.49% for the total travel time. The *LNS-SA-GG* found about 43% of the b.k. solutions, with an average of 0.13 working days and 0.77% for the total travel time. The *LNS-SA-RG* achieved 40% of the b.k. solutions, with an average of 0.10 working days and 0.82% for the total travel time. The methods *LNS-GG* and *RCH* found respectively 18.3% and 3.3% of the b.k. solutions. *GCH* did not find any b.k. solution. As for time consumption, *LNS-SA-GG* and *LNS-SA-RG* performed faster than *LNS-GG* and *LNS-RG* for the instances with $|D| = 400$ and $|D| = 500$, thus, the overall average running time was smaller for the hybrid metaheuristics. Considering only the set S1, the hybrid metaheuristics performed better with 77% of the b.k. solutions found by *LNS-SA-RG* and 60% of the b.k. solution found by *LNS-SA-GG*.

	Instances	GCH				RCH				LNS-GG				LNS-RG				LNS-SA-GG				LNS-SA-RG			
	$\|D\|$	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
S1	10	0	0.33	1.81	~ 0	2	0.00	0.15	20	1	0.33	0.68	~ 0	2	0.00	0.13	20	3	0.33	0.15	0.23	**6**	**0.00**	**0.00**	20
	20	0	2.33	7.81	~ 0	0	1.00	3.18	20	4	0.17	0.39	0.03	3	0.33	0.80	20	**6**	**0.00**	**0.00**	0.77	**6**	**0.00**	**0.00**	20
	30	0	1.50	2.99	~ 0	0	0.67	1.62	20	2	0.33	0.24	0.15	0	0.67	1.62	20	1	0.33	0.26	1.50	**4**	**0.17**	**0.17**	21
	40	0	4.83	7.80	~ 0	0	3.67	5.87	20	3	0.50	0.08	0.30	3	0.33	0.37	20	4	0.17	0.07	2.57	**5**	**0.00**	**0.00**	22.20
	50	0	7.33	7.27	~ 0	0	4.33	4.10	20	1	0.67	0.30	0.63	0	4.00	3.58	20	**4**	**0.00**	**0.03**	4.20	2	0.17	0.18	23.63
S1: Avg. [Total]		[0]	3.27	5.53	~ 0	[2]	1.93	2.98	20	[11]	0.40	0.34	0.22	[8]	1.07	1.30	20	[18]	0.17	0.10	1.85	**[23]**	**0.07**	**0.07**	21.37
S2	100	0	0.67	7.77	~ 0	0	1.00	16.59	20	0	0.33	1.49	8.35	**4**	**0.17**	**0.03**	27.12	2	0.00	0.43	26.73	0	0.17	0.57	46.18
	200	0	1.17	9.56	~ 0	0	2.67	23.85	20	0	0.50	2.83	69.38	**4**	**0.17**	**-0.19**	88.47	2	0.00	1.66	125.13	0	0.00	2.11	143.33
	300	0	2.17	12.01	0.17	0	4.17	22.33	20	0	1.17	4.31	331.30	**6**	**0.00**	**0.00**	258.07	0	0.17	1.92	323.13	0	0.00	1.81	352.73
	400	0	3.17	7.91	0.17	0	8.00	22.58	20	0	1.67	2.39	920.88	**4**	**0.50**	**-0.64**	799.30	1	0.33	1.90	656.97	1	0.33	1.91	668.77
	500	0	3.83	7.25	~ 0	0	9.00	22.20	20	0	1.83	3.46	2136.78	3	0.50	-0.84	1727.65	**3**	**0.00**	**1.30**	1133.53	0	0.17	1.41	1192.70
S2: Avg. [Total]		[0]	2.20	8.90	0.07	[0]	4.97	21.51	20	[0]	1.10	2.90	693.34	**[21]**	**0.27**	**-0.33**	580.12	[8]	0.10	1.44	453.10	[1]	0.13	1.56	480.74
C: Avg. [Total]		[0]	2.73	7.22	0.03	[2]	3.45	12.25	20	[11]	0.75	1.62	346.78	**[29]**	**0.67**	**0.49**	300.06	[26]	0.13	0.77	227.48	[24]	0.10	0.82	251.06

Table 5.10: Results for the approaches in the instances with cluster topology.

Considering the set S2, *LNS-RG* achieved 70% of the b.k. solutions, with an average deviation of 0.27 working days and −0.33% for the total travel time of the dump trucks. *LNS-SA-GG* obtained about 27% of the b.k. solutions, but excelled for $|D| = 500$ with the smaller value of α. Since α and β are computed independently, a negative percentage deviation for β means that a better routing was found, but with a worst α. These results can be seen in Table B.2.

Results for random topology in Table 5.11 also indicate *LNS-RG* as the best method finding about 43% of the b.k. solutions, with an average of 0.97 working days and 1.35% for the total travel time. The *LNS-SA-GG* obtained 40% of the b.k. solutions, with an average of 0.08 working days and 0.71% for the total travel time. The *LNS-SA-RG* achieved 35% of the b.k. solutions, with an average of 0.10 working days and 0.67% for the total travel time. The methods *LNS-GG* and *RCH* found respectively 5% and 10% of the b. k. solutions. *GCH* did not find any b.k. solution. Considering the time consumption, the hybrid metaheuristics also performed better. Nevertheless, the running times for *LNS-GG* and *LNS-RG* were better in these instances compared to the cluster ones. In the set S1, *LNS-SA-GG* found about 63% of the b.k. solutions and the *LNS-SA-RG* obtained about 53% of the b.k. solutions. Concerning the set S2, *LNS-RG* performed better, finding 67% of the b.k. solutions, while *LNS-SA-GG* and *LNS-SA-RG* found about 17% of the b.k. solutions each.

In Table 5.12, results indicate that *LNS-SA-RG* performed globally better compared to the other methods. It obtained 40% of the b.k. solutions with an average deviation of 0.08 for working days and 0.72% for the total travel time. *LNS-RG* achieved 40% of the b.k. solutions with an average deviation of 0.77 for working days and 2.04% for the total travel time. *LNS-SA-GG* found about 37% of the b.k. solutions, with an average of 0.08 working days and 0.86% for the total travel time. *LNS-GG* obtained about 13% of the b. k. solutions. *RCH* found 10% of the b. k. solutions and *GCH* did not find any b.k. solution. Regarding the time consumption, the methods kept a performance similar to the results in Table 5.10. However, the results in the methods *LNS-GG* and *LNS-RG* indicate that they are sensitive to the clustered nodes present in topologies C and M. Considering the

Instances		GCH				RCH				LNS-GG				LNS-RG				LNS-SA-GG				LNS-SA-RG			
	$\|D\|$	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
S1	10	0	1.00	3.59	~ 0	**6**	**0.00**	**0.00**	20	1	0.33	1.33	0.03	**6**	**0.00**	**0.00**	20	1	0.33	0.77	0.25	**6**	**0.00**	**0.00**	20
	20	0	3.00	9.88	~ 0	0	1.50	5.22	20	1	0.50	1.39	0.07	0	0.33	1.31	20	**6**	**0.00**	**0.00**	0.80	2	0.17	0.39	20.07
	30	0	4.33	11.61	~ 0	0	3.00	8.34	20	0	0.83	1.20	0.15	0	1.33	2.85	20	**3**	**0.00**	**0.16**	1.47	3	0.17	0.04	21
	40	0	6.00	7.75	~ 0	0	4.00	5.25	20	1	0.50	0.50	0.28	0	2.67	4.35	20	**4**	**0.00**	**0.15**	2.80	2	0.00	0.11	22.25
	50	0	8.33	11.84	~ 0	0	5.17	7.26	20	0	1.00	0.75	0.55	0	3.67	5.95	20	**5**	**0.00**	**0.02**	3.80	3	0.33	0.09	23.28
S1: Avg. [Total]		[0]	4.53	8.93	~ 0	[6]	2.73	5.21	20	[3]	0.63	1.04	0.22	[6]	1.60	2.89	20	**[19]**	**0.07**	**0.22**	1.82	[16]	0.13	0.12	21.32
S2	100	0	0.67	9.74	~ 0	0	1.00	18.89	20	0	0.17	2.45	8.12	**4**	**0.00**	**0.47**	26.35	1	0.00	0.78	27.75	1	0.00	0.70	46.73
	200	0	1.67	9.65	~ 0	0	2.50	19.62	20	0	1.67	6.35	58.78	**4**	**0.33**	**-0.24**	75.20	0	0.17	0.95	122.50	2	0.00	0.92	146.63
	300	0	2.67	11.26	0.17	0	4.50	21.66	20	0	2.17	5.51	266.47	**4**	**0.33**	**-0.41**	185.58	0	0.33	1.16	321.95	2	0.00	1.18	345.00
	400	0	2.50	6.69	~ 0	0	6.50	19.53	20	0	1.67	4.02	701.35	**4**	**0.50**	**-0.22**	648.50	2	0.00	1.35	628.90	0	0.17	1.53	678.38
	500	0	3.33	8.03	~ 0	0	9.00	22.53	20	0	2.17	4.52	1696.15	**4**	**0.50**	**-0.58**	1563.38	2	0.00	1.77	1169.32	0	0.17	1.73	1215.15
S2: Avg. [Total]		[0]	2.17	9.08	0.03	[0]	4.70	20.45	20	[0]	1.57	4.57	546.17	**[20]**	**0.33**	**-0.20**	499.80	[5]	0.10	1.20	454.08	[5]	0.07	1.21	486.38
R: Avg. [Total]		[0]	3.35	9.01	0.02	[6]	3.72	12.83	20	[3]	1.10	2.80	273.20	**[26]**	**0.97**	**1.35**	259.90	[24]	0.08	0.71	227.95	[21]	0.10	0.67	253.85

Table 5.11: Results for the approaches in the instances with random topology.

Instances		GCH				RCH				LNS-GG				LNS-RG				LNS-SA-GG				LNS-SA-RG			
	$\|D\|$	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
S1	10	0	0.83	6.11	~ 0	**6**	**0.00**	**0.00**	20	3	0.17	1.15	~ 0	**6**	**0.00**	**0.00**	20	3	0.00	1.12	0.25	**6**	**0.00**	**0.00**	20
	20	0	1.67	4.84	~ 0	0	1.17	3.56	20	1	0.67	1.03	0.08	1	0.67	0.90	20	**6**	**0.00**	**0.00**	0.78	4	0.17	0.31	20
	30	0	3.50	10.86	~ 0	0	2.33	8.62	20	1	2.00	3.83	0.17	0	1.33	5.97	20	**3**	**0.00**	**0.31**	1.45	2	0.17	0.28	21
	40	0	5.17	14.66	~ 0	0	3.67	12.15	20	0	1.67	6.63	0.35	0	1.00	2.56	20	**4**	**0.17**	**0.09**	2.53	2	0.00	0.15	22.13
	50	0	5.67	11.78	~ 0	0	3.00	8.24	20	**3**	**0.17**	**0.55**	0.68	0	2.17	6.87	20	1	0.17	0.17	3.97	2	0.17	0.10	23.37
S1: Avg. [Total]		[0]	3.37	9.65	~ 0	[6]	2.03	6.51	20	[8]	0.93	2.64	0.26	[7]	1.03	3.26	20	**[17]**	**0.07**	**0.34**	1.80	[16]	0.10	0.17	21.30
S2	100	0	0.67	11.78	~ 0	0	1.17	21.84	20	0	0.33	2.22	8.72	**4**	**0.17**	**-0.07**	26.92	0	0.00	0.94	27.15	2	0.00	0.77	46.85
	200	0	1.83	10.07	~ 0	0	2.33	20.03	20	0	1.17	2.74	101.55	**5**	**0.17**	**-0.07**	73.58	1	0.00	1.04	126.98	0	0.17	1.04	147.32
	300	0	4.00	9.21	~ 0	0	4.50	20.08	20	0	5.50	8.53	213.85	2	0.50	5.82	197.00	2	0.17	1.01	329.83	**2**	**0.17**	**0.79**	342.37
	400	0	4.50	11.06	0.17	0	8.50	28.00	20	0	2.67	2.85	875.78	**3**	**0.83**	**-0.71**	833.32	1	0.33	1.84	649.22	2	0.00	1.76	679.78
	500	0	4.67	11.13	~ 0	0	9.67	27.12	20	0	2.67	5.23	2241.28	**3**	**0.83**	**-0.87**	1807.63	1	0.00	2.09	1173.40	2	0.00	1.98	1212.33
S2: Avg. [Total]		[0]	3.13	10.65	0.03	[0]	5.23	23.41	20	[0]	2.47	4.31	688.24	**[17]**	**0.50**	**0.82**	587.69	[5]	0.10	1.38	461.32	[8]	0.07	1.27	485.73
M: Avg. [Total]		[0]	3.25	10.15	0.02	[6]	3.63	14.96	20	[8]	1.70	3.48	344.25	[24]	0.77	2.04	303.85	[22]	0.08	0.86	231.56	**[24]**	**0.08**	**0.72**	253.52

Table 5.12: Results for the approaches in the instances with mix topology.

set S1, the hybrid metaheuristics performed better than excepted for the instances with $|D| = 50$, where *LNS-GG* obtained the best results. In the set S2, *LNS-RG* also performed better, finding about 57% of the b.k. solutions. *LNS-SA-RG* found about 27% and excelled for the instances with $|D| = 300$ with the smaller values for α and β. *LNS-SA-GG* found about 17% of the b.k. solutions.

We noticed that the improvements obtained using the *LNS-GG*, *LNS-RG*, *LNS-SA-GG* and *LNS-SA-RG* methods when compared with the constructive heuristics became very relevant, in particular for S2. For instance, on average, the solutions obtained by the metaheuristics are for the cluster topology up to 4.97 better that the others considering the total number of working days, and about 21.51% of total travel time. A quite similar results are found for the other topologies: improvements of up to 4.70 working days and 20.45% of total travel time for the random topology, and improvements of up to 5.23 working days and 23.41% of total travel time for mix topology. But not surprisingly, the metaheuristics require more execution time, according to instance size. The more time consuming instances were the ones with $|D| = 500$, with up to 2241.28 seconds for the instances with mix topology in *LNS-GG*.

LNS-RG found the largest number of b.k. solutions for the instances in C, R and M. Nevertheless, the overall values for α and β in the methods *LNS-SA-GG* and *LNS-SA-RG* indicates that even when they did not find the best solution, they were still able to find solutions very close to the b.k. solutions, often with the same number of working days and a routing with less than or equals to 0.86% of difference in average.

The ranking analysis described in Section 5.2.2 were computed for the methods. Such the results are presented in Table 5.13 for each topology. Values in bold represent the best results. In order to perform these ranking analysis, the best solutions of each method in each instance were considered. According to these results, as expected, the absolute rank (Abs-Rank) follows the same values found in Tables 5.10, 5.11, and 5.12. On the other hand, the results for average rank (Avg-Rank) indicate that the hybrid metaheuristics often find better ranked solutions than other methods in average, where *LNS-SA-GG* excelled for C and *LNS-SA-RG*

excelled for R and M.

		GCH	RCH	LNS-GG	LNS-RG	LNS-SA-GG	LNS-SA-RG
C	Avg-Rank	5.52	5.17	3.42	2.54	**2.17**	2.18
	Abs-Rank	0	2	11	**29**	26	24
R	Avg-Rank	5.57	4.96	3.73	2.59	2.10	**2.06**
	Abs-Rank	0	6	3	**26**	24	21
M	Avg-Rank	5.52	4.88	3.66	2.65	2.17	**2.12**
	Abs-Rank	0	6	8	**24**	22	**24**

Table 5.13: Ranking analysis for all methods in each topology.

To summarize, from the point of view of the sets S1 and S2, the methods *LNS-SA-GG* and *LNS-SA-RG* worked better for the S1 instances, where *LNS-SA-RG* was better for cluster topology and *LNS-SA-GG* was better for random and mix topologies. For S2 instances, the *LNS-RG* results overcame the other heuristics in the three topologies. From the point of view of the topologies, *LNS-RG* worked better for cluster and random topologies, while the *LNS-SA-RG* worked better for mix topology.

5.2.5 Robustness of the proposed metaheuristics

Time-to-target (TTT plots) proposed by [Aiex et al., 2002, 2007] are used in this section to analyse the robustness of the metaheuristics. TTT plots map the running time for a heuristic to find a solution at least as good as a targeted solution, varying the random generator seed. This way, the method robustness is measured according to the sensibility of a random generation.

Given an instance and a target value, the methods are run 100 times until met the target value. Then, the runs are sorted in an ascending order of running time. In the following, a TTT graphics is built by taking each point (τ_i, ρ_i), $i = \{1, ..., 100\}$, where τ_i is the running time for the ith solution, and $\rho_i = (i - 0.5)/100, i = \{1, ..., 100\}$.

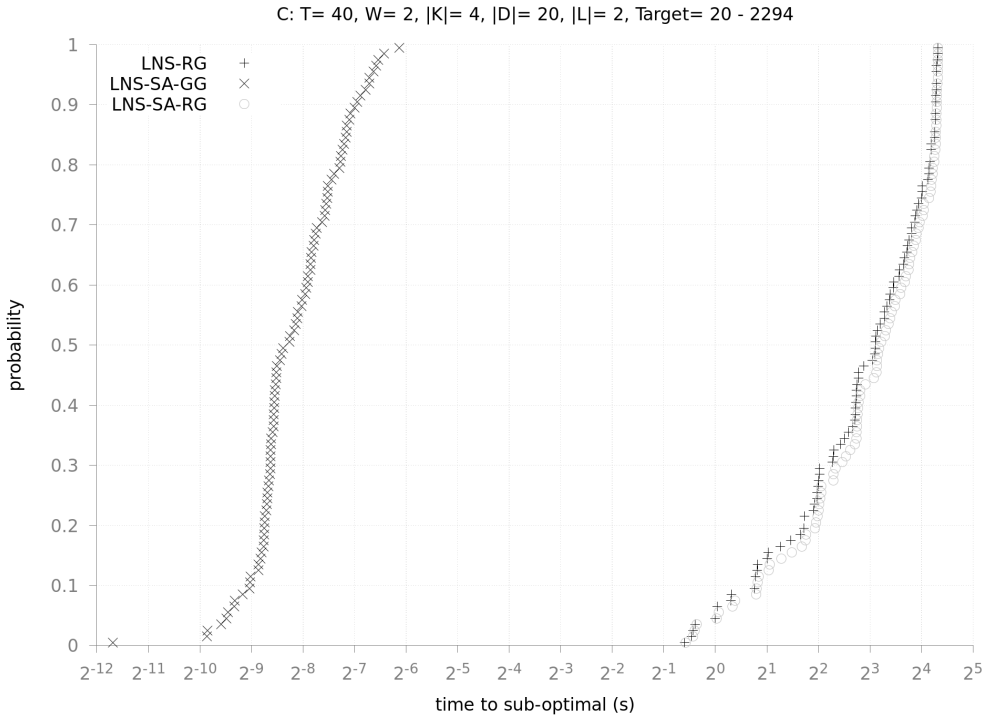
The three metaheuristics *LNS-RG*, *LNS-SA-GG* and *LNS-SA-RG* that obtained the best results were used in the TTT plot experiments. Figures 5.4, 5.5 and 5.6 depicts TTT plots for six instances representative of the instance sets: one instance from S1 and another from S2, for each of each topology. The targeted

solution should not be too easy, nor too difficult to reach, because these cases prevent a fair comparison between the methods. Hence, for each instance, a deviation in the best known solution up to 10% for $\mathcal{F}_1$ and $\mathcal{F}_2$ was fixed to obtain the target values. For example, if for some instance the best known solution has $\mathcal{F}_1 = 19$ and $\mathcal{F}_2 = 2144$ and a targeted solution is set to $\mathcal{F}_1 = 20$ and $\mathcal{F}_2 = 2294$, the CPU running times to reach this targeted solution are computed for 100 different runs in each method.

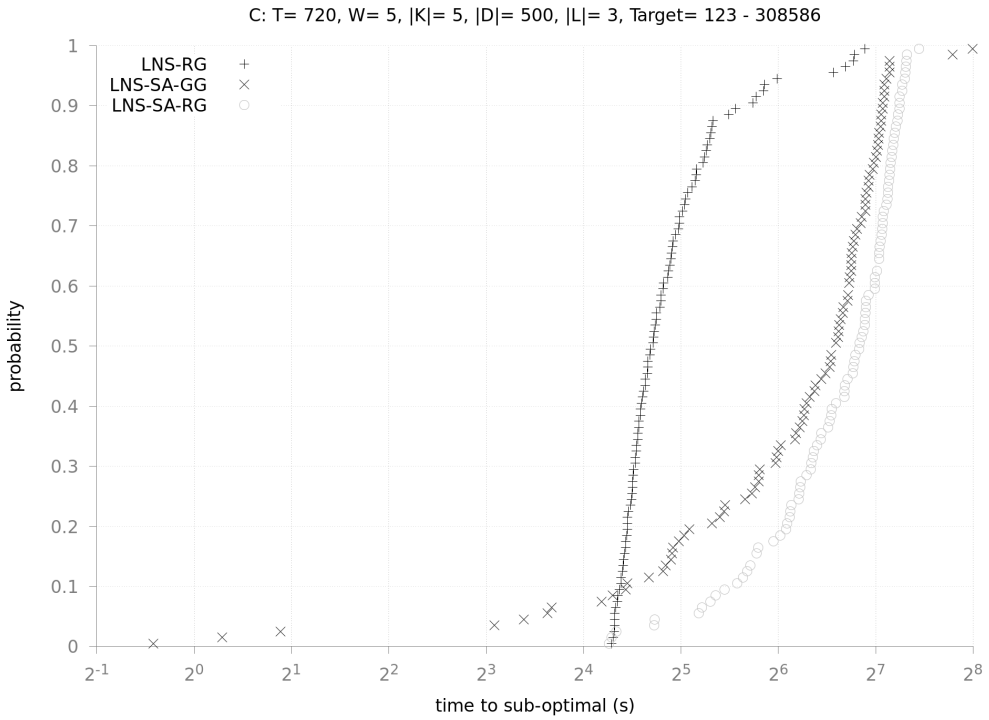
In the plots, each method is represented according to legend and the results of the 100 runs are plotted in the image. In addition, the title of each plot indicates the instance used and the targeted solution in terms of $\mathcal{F}_1$ and $\mathcal{F}_2$. The axes x and y indicate respectively τ_i and ρ_i to reach the target value. The straighter the plot, the higher probability to reach the target value. Moreover, the closer to the y axis the plot, the faster the method.

For cluster topology, in S1, the results in Figure 5.4-(a) indicate that *LNS-SA-GG* performs globally better than *LNS-RG* and *LNS-SA-RG*, since plots are closer to the y axis. Since a max running time for each run is set to prevent time consuming experiments, a disruption can occurs when a method does not reach the target solution in this time. According to the data of the results, 7 disruptions out of 100 runs occurred for *LNS-RG* and no disruptions for the others. A disruption indicates the method was sensitive to some seeds and did not found a solution that reached the target value. For S2, the results in Figure 5.4-(b) indicate that *LNS-RG* dominates the other metaheuristics. The data showed that no disruptions appeared for *LNS-RG*, and *LNS-SA-GG* and *LNS-SA-RG* had 2 disruptions each. Regarding the time consumption, plots for *LNS-RG* also indicate that the method is more stable than the others with a shorter interval considering the fastest and the slowest runs.

For random topology, in S1, the plots in Figure 5.5-(a) show that *LNS-SA-GG* performed better than the other methods. According to the data, no disruptions occurred for both *LNS-SA-based* methods and *LNS-RG* met 8 disruptions. For S2, in Figure 5.5-(b), the *LNS-RG* performed globally better than the hybrid metaheuristcs, once it reached the target solution faster. In addition, no disruptions



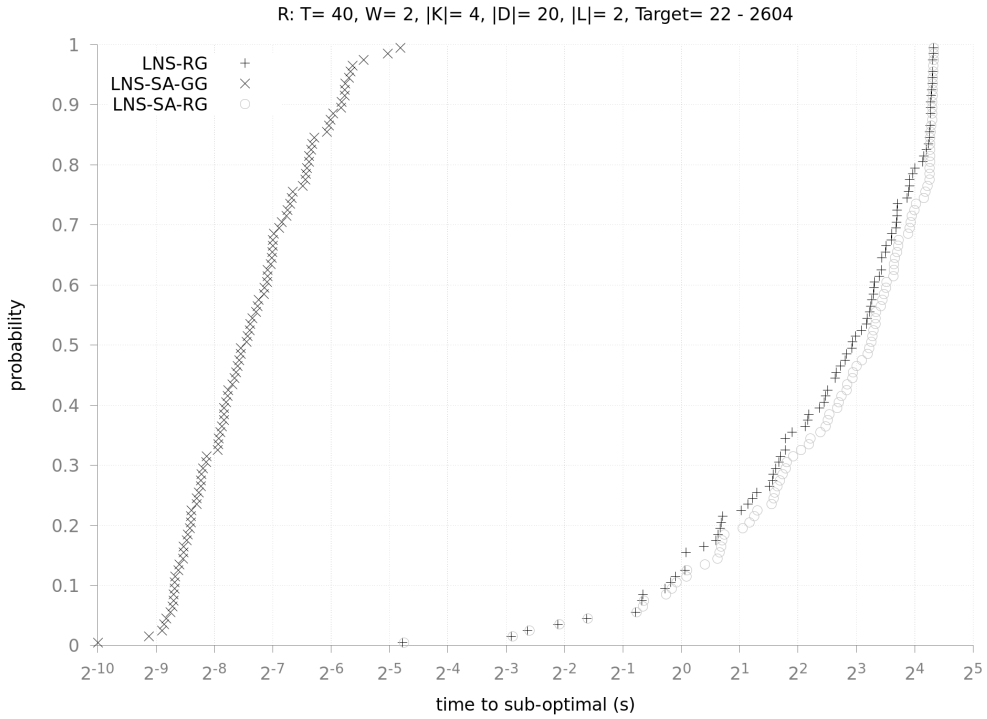
(a) TTT plot for S1 instance.



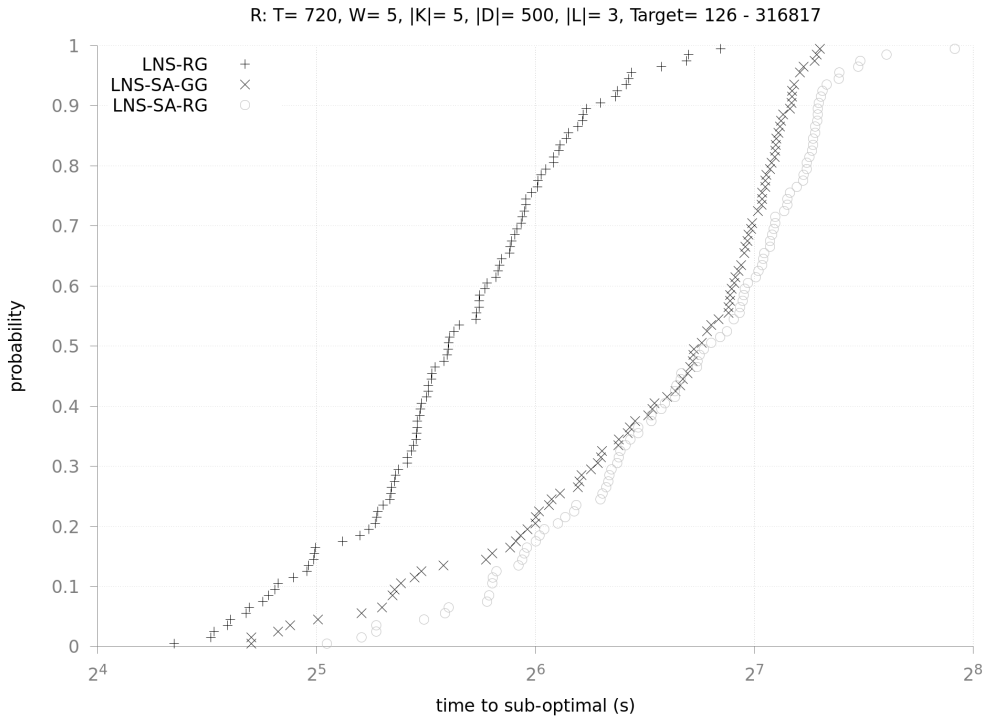
(b) TTT plot for S2 instance.

Figure 5.4: Analysis of TTT plots of instances with cluster topology.

appeared in the *LNS-RG*, while *LNS-SA-GG* and *LNS-SA-RG* had 5 and 2 disruptions respectively.



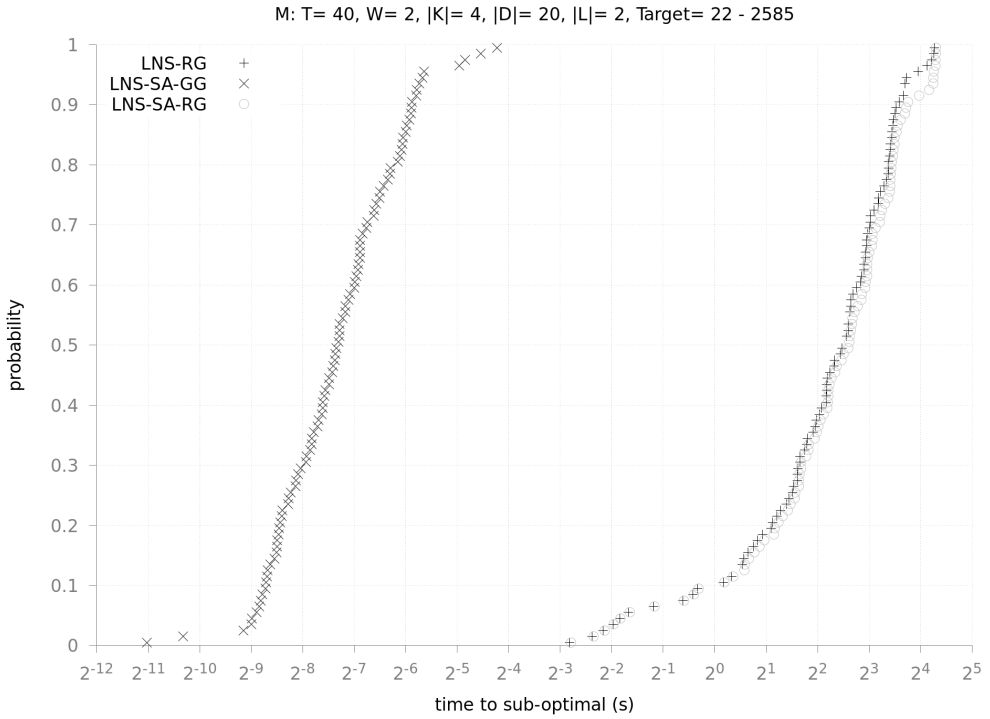
(a) TTT plot for S1 instance.



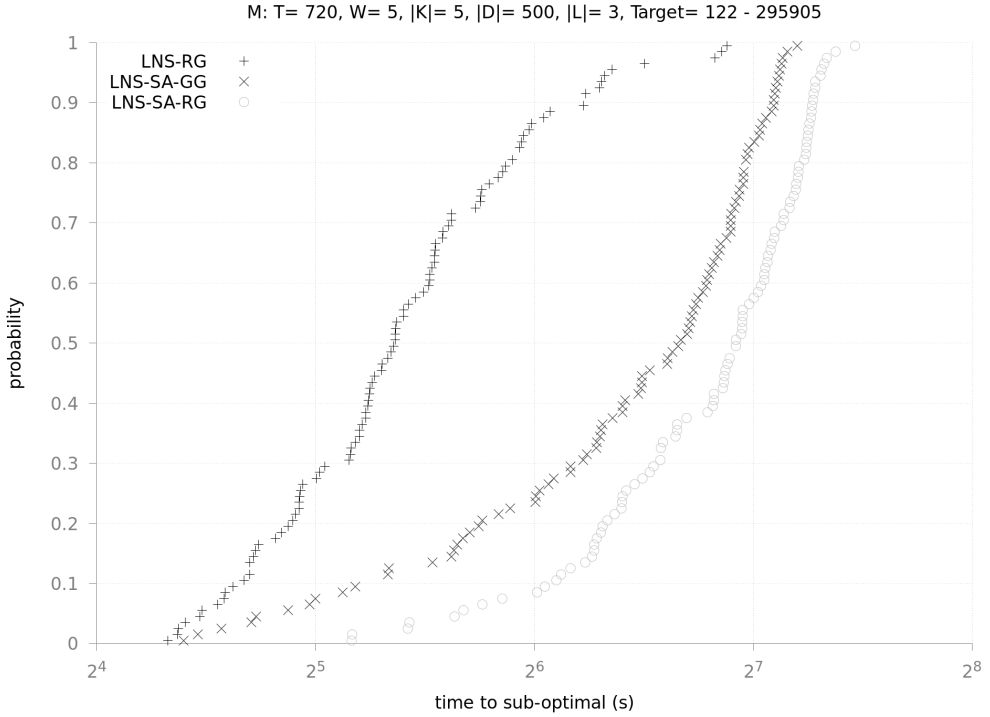
(b) TTT plot for S2 instance.

Figure 5.5: Analysis of TTT plots of instances with random topology.

In the results for mix topology, in Figure 5.6-(a) the plots for S1 also illustrate that *LNS-SA-GG* performed better than the other methods. The data of the



(a) TTT plot for S1 instance.



(b) TTT plot for S2 instance.

Figure 5.6: Analysis of TTT plots of instances with mix topology.

results showed that no disruptions were observed in *LNS-SA-GG* nor *LNS-SA-RG*, and 4 disruptions were met by *LNS-RG*. For S2, in Figure 5.6-(b), *LNS-RG* also

dominates the hybrid metaheuristics in the experiments. For this case, the data showed that *LNS-RG*, *LNS-SA-GG* and *LNS-SA-RG* had respectively 3, 2 and 2 disruptions out of 100 runs each.

For all topologies of S1 instances, it is noticeable that LNS-RG and LNS-SA-RG reached target values very fast after the 20 seconds necessary to generate the initial solution. Last but not least, clearly results indicate the *LNS-SA-GG* being more robust (fast and less sensible to the seeds) than *LNS-RG* and *LNS-SA-RG* for S1 instances; and the results indicate *LNS-RG* being more robust than the other metaheuristics for S2 instances.

5.2.6 Results for the case study

In this section, the results and discussions are done on the realistic instances proposed in Section 5.1.2 for the case study of Port-au-Prince, Haiti earthquake in 2010.

The method *LNS-RG* that obtained the best results for the biggest instances (S2) was used in this section. In addition, the number of iterations and runs of the method was reduced in order to prevent time-consuming experiments. The values of $\mathcal{F}_1$ are presented in *years, months and days*, while the values of $\mathcal{F}_2$ are presented in *hours*. Column *t(s)* are the running time in seconds. The results are presented in Table 5.14.

Regarding the values of $\mathcal{F}_1$, as expected, the instance using the smaller number of WTs, dump trucks and the working day of $8h$ obtained the longest amount of time. That is, the solution required 4 years and 5 months and 17 days to clean the overall area. Moreover, the dump trucks consumed 108 551 hours, which represents 500 000 trips. In addition, as the number of vehicles and working days increase, the time for the overall cleaning decreases. For instance, the solution with $W = 21$, $|K| = 83$ and $T = 12h$ required 1 year, 3 months and 9 days and 108 967 hours of travel.

Concerning the values for $\mathcal{F}_2$, they increase as the number of trucks increases for each working day. This is due the fact that more dump trucks being used, then more daily departures and returns are done at the depot. However, by analyzing

T	W	$\|K\|$	LNS-RG				
			$\mathcal{F}_1$			$\mathcal{F}_2$	t(s)
			Years	*Months*	*Days*	*Total travel hours*	
		21	4	5	17	108 551	2558
8h	21	42	2	7	17	109 119	4154
		83	1	10	26	110 740	3774
		21	3	6	27	107 910	2000
10h	21	42	2	1	3	108 351	2294
		83	1	6	8	109 659	3879
		21	2	11	26	107 470	2005
12h	21	42	1	9	0	107 845	2202
		83	1	3	9	108 967	4602

Table 5.14: Results applying LNS-RG in the realistic instances of Port-au-Prince, Haiti.

the same number of dump trucks, as the value of the working day increases, the value of $\mathcal{F}_2$ decreases. This is due the fact that dump trucks remove more debris before returning to the depot.

The studies of Sakuraba et al. [2016] and Barbalho et al. [2020] showed that 3 months and 16 days were necessary to push aside debris and establish the traffic for relief teams and distribution. The results obtained here highlight the relevance of cleaning the overall area, as well as the operational level (routing). According to the scenarios, the method provides a solution that take a little more than 1 to 4 years, dependent on the available resources like WTs and dump trucks. This is an interesting result when compared with what happens in practice. The report of UNDP [2013] presented that, after two years, approximately 1 million cubic meters had been removed.

Chapter 6

Concluding remarks and perspectives

This thesis brings several contributions for the integrated Scheduling vehicle Routing Problem to Clean Debris (SRP-CD) in the aftermath of major disasters: a mathematical formulation with two levels of synchronization (between WT and dump trucks, and among dump trucks), including two temporal scales (working day and time horizon), and having two objectives functions (one to minimize the total number of working days and another to minimize the vehicle routing costs); several constructive heuristics, metaheuristics, and a data generation mechanism for which three topologies (random, cluster, and mix) are proposed. Moreover, extensive computational experiments have been done to measure performance and robustness of the proposed methods. The metaheuristic parameters were calibrated using IRACE that is a machine learning based method.

Concerning the mathematical model, it is based on dynamic multi-flow and it was validated on small instances. To our knowledge, this is the first formulation integrating scheduling and routing for SRP-CD, including all parameters focused on this thesis. In spite of its limits on solving medium instances, the temporal flow allowed us to further address the two levels of synchronization.

Results indicate that *LNS-RG* overcame the other heuristics for cluster and random topologies and the *LNS-SA-RG* and *LNS-RG* performed very well for the mix topology. For cluster instances, *LNS-RG* found about 48% of the best known

solutions, while *LNS-SA-GG* reached about 43% of the best known solutions and *LNS-SA-RG* reached about 40% of the best known solutions. For random instances, *LNS-RG* achieved about 43% of the best known solutions, *LNS-SA-GG* reached about 40% of the best known solutions, and *LNS-GG* obtained 21% of the best known solutions. And, for the mix instances, *LNS-SA-RG* and *LNS-RG* found 40% of the best known solutions, while *LNS-SA-GG* achieved about 37% of the best known solutions. *LNS-RG* has been shown to perform better and be more robust for larger instances regarding the proposed methods.

The results for the case study of Port-au-Prince, Haiti earthquake in 2010 considered the method *LNS-RG* that obtained the best results for the larger instances in the benchmark. According to the scenarios, to clean an amount of 10 million cubic meters of debris, the method provides a solution that take a little more than 1 to 4 years, dependent on the available resources like WTs and dump trucks. This is an interesting result when compared with what happens in practice, where, after two years as reported in UNDP [2013], approximately 1 million cubic meters had been removed.

Some managerial insights were raised from the results: the planned operations allow for more efficient resource management and faster recovery; using less vehicles leads to a smaller cost, but increases the number of working days for the overall cleaning; due to the fact that the volume of debris is very important, a multi-year governmental budgetary and operational planning is strongly required; and for the cases where the volume of debris is very important, it can be interesting to think about sorting and valorization strategies (direct use or recycling).

This thesis opens several avenues of research. Open questions were raised concerning the integration of other optimization problems according to the volume of debris. This is the case of the scheduling coupled to the SDVRP problem, or either the transition between SRP-CD (RCPSP and FTVRP) and the integrated RCPSP and SDVRP. Another hypothesis raised, concerns a WT performing loads on different nodes on the same working day, this could bring gains in the number of days. Nevertheless, this implies integrating WTs' displacement constraints with the two time scales to ensure that the working day is respected. In addition, other

issues can also be considered such as: prioritizing debris nodes, for instance, a site where debris brings a toxicity risk, or including precedence relationships between debris nodes, for example, when a debris node is not accessible until others nodes are cleaned.

In terms of methods, hybrid ones and new local searchers can be proposed, for instance, the Adaptive Large Neighborhood Search proposing a set of removal operators and a set of insertion operators. For instance, the insertion operators could be developed as the "regret insertion" [Potvin and Rousseau, 1993; Ropke and Pisinger, 2006; Tillman and Cain, 1972]; "regret insertion with noise" of Ropke and Pisinger [2006],"cheapest insertion" and "cheapest insertion with noise". A number of removal operators are also available in the literature such as "related removal" of Shaw [1998], "worst removal" of Ropke and Pisinger [2006], "route removal" of Demir et al. [2012]; Mancini [2016]. Additionally, different acceptance criteria also could be developed as the ones described in the study of Santini et al. [2018]. In particular, there is room to improve the routing part of the SRP-CD.

Appendix A

Résumé étendu en français

Modèles et méthodes pour le problème intégré de planification et tournées appliqué aux catastrophes majeures

Chapitre 1 : Introduction

Les travaux publics et la gestion de crise suite à des catastrophes majeures (tremblements de terre, ouragans, inondations, etc.) sont deux situations qui nécessitent un effort important pour nettoyer une zone urbaine. La première peut être planifiée à l'avance, tandis que la seconde peut devenir très complexe si une large zone urbaine est affectée de manière inattendue. Cependant, le nettoyage des zones urbaines après des catastrophes majeures est crucial pour que les habitants se remettent des effets des catastrophes. Lorsque les catastrophes frappent des régions habitées, elles impactent fortement la population, l'environnement et les infrastructures urbaines. En général, une catastrophe peut être divisée en quatre phases : anticipation, préparation, intervention et récupération [Altay and Green III, 2006].

Cette thèse aborde le problème intégré de planification et de l'organisation de tournées de nettoyage des débris (en anglais *Scheduling Routing Problem to clean debris (SRP-CD)*) dans une zone urbaine après des catastrophes majeures. Ce problème considère deux niveaux d'optimisation stratégique et opérationnel. SRP-CD apparaît dans la phase de récupération, qui peut prendre des mois ou des

années après les événements. Le problème considère des ressources limitées telles que des équipes de travail (*work-troops (WT)*) et une flotte de camions à benne basculantes, chargée de transporter les débris vers les décharges. Le réseau de transport est modélisé sous la forme d'un graphe et dans tout le résumé en français, le terme *sommet de débris* correspond à un nœud du graphe où des débris doivent être nettoyés, et le terme *sommet de décharge* correspond à un nœud du graphe où les camions vont vider leurs contenus. De plus, un horizon temporel global et une journée de travail limitée sont pris en compte. Les opérations sont effectuées aux sommets de débris affectant les *WTs*, les camions, la synchronisation du chargement des camions, et le déchargement des camions dans les décharges. Du point de vue stratégique, la décision majeure concerne l'attribution des *WTs* aux sommets de débris dans l'horizon temporel. Dans le contexte opérationnel, les camions effectueront plusieurs trajets dans chaque journée pour transporter les débris de leur origine vers les décharges. Il existe donc deux niveaux de synchronisation, entre les *WTs* et les camions, et entre les camions au moment du chargement. L'objectif est double : au niveau stratégique, minimiser le temps total, en jours, pour réaliser les opérations de nettoyage de l'ensemble de la zone urbaine ; et au niveau opérationnel, minimiser les coûts totaux des tournées de véhicules.

L'intégration de différents niveaux de décision dans les problèmes d'optimisation est une tendance croissante dans la littérature scientifique. Un point d'entrée intéressant sur l'intégration des problèmes de tournées de véhicules (en anglais *Vehicle Routing Problem - VRP*) est l'étude de Coelho et al. [2016]. Les problèmes intégrés sont un domaine difficile car la complexité globale de l'intégration peut augmenter en fonction de la complexité de chaque problème. Cependant, comme indiqué dans Prodhon and Prins [2014], les gains à résoudre un problème de manière intégré sont d'environ 15 % par rapport à la résolution des problèmes indépendants. De plus, les problèmes intégrés d'optimisation permettent de travailler avec des scénarios plus réalistes considerant des contraintes pertinentes impliquées dans l'application réelle. C'est le cas du SRP-CD avec les deux niveaux de synchronisation : un entre les *WTs* assignés et les camions et l'autre entre les camions eux-mêmes.

Le SRP-CD est lié à trois problèmes d'optimisation classiques : au niveau stratégique, il est étroitement lié au problème d'ordonnancement de projets avec contraintes de ressources (en anglais *Resource-Constrained Project Scheduling Problem - (RCPSP)*) [Brucker et al., 1999; Hartmann and Briskorn, 2010]. Selon le volume de débris, deux problèmes de tournées de véhicules peuvent aborder le niveau opérationnel du SRP-CD : le problème de tournées de véhicules avec pleine charge (en anglais *Full Truckload Vehicle Routing Problem - (FTVRP)*) [Desrosiers et al., 1988] et le problème de tournées de véhicules avec livraisons fractionnées (en anglais *Split Delivery Vehicle Routing Problem - (SDVRP)*) [Archetti and Speranza, 2008; Golden et al., 2008].

Le volume de débris a un rôle clé au niveau opérationnel et définit quel problème d'optimisation doit être utilisé pour modéliser la situation. Trois cas peuvent être distingués : (1) le volume de débris est très important et dans ce cas, le niveau opérationnel repose sur le FTVRP ; (2) le nombre de trajets de camions avec pleine charge n'est pas élevé et l'utilisation du FTVRP et du SDVRP peut apporter des gains en termes de coûts totaux ; et (3) le volume de débris est inférieur ou égal à la capacité de chaque véhicule et le SDVRP peut être utilisé. Tous les cas mentionnés ci-dessus présentent des problèmes $\mathcal{NP}$-difficiles [Archetti et al., 2011; Blazewicz et al., 1983]. Ainsi, le SRP-CD est $\mathcal{NP}$-difficile. Cette thèse aborde le premier cas (1), qui nous permet de produire les premiers modèles et algorithmes intégrés dans le contexte des catastrophes.

Les principales contributions de cette thèse sont l'intégration des niveaux stratégique (ordonnancement) et opérationnel (tournée de véhicules) du problème d'optimisation SRP-CD, motivé par le nettoyage des débris après des catastrophes majeures. Nous proposons un nouveau modèle mathématique basé sur une formulation de flot dynamique, incluant des synchronisations entre les *WTs* et les camions ; synchronisations entre les camions ; deux discrétisations temporelles, journées de travail et horizon temporel, en jours. En termes de méthodes, des heuristiques constructives, des métaheuristiques basées sur *Large Neighborhood Search -(LNS)* et des métaheuristiques hybrides basées sur LNS et *Simulated Annealing - (SA)* sont proposées. De plus, un benchmark de données intéressantes pour le SRP-CD est décrit avec un mécanisme de génération d'instances. En outre, une étude de

cas sur le tremblement de terre qui a frappé Port-au-Prince, en Haïti, en 2010, est également fournie. A notre connaissance, ce sont les premières approches, centrées sur la problématique intégrée avec toutes les contraintes évoquées, dans le contexte des catastrophes.

Chapitre 2 : L'état de l'art

Le SRP-CD est lié aux problèmes classiques et aux variantes des problèmes de tournées de véhicules et de planification ; ainsi qu'à plusieurs applications dans la logistique de crise, l'exploitation minière à ciel ouvert (en anglais *Open pit Mining - OPM*), entre autres. Une vision générale des études associées et des similitudes et différences concernant le SRP-CD est fournie dans cette section.

Problèmes d'optimisation classiques associés

Le SRP-CD considère les aspects de différents problèmes d'optimisation classiques connus, tant au niveau stratégique qu'opérationnel.

Le RCPSP, comme décrit dans Brucker et al. [1999]; Hartmann and Briskorn [2010]; Pritsker et al. [1969], peut être défini comme un projet contenant un ensemble de tâches à traiter, qui doivent être terminées sans préemption. Chaque tâche a des besoins en ressources pour être réalisée, une durée et certaines d'entre elles peuvent avoir des relations de priorité ou de précédence. L'objectif de minimiser le *makespan* est généralement utilisé. Comparé au SRP-CD, le RCPSP apparaît au niveau stratégique, où le projet pourrait être considéré comme l'ensemble de la zone affectée et les tâches comme les sommets de débris à nettoyer. Les WT pourraient être les machines effectuant les tâches (nettoyer les sommets de débris). Les périodes peuvent être considérées comme des journées de travail et il n'y a pas de date d'échéance pour chaque sommet de débris. Les sommets doivent être nettoyés pendant une journée de travail limitée, mais le nombre de jours pour terminer le nettoyage complet peut varier en fonction des itinéraires des camions. L'objectif principal du SRP-CD est de trouver le *makespan* minimal, similaire au RCPSP. Certaines différences existent entre RCPSP et SRP-CD comme l'intégration du

VRP dans le SRP-CD, et le fait que le SRP-CD soit modélisé sur un graphe en raison du réseau routier.

Le problème de tournées de véhicules avec collectes et livraisons (en anglais *Pickup and Delivery Problems -PDP*) a été référencé dans Parragh et al. [2008b,a] comme une variante spéciale du VRP, où un véhicule effectuait les collectes et livraisons pour les clients dans une période de temps, en commençant et en terminant l'itinéraire au dépôt. Les auteurs ont montré d'autres variantes comprenant plusieurs véhicules, plusieurs dépôts, des sommets de collecte et de livraison appariés ou non, des fenêtres horaires, etc. Par rapport au SRP-CD, au niveau opérationnel, les sommets de débris et les décharges pourraient être considérés comme des sommets de collecte et de livraison, mais le SRP-CD est plus complexe que les variantes du PDP puisqu'il comprend simultanément la synchronisation entre les *WTs* et les camions, la synchronisation entre les camions, les temps de chargement et de déchargement, une journée de travail limitée et plusieurs trajets pour un même sommet.

Certaines variantes du VRP sont liées au SRP-CD comme le FTVRP [Desrosiers et al., 1988], où le problème consiste à désigner des itinéraires pour les camions afin d'effectuer un certain nombre de trajets à pleine charge entre les sites. Le SDVRP [Archetti and Speranza, 2008; Golden et al., 2008], où le problème vise à définir des itinéraires pour les véhicules afin de rendre visite aux clients, et la demande de chaque client est inférieure ou égale à la capacité d'un camion. Chaque client peut être visité plus d'une fois et chaque demande peut être divisée en différents véhicules. L'objectif du FTVRP ainsi que du SDVRP est de minimiser la distance totale parcourue par les camions. La troisième variante concerne le problème de tournées de véhicules multi-dépôts avec routes inter-dépôts (en anglais *Multi-Depot Vehicle Routing Problem with Inter-Depot Routes - MDVRPI* [Crevier et al., 2007], où une flotte de véhicules dessert un ensemble de clients une fois chacun avec une demande et un temps de service. Les itinéraires commencent et se terminent dans le même dépôt en une journée de travail prédéfinie. Cependant, les véhicules sont autorisés à visiter des dépôts intermédiaires le long de leurs itinéraires afin de recharger leur capacité. L'objectif est de minimiser la durée totale des itinéraires.

Par rapport à ces variantes du VRP, dans le SRP-CD, les véhicules ne peuvent visiter que les sommets de débris avec des *WTs* assignés et l'objectif vise à effectuer toutes les opérations de nettoyage le plus rapidement possible tout en minimisant le temps de trajet total des camions.

Lacomme et al. [2019] ont abordé le RCPSP avec tournées (*RCPSPR*), où chaque activité d'un projet exigeait une quantité de ressources et les camions effectuaient des collectes et livraisons de ressources entre les activités. Le problème est représenté par un Programme Linéaire en Nombres Entiers (en anglais *Mixed Integer Linear Programming - MILP*) et par la programmation par contraintes, et une heuristique hybride a été proposée pour la résolution. Par rapport au SRP-CD, le RCPSPR est différent parce qu'il ne décrit pas les *WTs* pour les activités, la minimisation du temps de trajet des camions n'est pas un objectif, les opérations de collecte et de livraison sont effectuées sans temps de chargement ou de déchargement, il n'y a pas de points décharges et il n'y a pas de journées de travail.

Une application dans une chaîne d'approvisionnement en biomasse a été trouvée dans Soares et al. [2019]. Ils ont abordé un problème de tournées de véhicules avec collectes et livraisons avec pleine charge (en anglais *Full Truckload Pickup and Delivery Problem - FTPDP*) et la synchronisation de multiples véhicules. Trois types de véhicules ont été utilisés pour effectuer le transport des matériaux des sommets de collecte aux sommets de livraison dans un horizon de planification d'une seule journée de travail. L'objectif était de minimiser les coûts de transport pour les véhicules, les distances et le temps. L'étude Grimault et al. [2017] aborde le FTPDP avec la synchronisation de ressources pour une application dans le cadre des travaux publics et de la construction de routes. Le problème a été décrit dans un horizon temporel d'une journée de travail, où un ensemble de camions répondait aux demandes de transport de matériel. L'objectif était de minimiser le coût global des itinéraires. Dans le SRP-CD, l'horizon temporel est composé de plusieurs jours. Une autre différence réside dans l'objectif, puisque le SRP-CD a la minimisation des jours comme objectif principal et la minimisation des coûts des itinéraires comme objectif secondaire.

Applications en logistique de crise

Plusieurs applications en logistique de crise, nettoyage et en réparation de réseaux urbains abordent des aspects liés au SRP-CD.

L'étude Duque et al. [2016] a abordé le problème de planification de tournée d'équipes de réparation de réseau (en anglais *Network Repair Crew Scheduling and Routing Problem - NRCSRP*) dans les opérations de secours post-catastrophe, avec une seule équipe de réparation. L'idée était de partir d'un dépôt unique pour rétablir la connectivité du réseau et fournir une aide humanitaire. L'objectif était de minimiser le temps d'inaccessibilité des sommets bloqués. Le problème de planification de tournées d'équipages (en anglais *Crew Scheduling and Routing Problem - CSRP*) étudié dans Moreno et al. [2019] a pris en compte un seul équipage, avec pour but de minimiser le temps d'inaccessibilité des zones affectées. Un autre travail contenant un CSRP a été trouvé dans Shin et al. [2019], où une seule équipe, intégrée à un seul véhicule, distribuait des secours après la réparation des lieux endommagés.

Le problème stochastique de nettoyage de débris (en anglais *Stochastic Debris Clearance Problem - SDCP*) a été étudié dans Çelik et al. [2015], dont l'objectif principal était de débloquer les routes dans la phase de réponse, en déplaçant les débris afin d'atteindre d'autres sommets de demande et afin de maximiser les opérations de secours. Le problème du déblocage des routes afin de reconnecter le réseau affecté dans la phase de réponse, tout en minimisant le temps des opérations, a été résolu par Akbari and Salman [2017]. Li and Teo [2019] abordent le problème de planification de la réparation du réseau et logistique de secours, sur la phase de réponse, en deux étapes : une première où un problème de planification sur plusieurs périodes avec des ressources limitées a été résolu ; et une deuxième où les itinéraires pour la livraison des matériaux de secours ont été définis. Le problème de la réhabilitation routière d'urgence de Sakuraba et al. [2016] comprend le problème d'accessibilité du réseau routier où l'objectif était de trouver des voies pour que les équipes de secours atteignent la population le plus rapidement possible ; et le problème d'ordonnancement des équipes de travail, où l'objectif était de planifier les réparations pour améliorer l'accès aux zones de réfugiés, et

par conséquent, améliorer la distribution de l'aide humanitaire. Une extension de ce travail a également été trouvée dans Barbalho et al. [2020].

Par rapport au SRP-CD, dans les études précitées [Akbari and Salman, 2017; Barbalho et al., 2020; Çelik et al., 2015; Duque et al., 2016; Li and Teo, 2019; Moreno et al., 2019; Sakuraba et al., 2016; Shin et al., 2019], l'intégration de l'ordonnancement aux tournées pour le transport des débris, qui incluent une synchronisation entre les *WTs* et les camions, n'est pas ciblée car leurs objectifs résident dans la phase de réponse pour retrouver l'accessibilité ou améliorer l'accès des équipes de secours. De plus, les zones de décharge pour les débris ou temps de service pour les opérations ne sont pas prises en compte.

Le problème de collecte des débris a été traité dans Pramudita and Taniguchi [2014]; Pramudita et al. [2014] comme un problème intégré de *location-routing* avec pour but de minimiser les coûts. Le nettoyage des débris a été décrit par Lorca et al. [2017], comme un ensemble d'opérations telles que la collecte, le transport, la réduction, le recyclage et l'élimination des débris, avec pour but de minimiser les coûts de gestion. Le problème de nettoyage post-catastrophe en tant que modèle de logistique de la chaîne d'approvisionnement des déchets a été proposé par Boonmee et al. [2018], où l'objectif était de minimiser les coûts totaux. Ces travaux de nettoyage de débris diffèrent du SRP-CD car ils ne gèrent pas les décisions stratégiques (ordonnancement) intégrées aux décisions opérationnelles (tournées), et ils ont des objectifs différents.

Applications dans l'exploitation minière à ciel ouvert

Le problème d'exploitation minière (en anglais *Open pit Mining - OPM*) [Alarie and Gamache, 2002] consiste à définir des itinéraires pour le transport des matériaux dans les champs miniers. Malgré le contexte et le fait que l'OPM a été plus résolu dans la littérature par le biais de la simulation, plusieurs similitudes existent entre l'OPM et le SRP-CD. Par exemple, OPM a des points de chargement et de déchargement, et des décisions stratégiques et opérationnels. Les différences reposent sur les points suivants : l'OPM a un temps fixe pour planifier les opérations, et la fonction objectif cherche à minimiser les coûts des ressources. De plus, en

général, l'OPM est abordé dans une zone connue et de taille moyenne, ce qui diffère d'une catastrophe qui peut frapper partout et couvrir une zone inattendue et étendue.

Les auteurs dans Tan and Takakuwa [2016] ont traité l'OPM comme un processus de calcul d'un nombre optimal de pelles, de camions et le volume de production pour chaque pelle à partir d'un plan d'excavation. Ils ont appliqué un algorithme de simulation pour répartir chaque camion avec chaque pelle en fonction des données pour réaliser le plan. L'étude de Subtil et al. [2011] a traité l'OPM par un modèle de programmation linéaire (LP) appliqué pour maximiser la productivité en utilisant un algorithme par séparation et évaluation (*Branch-and-Bound*). Ensuite, une simulation a été réalisée afin d'affecter les camions aux points de chargement selon une stratégie exhaustive.

Alors que l'OPM est le plus souvent résolu par simulation [Alarie and Gamache, 2002], quelques études ont étudié le problème en utilisant une approche d'optimisation. Dans Fioroni et al. [2008], les auteurs se sont concentrés sur la réduction des coûts d'exploitation minière, en couplant un modèle mathématique d'optimisation à un modèle de simulation. Dans Ercelebi and Bascetin [2009], une stratégie de réseau de files d'attente a été proposée pour optimiser le nombre d'affectations de camions aux pelles, qui analysait tous les chemins possibles et réalisables entre les points de chargement et de déchargement. Une autre étude utilisant l'optimisation a été trouvée dans Souza et al. [2010], où un MILP optimisait les déviations de la production, les objectifs de qualité et le nombre de véhicules nécessaires aux opérations. Une heuristique hybride a également été proposée pour résoudre le problème.

Position de cette thèse

Le Tableau A.1 présente un résumé des travaux les plus liés au SRP-CD en ce qui concerne les caractéristiques des problèmes abordés trouvés dans la littérature et les approches proposées. Les travaux sont classés par année et par application. Comme on peut l'observer, aucune étude dans la littérature ne couvre toutes les caractéristiques du SRP-CD.

Applications	Études	Caractéristiques: Dépôt: Unique	Dépôt: Multiple	WT/Equipe: Unique	WT/Equipe: Plusieurs	Flotte: Homogène	Flotte: Hétérogène	Collection de matériaux: Caché	Collection de matériaux: Explicite	Déchargement	Temps de service	Sync.: WT-WT	Sync.: WT-Camion	Sync.: Camion-Camion	Journée de travail: Multi-voyages	Journée de travail: Une	Journée de travail: Plusieurs	Objectif: Min. de temps	Objectif: Min. de coût	Objectif: Max. de satisf.	Approches: Modèle mathématique	Approches: Méthode exacte	Approches: Heuristique	Approches: Simulation
	SRP-CD	●			●	●			●	●	●		●	●	●		●	●	●		●		●	
Classique	Lacomme et al. [2019]	●				●			●						●			●			●		●	
	Soares et al. [2019]		●		●	●			●	●	●		●	●	●	●			●		●		●	
	Grimault et al. [2017]		●		●		●		●	●	●		●			●			●		●		●	
Logistique de crise	Barbalho et al. [2020]				●			●										●					●	
	Moreno et al. [2019]	●		●				●										●			●	●	●	
	Shin et al. [2019]	●		●				●										●			●		●	
	Li and Teo [2019]		●		●			●									●			●	●		●	
	Boonmee et al. [2018]								●	●									●		●		●	
	Akbari and Salman [2017]		●		●			●				●						●			●		●	
	Lorca et al. [2017]				●				●	●		●					●		●		●			●
	Sakuraba et al. [2016]				●			●										●			●		●	
	Duque et al. [2016]	●		●				●										●			●	●	●	
	Çelik et al. [2015]				●			●									●			●	●		●	
	Pramudita et al. [2014a, 2014b]	●				●			●	●									●		●		●	
L'exploitation minière	Tan and Takakuwa [2016]				●	●			●	●			●	●			●			●		●		●
	Subtil et al. [2011]				●		●		●	●			●			●				●	●	●		●
	Souza et al. [2010]				●		●		●	●			●		●				●		●		●	
	Ercelebi and Bascetin [2009]				●		●		●	●			●						●		●	●		
	Fioroni et al. [2008]				●	●			●	●			●			●			●		●			●

Table A.1: Résumé des travaux connexes au SRP-CD.

Chapitre 3 : Définition formelle du SRP-CD

Dans le problème SRP-CD, le réseau de transport est modélisé sous la forme d'un graphe simple, non orienté et connecté $G = (V, E)$, où V est l'ensemble des n sommets et E l'ensemble des m arêtes. Soit $O \subset V$, $D \subset V$, $L \subset V$ respectivement, le dépôt unique ($O = \{0\}$), l'ensemble des sommets de débris et l'ensemble des décharges. Ainsi, $V = D \cup L \cup O$ et $D \cap L \cap O = \emptyset$. Chaque sommet $i \in D$ a un volume $v_i \in \mathbb{R}_+^*$ de débris à nettoyer. Dans l'ensemble E, chaque arête $[i, j], i, j \in V$ modélise un chemin le plus court en termes de temps reliant les sommets i et j dans le réseau réel, avec un temps de trajet pré-calculé $c_{ij} \in \mathbb{R}_+^*$. De plus, $A = \{1...H\}$ représente un horizon temporel (avec un pas d'une journée) considéré dans le modèle, et chaque journée de travail est discrétisé par une unité de temps $B = \{1...T\}$, avec une limite de temps T. Soit W et $K = \{1, ..., k\}$ respectivement le nombre de *WTs* (chacun est composé de bulldozers, d'excavateurs et des ressources humaines) et une flotte homogène de camions à benne disponibles au dépôt, avec une capacité Q chacun, en tonnes.

L'objectif du niveau stratégique est d'affecter les *WTs* aux sommets de débris à chaque période de l'horizon temporel ; tandis que le niveau opérationnel vise à définir des trajets multiples synchronisés pour les camions pendant chaque journée de travail, en satisfaisant leur capacité. Chaque WT est affecté à un sommet $i \in D$ si et seulement s'il n'y a pas de WT au sommet i pour un jour donné, et une WT reste au sommet i jusqu'à que tous les débris soient nettoyés. De plus, une WT commence une journée de travail en se déplaçant vers un sommet de débris et en enlevant les débris ou en enlevant les débris si elle est déjà affecté à un sommet. Par rapport aux camions, ils visitent les sommets $i \in D$ si et seulement s'il existe une WT au sommet i. En plus, les opérations de chargement et de déchargement ont un temps de service fixe, qui sont représentés respectivement par $t_l, t_u \in B$. Les camions commencent et finissent une journée de travail au dépôt. SRP-CD aborde ces deux objectifs de manière lexicographique : le premier objectif est de minimiser le nombre total de jours de travail pour nettoyer tous les sites contenant des débris, et le second cherche à minimiser le temps de trajet total des camions.

Formulation mathématique

La formulation mathématique proposée est donnée par les équations (A.1) à (A.20). Elle est basée sur une formulation multi-flot [Ahuja et al., 1989], et utilise un flot dynamique. Comme décrit dans Aronson [1989]; Wang [2018], la représentation la plus fréquente pour les flot dynamiques est une discrétisation du temps. Les journées et l'horizon temporel sont indexés sur le temps, selon une discrétisation considérant respectivement unités de temps et jours.

Malgré le grand nombre de variables et de contraintes de cette stratégie, elle permet d'aborder les deux niveaux de synchronisation de manière simple. A notre connaissance, la formulation dynamique multi-flot proposée est la première trouvée dans la littérature pour SRP-CD. Elle comporte deux flottes de véhicules : une de *WTs* et une autre de camions à benne. Les variables binaires utilisées sont : y_{ij}^h détermine si le WT visite le sommet j après i en début de journée de travail h ($y_{ij}^h = 1$), sinon ($y_{ij}^h = 0$) ; f_{ijk}^{th} indique si le camion k arrive au sommet j en provenance de i dans la période t du jour h ($f_{ijk}^{th} = 1$), sinon ($f_{ijk}^{th} = 0$) ; x_i^h établit si une WT est affectée au sommet i au cours du jour h ($x_i^h = 1$), sinon ($x_i^h = 0$).

De plus, les variables $0 \leq l_{ik}^{th} \leq Q$ déterminent la charge accumulée de chaque camion k au sommet i dans la période t du jour h ; tandis que les variables $0 \leq u_{ik}^{th} \leq Q$ indiquent le volume de débris nettoyés en i par le camion k dans la période t du jour h. La variable $\delta \geq 0$ représente un nombre de jours utilisés dans l'horizon temporel.

Le modèle est composé de deux fonctions objectifs $\mathcal{F}_1$ et $\mathcal{F}_2$ (respectivement données dans les équations (A.1) et (A.2)), qui sont réalisées dans les deux phases d'optimisation, de manière lexicographique. La première avec $\mathcal{F}_1$, qui minimise le nombre de jours pour nettoyer tous les sites de débris, et la seconde avec $\mathcal{F}_2$ qui optimise le temps de trajet total des camions, en satisfaisant l'objectif primaire. Le modèle est désigné comme multi-flot multi-période bi-objectif lexicographique.

$$\mathcal{F}_1 = min \quad \delta \quad s.t. \tag{A.1}$$

$$\mathcal{F}_2 = min \sum_{h=1}^{H'} \sum_{t=1}^{T} \sum_{k \in K} \sum_{(i,j) \in E} c_{ij} f_{ijk}^{th} \quad s.t. \tag{A.2}$$

$$\delta \geq h x_i^h \qquad \forall i \in D, h = \{1...H\} \tag{A.3}$$

$$\sum_{h=1}^{H} \sum_{j \in D} y_{0j}^h \leq W \tag{A.4}$$

$$x_i^h = \sum_{j \in \{0\} \cup D} y_{ji}^h \qquad \forall i \in D, h = \{1...H\} \tag{A.5}$$

$$\sum_{j \in \{0\} \cup D} y_{ji}^h = \sum_{j \in \{0\} \cup D} y_{ij}^{h+1} \qquad \forall i \in D, h = \{1...(H-1)\} \tag{A.6}$$

$$\sum_{h=1}^{H} \sum_{\substack{j \in \{0\} \cup D, \\ j \neq i}} y_{ji}^h = 1 \qquad \forall i \in D \tag{A.7}$$

$$\sum_{i \in D} \sum_{t=1}^{T - c_{0i}} f_{0ik}^{(t+c_{0i})h} \leq 1 \qquad \forall k \in K, h = \{1...H\} \tag{A.8}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{th} = \sum_{j:(i,j) \in E} f_{ijk}^{(t+s)h}, \begin{cases} s = c_{ij} + t_l, \text{if } i \in D \\ s = c_{ij} + t_u, \text{if } i \in L \end{cases} \qquad \begin{matrix} \forall k \in K, t = \{1...(T-s)\}, \\ h = \{1...H\} \end{matrix} \tag{A.9}$$

$$\sum_{j:(j,i) \in E} f_{jik}^{(t+c_{ji})h} \leq x_i^h \qquad \begin{matrix} \forall i \in D, \forall k \in K, \\ t = \{1...(T-c_{ji})\}, \\ h = \{1...H\} \end{matrix} \tag{A.10}$$

$$f_{jik}^{th} + \sum_{j' \in \{0\} \cup L} \sum_{\substack{k' \in K, \\ k' \neq k}} \sum_{t'=t}^{t+t_l-1} f_{j'ik'}^{t'h} \leq 1 \qquad \begin{matrix} \forall i \in D, \forall j \in \{0\} \cup L, \\ \forall k \in K, \\ t = \{1...(T-t_l+1)\}, \\ h = \{1...H\} \end{matrix} \tag{A.11}$$

$$l_{jk}^{(t+s)h} \geq l_{ik}^{th} + u_{jk}^{(t+s)h} - Q(1 - f_{ijk}^{(t+c_{ij})h}), \Big\{ s = c_{ij} + t_l \qquad \begin{matrix} \forall i \in V, \forall j \in D, \forall k \in K, \\ t = \{1...(T-s)\}, \\ h = \{1...H\} \end{matrix} \tag{A.12}$$

$$u_{jk}^{th} \leq Q \sum_{\substack{i:(j,i) \in E, \\ i \in L}} f_{jik}^{(t+c_{ji})h} \qquad \begin{matrix} \forall j \in D, \forall k \in K, \\ t = \{1...(T-c_{ji})\}, \\ h = \{1...H\} \end{matrix} \tag{A.13}$$

$$\sum_{h=1}^{H} \sum_{t=1}^{T} \sum_{k \in K} u_{jk}^{th} = v_j \qquad \forall j \in D \tag{A.14}$$

$$y_{ij}^{h} \in \{0,1\} \qquad \forall i,j \in D \cup \{0\}, h = \{1...H\} \tag{A.15}$$

$$x_{i}^{h} \in \{0,1\} \qquad \forall i \in D \cup \{0\}, h = \{1...H\} \tag{A.16}$$

$$f_{ijk}^{th} \in \{0,1\} \qquad \forall i,j \in V, \forall k \in K, t = \{1...T\}, h = \{1...H\} \tag{A.17}$$

$$0 \leq l_{ik}^{th} \leq Q \qquad \forall i \in V, \forall k \in K, t = \{1...T\}, h = \{1...H\} \tag{A.18}$$

$$0 \leq u_{jk}^{th} \leq Q \qquad \forall j \in D, \forall k \in K, t = \{1...T\}, h = \{1...H\} \tag{A.19}$$

$$\delta \geq 0 \tag{A.20}$$

Les contraintes (A.3), ainsi que la fonction objectif $\mathcal{F}_1$ déterminent le nombre minimum δ de jours de travail nécessaires pour nettoyer tous les sommets de débris. La conception du flot pour les *WTs* est adaptée du modèle de Sakuraba et al. [2016]. Ce flot permet à la formulation de s'assurer qu'une *WT* reste à un sommet de débris jusqu'à ce que celui-ci soit totalement nettoyé. Sans de telles contraintes, le modèle peut affecter chaque *WT* dans différents sommets de débris chaque jour avant de finir le nettoyage complet. Ce n'est pas une situation appropriée dans le contexte de la récupération. Les inégalités (A.4) garantissent que jusqu'à W *WTs* quittent le dépôt $\{0\}$. Les inégalités (A.5) relient les variables x à y et garantissent l'affectation de *WTs* à un sommet de débris i s'il existe un flot entrant au sommet i. Ces *WTs* peuvent provenir du dépôt ou d'un sommet de débris. Les contraintes (A.6) assurent la conservation du flot pour les *WTs*. L'idée générale reste similaire aux contraintes classiques de conservation de flot, où tout flot entrant vers un sommet de débris doit quitter ce sommet. La principale différence repose sur l'indexation dans le temps. Précisément, le flot doit sortir à une date supérieure à la date d'entrée plus le temps de trajet et le temps de temps de chargement pour les sommets de débris (resp. le temps de déchargement pour les décharges). Les contraintes (A.7) déterminent qu'un seul *WT* est attribué à chaque sommet de débris à une date spécifique.

Les contraintes (A.8) à (A.14) concernent les itinéraires des camions pour chaque journée de travail. Les inégalités (A.8) établissent que pour chaque jour, chaque camion peut être utilisé ou non. Les contraintes (A.9) indiquent la con-

servation du flot pour les camions lors de chaque jour. Comme pour les *WTs*, le flot est indexé dans le temps. La valeur s correspond à la somme du temps de déplacement (c_{ij}) et du temps de service (resp. t_l temps de chargement et t_u temps de déchargement). En conséquence, cela permet de contrôler la date des flot entrants et sortants. Les inégalités (A.10) relient les variables f_{ijk}^{th} et x_i^h. Elles garantissent qu'un flot entre dans un sommet de débris i si et seulement s'il y a une WT assigné à i. Les contraintes (A.11) garantissent la synchronisation entre les camions. Par exemple, si un camion (k) est en train de charger à un sommet, les autres camions (k') doivent attendre que l'opération de chargement (k) soit terminée. Pour ce faire, les variables de flot utilisent l'indexation sur le temps pour contrôler la date des flot entrants et sortants. Les inégalités (A.12) déterminent le volume accumulée de débris, chargée dans un camion. Ces contraintes sont inspirées des contraintes Miller, Tucker and Zemlin [Miller et al., 1960]. Les équations (A.13) établissent que la capacité des camions est satisfaite. Les contraintes (A.14) garantissent que tous les sommets de débris doivent être nettoyés jusqu'au dernier jour utilisé. Le domaine des variables est donné par les équations (A.15) à (A.20).

Chapitre 4 : Méthodes pour le SRP-CD

Ce chapitre est dédié à la description des méthodes proposées pour la résolution du SRP-CD. Plusieurs heuristiques constructives sont présentées. Ensuite, des métaheuristiques basées sur le *Large Neighborhood Search - LNS* sont décrites. Enfin, une métaheuristique hybride basée sur la LNS et le *Simulated Annealing - SA* est proposée.

Heuristiques constructives

Comme décrit dans Dorigo and Stützle [2003], les algorithmes d'approximation sont souvent capables de générer des solutions avec une qualité raisonnable à faible coût de calcul. Dans les méthodes constructives, une solution est construite à partir de rien et de manière itérative, les méthodes ajoutent des composants à la solution sans revenir en arrière jusqu'à obtenir une solution complète.

Les heuristiques constructives proposées utilisent un des six critères gloutons pour le niveau de décision stratégique, appelé *cWT* (*LDF, MDF, STTG, GTTF, SDTTF, GDTTF*), un des sept critères gloutons pour le niveau de décision opérationnel, appelé *cK* (*LDF, MDF, STTG, GTTF, SDTTF, GDTTF, LTF*), ou bien un choix aléatoire à la place des critères gloutons pour chaque décision. Il en résulte 42 combinaisons possibles de critères gloutons ou le choix aléatoire pour produire une solution complète réalisable pour le SRP-CD.

Les heuristiques constructives proposées passent par quatre étapes : l'étape (a) se concentre sur les décisions stratégiques de l'ordonnancement des *WTs*, tandis que les étapes (b), (c) et (d) sont liées aux décisions opérationnelles des trajets des camions. Selon les critères choisis, un ordre d'assignation est appliqué dans la construction pendant les étapes (a) et (b). L'étape (c) réalise l'opération de chargement de chaque camion au sommet de débris correspondant. Lors de l'étape (d), les camions chargés sont dirigés vers la décharge la plus proche et l'opération de déchargement est effectuée. Lorsque tous les camions sont rentrés au dépôt parce que la journée de travail est terminée, une nouvelle journée de travail commence.

La première heuristique constructive, appelée *GCH*, choisit la meilleure solution construite parmi les 42 critères gloutons. La seconde heuristique constructive, appelée *RCH*, prend la meilleure solution construite de manière aléatoire pendant 20 secondes d'exécution.

Métaheuristique basée sur LNS

La LNS est une métaheuristique qui effectue des améliorations itératives basées sur trois procédures principales : suppression (cR), insertion (cI) et acceptation. Les itérations se terminent lorsqu'un critère d'arrêt est atteint. La LNS a été proposée pour la première fois par Shaw [1998]. La procédure de suppression est une stratégie employée pour supprimer les composants d'une solution actuelle, tandis que l'insertion est un processus qui permet de réinsérer les composants supprimés dans la solution partielle, ce qui permet de générer une nouvelle solution voisine. Ensuite, la procédure d'acceptation décide comment la solution voisine trouvée est acceptée comme étant la nouvelle solution actuelle.

Dans le contexte du SRP-CD, la métaheuristique proposée utilise les procédures de suppression et d'insertion dans l'ordonnancement des *WTs* comme suit : Étant donné une solution réalisable, la procédure de suppression retire de manière aléatoire un sommet de débris de la solution. Après la suppression d'un sommet de débris, les itinéraires des véhicules sont également supprimés et doivent être refaits après l'insertion avec l'un des critères gloutons (cK'). L'insertion évalue l'une des deux recherches locales (Meilleure amélioration - BI ou Première amélioration - FI) pour réinsérer le sommet supprimé. La procédure d'acceptation accepte les solutions améliorées après la suppression et l'insertion de la même manière que dans l'article [Shaw, 1998]. Les procédures de suppression, d'insertion et d'acceptation sont appliquées de manière itérative jusqu'à ce qu'un critère d'arrêt soit satisfait : un nombre fixe d'itérations, un temps d'exécution fixe ou un nombre fixe d'itérations sans aucune amélioration.

Cette méthode est appelée *LNS-GG* si la solution initiale est construite à partir de critères de type glouton ou *LNS-RG* si la solution initiale est construite à l'aide du *RCH*. Des expérimentations pour le réglage des paramètres sont effectuées pour cette méthode dans le chapitre suivant.

Méthode hybride basée sur LNS et SA

La métaheuristique SA est une métaheuristique inspirée du processus de recuit physique pour les solides cristallins, où après avoir été chauffé, le solide est lentement refroidi pour réduire les effets sur le matériau, ce qui permet d'obtenir un solide avec une intégrité structurelle supérieure [Henderson et al., 2003]. SA a été initialement proposé par Kirkpatrick et al. [1983], où le concept physique de recuit a été adapté pour certains problèmes d'optimisation combinatoire en amélioration itérative. Le solide est analogue à une solution du problème correspondant, et le processus de refroidissement est analogue à la recherche d'une solution de meilleure qualité au cours des itérations de la méthode.

Comme décrit par les auteurs dans Henderson et al. [2003], SA part d'une solution initiale réalisable et d'une température initiale. La température est refroidie de manière itérative et à chaque itération, de nouvelles solutions voisines

sont générées. Chaque fois qu'une solution voisine améliore la solution courante, elle est acceptée comme la nouvelle solution courante. Lorsqu'une solution voisine n'améliore pas la solution courante, le critère de Boltzmann est utilisé. Ceci est fait pour essayer d'échapper aux optima locaux, et lorsque la température s'approche de zéro, le critère de Boltzmann se produit moins fréquemment [Kirkpatrick et al., 1983] que lorsque la température est proche de sa valeur initiale.

Dans le contexte du SRP-CD, la méthode hybride couple l'idée des opérations de suppression-insertion de LNS et le schéma de refroidissement du SA avec le critère de Boltzmann comme suit : Étant donné une solution réalisable, une température initiale et un taux de refroidissement (θ), plusieurs iterations sont faites jusqu'à un critère d'arrêt. À chaque itération, pour générer une solution de voisinage, la procédure de suppression est appliquée pour retirer de manière aléatoire un sommet de débris de la solution. Les itinéraires des véhicules sont également supprimés et doivent être refaits après l'insertion avec l'un des critères gourmands. L'insertion réinsère de manière aléatoire le sommet supprimé. La meilleure solution trouvée est conservée, mais, si après $|D|$ itérations cette solution ne change pas, on applique une forte pertubation en supprimant entre 1 et $|D|/2$ sommets et en les réinsérant selon une des recherches locales (BI ou FI). La solution voisine est acceptée comme la nouvelle solution courante lorsqu'elle est meilleure ou lorsque le critère de Boltzmann susmentionné a lieu. Ensuite, la température actuelle est refroidie par le taux θ. La méthode s'arrête lorsque la température atteint un seuil $\epsilon = 10^{-4}$.

Cette méthode est appelée *LNS-SA-GG* si la solution initiale est construite à partir de critères de type glouton et *LNS-SA-RG* si la solution initiale est construite à l'aide du *RCH*. Des expérimentations pour le réglage des paramètres sont effectuées pour cette méthode dans le chapitre suivant.

Chapitre 5 : Résultats

Ce chapitre est consacré aux expérimentations de calcul impliquant les méthodes proposées afin de tester leurs performances en termes de qualité des solutions et de temps d'exécution. Un mécanisme de génération de données est introduit pour les instances, et un ensemble de scénarios réalistes est proposé par une étude de

cas du tremblement de terre de Port-au-Prince, en Haïti, en 2010. Les résultats numériques sont présentés, où les résultats pour le modèle mathématique sont discutés, les heuristiques constructives gloutonnes sont classées, les expérimentations de réglage des paramètres pour les métaheuristiques sont décrites et les résultats et l'analyse pour les heuristiques proposées en utilisant toutes les instances sont détaillés.

Les tests ont été effectués sur un processeur Intel(R) Core(TM) i5-8350U @1.70-GHz avec 16GB de RAM et 2GB de swap, utilisant le système opérationnel Ubuntu 18.04.1 LTS. Tous les cœurs et la RAM ont été utilisés. Les approches proposées ont été développées en C++ et de plus, *IBM ILOG CPLEX* 12.8, avec les paramètres par défaut, et *CPLEX Concert Technology* en C++ ont été utilisés pour résoudre la formulation mathématique.

Génération des données

Aucune base de données spécifique n'a été trouvée pour ce problème, ainsi, un mécanisme de génération de données a été proposé, où les instances contiennent toutes les caractéristiques pertinentes du SRP-CD : W, $|K|$, Q, t_l, t_u, T et le graphe $G = (V, E)$, où $D, L, O \in V$. Le volume initial des débris $v_i, i \in D$, et, pour chaque arête, le temps de parcours entre les sommets $(c_{ij}, i, j \in V)$. Le graphe n'est pas nécessairement complet, dans ce cas, une matrice de distance est créée lors d'une phase de pré-traitement en utilisant l'algorithme de Dijkstra [Dijkstra, 1959].

La méthode de génération de données effectue une distribution géographique des sommets de débris sur un plan cartésien inspiré de l'organisation spatiale des quartiers urbains, ainsi que du benchmark de Solomon [Solomon, 1987]. Il en résulte trois topologies de réseau, appelées ici *(C)luster*, *(R)andom*, et *(M)ix*. La distance euclidienne est associée à chaque arête représentant le temps de parcours entre deux sommets.

Afin d'effectuer les expérimentations, trois ensembles d'instances ont été créés : S0 (petite taille), S1 (taille moyenne) et S2 (grande taille). Pour l'ensemble S0, 36 instances ont été générées, 12 avec une topologie C, 12 pour la topologie R et 12 avec une topologie M. Pour chaque ensemble S1 et S2, 90 instances ont été

générées, 30 avec une topologie C, 30 pour la topologie R et 30 avec une topologie M. Les caractéristiques des ensembles d'instances sont présentées dans le Tableau A.2.

	# Inst.	$\|D\|$	$\|L\|$	W	$\|K\|$	Q	t_l t_u	T
S0	36	5	1	2, 3	2, 3, 4	2	1	30, 35, 40
S1	90	10, 20, 30, 40, 50	2	2, 3, 4	2, 3, 4, 6, 8	2	1	40, 55, 60, 65
S2	90	100, 200, 300, 400, 500	3	5, 6, 7	5, 6, 7, 10, 12, 14	2	10	480, 720

Table A.2: Caractéristiques des ensembles d'instances.

L'étude de cas du tremblement de terre de Port-au-Prince rassemble des informations provenant de différentes sources [ICSMD, 2021; SERTIT, 2021; UNDP, 2013; Gorelick et al., 2017; Sinotruk, 2021]. Les données sont basées sur un graphe avec $|V| = 5,764$, $|E| = 7,301$, un dépôt et une décharge. Le volume total des débris utilisés est de 10 millions de mètres cubes. Les caractéristiques de chaque scénario sont résumées dans le tableau A.3.

T	$\|D\|$	$\|L\|$	W	$\|K\|$	Q	t_l	t_u
8h	390	1	21	21 42 83	$20m^3$	10 $min.$	5 $min.$
10h	390	1	21	21 42 83	$20m^3$	10 $min.$	5 $min.$
12h	390	1	21	21 42 83	$20m^3$	10 $min.$	5 $min.$

Table A.3: Caractéristiques de chaque scénario de l'étude de cas.

Résultats numériques

Les benchmarks de l'ensemble d'instances S0 a été utilisé pour tester le modèle mathématique. Les benchmarks des ensembles d'instances S1 et S2 sont utilisés pour mesurer la robustesse et l'évolution des heuristiques proposées. Des expérimentations numériques approfondies ont été réalisées avec les méthodes proposées et sont détaillées dans cette section.

Résultats du modèle mathématique

La formulation mathématique a été validée avec 9 petites instances (resp. 3 instances pour chaque topologie C, R et M) avec $|D| = 5$, $W = 2$ et $|K| = 3$, pour lesquelles elle s'est avérée optimale pour les deux fonctions objectifs.

Cependant, le plus pertinent est de montrer les limites du modèle. Pour cela, nous avons pris les 18 plus petites instances (resp. 6 pour chaque topologie) de S1 avec $|D| = 10$. Le temps d'exécution a été limité à 2 heures pour chaque fonction objectif, basé sur les études de Soares et al. [2019]; Souza et al. [2010]. La formulation mathématique a permis d'obtenir un résultat optimal pour la première fonction objectif pour 7 instances sur 18. Évidemment, lorsque les W et $|K|$ augmentent, un grand nombre de variables et de contraintes sont nécessaires. Dans certains cas, la combinatoire a augmenté, et la formulation a dépassé le temps limite sans prouver l'optimalité.

Classement des heuristiques constructives gloutonnes

Les 42 combinaisons différentes de critères gloutons ont été comparées par analyse du classement, du rang absolu et du rang moyen. Les résultats ont mis en évidence quatre critères parmi les combinaisons considérées, LDF, MDF, STTF et GDTTF. Ainsi, ces critères sont utilisés pour réduire le nombre de possibilités dans le réglage de la métaheuristique dans la section suivante.

Prétraitement et réglage des métaheuristiques

Le réglage a été effectué à l'aide du package IRACE (Iterated racing for automatic algorithm configuration) [López-Ibánez et al., 2016], qui fonctionne avec un

mécanisme d'apprentissage automatique.

Pour la métaheuristique *LNS-based*, des tests de prétraitement ont été effectués pour choisir le critère d'arrêt parmi l'ensemble des critères considérés, et les résultats ont pointé vers le nombre d'itérations sans amélioration. Ainsi, les paramètres suivants ont été réglés avec l'ensemble d'instances S1 : cWT, cK, cK' considérant les critères LDF, MDF, STTF et GDTTF ; la recherche locale pour le critère d'insertion cI considérant BI et FI ; le critère d'arrêt considérant $\{50, 60, 70, 80, 90, 100\}$ itérations sans amélioration. Les résultats sont présentés dans le Tableau A.4 afin de trouver les paramètres qui donnent les meilleurs résultats pour chacune topologie.

Topology	cWT	cK	cI	cK'	critère d'arrêt
C	STTF	STTF	FI	STTF	100
R	STTF	GDTTF	BI	STTF	90
M	STTF	LDF	FI	STTF	100

Table A.4: Paramètres de la métaheuristique basée sur la LNS, calibrés par l'IRACE.

Pour la métaheuristique *basée sur LNS et SA*, les paramètres suivants ont été réglés avec l'ensemble d'instances S1 : cWT, cK, cK' considérant les critères LDF, MDF, STTF et GDTTF ; et la recherche locale pour le critère d'insertion cI dans la forte pertubation considérant BI et FI. Les résultats du réglage sont présentés par topologie dans le Tableau A.5.

Topology	cWT	cK	cI	cK'
C	LDF	GDTTF	BI	STTF
R	STTF	LDF	BI	STTF
M	LDF	GDTTF	BI	STTF

Table A.5: Paramètres de la métaheuristique basée sur LNS et SA, calibrés par l'IRACE.

Comparaison des heuristiques

Les résultats décrits dans cette section prennent en compte les performances en termes de qualité des solutions et de temps d'exécution de chaque méthode dans les ensembles S1 et S2. Les méthodes évaluées sont les heuristiques constructives GCH et

RCH (en utilisant un temps d'exécution fixe de 20 secondes comme critère d'arrêt), et les métaheuristiques utilisant la calibration IRACE décrite, *LNS-GG*, *LNS-RG*, *LNS-SA-GG* et *LNS-SA-RG*. Pour la *RCH* et les quatre métaheuristiques, les résultats considèrent le meilleur résultat de 10 exécutions différentes pour chaque instance, en utilisant différentes graines.

Les résultats sont présentés dans les Tableaux A.6, A.7 et A.8 respectivement pour les topologies *cluster*, *random* et *mix*, où les meilleurs résultats sont mis en évidence en gras. Pour chaque topologie, S1 et S2 contiennent chacun 30 instances, ce qui donne un total de 60 instances ($S1 + S2$). Dans les tableaux, chaque ligne représente les méthodes, et pour chaque méthode, les résultats considèrent les 30 instances de S1, les 30 instances de S2 et les 60 instances de S1 + S2. Il convient de mentionner qu'une solution est considérée comme meilleure lorsqu'elle a la plus petite valeur de $\mathcal{F}_1$ (jours), ou en cas d'égalité, lorsqu'elle a la plus petite valeur de $\mathcal{F}_2$ (temps de trajet total). Ainsi, un écart et un écart relatif sont calculés entre la solution obtenue ($\mathcal{F}(m)$) pour chaque méthode et la meilleure solution connue ($\mathcal{F}(b)$) respectivement pour le premier et le second objectif, comme indiqué dans les équations (A.21) et (A.22).

$$Dev = \mathcal{F}_1(m) - \mathcal{F}_1(b) \tag{A.21}$$

$$Dev(\%) = \frac{\mathcal{F}_2(m) - \mathcal{F}_2(b)}{\mathcal{F}_2(b)} \times 100 \tag{A.22}$$

En considérant les colonnes intérieures des tableaux, *Nb_best* représente le nombre de meilleures solutions connues parmi les méthodes évaluées ; α montre la moyenne Dev des instances de chaque colonne ; β fournit la moyenne $Dev(\%)$ des instances pour chaque colonne, et $t(s)$ donne le temps d'exécution moyen en secondes. Par conséquent, une méthode est considérée comme meilleure selon un ordre lexicographique de *Nb_best*, α et β.

Les résultats du Tableau A.6 indiquent que la méthode *LNS-RG* a donné de meilleurs résultats que les autres méthodes et a trouvé environ 48% (29 sur 60) des meilleures solutions connues, avec une moyenne de 0.67 jours et de 0.49% pour le temps de trajet total. Cependant, en S1, la meilleure méthode était *LNS-SA-RG*

avec 77% des meilleures solutions connues et, en S2, la meilleure méthode était *LNS-RG* avec 70% des meilleures solutions connues.

Méthodes	S1				S2				S1 + S2			
	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
GCH	0	3.27	5.53	~ 0	0	2.20	8.90	0.07	0	2.73	7.22	0.03
RCH	2	1.93	2.98	20	0	4.97	21.51	20	2	3.45	12.25	20
LNS-GG	11	0.40	0.34	0.22	0	1.10	2.90	693.34	11	0.75	1.62	346.78
LNS-RG	8	1.07	1.30	20	**21**	**0.27**	**-0.33**	580.12	**29**	**0.67**	**0.49**	300.06
LNS-SA-GG	18	0.17	0.10	1.85	8	0.10	1.44	453.10	26	0.13	0.77	227.48
LNS-SA-RG	**23**	**0.07**	**0.07**	21.37	1	0.13	1.56	480.74	24	0.10	0.82	251.06

Table A.6: Résultats pour les méthodes dans les instances avec la topologie *cluster*.

Les résultats du Tableau A.7 indiquent que la méthode *LNS-RG* a donné de meilleurs résultats, où elle a trouvé environ 43% des meilleures solutions connues, avec une moyenne de 0.97 jours et de 1.35% pour le temps de trajet total. Cependant, en S1, la meilleure méthode était *LNS-SA-GG* avec 53% des meilleures solutions connues et, en S2, la meilleure méthode était *LNS-RG* avec 67% des meilleures solutions connues.

Méthodes	S1				S2				S1 + S2			
	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
GCH	0	4.53	8.93	~ 0	0	2.17	9.08	0.03	0	3.35	9.01	0.02
RCH	6	2.73	5.21	20	0	4.70	20.45	20	6	3.72	12.83	20
LNS-GG	3	0.63	1.04	0.22	0	1.57	4.57	546.17	3	1.10	2.80	273.20
LNS-RG	6	1.60	2.89	20	**20**	**0.33**	**-0.20**	499.80	**26**	**0.97**	**1.35**	259.90
LNS-SA-GG	**19**	**0.07**	**0.22**	1.82	5	0.10	1.20	454.08	24	0.08	0.71	227.95
LNS-SA-RG	16	0.13	0.12	21.32	5	0.07	1.21	486.38	21	0.10	0.67	253.85

Table A.7: Résultats pour les méthodes dans les instances avec la topologie *random*.

Les résultats du Tableau A.8 indiquent que la méthode *LNS-SA-RG* a donné de meilleurs résultats que les autres méthodes et a trouvé environ 40% des meilleures solutions connues, avec une moyenne de 0.08 jours et de 0.72% pour le temps de trajet total. Cependant, en S1, la meilleure méthode était *LNS-SA-GG* avec 57% des meilleures solutions connues et, en S2, la meilleure méthode était *LNS-RG* avec 57% des meilleures solutions connues.

Nous avons remarqué que les améliorations obtenues en utilisant les méthodes *LNS-GG*, *LNS-RG*, *LNS-SA-GG* et *LNS-SA-RG* par rapport aux heuristiques constructives deviennent très pertinentes, en particulier pour S2. Sans surprise, les métaheuristiques nécessitent un temps d'exécution plus long que les heuristiques

Méthodes	S1				S2				S1 + S2			
	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)	Nb_best	α	β	t(s)
GCH	0	3.37	9.65	~ 0	0	3.13	10.65	0.03	0	3.25	10.15	0.02
RCH	6	2.03	6.51	20	0	5.23	23.41	20	6	3.63	14.96	20
LNS-GG	8	0.93	2.64	0.26	0	2.47	4.31	688.24	8	1.70	3.48	344.25
LNS-RG	7	1.03	3.26	20	**17**	**0.50**	**0.82**	587.69	24	0.77	2.04	303.85
LNS-SA-GG	**17**	**0.07**	**0.34**	1.80	5	0.10	1.38	461.32	22	0.08	0.86	231.56
LNS-SA-RG	16	0.10	0.17	21.30	8	0.07	1.27	485.73	**24**	**0.08**	**0.72**	253.52

Table A.8: Résultats pour les méthodes dans les instances avec la topologie *mix*.

constructives. De plus, le temps d'exécution moyen augmente avec la taille de l'instance.

En résumé, du point de vue des ensembles S1 et S2, les méthodes *LNS-SA-GG* et *LNS-SA-RG* ont mieux fonctionné pour les instances S1, où *LNS-SA-RG* était meilleure pour la topologie *cluster* et *LNS-SA-GG* était meilleure pour les topologies *random* et *mix*. Pour les instances S2, les résultats de *LNS-RG* ont surpassé les autres heuristiques dans les trois topologies. Du point de vue des topologies, *LNS-RG* a mieux fonctionné pour les topologies *cluster* et *random*, tandis que *LNS-SA-RG* a mieux fonctionné pour la topologie *mix*.

Pour l'étude de cas, la méthode *LNS-RG* a été choisie car elle a obtenu les meilleurs résultats pour les plus grandes instances (S2). En outre, le nombre d'itérations et d'exécutions de la méthode a été réduit afin d'éviter les expérimentations chronophages. Les valeurs de $\mathcal{F}_1$ sont présentées en années, mois et jours, tandis que les valeurs de $\mathcal{F}_2$ sont présentées en heures. La colonne *t(s)* représente le temps d'exécution en secondes. Les résultats sont présentés dans le tableau A.9.

T	W	$\|K\|$	LNS-RG				
			$\mathcal{F}_1$			$\mathcal{F}_2$	t(s)
			Années	*Mois*	*Jours*	*Heures totales de voyage*	
8h	21	21	4	5	17	108 551	2558
		42	2	7	17	109 119	4154
		83	1	10	26	110 740	3774
10h	21	21	3	6	27	107 910	2000
		42	2	1	3	108 351	2294
		83	1	6	8	109 659	3879
12h	21	21	2	11	26	107 470	2005
		42	1	9	0	107 845	2202
		83	1	3	9	108 967	4602

Table A.9: Résultats pour les instances de l'étude de cas.

Selon les scénarios, la méthode fournit une solution qui prend un peu plus d'un

an à quatre ans pour enlever 10 million de mètres cubes de débris, en fonction des ressources disponibles telles que les *WTs* et les camions à benne. C'est un résultat intéressant si on le compare à ce qui se passe dans la pratique. Le rapport de UNDP [2013] présente qu'après deux ans, seulement 1 million de mètres cubes ont été enlevés sur les 10 millions estimés.

Chapitre 6 : Conclusions et perspectives futures

Cette thèse apporte plusieurs contributions pour le problème intégré de Planification et d'organisation de Tournées de nettoyage des débris (SRP-CD) à la suite de catastrophes majeures : une formulation mathématique avec deux niveaux de synchronisation (entre les véhicules de travail (*WTs*) et les camions à benne, et entre les camions à benne eux-mêmes, comprenant deux dimensions temporelles (journées de travail et horizon temporel), et ayant deux fonctions objectifs (une pour minimiser le nombre total de jours et une autre pour minimiser le temps de trajet total des camions) ; plusieurs heuristiques constructives, des métaheuristiques et un mécanisme de génération de données suivant trois topologies différentes (*cluster*, *random* et *mix*) ont été proposées. De plus, des expérimentations ont été réalisées pour mesurer la performance et la robustesse des méthodes proposées. Les paramètres des métaheuristiques ont été calibrés en utilisant IRACE, une méthode basée sur l'apprentissage automatique.

En ce qui concerne le modèle mathématique, il est basé sur une formulation multi-flot dynamique et il a été validé sur de petites instances. A notre connaissance, il s'agit de la première formulation intégrant la planification et l'organisation des tournées pour le SRP-CD, incluant tous les paramètres étudiés dans cette étude. Malgré ses limites pour la résolution d'instances moyennes, le flot temporel nous a tout de même permis d'aborder les deux niveaux de synchronisation.

Les résultats indiquent que *LNS-RG* surpasse les autres heuristiques pour les topologies *cluster* et *random* et que *LNS-SA-RG* et *LNS-RG* ont donné de très bons résultats pour la topologie *mix*. Pour les instances *cluster*, *LNS-RG* a trouvé environ 48% des meilleures solutions connues, tandis que *LNS-SA-GG* a atteint environ 43% des meilleures solutions connues et *LNS-SA-RG* a atteint environ 40%

des meilleures solutions connues. Pour les instances *random*, *LNS-RG* a atteint environ 43% des meilleures solutions connues, *LNS-SA-GG* a atteint environ 40% des meilleures solutions connues, et *LNS-GG* a obtenu 21% des meilleures solutions connues. Finalement, pour les instances *mix*, *LNS-SA-RG* et *LNS-RG* ont trouvé 40% des meilleures solutions connues, tandis que *LNS-SA-GG* a obtenu environ 37% des meilleures solutions connues. Les résultats ont montré que la méthode *LNS-RG* est plus performante et plus robuste pour les plus grandes instances que les méthodes proposées.

Dans l'étude de cas, la méthode *LNS-RG* fournit une solution qui prend un peu plus d'un an à quatre ans pour enlever 10 million de mètres cubes de débris, en fonction des ressources disponibles telles que les *WTs* et les camions à benne. C'est un résultat intéressant si on le compare à ce qui se passe dans la pratique, où le rapport de UNDP [2013] présente qu'après deux ans, seulement 1 million de mètres cubes ont été enlevés sur les 10 millions estimés.

Cette étude ouvre plusieurs directions de recherche. Plusieurs questions ouvertes ont été soulevées concernant l'intégration d'autres problèmes d'optimisation en fonction du volume des débris. C'est le cas de l'ordonnancement couplé au problème SDVRP, ou encore de la transition entre SRP-CD (RCPSP et FTVRP) et le RCPSP et SDVRP intégrés. De plus, des questions supplémentaires peuvent également être prises en compte telles que : donner une priorité aux sommets de débris, par exemple un sommet où les débris apportent un risque de toxicité, ou bien inclure des relations de précédences entre les sommets de débris, par exemple lorsque des débris ne sont pas accessibles tant que les autres sommets ne sont pas déblayés.

En ce qui concerne les approches de résolution, des méthodes hybrides et de nouvelles manières d'explorer les voisinages peuvent être proposés, par exemple, la *Adptive Large Neighborhood Search* proposant un ensemble d'opérateurs de suppression et d'opérateurs d'insertion, ce qui permet donc une meilleure exploration du voisinage d'une solution. En particulier, il serait possible d'améliorer la partie des tournées pour le SRP-CD.

Appendix B

Full tables

This appendix is devoted to present the complete results obtained from the proposed methods in each instance generated for the sets S1 and S2. Results in Tables B.1, B.2, B.3, B.4, B.5 and B.6 include the best solution found among the methods, and for each method, the results for $\mathcal{F}_1$, $\mathcal{F}_2$ and the running time $t(s)$ in seconds.

In the following, the complete results are showed in terms of the best, the worst and the average solutions after the 10 runs for each method and each instance from cluster, random and mix topologies. *RCH* results are showed in the Tables B.7, B.8, B.9. *LNS-GG* results are showed in the Tables B.10, B.11, B.12. *LNS-RG* results are showed in the Tables B.13, B.14, B.15. *LNS-SA-GG* results are showed in the Tables B.16, B.17, B.18. And *LNS-SA-RG* results are showed in the Tables B.19, B.20, B.21.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S1	10	2	40	2	2	20	1126	20	1147	~ 0	20	1129	20	20	1140	~ 0	20	1128	20	**20**	**1126**	~ 0	**20**	**1126**	20
				3	3	13	1122	13	1129	~ 0	13	1125	20	**13**	**1122**	~ 0	13	1125	20	**13**	**1122**	0.2	**13**	**1122**	20
				4	4	10	1122	10	1134	~ 0	10	1123	20	10	1130	~ 0	10	1123	20	**10**	**1122**	0.2	**10**	**1122**	20
				2	4	10	1143	11	1189	~ 0	**10**	**1143**	20	11	1150	~ 0	**10**	**1143**	20	11	1142	0.3	**10**	**1143**	20
				3	6	7	1137	7	1154	~ 0	**7**	**1137**	20	7	1142	~ 0	**7**	**1137**	20	7	1140	0.5	**7**	**1137**	20
				4	8	5	1130	6	1150	~ 0	5	1133	20	6	1142	~ 0	5	1133	20	6	1138	0.2	**5**	**1130**	20
	20	2	40	2	2	36	2102	41	2308	~ 0	38	2187	20	**36**	**2102**	~ 0	**36**	**2102**	20	**36**	**2102**	0.6	**36**	**2102**	20
				3	3	24	2102	27	2274	~ 0	25	2148	20	**24**	**2102**	~ 0	**24**	**2102**	20	**24**	**2102**	0.5	**24**	**2102**	20
				4	4	18	2102	19	2180	~ 0	19	2130	20	**18**	**2102**	0.1	**18**	**2102**	20	**18**	**2102**	0.7	**18**	**2102**	20
				2	4	19	2144	22	2352	~ 0	20	2241	20	19	2180	~ 0	20	2204	20	**19**	**2144**	0.8	**19**	**2144**	20
				3	6	13	2130	14	2322	~ 0	13	2210	20	**13**	**2130**	~ 0	13	2144	20	**13**	**2130**	1	**13**	**2130**	20
				4	8	9	2116	10	2252	~ 0	10	2185	20	10	2130	0.1	10	2144	20	**9**	**2116**	1	**9**	**2116**	20
	30	2	55	2	2	55	4692	57	4825	~ 0	56	4777	20	55	4707	0.2	56	4777	20	55	4697	1.3	**55**	**4692**	21
				3	3	36	4677	38	4782	~ 0	37	4768	20	37	4690	0.3	37	4768	20	37	4693	1.6	**36**	**4677**	21
				4	4	28	4701	29	4820	~ 0	28	4747	20	28	4712	0.1	28	4747	20	28	4710	1.7	**28**	**4701**	21
				2	4	28	4795	31	5039	~ 0	29	4857	20	29	4823	~ 0	29	4857	20	**28**	**4795**	1.1	28	4796	21
				3	6	19	4747	19	4801	~ 0	19	4837	20	**19**	**4747**	0.2	19	4837	20	19	4749	1.3	**19**	**4747**	21
				4	8	14	4745	15	4939	~ 0	15	4831	20	**14**	**4745**	0.1	15	4831	20	15	4788	2	15	4791	21
	40	2	60	2	2	75	6801	84	7366	~ 0	82	7275	20	**75**	**6801**	0.2	75	6823	20	**75**	**6801**	2.1	**75**	**6801**	22
				3	3	50	6801	55	7189	~ 0	54	7146	20	**50**	**6801**	0.3	**50**	**6801**	20	**50**	**6801**	2.4	**50**	**6801**	22
				4	4	38	6801	41	7081	~ 0	40	7097	20	**38**	**6801**	0.3	**38**	**6801**	20	**38**	**6801**	3	**38**	**6801**	22.2
				2	4	39	7012	45	7795	~ 0	43	7392	20	41	6978	0.4	40	7051	20	39	7018	2.2	**39**	**7012**	22
				3	6	25	6823	29	7570	~ 0	28	7328	20	26	6867	0.2	**25**	**6823**	20	26	6845	2.6	25	6825	22
				4	8	19	6845	21	7294	~ 0	21	7255	20	19	6867	0.4	20	6937	20	**19**	**6845**	3.1	**19**	**6845**	23
	50	2	65	2	2	98	11019	113	11873	~ 0	108	11561	20	100	11074	0.6	108	11561	20	**98**	**11019**	3.7	98	11035	23
				3	3	67	11055	75	11829	~ 0	71	11491	20	67	11083	0.4	71	11491	20	**67**	**11055**	3.7	**67**	**11055**	23
				4	4	50	11043	56	11788	~ 0	53	11398	20	51	11111	0.6	53	11398	20	50	11055	4.3	**50**	**11043**	24
				2	4	52	11255	59	11974	~ 0	56	11723	20	53	11220	0.7	54	11373	20	**52**	**11255**	3.7	53	11275	23
				3	6	34	11112	39	12058	~ 0	37	11626	20	34	11196	0.5	37	11626	20	**34**	**11112**	4.3	34	11168	23.9
				4	8	26	11167	29	11971	~ 0	28	11585	20	**26**	**11167**	1	28	11585	20	26	11172	5.5	26	11197	24.9

Table B.1: Results for the approaches in all instances of S1 with cluster topology.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S2	100	3	480	5	5	24	34263	25	36996	~ 0	26	40487	20	24	34943	9	**24**	**34263**	27.5	24	34601	23.9	24	34622	42.3
				6	6	20	34543	21	36556	~ 0	22	40265	20	21	35002	7.5	20	34876	25.8	**20**	**34543**	25	20	34559	44
				7	7	17	34736	19	36235	~ 0	19	40718	20	18	34894	10.3	18	34474	28.9	**17**	**34736**	27.9	18	34761	46.8
				5	10	20	38376	20	43131	~ 0	20	43799	20	20	39526	6.3	**20**	**38376**	25.7	20	38511	25	20	38567	45
				6	12	17	38284	17	41647	~ 0	17	44489	20	17	38816	8.7	**17**	**38284**	27.8	17	38570	28.6	17	38658	48.4
				7	14	15	38358	15	41176	~ 0	15	44983	20	15	38659	8.3	**15**	**38358**	27	15	38548	30	15	38666	50.6
	200	3	480	5	5	47	67603	50	73243	~ 0	54	84333	20	49	68731	72.2	48	66651	92.1	**47**	**67603**	112.5	47	67629	133.4
				6	6	40	66265	42	71950	~ 0	45	84293	20	41	68562	63.8	**40**	**66265**	92.8	40	67467	118.5	40	68257	137.1
				7	7	35	66896	37	70894	~ 0	39	84444	20	35	67981	80	35	67079	106.4	**35**	**66896**	130.2	35	67577	143.9
				5	10	40	77081	40	85824	~ 0	40	92240	20	40	79141	71.2	**40**	**77081**	71	40	78588	120	40	79229	136.7
				6	12	34	76560	34	85592	~ 0	34	93368	20	34	79465	64.9	**34**	**76560**	79.3	34	78898	129.7	34	79011	153.1
				7	14	29	76147	29	84774	~ 0	29	93872	20	29	79016	64.2	**29**	**76147**	89.2	29	78529	139.9	29	78115	155.8
	300	3	480	5	5	80	120301	85	130172	1	91	148374	20	82	124016	361.4	**80**	**120301**	363.1	80	123006	311.5	80	121927	331.4
				6	6	68	121922	72	129438	~ 0	76	148463	20	70	125207	377.8	**68**	**121922**	343.6	68	122137	331.3	68	122520	339.6
				7	7	59	121077	62	130139	~ 0	65	149470	20	61	126199	315.4	**59**	**121077**	308	60	122361	321	59	123479	343.4
				5	10	60	131179	60	152865	~ 0	60	159086	20	60	137931	281.5	**60**	**131179**	165.6	60	135225	303.1	60	134323	358.8
				6	12	50	130958	51	149617	~ 0	50	159538	20	51	137552	331.3	**50**	**130958**	192.1	50	134304	332.4	50	134095	357.7
				7	14	43	130308	43	155636	~ 0	43	159378	20	43	137654	320.4	**43**	**130308**	176	43	133433	339.5	43	133213	385.5
	400	3	720	5	5	97	254506	105	271354	~ 0	117	314178	20	101	255646	1265.1	99	249391	976.1	**97**	**254506**	658	98	256455	635
				6	6	82	248549	88	269446	~ 0	98	312422	20	85	257986	1130.1	**82**	**248549**	872	83	255747	649.7	83	255628	690.1
				7	7	71	256036	75	265629	~ 0	83	319352	20	74	259760	865.3	72	251401	854.9	72	255681	629.9	**71**	**256036**	667.2
				5	10	80	284626	81	311266	~ 0	80	340643	20	80	293969	733.7	**80**	**284626**	633.4	80	293238	663	80	291456	648
				6	12	67	284409	67	310565	~ 0	67	342250	20	67	291604	683.7	**67**	**284409**	638.7	67	292140	666.6	67	292258	665.2
				7	14	58	282679	58	311265	1	58	343770	20	58	290643	847.4	**58**	**282679**	820.7	58	290850	674.6	58	290205	707.1
	500	3	720	5	5	116	295298	125	309042	~ 0	138	364775	20	120	299883	2809.8	117	289758	2210	**116**	**295298**	1054.6	117	295978	1116.5
				6	6	97	294385	104	315911	~ 0	115	364095	20	101	302165	2842.7	98	290019	1999.9	**97**	**294385**	1125.4	97	294731	1137.7
				7	7	84	293859	90	305583	~ 0	98	365888	20	87	301205	2046.8	85	288885	1894.7	**84**	**293859**	1131.9	84	294619	1128.3
				5	10	100	328747	101	360858	~ 0	100	393447	20	100	341699	1887.9	**100**	**328747**	1273.8	100	337781	1038.6	100	337343	1236.2
				6	12	84	327984	84	357427	~ 0	84	395041	20	84	344674	1649.1	**84**	**327984**	1523.6	84	335741	1212.6	84	336610	1255.9
				7	14	72	326016	72	354736	~ 0	72	395647	20	72	342505	1584.4	**72**	**326016**	1463.9	72	334859	1238.1	72	334596	1281.6

Table B.2: Results for the approaches in all instances of S2 with cluster topology.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S1	10	2	40	2	2	20	1299	21	1363	~ 0	**20**	**1299**	20	**20**	**1299**	~ 0	**20**	**1299**	20	**20**	**1299**	0.1	**20**	**1299**	20
				3	3	13	1291	14	1343	~ 0	**13**	**1291**	20	14	1299	0.1	**13**	**1291**	20	14	1299	0.2	**13**	**1291**	20
				4	4	10	1291	11	1323	~ 0	**10**	**1291**	20	10	1299	0.1	**10**	**1291**	20	10	1299	0.2	**10**	**1291**	20
				2	4	11	1327	12	1383	~ 0	**11**	**1327**	20	11	1339	~ 0	**11**	**1327**	20	11	1339	0.2	**11**	**1327**	20
				3	6	7	1299	8	1343	~ 0	**7**	**1299**	20	7	1351	~ 0	**7**	**1299**	20	7	1307	0.5	**7**	**1299**	20
				4	8	5	1291	6	1323	~ 0	**5**	**1291**	20	6	1315	~ 0	**5**	**1291**	20	6	1315	0.3	**5**	**1291**	20
	20	2	40	2	2	36	2317	43	2633	~ 0	40	2469	20	37	2353	~ 0	37	2335	20	**36**	**2317**	0.7	36	2321	20
				3	3	25	2335	27	2489	~ 0	26	2430	20	**25**	**2335**	~ 0	25	2353	20	**25**	**2335**	0.5	**25**	**2335**	20
				4	4	19	2335	21	2498	~ 0	20	2421	20	19	2371	0.1	19	2353	20	**19**	**2335**	1	**19**	**2335**	20
				2	4	20	2485	24	2738	~ 0	21	2581	20	21	2532	0.1	21	2523	20	**20**	**2485**	0.8	21	2485	20
				3	6	13	2389	15	2657	~ 0	14	2557	20	14	2451	~ 0	13	2415	20	**13**	**2389**	0.8	13	2425	20
				4	8	10	2391	11	2646	~ 0	11	2537	20	10	2409	0.2	10	2461	20	**10**	**2391**	1	10	2407	20.4
	30	2	55	2	2	40	3623	48	4025	~ 0	47	3991	20	42	3688	0.1	43	3707	20	40	3631	1.1	**40**	**3623**	21
				3	3	27	3640	32	3976	~ 0	31	3974	20	28	3662	0.1	28	3663	20	**27**	**3640**	1.4	28	3617	21
				4	4	21	3643	23	3922	~ 0	23	3935	20	21	3682	0.1	22	3727	20	21	3660	1.7	**21**	**3643**	21
				2	4	24	3912	29	4463	~ 0	26	4113	20	25	3985	0.2	26	4113	20	24	3923	1.3	**24**	**3912**	21
				3	6	15	3788	19	4224	~ 0	17	4109	20	16	3844	0.1	16	3910	20	**15**	**3788**	1.5	15	3796	21
				4	8	12	3797	14	4407	~ 0	13	4142	20	12	3812	0.3	12	3930	20	**12**	**3797**	1.8	12	3821	21
	40	2	60	2	2	78	7949	91	8611	~ 0	88	8464	20	79	7981	0.3	80	8035	20	**78**	**7949**	2.4	78	7961	22
				3	3	53	7971	60	8574	~ 0	57	8375	20	54	8019	0.4	57	8375	20	53	7979	2.5	**53**	**7971**	22
				4	4	40	8005	44	8539	~ 0	43	8317	20	**40**	**8005**	0.3	43	8317	20	**40**	**8005**	3.1	40	8007	22.5
				2	4	43	8183	48	8829	~ 0	46	8580	20	43	8263	~ 0	46	8580	20	43	8247	2.5	**43**	**8183**	22
				3	6	27	8035	32	8811	~ 0	30	8533	20	28	8093	0.3	30	8533	20	**27**	**8035**	2.5	27	8053	22
				4	8	21	8089	23	8607	~ 0	22	8493	20	21	8115	0.4	22	8493	20	**21**	**8089**	3.8	21	8109	23
	50	2	65	2	2	84	8984	100	10123	~ 0	94	9739	20	86	9070	0.5	85	9033	20	**84**	**8984**	3.1	**84**	**8984**	23
				3	3	56	8938	66	10022	~ 0	62	9606	20	56	9002	0.7	62	9606	20	**56**	**8938**	4	56	8952	23
				4	4	42	8990	49	10016	~ 0	46	9559	20	43	9038	0.6	46	9559	20	**42**	**8990**	4	43	8992	23.8
				2	4	45	9353	53	10464	~ 0	50	9969	20	46	9357	0.4	50	9969	20	**45**	**9353**	3	46	9386	22.9
				3	6	30	9246	35	10348	~ 0	33	9957	20	31	9372	0.5	33	9957	20	30	9259	3.9	**30**	**9246**	23
				4	8	22	9201	26	10218	~ 0	25	9852	20	23	9283	0.6	25	9852	20	**22**	**9201**	4.8	**22**	**9201**	24

Table B.3: Results for the approaches in all instances of S1 with random topology.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S2	100	3	480	5	5	26	38499	28	41509	~ 0	28	46245	20	26	38982	9.6	26	39211	26.9	26	38745	23.9	**26**	**38499**	43.2
				6	6	22	38355	23	40760	~ 0	24	45945	20	23	39370	6.4	**22**	**38355**	26.1	22	38562	26.6	22	38591	45.1
				7	7	19	38197	20	40710	~ 0	21	45783	20	19	38998	8.6	19	38570	26.7	**19**	**38197**	28	19	38320	47.5
				5	10	20	41823	20	49733	~ 0	20	49107	20	20	43336	8.3	**20**	**41823**	25.5	20	42768	26.7	20	42804	45.5
				6	12	17	42224	17	46490	~ 0	17	49638	20	17	43161	8.2	**17**	**42224**	25.9	17	42603	29.3	17	42385	48.5
				7	14	15	42138	15	45836	~ 0	15	49984	20	15	43355	7.6	**15**	**42138**	27	15	42278	32	15	42365	50.6
	200	3	480	5	5	51	77528	55	81312	~ 0	58	91972	20	54	80340	66.3	52	76275	75.7	52	77460	109.7	**51**	**77528**	137.3
				6	6	43	76790	46	80847	~ 0	48	92282	20	46	79792	71.4	44	76925	76.9	43	76974	110.8	**43**	**76790**	140
				7	7	38	76867	40	80307	~ 0	41	92591	20	42	88356	8.7	**38**	**76867**	79.3	38	77507	127.9	38	77386	149.1
				5	10	40	83771	41	90125	~ 0	40	99242	20	40	88462	64.7	**40**	**83771**	68.5	40	84962	120	40	85057	142
				6	12	34	83810	34	96155	~ 0	34	100143	20	34	87909	72.3	**34**	**83810**	76.1	34	85175	127.2	34	85154	154
				7	14	29	83245	29	100697	~ 0	29	100300	20	29	87513	69.3	**29**	**83245**	74.7	29	84644	139.4	29	84660	157.4
	300	3	480	5	5	82	127088	86	132346	~ 0	93	154683	20	86	131476	265.4	83	125995	239	83	127827	294.9	**82**	**127088**	317.4
				6	6	69	127500	73	130937	~ 0	78	154745	20	73	131724	276.5	70	125466	233.1	70	126959	308.2	**69**	**127500**	337.9
				7	7	60	125451	63	131168	1	67	154840	20	62	132705	244.9	**60**	**125451**	294.7	60	127205	322.8	60	127040	339.7
				5	10	60	135952	62	162920	~ 0	60	164718	20	61	146747	267.2	**60**	**135952**	85.6	60	139257	325.3	60	139922	335.1
				6	12	50	136979	52	156281	~ 0	50	165073	20	51	145281	255.5	**50**	**136979**	66.3	50	138924	338.4	50	138856	365.6
				7	14	43	135709	44	165911	~ 0	43	165292	20	44	144520	289.3	**43**	**135709**	194.8	43	137848	342.1	43	137794	374.3
	400	3	720	5	5	101	262066	107	274106	~ 0	117	315511	20	104	269731	737.4	103	261004	630.2	**101**	**262066**	582.9	101	262706	646.1
				6	6	84	262894	89	272807	~ 0	97	315158	20	88	271410	619	85	260457	638.7	**84**	**262894**	591.6	85	263469	708.8
				7	7	73	258502	76	270490	~ 0	83	316343	20	76	270164	678.3	**73**	**258502**	849.4	73	262979	646	73	263741	662
				5	10	80	286304	81	308927	~ 0	80	335228	20	80	299399	638.5	**80**	**286304**	594	80	292663	622.3	80	293135	660.3
				6	12	67	285232	67	316696	~ 0	67	337172	20	67	297713	779.9	**67**	**285232**	580.1	67	291390	655.3	67	291045	685.6
				7	14	58	284142	58	307483	~ 0	58	338746	20	58	296861	755	**58**	**284142**	598.6	58	289816	675.3	58	290562	707.5
	500	3	720	5	5	120	307590	129	321155	~ 0	143	376705	20	125	313983	2174.6	122	300937	1619.9	**120**	**307590**	1040.7	121	307403	1188.9
				6	6	101	300270	107	318968	~ 0	118	378560	20	105	315684	1811.8	**101**	**300270**	1752.6	101	307733	1053.7	101	306820	1178.8
				7	7	87	305598	92	315438	~ 0	101	376844	20	91	316458	1621.6	88	301550	1767.3	**87**	**305598**	1059.9	87	305659	1173.7
				5	10	100	337103	100	398658	~ 0	100	404412	20	100	355875	1289.2	**100**	**337103**	1263	100	347056	1230.5	100	346855	1206.7
				6	12	84	335356	84	364345	~ 0	84	406361	20	84	354263	1573.8	**84**	**335356**	1519	84	344606	1366.9	84	344308	1262.1
				7	14	72	334462	72	359234	~ 0	72	408755	20	72	351626	1705.9	**72**	**334462**	1458.5	72	342475	1264.2	72	343470	1280.7

Table B.4: Results for the approaches in all instances of S2 with random topology.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S1	10	2	40	2	2	18	1031	20	1114	~ 0	**18**	**1031**	20	18	1040	~ 0	**18**	**1031**	20	18	1040	0.1	**18**	**1031**	20
				3	3	12	1012	12	1044	~ 0	**12**	**1012**	20	**12**	**1012**	~ 0	**12**	**1012**	20	**12**	**1012**	0.2	**12**	**1012**	20
				4	4	9	1012	9	1050	~ 0	**9**	**1012**	20	**9**	**1012**	~ 0	**9**	**1012**	20	**9**	**1012**	0.2	**9**	**1012**	20
				2	4	10	1078	12	1174	~ 0	**10**	**1078**	20	11	1114	~ 0	**10**	**1078**	20	10	1112	0.2	**10**	**1078**	20
				3	6	6	1040	7	1116	~ 0	**6**	**1040**	20	**6**	**1040**	~ 0	**6**	**1040**	20	**6**	**1040**	0.5	**6**	**1040**	20
				4	8	5	1040	5	1097	~ 0	**5**	**1040**	20	5	1068	~ 0	**5**	**1040**	20	5	1068	0.3	**5**	**1040**	20
	20	2	40	2	2	40	2350	43	2470	~ 0	42	2434	20	41	2372	0.1	41	2372	20	**40**	**2350**	0.7	**40**	**2350**	20
				3	3	27	2350	28	2426	~ 0	28	2418	20	27	2368	0.1	**27**	**2350**	20	**27**	**2350**	0.7	**27**	**2350**	20
				4	4	20	2350	21	2442	~ 0	21	2412	20	21	2372	0.1	21	2372	20	**20**	**2350**	0.9	**20**	**2350**	20
				2	4	21	2416	24	2580	~ 0	22	2496	20	22	2456	0.1	22	2438	20	**21**	**2416**	0.7	21	2438	20
				3	6	14	2394	15	2474	~ 0	15	2482	20	**14**	**2394**	0.1	14	2416	20	**14**	**2394**	0.7	**14**	**2394**	20
				4	8	10	2350	11	2506	~ 0	11	2474	20	11	2394	~ 0	11	2390	20	**10**	**2350**	1	11	2372	20
	30	2	55	2	2	36	3235	43	3634	~ 0	41	3569	20	45	3819	~ 0	37	3284	20	**36**	**3235**	1.1	37	3252	21
				3	3	24	3220	28	3507	~ 0	27	3531	20	**24**	**3220**	0.2	25	3302	20	24	3227	1.4	24	3230	21
				4	4	19	3247	21	3555	~ 0	20	3448	20	19	3252	0.2	20	3448	20	**19**	**3247**	1.8	19	3270	21
				2	4	21	3481	25	3945	~ 0	23	3717	20	22	3545	0.3	23	3717	20	21	3532	1.3	**21**	**3481**	21
				3	6	13	3356	15	3740	~ 0	15	3701	20	14	3450	0.1	15	3701	20	13	3361	1.3	**13**	**3356**	21
				4	8	10	3416	12	3745	~ 0	11	3706	20	11	3420	0.2	11	3706	20	**10**	**3416**	1.8	10	3420	21
	40	2	60	2	2	51	4987	60	5656	~ 0	59	5696	20	52	5034	0.4	52	5019	20	**51**	**4987**	2.1	51	5012	21.8
				3	3	34	4940	39	5606	~ 0	38	5555	20	34	4969	0.4	34	4970	20	**34**	**4940**	2.5	34	4943	22
				4	4	25	4922	30	5498	~ 0	29	5531	20	27	5061	0.3	26	4983	20	**25**	**4922**	2.9	25	4937	22.1
				2	4	30	5556	35	6288	~ 0	32	5998	20	31	5508	0.2	32	5998	20	31	5498	2.1	**30**	**5556**	22
				3	6	19	5177	23	6344	~ 0	21	5846	20	19	5178	0.5	20	5322	20	**19**	**5177**	2.4	19	5178	22
				4	8	14	5193	17	5900	~ 0	16	5866	20	20	7078	0.3	15	5303	20	14	5276	3.2	**14**	**5193**	22.9
	50	2	65	2	2	64	6745	75	7626	~ 0	71	7411	20	65	6808	0.4	66	6856	20	64	6764	3.4	**64**	**6745**	23
				3	3	43	6711	49	7511	~ 0	46	7283	20	**43**	**6711**	0.5	46	7283	20	43	6766	3.9	43	6735	23.1
				4	4	33	6793	36	7421	~ 0	34	7194	20	**33**	**6793**	0.9	34	7194	20	33	6811	4.5	33	6828	24
				2	4	35	7152	44	8011	~ 0	39	7714	20	**35**	**7152**	0.9	39	7714	20	36	7096	3.4	36	7115	23
				3	6	24	7006	27	7950	~ 0	26	7618	20	24	7124	0.6	26	7618	20	**24**	**7006**	3.7	24	7025	23.1
				4	8	18	6984	20	7750	~ 0	19	7582	20	18	7031	0.8	19	7582	20	18	7014	4.9	**18**	**6984**	24

Table B.5: Results for the approaches in all instances of S1 with mix topology.

	Instance					Best sol.		GCH			RCH			LNS-GG			LNS-RG			LNS-SA-GG			LNS-SA-RG		
	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
S2	100	3	480	5	5	24	35056	26	37984	~ 0	27	43353	20	25	35510	9.4	25	34862	26.9	24	35271	23.5	**24**	**35056**	43.6
				6	6	21	34917	22	38360	~ 0	23	43042	20	21	36032	8.3	**21**	**34917**	27.2	21	35212	25.5	21	35313	45.4
				7	7	18	34593	19	38321	~ 0	20	43262	20	19	36104	9.8	**18**	**34593**	28.6	18	35216	27.8	18	35138	47.7
				5	10	20	39154	20	45546	~ 0	20	46340	20	20	39444	6.6	**20**	**39154**	25.9	20	39534	26.3	20	39586	43.9
				6	12	17	39202	17	44133	~ 0	17	46880	20	17	39544	10.1	**17**	**39202**	27	17	39484	28.5	17	39509	48.1
				7	14	15	39129	15	44128	~ 0	15	47386	20	15	40235	8.1	15	39179	25.9	15	39397	31.3	**15**	**39129**	52.4
	200	3	480	5	5	56	87461	60	92546	~ 0	63	104197	20	58	89942	94.4	57	87113	103	**56**	**87461**	121.6	57	87600	138
				6	6	48	86370	50	92437	~ 0	52	105701	20	50	90812	94.1	**48**	**86370**	88.8	48	87312	124.2	48	87585	143.3
				7	7	42	86715	44	92242	~ 0	45	105697	20	44	90946	89.5	**42**	**86715**	93.3	42	87318	126.1	42	86927	146
				5	10	40	94073	42	104632	~ 0	40	111097	20	41	95115	97.6	**40**	**94073**	38.7	40	95194	119.5	40	95086	142.7
				6	12	34	93129	34	108959	~ 0	34	111302	20	34	94599	122	**34**	**93129**	65.6	34	95336	128.7	34	95120	154.6
				7	14	29	93804	30	105999	~ 0	29	111758	20	29	94616	111.7	**29**	**93804**	52.1	29	94654	141.8	29	94945	159.3
	300	3	480	5	5	86	134349	92	142090	~ 0	98	162753	20	96	150698	26.5	87	133370	343.6	**86**	**134349**	318.1	87	134759	332.8
				6	6	74	133212	78	143371	~ 0	81	163436	20	82	152982	27	**74**	**133212**	359.9	74	135549	316.9	74	135973	334.6
				7	7	64	134605	67	142703	~ 0	70	163958	20	73	155756	27.1	65	134153	296.6	65	136548	332.8	**64**	**134605**	339.2
				5	10	60	146215	64	165308	~ 0	61	173837	20	62	153775	390.1	**60**	**146215**	64.1	60	149707	322	60	149631	346.6
				6	12	50	149025	54	164981	~ 0	51	174774	20	52	151336	408.1	51	174774	48	50	149708	343.1	**50**	**149025**	334.3
				7	14	43	148690	46	166672	~ 0	43	176498	20	45	151270	404.3	43	176498	69.8	**43**	**148690**	346.1	43	148765	366.7
	400	3	720	5	5	96	237592	106	256103	~ 0	116	309796	20	100	241415	995.2	98	233048	913.7	**96**	**237592**	653.8	96	238515	663.5
				6	6	80	238851	88	254881	~ 0	97	307300	20	85	241846	1260	82	234334	880.8	81	237420	655.3	**80**	**238851**	668.1
				7	7	69	235838	76	252229	1	83	307811	20	74	241142	390.6	70	234814	1028.7	70	237939	640.6	**69**	**235838**	687
				5	10	80	267413	81	306218	~ 0	80	336384	20	81	280748	948.4	**80**	**267413**	715.7	80	279297	603.7	80	278195	679.1
				6	12	67	267634	68	303079	~ 0	67	337957	20	68	279778	774.7	**67**	**267634**	702.6	67	277106	641.7	67	275909	674.6
				7	14	58	268467	58	314553	~ 0	58	339263	20	58	275059	885.8	**58**	**268467**	758.4	58	275829	700.2	58	276720	706.4
	500	3	720	5	5	115	283163	127	304990	~ 0	139	359017	20	121	288282	3577.3	117	278882	2152.4	**115**	**283163**	1163.3	115	285144	1117.1
				6	6	97	283389	105	303525	~ 0	115	361519	20	101	290897	2406.5	99	277250	2114.4	97	288727	1045.9	**97**	**283389**	1221.1
				7	7	83	282153	91	301213	~ 0	99	362032	20	89	293368	2383.1	84	277863	2026.5	83	282890	1187.7	**83**	**282153**	1213.2
				5	10	100	310085	100	362887	~ 0	100	387528	20	100	332664	1591	**100**	**310085**	1540.4	100	321947	1252.8	100	322012	1218.3
				6	12	84	308406	84	355406	~ 0	84	391407	20	84	332561	1791.3	**84**	**308406**	1460.6	84	318101	1216.7	84	320537	1247.6
				7	14	72	306848	72	346498	~ 0	72	393298	20	72	330962	1698.5	**72**	**306848**	1551.5	72	317384	1174	72	317254	1256.7

Table B.6: Results for the approaches in all instances of S2 with mix topology.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1129	20	1134	20	1131.8	20	100	3	480	5	5	26	40487	26	41076	26	40695.3	20
			3	3	13	1125	13	1128	13	1126.4	20				6	6	22	40265	22	40930	22	40674.3	20
			4	4	10	1123	10	1127	10	1125.9	20				7	7	19	40718	19	41169	19	40976	20
			2	4	10	1143	10	1148	10	1146.1	20				5	10	20	43799	20	44411	20	44175.4	20
			3	6	7	1137	7	1141	7	1139.1	20				6	12	17	44489	17	44850	17	44683.4	20
			4	8	5	1133	5	1139	5	1135.7	20				7	14	15	44983	15	45241	15	45124.4	20
20	2	40	2	2	38	2187	38	2193	38	2190	20	200	3	480	5	5	54	84333	55	84547	54.6	84302.7	20
			3	3	25	2148	25	2171	25	2159.9	20				6	6	45	84293	46	85407	45.8	84719	20
			4	4	19	2130	19	2158	19	2143.9	20				7	7	39	84444	39	86100	39	85250.2	20
			2	4	20	2241	21	2245	20.2	2252.1	20				5	10	40	92240	40	92902	40	92689.1	20
			3	6	13	2210	14	2235	13.3	2225.7	20				6	12	34	93368	34	94201	34	93868.8	20
			4	8	10	2185	10	2227	10	2209.7	20				7	14	29	93872	29	94429	29	94187.4	20
30	2	55	2	2	56	4777	56	4812	56	4796.6	20	300	3	480	5	5	91	148374	92	149036	91.8	148240.9	20
			3	3	37	4768	37	4789	37	4776.8	20				6	6	76	148463	77	149003	76.4	149066.6	20
			4	4	28	4747	28	4783	28	4764.6	20				7	7	65	149470	66	149897	65.2	149880.2	20
			2	4	29	4857	29	4903	29	4879.6	20				5	10	60	159086	60	160182	60	159568.7	20
			3	6	19	4837	19	4888	19	4865.2	20				6	12	50	159538	50	160746	50	160173.1	20
			4	8	15	4831	15	4869	15	4849.7	20				7	14	43	159378	43	161058	43	160589.1	20
40	2	60	2	2	82	7275	83	7313	82.5	7293.7	20	400	3	720	5	5	117	314178	118	317128	117.9	315492.3	20
			3	3	54	7146	54	7205	54	7175.1	20				6	6	98	312422	98	318392	98	316174.6	20
			4	4	40	7097	40	7152	40	7122.3	20				7	7	83	319352	84	319043	83.9	317603.2	20
			2	4	43	7392	43	7497	43	7451.9	20				5	10	80	340643	80	342552	80	341692.5	20
			3	6	28	7328	28	7391	28	7363.6	20				6	12	67	342250	67	344341	67	343249.2	20
			4	8	21	7255	21	7341	21	7310.2	20				7	14	58	343770	58	345643	58	344936.3	20
50	2	65	2	2	108	11561	109	11610	108.7	11595	20	500	3	720	5	5	138	364775	140	363342	139.1	364092.7	20
			3	3	71	11491	72	11525	71.8	11510.6	20				6	6	115	364095	116	365572	115.4	366215.6	20
			4	4	53	11398	53	11467	53	11451.1	20				7	7	98	365888	99	367174	98.7	366090.7	20
			2	4	56	11723	57	11730	56.6	11720.7	20				5	10	100	393447	100	394333	100	393887.6	20
			3	6	37	11626	37	11714	37	11665.2	20				6	12	84	395041	84	396666	84	396009.5	20
			4	8	28	11585	28	11657	28	11628.2	20				7	14	72	395647	72	397919	72	397174.9	20

Table B.7: Results for the RCH in all instances with cluster topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1299	20	1299	20	1299	20	100	3	480	5	5	28	46245	29	46065	28.9	45975.6	20
			3	3	13	1291	13	1291	13	1291	20				6	6	24	45945	24	46607	24	46195	20
			4	4	10	1291	10	1291	10	1291	20				7	7	21	45783	21	46641	21	46377.9	20
			2	4	11	1327	11	1327	11	1327	20				5	10	20	49107	20	49465	20	49270.2	20
			3	6	7	1299	7	1307	7	1300.6	20				6	12	17	49638	17	50006	17	49833.7	20
			4	8	5	1291	5	1299	5	1298.2	20				7	14	15	49984	15	50366	15	50248.3	20
20	2	40	2	2	40	2469	40	2492	40	2486	20	200	3	480	5	5	58	91972	58	92861	58	92553.9	20
			3	3	26	2430	27	2466	26.2	2452	20				6	6	48	92282	48	93234	48	92863.9	20
			4	4	20	2421	20	2443	20	2431.5	20				7	7	41	92591	41	93921	41	93341.6	20
			2	4	21	2581	22	2615	21.9	2601	20				5	10	40	99242	40	99805	40	99584.5	20
			3	6	14	2557	14	2571	14	2566.9	20				6	12	34	100143	34	100536	34	100313.5	20
			4	8	11	2537	11	2561	11	2549	20				7	14	29	100300	29	100801	29	100607.2	20
30	2	55	2	2	47	3991	48	4091	47.7	4055.1	20	300	3	480	5	5	93	154683	94	154978	93.9	154511.7	20
			3	3	31	3974	31	4023	31	3996.1	20				6	6	78	154745	78	156062	78	155417.1	20
			4	4	23	3935	23	3975	23	3954.2	20				7	7	67	154840	67	155975	67	155630.9	20
			2	4	26	4113	26	4270	26	4221.7	20				5	10	60	164718	60	166492	60	165704.7	20
			3	6	17	4109	17	4206	17	4177.8	20				6	12	50	165073	50	166609	50	165838.8	20
			4	8	13	4142	13	4184	13	4165.9	20				7	14	43	165292	43	166447	43	166165.6	20
40	2	60	2	2	88	8464	89	8493	88.4	8479.9	20	400	3	720	5	5	117	315511	118	315835	117.8	314641.1	20
			3	3	57	8375	58	8443	57.9	8402.8	20				6	6	97	315158	98	316564	97.9	315557.1	20
			4	4	43	8317	43	8435	43	8364.9	20				7	7	83	316343	84	316484	83.5	317075.4	20
			2	4	46	8580	47	8602	46.2	8647.7	20				5	10	80	335228	80	337066	80	336178.2	20
			3	6	30	8533	30	8638	30	8590.6	20				6	12	67	337172	67	338542	67	337759.3	20
			4	8	22	8493	23	8549	22.8	8510.8	20				7	14	58	338746	58	339507	58	339124.6	20
50	2	65	2	2	94	9739	96	9842	95.4	9812.1	20	500	3	720	5	5	143	376705	144	377490	143.6	376845.4	20
			3	3	62	9606	63	9710	62.5	9669.8	20				6	6	118	378560	120	379339	119	379568.5	20
			4	4	46	9559	47	9630	46.3	9597	20				7	7	101	376844	102	380842	101.8	379717.6	20
			2	4	50	9969	51	10109	50.7	10074.3	20				5	10	100	404412	100	406097	100	405536.9	20
			3	6	33	9957	33	10039	33	9996.9	20				6	12	84	406361	84	408656	84	407704.9	20
			4	8	25	9852	25	9940	25	9902.1	20				7	14	72	408755	72	410352	72	409601.9	20

Table B.8: Results for the RCH in all instances with random topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	18	1031	18	1044	18	1040.3	20	100	3	480	5	5	27	43353	28	43331	27.9	43021.9	20
			3	3	12	1012	12	1026	12	1013.4	20				6	6	23	43042	23	43672	23	43417.7	20
			4	4	9	1012	9	1012	9	1012	20				7	7	20	43262	20	44042	20	43589.9	20
			2	4	10	1078	10	1092	10	1087	20				5	10	20	46340	20	47059	20	46755.8	20
			3	6	6	1040	6	1040	6	1040	20				6	12	17	46880	17	47531	17	47302.3	20
			4	8	5	1040	5	1040	5	1040	20				7	14	15	47386	15	48023	15	47717.7	20
20	2	40	2	2	42	2434	43	2456	42.4	2444.4	20	200	3	480	5	5	63	104197	64	105354	63.7	104934.4	20
			3	3	28	2418	28	2434	28	2431.6	20				6	6	52	105701	53	105876	52.9	105275.2	20
			4	4	21	2412	21	2434	21	2417.6	20				7	7	45	105697	46	105898	45.3	105913.6	20
			2	4	22	2496	22	2522	22	2512.4	20				5	10	40	111097	40	111698	40	111476.6	20
			3	6	15	2482	15	2514	15	2499	20				6	12	34	111302	34	111905	34	111665.2	20
			4	8	11	2474	11	2496	11	2484.4	20				7	14	29	111758	29	112269	29	111986.6	20
30	2	55	2	2	41	3569	42	3605	41.9	3592.4	20	300	3	480	5	5	98	162753	98	164286	98	163514	20
			3	3	27	3531	28	3565	27.3	3550	20				6	6	81	163436	82	164183	81.4	164105.5	20
			4	4	20	3448	20	3525	20	3490.9	20				7	7	70	163958	70	164616	70	164366.3	20
			2	4	23	3717	23	3781	23	3752.5	20				5	10	61	173837	61	175627	61	175057.6	20
			3	6	15	3701	15	3751	15	3733.4	20				6	12	51	174774	51	175458	51	175161.3	20
			4	8	11	3706	12	3753	11.3	3725	20				7	14	43	176498	44	175540	43.3	176614.8	20
40	2	60	2	2	59	5696	60	5749	59.7	5714.8	20	400	3	720	5	5	116	309796	117	310196	116.9	308591.8	20
			3	3	38	5555	39	5658	38.8	5602.8	20				6	6	97	307300	98	310926	97.6	309594.3	20
			4	4	29	5531	29	5597	29	5575	20				7	7	83	307811	84	310603	83.3	311159.6	20
			2	4	32	5998	33	6047	32.8	6001	20				5	10	80	336384	80	338855	80	337920.1	20
			3	6	21	5846	22	5951	21.7	5893.1	20				6	12	67	337957	67	340329	67	338990.8	20
			4	8	16	5866	16	5934	16	5902.2	20				7	14	58	339263	58	341889	58	340611.3	20
50	2	65	2	2	71	7411	72	7447	71.2	7481.3	20	500	3	720	5	5	139	359017	140	361690	139.6	360635.7	20
			3	3	46	7283	47	7353	46.1	7359.5	20				6	6	115	361519	116	367015	115.7	363681	20
			4	4	34	7194	34	7370	34	7283.3	20				7	7	99	362032	99	366978	99	364861.8	20
			2	4	39	7714	40	7735	39.6	7730.9	20				5	10	100	387528	100	390550	100	389604.4	20
			3	6	26	7618	26	7700	26	7661.9	20				6	12	84	391407	84	393235	84	392438	20
			4	8	19	7582	19	7693	19	7640.9	20				7	14	72	393298	72	395544	72	394522.7	20

Table B.9: Results for the RCH in all instances with mix topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1140	20	1140	20	1140	~ 0	100	3	480	5	5	24	34943	25	36333	24.6	35526.3	9
			3	3	13	1122	14	1132	13.1	1125.9	~ 0				6	6	21	35002	21	35996	21	35625.5	7.50
			4	4	10	1130	11	1150	10.5	1139	~ 0				7	7	18	34894	19	35350	18.1	35369.8	10.30
			2	4	11	1150	11	1174	11	1165	~ 0				5	10	20	39526	20	40581	20	39908.1	6.30
			3	6	7	1142	7	1158	7	1147.4	~ 0				6	12	17	38816	17	39895	17	39552.1	8.70
			4	8	6	1142	6	1167	6	1152.4	~ 0				7	14	15	38659	15	40714	15	39665.2	8.30
20	2	40	2	2	36	2102	37	2137	36.2	2114.3	~ 0	200	3	480	5	5	49	68731	50	70166	49.1	69107.2	72.20
			3	3	24	2102	25	2171	24.4	2122.9	~ 0				6	6	41	68562	42	70596	41.5	69390.9	63.80
			4	4	18	2102	19	2144	18.5	2122	0.10				7	7	35	67981	37	70595	36.4	69388	80
			2	4	19	2180	20	2270	19.9	2215.8	~ 0				5	10	40	79141	40	79991	40	79651.2	71.20
			3	6	13	2130	13	2194	13	2165	~ 0				6	12	34	79465	34	80668	34	80120.2	64.90
			4	8	10	2130	10	2190	10	2161.2	0.10				7	14	29	79016	29	80746	29	79734.2	64.20
30	2	55	2	2	55	4707	56	4751	55.2	4727.4	0.20	300	3	480	5	5	82	124016	84	127384	83.3	125793.3	361.40
			3	3	37	4690	37	4728	37	4710.9	0.30				6	6	70	125207	71	127620	70.7	126554.4	377.80
			4	4	28	4712	29	4773	28.1	4743.9	0.10				7	7	61	126199	63	129050	61.8	127362.4	315.40
			2	4	29	4823	30	4896	29.2	4854.8	~ 0				5	10	60	137931	61	139394	60.7	138739.5	281.50
			3	6	19	4747	19	4780	19	4768.3	0.20				6	12	51	137552	51	140344	51	139209.4	331.30
			4	8	14	4745	15	4881	14.9	4807.6	0.10				7	14	43	137654	44	140355	43.9	138802	320.40
40	2	60	2	2	75	6801	78	6963	75.8	6852.4	0.20	400	3	720	5	5	101	255646	104	261999	102	258382.8	1265.10
			3	3	50	6801	52	6913	50.6	6849.4	0.30				6	6	85	257986	88	265004	86.2	260390.7	1130.10
			4	4	38	6801	39	6893	38.1	6812.4	0.30				7	7	74	259760	76	264623	75	262599.1	865.30
			2	4	41	6978	43	7270	41.4	7097.6	0.40				5	10	80	293969	81	297629	80.8	296054.2	733.70
			3	6	26	6867	27	7175	26.3	6973.3	0.20				6	12	67	291604	67	296122	67	294440.3	683.70
			4	8	19	6867	20	7017	19.9	6946	0.40				7	14	58	290643	58	297085	58	294902.2	847.40
50	2	65	2	2	100	11074	102	11260	101.1	11160.1	0.60	500	3	720	5	5	120	299883	123	304985	121.9	303864	2809.80
			3	3	67	11083	69	11224	67.6	11139.8	0.40				6	6	101	302165	103	308106	102.4	305678.8	2842.70
			4	4	51	11111	52	11371	51.6	11226.7	0.60				7	7	87	301205	90	309154	89.1	307343.5	2046.80
			2	4	53	11220	56	11426	54.5	11350.3	0.70				5	10	100	341699	101	348657	100.7	346055.3	1887.90
			3	6	34	11196	36	11481	35.1	11286.9	0.50				6	12	84	344674	84	347965	84	346305.8	1649.10
			4	8	26	11167	28	11617	26.5	11310.8	1				7	14	72	342505	72	347351	72	345925.9	1584.40

Table B.10: Results for the LNS-GG in all instances with cluster topology after 10 runs.

S1												S2										
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1299	21	1327	20.8	1318.2	~ 0	100	3	480	5	5	26	38982	28	40425	27	40086.3	9.60
			3	3	14	1299	14	1319	14	1303.4	0.10				6	6	23	39370	24	40740	23.1	39993.3	6.40
			4	4	10	1299	11	1347	10.9	1322.2	0.10				7	7	19	38998	20	40986	19.7	39855.5	8.60
			2	4	11	1339	12	1363	11.5	1352.6	~ 0				5	10	20	43336	21	44475	20.1	43816.7	8.30
			3	6	7	1351	8	1363	7.8	1340.6	~ 0				6	12	17	43161	17	44375	17	43849.6	8.20
			4	8	6	1315	6	1339	6	1325.4	~ 0				7	14	15	43355	15	44298	15	43763	7.60
20	2	40	2	2	37	2353	38	2399	37.3	2364.7	~ 0	200	3	480	5	5	54	80340	55	81932	54.6	80989.9	66.30
			3	3	25	2335	26	2401	25.5	2375	~ 0				6	6	46	79792	47	81657	46.2	80964.7	71.40
			4	4	19	2371	20	2429	19.2	2390.6	0.10				7	7	42	88356	42	88356	42	88356	8.70
			2	4	21	2532	22	2581	21.3	2553.8	0.10				5	10	40	88462	41	90998	40.9	89827.2	64.70
			3	6	14	2451	14	2527	14	2495.2	~ 0				6	12	34	87909	35	90707	34.2	89141.3	72.30
			4	8	10	2409	11	2533	10.4	2474.4	0.20				7	14	29	87513	30	90211	29.4	88891.8	69.30
30	2	55	2	2	42	3688	48	3978	44	3790	0.10	300	3	480	5	5	86	131476	87	134460	86.4	133004.5	265.40
			3	3	28	3662	29	3796	28.6	3733.8	0.10				6	6	73	131724	74	132683	73.1	133063.2	276.50
			4	4	21	3682	23	3906	22	3767	0.10				7	7	62	132705	64	134477	63.2	133327	244.90
			2	4	25	3985	28	4125	25.7	4068.4	0.20				5	10	61	146747	62	148151	61.9	147483.5	267.20
			3	6	16	3844	17	4074	16.2	3942.1	0.10				6	12	51	145281	52	147666	51.7	146673.2	255.50
			4	8	12	3812	13	4119	12.3	3936.5	0.30				7	14	44	144520	45	148566	44.2	146301.9	289.30
40	2	60	2	2	79	7981	82	8143	80.2	8049.9	0.30	400	3	720	5	5	104	269731	109	275147	106.6	272220	737.40
			3	3	54	8019	55	8140	54.6	8080.8	0.40				6	6	88	271410	91	276267	89.2	273112.6	619
			4	4	40	8005	42	8115	40.8	8053.4	0.30				7	7	76	270164	78	274125	77	272896.4	678.30
			2	4	43	8263	45	8482	44.4	8394.6	~ 0				5	10	80	299399	80	302838	80	300449	638.50
			3	6	28	8093	30	8426	28.7	8246.6	0.30				6	12	67	297713	67	299894	67	298814.7	779.90
			4	8	21	8115	23	8453	21.7	8252.6	0.40				7	14	58	296861	58	302101	58	299225.4	755
50	2	65	2	2	86	9070	89	9364	87	9195.1	0.50	500	3	720	5	5	125	313983	127	316707	125.7	315857.9	2174.60
			3	3	56	9002	59	9274	57.5	9114.5	0.70				6	6	105	315684	107	319012	105.9	317412.6	1811.80
			4	4	43	9038	46	9413	44.1	9180.8	0.60				7	7	91	316458	94	321744	92.4	319153.1	1621.60
			2	4	46	9357	48	9609	47.1	9500.1	0.40				5	10	100	355875	101	357414	100.2	356770.6	1289.20
			3	6	31	9372	32	9674	31.4	9507.7	0.50				6	12	84	354263	84	356939	84	355261	1573.80
			4	8	23	9283	24	9549	23.3	9417.8	0.60				7	14	72	351626	72	355286	72	353987.3	1705.90

Table B.11: Results for the LNS-GG in all instances with random topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	18	1040	19	1077	18.4	1053.3	~ 0	100	3	480	5	5	25	35510	27	37406	25.6	36368.9	9.40
			3	3	12	1012	12	1044	12	1025.4	~ 0				6	6	21	36032	22	37434	21.9	36643.8	8.30
			4	4	9	1012	9	1040	9	1020.4	~ 0				7	7	19	36104	21	37790	19.6	36970.8	9.80
			2	4	11	1114	11	1142	11	1130.8	~ 0				5	10	20	39444	21	40968	20.1	40216.7	6.60
			3	6	6	1040	7	1086	6.2	1049.2	~ 0				6	12	17	39544	17	41098	17	40399.3	10.10
			4	8	5	1068	5	1096	5	1075	~ 0				7	14	15	40235	15	41680	15	40795	8.10
20	2	40	2	2	41	2372	41	2394	41	2378.2	0.10	200	3	480	5	5	58	89942	60	92603	59	90661	94.40
			3	3	27	2368	28	2438	27.5	2393.2	0.10				6	6	50	90812	51	92151	50.1	91347.7	94.10
			4	4	21	2372	22	2444	21.1	2405.2	0.10				7	7	44	90946	46	92659	44.8	92071.9	89.50
			2	4	22	2456	23	2500	22.3	2475	0.10				5	10	41	95115	42	97733	41.3	96275.6	97.60
			3	6	14	2394	16	2616	14.7	2474.6	0.10				6	12	34	94599	35	96851	34.5	95675.8	122
			4	8	11	2394	11	2510	11	2452.4	~ 0				7	14	29	94616	30	97312	29.7	96091.1	111.70
30	2	55	2	2	45	3819	45	3819	45	3819	~ 0	300	3	480	5	5	96	150698	96	150698	96	150698	26.50
			3	3	24	3220	26	3375	25.3	3295.1	0.20				6	6	82	152982	82	152982	82	152982	27
			4	4	19	3252	22	3536	20.1	3393.6	0.20				7	7	73	155756	73	155756	73	155756	27.10
			2	4	22	3545	24	3736	23	3654.6	0.30				5	10	62	153775	64	156038	62.9	154255	390.10
			3	6	14	3450	15	3614	14.4	3510.8	0.10				6	12	52	151336	53	155188	52.6	153982.9	408.10
			4	8	11	3420	12	3784	11.2	3550	0.20				7	14	45	151270	47	156036	45.6	153770.8	404.30
40	2	60	2	2	52	5034	54	5161	52.4	5086.5	0.40	400	3	720	5	5	100	241415	103	247714	102	244720.9	995.20
			3	3	34	4969	36	5141	35	5038.9	0.40				6	6	85	241846	90	254906	85.9	245474.9	1260
			4	4	27	5061	29	5293	27.6	5148.2	0.30				7	7	74	241142	80	281708	78.5	270451.4	390.60
			2	4	31	5508	33	5631	31.9	5627.2	0.20				5	10	81	280748	82	285541	81.1	282953	948.40
			3	6	19	5178	20	5425	19.7	5337.1	0.50				6	12	68	279778	69	284840	68.1	282558.9	774.70
			4	8	20	7078	20	7078	20	7078	0.30				7	14	58	275059	59	285513	58.6	281226	885.80
50	2	65	2	2	65	6808	67	6977	66.1	6861.8	0.40	500	3	720	5	5	121	288282	124	296057	122.4	293097.7	3577.30
			3	3	43	6711	45	6909	43.8	6807.5	0.50				6	6	101	290897	105	301687	103.4	296600.2	2406.50
			4	4	33	6793	35	6980	33.7	6870.4	0.90				7	7	89	293368	92	302020	90.7	298852.2	2383.10
			2	4	35	7152	38	7291	36.8	7191	0.90				5	10	100	332664	101	338612	100.5	335854.9	1591
			3	6	24	7124	27	7458	25.4	7297.5	0.60				6	12	84	332561	84	336827	84	335454.1	1791.30
			4	8	18	7031	21	7519	19.6	7294.6	0.80				7	14	72	330962	72	337722	72	334393	1698.50

Table B.12: Results for the LNS-GG in all instances with mix topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1128	20	1134	20	1131.4	20	100	3	480	5	5	24	34263	25	34896	24.5	34665.8	27.50
			3	3	13	1125	13	1128	13	1126.3	20				6	6	20	34876	21	35129	20.9	34903.7	25.80
			4	4	10	1123	10	1127	10	1125.9	20				7	7	18	34474	19	35545	18.1	34779.7	28.90
			2	4	10	1143	10	1148	10	1146.1	20				5	10	20	38376	20	38862	20	38604.7	25.70
			3	6	7	1137	7	1141	7	1139.1	20				6	12	17	38284	17	39031	17	38631.6	27.80
			4	8	5	1133	5	1139	5	1135.7	20				7	14	15	38358	15	39359	15	38731.9	27
20	2	40	2	2	36	2102	38	2192	37.1	2150.8	20	200	3	480	5	5	48	66651	50	69327	48.5	67465	92.10
			3	3	24	2102	25	2169	24.7	2141.9	20				6	6	40	66265	42	68706	41	67443.2	92.80
			4	4	18	2102	19	2158	18.4	2124.4	20				7	7	35	67079	36	69010	35.8	67831.8	106.40
			2	4	20	2204	21	2245	20.2	2242.4	20				5	10	40	77081	40	78361	40	77720.6	71
			3	6	13	2144	14	2233	13.1	2181	20				6	12	34	76560	34	78945	34	77730.5	79.30
			4	8	10	2144	10	2221	10	2180.4	20				7	14	29	76147	29	77291	29	76755.7	89.20
30	2	55	2	2	56	4777	56	4812	56	4796.6	20	300	3	480	5	5	80	120301	82	121733	81.1	120976.1	363.10
			3	3	37	4768	37	4789	37	4776.8	20				6	6	68	121922	70	123019	69.2	122148.9	343.60
			4	4	28	4747	28	4783	28	4764.6	20				7	7	59	121077	61	123476	60.2	122417.3	308
			2	4	29	4857	29	4903	29	4879.6	20				5	10	60	131179	60	160182	60	146051.7	165.60
			3	6	19	4837	19	4888	19	4865.2	20				6	12	50	130958	50	160657	50	143204.9	192.10
			4	8	15	4831	15	4869	15	4849.7	20				7	14	43	130308	43	160860	43	142720.3	176
40	2	60	2	2	75	6823	77	6934	75.9	6874.6	20	400	3	720	5	5	99	249391	101	252714	100	251944.5	976.10
			3	3	50	6801	52	6913	50.6	6846.2	20				6	6	82	248549	84	254319	83.6	251619.2	872
			4	4	38	6801	40	7120	39.2	6964.9	20				7	7	72	251401	73	254034	72.1	252712.7	854.90
			2	4	40	7051	43	7497	42.4	7343.8	20				5	10	80	284626	80	287152	80	286132.3	633.40
			3	6	25	6823	27	7018	26.2	6921.5	20				6	12	67	284409	67	287518	67	285705.9	638.70
			4	8	20	6937	21	7069	20.2	7001.8	20				7	14	58	282679	58	285852	58	284696.5	820.70
50	2	65	2	2	108	11561	109	11610	108.7	11595	20	500	3	720	5	5	117	289758	121	293262	118.8	292306	2210
			3	3	71	11491	72	11525	71.8	11510.6	20				6	6	98	290019	100	292751	98.8	291240.8	1999.90
			4	4	53	11398	53	11467	53	11451.1	20				7	7	85	288885	86	293248	85.3	291198.5	1894.70
			2	4	54	11373	57	11730	56.4	11683.7	20				5	10	100	328747	100	333747	100	330860.5	1273.80
			3	6	37	11626	37	11714	37	11666.3	20				6	12	84	327984	84	330458	84	329150.9	1523.60
			4	8	28	11585	28	11657	28	11628.2	20				7	14	72	326016	72	328627	72	327946.5	1463.90

Table B.13: Results for the LNS-RG in all instances with cluster topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1299	20	1299	20	1299	20	100	3	480	5	5	26	39211	27	39718	26.9	39029.1	26.90
			3	3	13	1291	13	1291	13	1291	20				6	6	22	38355	23	40606	22.6	38983.1	26.10
			4	4	10	1291	10	1291	10	1291	20				7	7	19	38570	20	39350	19.9	38939.5	26.70
			2	4	11	1327	11	1327	11	1327	20				5	10	20	41823	20	43262	20	42577.2	25.50
			3	6	7	1299	7	1307	7	1300.6	20				6	12	17	42224	17	42901	17	42583.2	25.90
			4	8	5	1291	5	1299	5	1298.2	20				7	14	15	42138	15	43135	15	42540.2	27
20	2	40	2	2	37	2335	40	2492	38.5	2411	20	200	3	480	5	5	52	76275	53	78234	52.3	77084	75.70
			3	3	25	2353	27	2459	26	2440.7	20				6	6	44	76925	45	78431	44.9	77430.3	76.90
			4	4	19	2353	20	2443	19.8	2419.5	20				7	7	38	76867	39	78167	38.9	77364.5	79.30
			2	4	21	2523	22	2615	21.8	2589	20				5	10	40	83771	40	85083	40	84305.9	68.50
			3	6	13	2415	14	2571	13.9	2543.7	20				6	12	34	83810	34	85086	34	84342	76.10
			4	8	10	2461	11	2561	10.8	2525.4	20				7	14	29	83245	29	84112	29	83885.9	74.70
30	2	55	2	2	43	3707	48	4086	46.1	3932.6	20	300	3	480	5	5	83	125995	85	127070	83.9	126587.6	239
			3	3	28	3663	31	4023	30.7	3961.9	20				6	6	70	125466	71	127135	70.3	126359.9	233.10
			4	4	22	3727	23	3975	22.9	3931.3	20				7	7	60	125451	62	127270	60.9	126148.5	294.70
			2	4	26	4113	26	4270	26	4221.7	20				5	10	60	135952	60	166492	60	160009.2	85.60
			3	6	16	3910	17	4206	16.9	4131.7	20				6	12	50	136979	50	166609	50	162984.2	66.30
			4	8	12	3930	13	4181	12.9	4125.8	20				7	14	43	135709	43	166447	43	145291.2	194.80
40	2	60	2	2	80	8035	89	8493	87.5	8437.1	20	400	3	720	5	5	103	261004	104	264249	103.6	262056.2	630.20
			3	3	57	8375	58	8443	57.9	8402.8	20				6	6	85	260457	87	263083	86.1	261271.6	638.70
			4	4	43	8317	43	8435	43	8370.9	20				7	7	73	258502	74	263343	73.7	260275.2	849.40
			2	4	46	8580	47	8602	46.2	8648.4	20				5	10	80	286304	80	289321	80	287447.9	594
			3	6	30	8533	30	8638	30	8590.6	20				6	12	67	285232	67	288061	67	286640.2	580.10
			4	8	22	8493	23	8549	22.8	8510.8	20				7	14	58	284142	58	288212	58	285950.8	598.60
50	2	65	2	2	85	9033	96	9842	93.2	9657.9	20	500	3	720	5	5	122	300937	125	305901	123.3	304571.9	1619.90
			3	3	62	9606	63	9710	62.5	9669.8	20				6	6	101	300270	104	306271	102.9	303051.1	1752.60
			4	4	46	9559	47	9630	46.3	9597	20				7	7	88	301550	90	304460	88.5	303198	1767.30
			2	4	50	9969	51	10109	50.7	10074.3	20				5	10	100	337103	100	338896	100	337837.8	1263
			3	6	33	9957	33	10039	33	9996.9	20				6	12	84	335356	84	337521	84	336655.2	1519
			4	8	25	9852	25	9940	25	9902.1	20				7	14	72	334462	72	338508	72	335639.8	1458.50

Table B.14: Results for the LNS-RG in all instances with random topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	18	1031	18	1044	18	1040.3	20	100	3	480	5	5	25	34862	25	36203	25	35535.7	26.90
			3	3	12	1012	12	1026	12	1013.4	20				6	6	21	34917	22	35996	21.2	35549.5	27.20
			4	4	9	1012	9	1012	9	1012	20				7	7	18	34593	19	36195	18.8	35546.1	28.60
			2	4	10	1078	10	1092	10	1087	20				5	10	20	39154	20	39898	20	39485.3	25.90
			3	6	6	1040	6	1040	6	1040	20				6	12	17	39202	17	39858	17	39516.3	27
			4	8	5	1040	5	1040	5	1040	20				7	14	15	39179	15	40275	15	39559.5	25.90
20	2	40	2	2	41	2372	42	2438	41.3	2397.6	20	200	3	480	5	5	57	87113	59	87675	58	87313.1	103
			3	3	27	2350	28	2434	27.5	2392.6	20				6	6	48	86370	50	88815	48.9	87540.7	88.80
			4	4	21	2372	21	2430	21	2411.4	20				7	7	42	86715	43	88188	42.3	87717	93.30
			2	4	22	2438	22	2522	22	2501.2	20				5	10	40	94073	40	111697	40	107995.7	38.70
			3	6	14	2416	15	2514	14.8	2479.8	20				6	12	34	93129	34	111695	34	100970.3	65.60
			4	8	11	2390	11	2492	11	2454	20				7	14	29	93804	29	112210	29	103180.1	52.10
30	2	55	2	2	37	3284	42	3605	40	3447.7	20	300	3	480	5	5	87	133370	89	135375	88.1	134178.1	343.60
			3	3	25	3302	27	3553	26.6	3477.9	20				6	6	74	133212	75	135685	74.5	134551	359.90
			4	4	20	3448	20	3525	20	3490.9	20				7	7	65	134153	66	136383	65.1	135785.1	296.60
			2	4	23	3717	23	3781	23	3752.9	20				5	10	60	146215	61	175627	60.9	172130.3	64.10
			3	6	15	3701	15	3751	15	3733.4	20				6	12	51	174774	51	175458	51	175161.3	48
			4	8	11	3706	12	3753	11.3	3725	20				7	14	43	176498	44	175540	43.3	173707.3	69.80
40	2	60	2	2	52	5019	54	5117	52.9	5080.8	20	400	3	720	5	5	98	233048	100	239358	98.8	236174.2	913.70
			3	3	34	4970	39	5628	36.4	5264.6	20				6	6	82	234334	84	237894	82.8	236329.2	880.80
			4	4	26	4983	29	5597	28.3	5415.9	20				7	7	70	234814	71	236647	70.8	235247.4	1028.70
			2	4	32	5998	33	6047	32.8	6001	20				5	10	80	267413	80	270874	80	269346.7	715.70
			3	6	20	5322	22	5911	21.1	5736.1	20				6	12	67	267634	67	272051	67	269881.7	702.60
			4	8	15	5303	16	5934	15.9	5845.7	20				7	14	58	268467	58	271694	58	270301	758.40
50	2	65	2	2	66	6856	72	7447	70.7	7414.5	20	500	3	720	5	5	117	278882	119	284832	117.8	280698.3	2152.40
			3	3	46	7283	47	7358	46.1	7360	20				6	6	99	277250	100	281897	99.4	279936.2	2114.40
			4	4	34	7194	34	7370	34	7283.3	20				7	7	84	277863	87	282633	85.6	280275.3	2026.50
			2	4	39	7714	40	7735	39.6	7731.6	20				5	10	100	310085	100	312830	100	311025.8	1540.40
			3	6	26	7618	26	7700	26	7663.1	20				6	12	84	308406	84	312404	84	310231.6	1460.60
			4	8	19	7582	19	7693	19	7640.9	20				7	14	72	306848	72	309875	72	308322.8	1551.50

Table B.15: Results for the LNS-RG in all instances with mix topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1126	20	1126	20	1126	~ 0	100	3	480	5	5	24	34601	24	34908	24	34795	23.90
			3	3	13	1122	13	1122	13	1122	0.20				6	6	20	34543	20	34929	20	34727.2	25
			4	4	10	1122	10	1126	10	1124.2	0.20				7	7	17	34736	18	34981	17.9	34829	27.90
			2	4	11	1142	11	1142	11	1142	0.30				5	10	20	38511	20	39129	20	38867.6	25
			3	6	7	1140	7	1140	7	1140	0.50				6	12	17	38570	17	38999	17	38852.8	28.60
			4	8	6	1138	6	1138	6	1138	0.20				7	14	15	38548	15	38945	15	38783.9	30
20	2	40	2	2	36	2102	36	2102	36	2102	0.60	200	3	480	5	5	47	67603	48	68981	47.6	68270	112.50
			3	3	24	2102	24	2102	24	2102	0.50				6	6	40	67467	41	68102	40.1	67982.5	118.50
			4	4	18	2102	18	2102	18	2102	0.70				7	7	35	66896	36	67772	35.2	67829.3	130.20
			2	4	19	2144	19	2166	19	2157.6	0.80				5	10	40	78588	40	80643	40	79731.1	120
			3	6	13	2130	13	2144	13	2133.8	1				6	12	34	78898	34	79899	34	79455.6	129.70
			4	8	9	2116	10	2144	9.7	2128.6	1				7	14	29	78529	29	79060	29	78738.4	139.90
30	2	55	2	2	55	4697	55	4705	55	4701.3	1.30	300	3	480	5	5	80	123006	82	124625	81.2	123468.9	311.50
			3	3	37	4693	37	4717	37	4706.2	1.60				6	6	68	122137	69	124835	68.9	123810.2	331.30
			4	4	28	4710	28	4750	28	4729.6	1.70				7	7	60	122361	61	124969	60.2	123786	321
			2	4	28	4795	29	4828	28.9	4803.5	1.10				5	10	60	135225	60	136345	60	135882.8	303.10
			3	6	19	4749	20	4803	19.1	4770.3	1.30				6	12	50	134304	50	135289	50	134816.5	332.40
			4	8	15	4788	15	4859	15	4821.3	2				7	14	43	133433	43	134806	43	134285.5	339.50
40	2	60	2	2	75	6801	75	6801	75	6801	2.10	400	3	720	5	5	97	254506	101	258513	99.5	256934.6	658
			3	3	50	6801	50	6801	50	6801	2.40				6	6	83	255747	85	257776	83.9	256672.6	649.70
			4	4	38	6801	38	6801	38	6801	3				7	7	72	255681	73	257790	72.1	256885	629.90
			2	4	39	7018	40	7105	39.9	7042.8	2.20				5	10	80	293238	80	295605	80	294248	663
			3	6	26	6845	26	6889	26	6858.4	2.60				6	12	67	292140	67	295194	67	293512.8	666.60
			4	8	19	6845	20	6913	19.5	6869.4	3.10				7	14	58	290850	58	294387	58	292335.6	674.60
50	2	65	2	2	98	11019	100	11071	99	11048.9	3.70	500	3	720	5	5	116	295298	121	300992	118.8	299264.5	1054.60
			3	3	67	11055	67	11127	67	11081.2	3.70				6	6	97	294385	101	296883	98.8	296273.3	1125.40
			4	4	50	11055	51	11141	50.5	11090.8	4.30				7	7	84	293859	86	299866	85.4	296524.8	1131.90
			2	4	52	11255	53	11346	52.9	11299.1	3.70				5	10	100	337781	100	343622	100	340079.5	1038.60
			3	6	34	11112	35	11261	34.6	11188.6	4.30				6	12	84	335741	84	339307	84	337718.6	1212.60
			4	8	26	11172	27	11340	26.5	11268.3	5.50				7	14	72	334859	72	338709	72	336443.3	1238.10

Table B.16: Results for the LNS-SA-GG in all instances with cluster topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1299	20	1311	20	1300.2	0.10	100	3	480	5	5	26	38745	27	39213	26.1	38928.2	23.90
			3	3	14	1299	14	1299	14	1299	0.20				6	6	22	38562	22	39486	22	38912.2	26.60
			4	4	10	1299	10	1299	10	1299	0.20				7	7	19	38197	20	39116	19.1	38786.4	28
			2	4	11	1339	11	1339	11	1339	0.20				5	10	20	42768	20	43262	20	43042.5	26.70
			3	6	7	1307	7	1307	7	1307	0.50				6	12	17	42603	17	43115	17	42924	29.30
			4	8	6	1315	6	1315	6	1315	0.30				7	14	15	42278	15	42898	15	42660.3	32
20	2	40	2	2	36	2317	37	2335	36.4	2326.2	0.70	200	3	480	5	5	52	77460	53	78812	52.1	78000.6	109.70
			3	3	25	2335	25	2335	25	2335	0.50				6	6	43	76974	44	79072	43.7	77805	110.80
			4	4	19	2335	19	2353	19	2349.4	1				7	7	38	77507	39	78370	38.5	78108.3	127.90
			2	4	20	2485	21	2528	20.9	2503.4	0.80				5	10	40	84962	40	86225	40	85805.3	120
			3	6	13	2389	14	2453	13.6	2433.9	0.80				6	12	34	85175	34	85734	34	85482	127.20
			4	8	10	2391	10	2427	10	2411.6	1				7	14	29	84644	29	85257	29	84941.7	139.40
30	2	55	2	2	40	3631	42	3658	41	3644.2	1.10	300	3	480	5	5	83	127827	84	129860	83.7	128684.1	294.90
			3	3	27	3640	28	3682	27.9	3656.9	1.40				6	6	70	126959	71	128978	70.1	127758.2	308.20
			4	4	21	3660	22	3723	21.4	3687.6	1.70				7	7	60	127205	62	128691	61	127914.8	322.80
			2	4	24	3923	25	3998	24.5	3971.5	1.30				5	10	60	139257	60	141762	60	140127.9	325.30
			3	6	15	3788	16	3838	15.1	3825.3	1.50				6	12	50	138924	50	140188	50	139433.8	338.40
			4	8	12	3797	12	3878	12	3829.9	1.80				7	14	43	137848	43	138853	43	138474.7	342.10
40	2	60	2	2	78	7949	79	7997	78.4	7977	2.40	400	3	720	5	5	101	262066	104	266981	102.5	265830.6	582.90
			3	3	53	7979	54	8005	53.1	7997.6	2.50				6	6	84	262894	86	265361	85.3	263653.5	591.60
			4	4	40	8005	41	8093	40.7	8043.4	3.10				7	7	73	262979	74	267461	73.7	264207.6	646
			2	4	43	8247	44	8330	43.5	8298	2.50				5	10	80	292663	80	294722	80	293930.9	622.30
			3	6	27	8035	28	8123	27.8	8086.7	2.50				6	12	67	291390	67	294753	67	292632.2	655.30
			4	8	21	8089	22	8195	21.3	8140.4	3.80				7	14	58	289816	58	292894	58	290901.3	675.30
50	2	65	2	2	84	8984	85	9025	84.6	9015.9	3.10	500	3	720	5	5	120	307590	124	311677	122.2	310602.1	1040.70
			3	3	56	8938	57	9027	56.3	8986.5	4				6	6	101	307733	104	311457	102.7	309471.3	1053.70
			4	4	42	8990	43	9052	42.9	9022.8	4				7	7	87	305598	89	309782	88	307648.7	1059.90
			2	4	45	9353	47	9527	46	9436.9	3				5	10	100	347056	100	351318	100	348946	1230.50
			3	6	30	9259	31	9385	30.2	9319.8	3.90				6	12	84	344606	84	348029	84	346698.9	1366.90
			4	8	22	9201	23	9420	22.9	9309.2	4.80				7	14	72	342475	72	347957	72	345003.9	1264.20

Table B.17: Results for the LNS-SA-GG in all instances with random topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	18	1040	18	1040	18	1040	0.10	100	3	480	5	5	24	35271	25	35845	24.7	35605.3	23.50
			3	3	12	1012	12	1012	12	1012	0.20				6	6	21	35212	21	35683	21	35497.2	25.50
			4	4	9	1012	9	1012	9	1012	0.20				7	7	18	35216	18	35785	18	35421.5	27.80
			2	4	10	1112	10	1112	10	1112	0.20				5	10	20	39534	20	40191	20	39828.1	26.30
			3	6	6	1040	6	1040	6	1040	0.50				6	12	17	39484	17	40281	17	39928.2	28.50
			4	8	5	1068	5	1068	5	1068	0.30				7	14	15	39397	15	40063	15	39714.2	31.30
20	2	40	2	2	40	2350	40	2350	40	2350	0.70	200	3	480	5	5	56	87461	58	88791	57.1	88347.5	121.60
			3	3	27	2350	27	2350	27	2350	0.70				6	6	48	87312	49	89193	48.5	88514.2	124.20
			4	4	20	2350	20	2350	20	2350	0.90				7	7	42	87318	42	89432	42	88473.6	126.10
			2	4	21	2416	22	2456	21.7	2440.8	0.70				5	10	40	95194	40	96758	40	95924.7	119.50
			3	6	14	2394	14	2394	14	2394	0.70				6	12	34	95336	34	96241	34	95843.9	128.70
			4	8	10	2350	11	2394	10.9	2377	1				7	14	29	94654	29	96232	29	95370.9	141.80
30	2	55	2	2	36	3235	38	3322	37	3273.3	1.10	300	3	480	5	5	86	134349	88	137927	87.3	136130.3	318.10
			3	3	24	3227	25	3254	24.7	3240	1.40				6	6	74	135549	75	138524	74.5	136759.6	316.90
			4	4	19	3247	19	3304	19	3281.8	1.80				7	7	65	136548	65	138872	65	137530.4	332.80
			2	4	21	3532	22	3568	21.7	3526	1.30				5	10	60	149707	61	149716	60.2	150642	322
			3	6	13	3361	14	3422	13.9	3402.3	1.30				6	12	50	149708	51	149688	50.2	150812.5	343.10
			4	8	10	3416	11	3492	10.9	3454.4	1.80				7	14	43	148690	44	150186	43.1	149959	346.10
40	2	60	2	2	51	4987	52	5049	51.6	5024.4	2.10	400	3	720	5	5	96	237592	99	246853	97.2	241644.6	653.80
			3	3	34	4940	34	4994	34	4959.8	2.50				6	6	81	237420	83	243037	81.8	240845.7	655.30
			4	4	25	4922	26	5017	25.9	4982.6	2.90				7	7	70	237939	71	240189	70.4	239251.3	640.60
			2	4	31	5498	31	5593	31	5546.5	2.10				5	10	80	279297	80	285809	80	281413.5	603.70
			3	6	19	5177	19	5250	19	5209.6	2.40				6	12	67	277106	67	283193	67	279311.9	641.70
			4	8	14	5276	15	5280	14.8	5259.3	3.20				7	14	58	275829	58	280179	58	278020.8	700.20
50	2	65	2	2	64	6764	65	6877	64.8	6796.9	3.40	500	3	720	5	5	115	283163	118	289576	116.4	287386.8	1163.30
			3	3	43	6766	44	6805	43.4	6789	3.90				6	6	97	288727	100	288993	98.8	287902.1	1045.90
			4	4	33	6811	34	6920	33.7	6876	4.50				7	7	83	282890	86	288755	84.6	285747.5	1187.70
			2	4	36	7096	37	7207	36.1	7168.7	3.40				5	10	100	321947	100	325795	100	323743.5	1252.80
			3	6	24	7006	24	7084	24	7047.7	3.70				6	12	84	318101	84	322674	84	320860.7	1216.70
			4	8	18	7014	19	7193	18.8	7123.8	4.90				7	14	72	317384	72	323090	72	320309.2	1174

Table B.18: Results for the LNS-SA-GG in all instances with mix topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1126	20	1126	20	1126	20	100	3	480	5	5	24	34622	24	34958	24	34786	42.30
			3	3	13	1122	13	1122	13	1122	20				6	6	20	34559	20	34917	20	34747.1	44
			4	4	10	1122	10	1126	10	1125	20				7	7	18	34761	18	34949	18	34831.1	46.80
			2	4	10	1143	10	1148	10	1146.1	20				5	10	20	38567	20	39308	20	38921.2	45
			3	6	7	1137	7	1141	7	1139.2	20				6	12	17	38658	17	39024	17	38863.7	48.40
			4	8	5	1130	5	1139	5	1135.1	20				7	14	15	38666	15	39058	15	38812.3	50.60
20	2	40	2	2	36	2102	36	2102	36	2102	20	200	3	480	5	5	47	67629	48	69599	47.5	68364.1	133.40
			3	3	24	2102	24	2102	24	2102	20				6	6	40	68257	41	68837	40.5	68530.5	137.10
			4	4	18	2102	18	2102	18	2102	20				7	7	35	67577	36	68445	35.3	68178.1	143.90
			2	4	19	2144	19	2166	19	2155.8	20				5	10	40	79229	40	80095	40	79760.5	136.70
			3	6	13	2130	13	2144	13	2134.2	20				6	12	34	79011	34	79877	34	79352.2	153.10
			4	8	9	2116	10	2137	9.7	2126.5	20				7	14	29	78115	29	79467	29	78715.5	155.80
30	2	55	2	2	55	4692	55	4703	55	4698.5	21	300	3	480	5	5	80	121927	82	123519	80.9	123114.2	331.40
			3	3	36	4677	37	4720	36.9	4706.5	21				6	6	68	122520	69	123596	68.4	123160.7	339.60
			4	4	28	4701	28	4744	28	4727.8	21				7	7	59	123479	60	124020	59.9	123547.7	343.40
			2	4	28	4796	29	4845	28.8	4806.5	21				5	10	60	134323	60	137217	60	135883.3	358.80
			3	6	19	4747	19	4888	19	4792.2	21				6	12	50	134095	50	135559	50	134944.9	357.70
			4	8	15	4791	15	4830	15	4810.7	21				7	14	43	133213	43	134630	43	134070.4	385.50
40	2	60	2	2	75	6801	75	6801	75	6801	22	400	3	720	5	5	98	256455	101	257452	99.6	256624.5	635
			3	3	50	6801	50	6801	50	6801	22				6	6	83	255628	85	257205	83.8	257163.3	690.10
			4	4	38	6801	38	6801	38	6801	22.20				7	7	71	256036	72	260511	71.9	256830.2	667.20
			2	4	39	7012	40	7038	39.9	7015.1	22				5	10	80	291456	80	296623	80	293950.5	648
			3	6	25	6825	26	6867	25.9	6850	22				6	12	67	292258	67	296275	67	293682.8	665.20
			4	8	19	6845	20	6905	19.6	6873.4	23				7	14	58	290205	58	294401	58	292534.6	707.10
50	2	65	2	2	98	11035	100	11119	99.2	11066	23	500	3	720	5	5	117	295978	120	301851	117.9	298182.3	1116.50
			3	3	67	11055	68	11140	67.1	11089.3	23				6	6	97	294731	101	297988	99.1	297923.2	1137.70
			4	4	50	11043	51	11198	50.3	11086.6	24				7	7	84	294619	88	297210	85.3	296372.3	1128.30
			2	4	53	11275	54	11392	53.2	11309.9	23				5	10	100	337343	100	341807	100	339872.8	1236.20
			3	6	34	11168	35	11225	34.7	11193.9	23.90				6	12	84	336610	84	342073	84	338215	1255.90
			4	8	26	11197	27	11336	26.7	11278.1	24.90				7	14	72	334596	72	337706	72	336185	1281.60

Table B.19: Results for the LNS-SA-RG in all instances with cluster topology after 10 runs.

S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	20	1299	20	1299	20	1299	20	100	3	480	5	5	26	38499	26	39247	26	38843.9	43.20
			3	3	13	1291	13	1291	13	1291	20				6	6	22	38591	22	39519	22	38912.8	45.10
			4	4	10	1291	10	1291	10	1291	20				7	7	19	38320	20	38894	19.1	38636.7	47.50
			2	4	11	1327	11	1327	11	1327	20				5	10	20	42804	20	43331	20	43040.9	45.50
			3	6	7	1299	7	1307	7	1300.6	20				6	12	17	42385	17	42966	17	42798.5	48.50
			4	8	5	1291	5	1299	5	1298.2	20				7	14	15	42365	15	42818	15	42654.3	50.60
20	2	40	2	2	36	2321	37	2335	36.2	2324.2	20	200	3	480	5	5	51	77528	53	78104	51.9	77794.9	137.30
			3	3	25	2335	25	2335	25	2335	20				6	6	43	76790	44	78574	43.7	77696	140
			4	4	19	2335	19	2353	19	2349.4	20				7	7	38	77386	39	78188	38.4	77764.1	149.10
			2	4	21	2485	21	2521	21	2504.1	20				5	10	40	85057	40	86327	40	85667.1	142
			3	6	13	2425	14	2463	13.6	2440.4	20				6	12	34	85154	34	85685	34	85411.6	154
			4	8	10	2407	10	2425	10	2411	20.40				7	14	29	84660	29	85013	29	84868.3	157.40
30	2	55	2	2	40	3623	41	3681	40.7	3650.3	21	300	3	480	5	5	82	127088	84	128257	83	127853.3	317.40
			3	3	28	3617	28	3679	28	3650	21				6	6	69	127500	71	128445	70	127489	337.90
			4	4	21	3643	22	3721	21.3	3693.3	21				7	7	60	127040	61	129190	60.9	128003.7	339.70
			2	4	24	3912	25	3991	24.6	3966.4	21				5	10	60	139922	60	140952	60	140451	335.10
			3	6	15	3796	16	3883	15.4	3839.3	21				6	12	50	138856	50	140268	50	139519	365.60
			4	8	12	3821	12	3849	12	3836.1	21				7	14	43	137794	43	139066	43	138423	374.30
40	2	60	2	2	78	7961	79	8001	78.6	7980.8	22	400	3	720	5	5	101	262706	104	266985	102.6	265703.1	646.10
			3	3	53	7971	54	8025	53.1	7996.9	22				6	6	85	263469	86	266376	85.8	264372	708.80
			4	4	40	8007	41	8047	40.7	8030.8	22.50				7	7	73	263741	74	266204	73.6	264025.6	662
			2	4	43	8183	44	8328	43.5	8280.8	22				5	10	80	293135	80	295811	80	294408.8	660.30
			3	6	27	8053	28	8103	27.9	8086	22				6	12	67	291045	67	294148	67	292825.4	685.60
			4	8	21	8109	22	8185	21.1	8142.2	23				7	14	58	290562	58	292467	58	291484.4	707.50
50	2	65	2	2	84	8984	85	9070	84.5	9021.6	23	500	3	720	5	5	121	307403	124	314512	122.6	311097.7	1188.90
			3	3	56	8952	57	9018	56.3	8984.5	23				6	6	101	306820	104	310550	102.4	308636	1178.80
			4	4	43	8992	43	9056	43	9030.5	23.80				7	7	87	305659	89	308909	88.2	307355.1	1173.70
			2	4	46	9386	47	9466	46.2	9456.1	22.90				5	10	100	346855	100	350538	100	348955.6	1206.70
			3	6	30	9246	31	9432	30.3	9335.5	23				6	12	84	344308	84	348075	84	346244.4	1262.10
			4	8	22	9201	23	9378	22.9	9301.4	24				7	14	72	343470	72	346928	72	345145.1	1280.70

Table B.20: Results for the LNS-SA-RG in all instances with random topology after 10 runs.

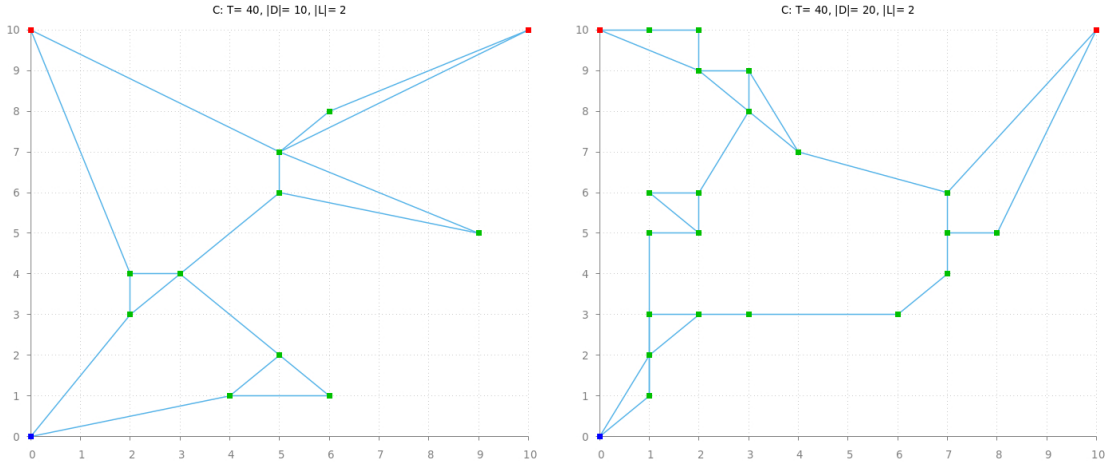
S1												S2											
Instance					Best		Worst		Average			Instance					Best		Worst		Average		
$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)	$\|D\|$	$\|L\|$	T	W	$\|K\|$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	$\mathcal{F}_1$	$\mathcal{F}_2$	t(s)
10	2	40	2	2	18	1031	18	1040	18	1039.1	20	100	3	480	5	5	24	35056	25	35830	24.6	35562.7	43.60
			3	3	12	1012	12	1012	12	1012	20				6	6	21	35313	21	35715	21	35492.7	45.40
			4	4	9	1012	9	1012	9	1012	20				7	7	18	35138	18	35601	18	35372.2	47.70
			2	4	10	1078	10	1092	10	1087	20				5	10	20	39586	20	40449	20	40027.7	43.90
			3	6	6	1040	6	1040	6	1040	20				6	12	17	39509	17	40234	17	39814.1	48.10
			4	8	5	1040	5	1040	5	1040	20				7	14	15	39129	15	40087	15	39666.6	52.40
20	2	40	2	2	40	2350	40	2350	40	2350	20	200	3	480	5	5	57	87600	58	88929	57.1	88256.1	138
			3	3	27	2350	27	2350	27	2350	20				6	6	48	87585	49	89110	48.3	88449.1	143.30
			4	4	20	2350	20	2350	20	2350	20				7	7	42	86927	43	88047	42.1	88250	146
			2	4	21	2438	22	2456	21.9	2443	20				5	10	40	95086	40	96365	40	95947.6	142.70
			3	6	14	2394	14	2416	14	2399.8	20				6	12	34	95120	34	96304	34	95875.7	154.60
			4	8	11	2372	11	2394	11	2383.2	20				7	14	29	94945	29	96152	29	95322.6	159.30
30	2	55	2	2	37	3252	38	3309	37.1	3274.2	21	300	3	480	5	5	87	134759	88	137649	87.6	136243.4	332.80
			3	3	24	3230	25	3272	24.9	3250.1	21				6	6	74	135973	75	137123	74.4	136574.7	334.60
			4	4	19	3270	19	3326	19	3291.2	21				7	7	64	134605	65	137765	64.9	136669.6	339.20
			2	4	21	3481	22	3556	21.5	3521.6	21				5	10	60	149631	61	150266	60.1	150741.1	346.60
			3	6	13	3356	14	3457	13.9	3418.1	21				6	12	50	149025	50	152592	50	150630.1	334.30
			4	8	10	3420	11	3523	10.9	3480	21				7	14	43	148765	43	151445	43	149899.5	366.70
40	2	60	2	2	51	5012	52	5059	51.5	5032.3	21.80	400	3	720	5	5	96	238515	99	247705	97.1	241881.6	663.50
			3	3	34	4943	34	5005	34	4969.6	22				6	6	80	238851	83	244719	81.5	240656	668.10
			4	4	25	4937	26	5036	25.9	4972.8	22.10				7	7	69	235838	71	242026	70.5	239302.3	687
			2	4	30	5556	31	5576	30.9	5539.6	22				5	10	80	278195	80	282991	80	280391.6	679.10
			3	6	19	5178	19	5296	19	5231.3	22				6	12	67	275909	67	280851	67	279180.2	674.60
			4	8	14	5193	15	5316	14.8	5259.2	22.90				7	14	58	276720	58	280163	58	278536.4	706.40
50	2	65	2	2	64	6745	65	6852	64.8	6824.5	23	500	3	720	5	5	115	285144	119	288859	116.9	287470.9	1117.10
			3	3	43	6735	44	6826	43.3	6783	23.10				6	6	97	283389	100	289864	98.4	287780.2	1221.10
			4	4	33	6828	34	6910	33.6	6876.6	24				7	7	83	282153	86	287267	84.5	285831.6	1213.20
			2	4	36	7115	37	7195	36.2	7181.9	23				5	10	100	322012	100	329271	100	324002.8	1218.30
			3	6	24	7025	24	7148	24	7064.2	23.10				6	12	84	320537	84	323421	84	322303.3	1247.60
			4	8	18	6984	19	7211	18.9	7158.7	24				7	14	72	317254	72	319699	72	318853.1	1256.70

Table B.21: Results for the LNS-SA-RG in all instances with mix topology after 10 runs.

Appendix C

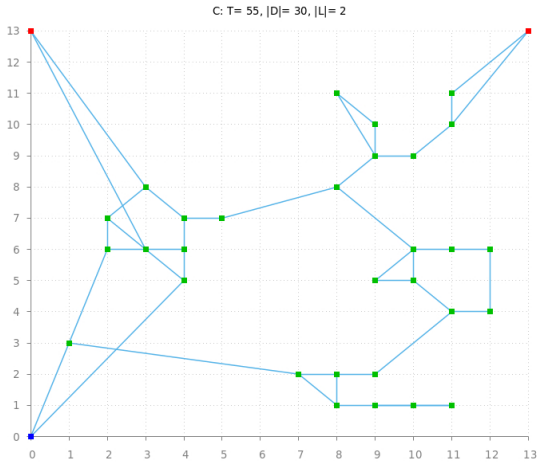
Graphs figures

For the sake of clarity, the generated graphs for the instances of S1 and S2 are illustrated in this appendix. The information about T, $|D|$ and $|L|$ is showed in the title of each figure. The depot, the landfills and the debris nodes are respectively the blue, the red and the green points. As mentioned before, each one of these graphs was the base for 6 different instances varying the number of WTs and dump trucks.

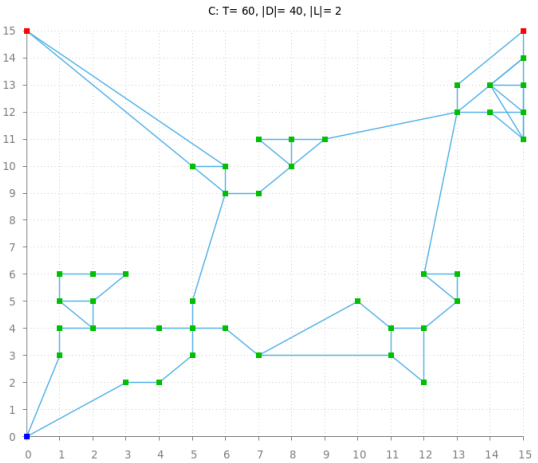


(a)

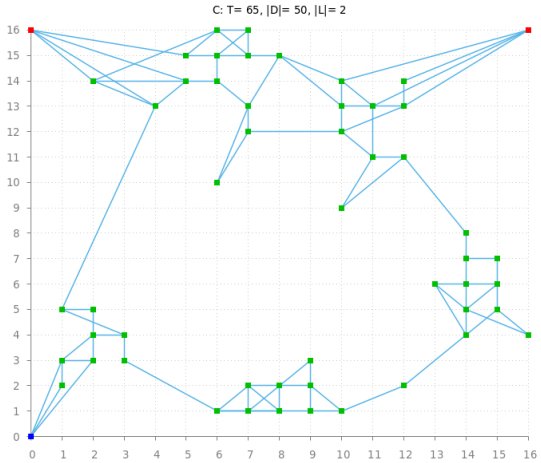
(b)



(c)

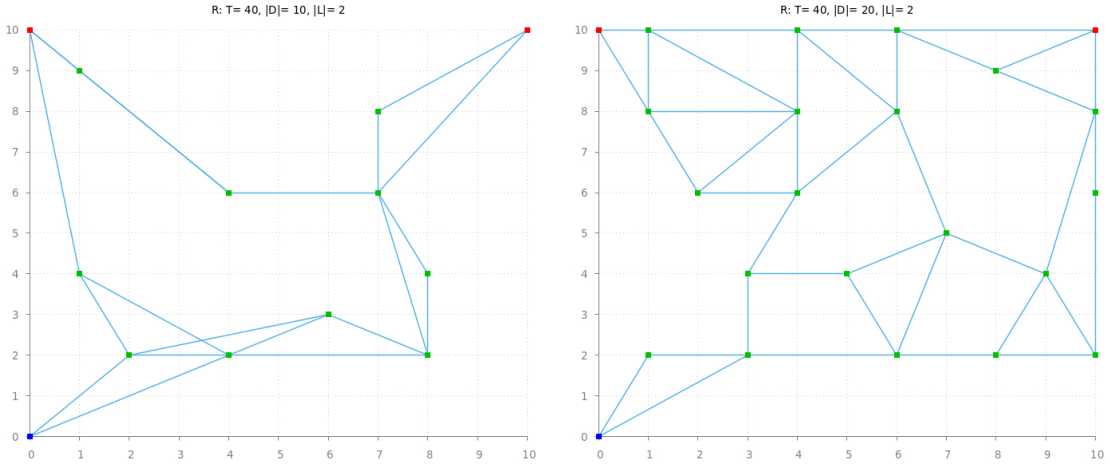


(d)



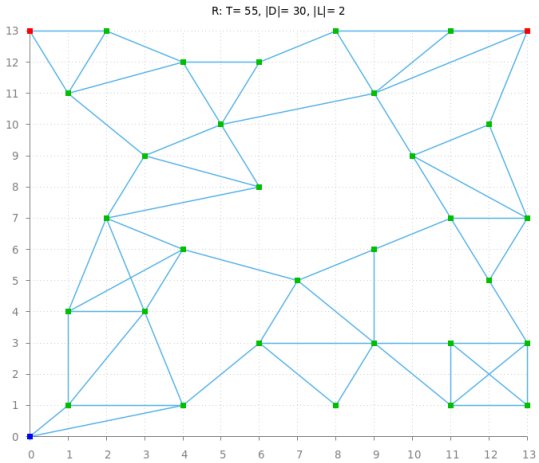
(e)

Figure C.1: Graphs used for the cluster topology instances in S1.

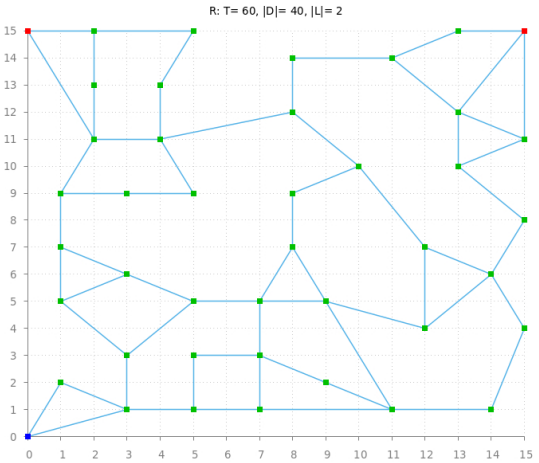


(a)

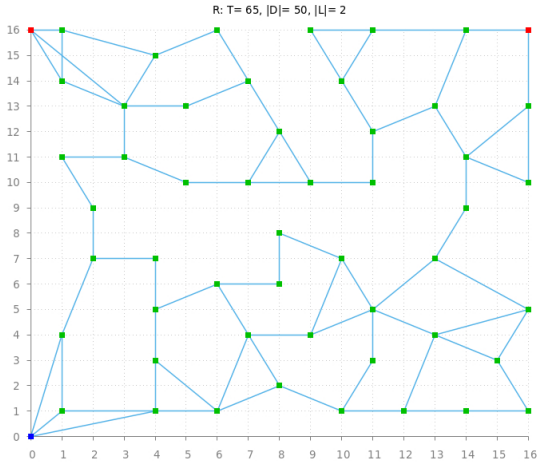
(b)



(c)

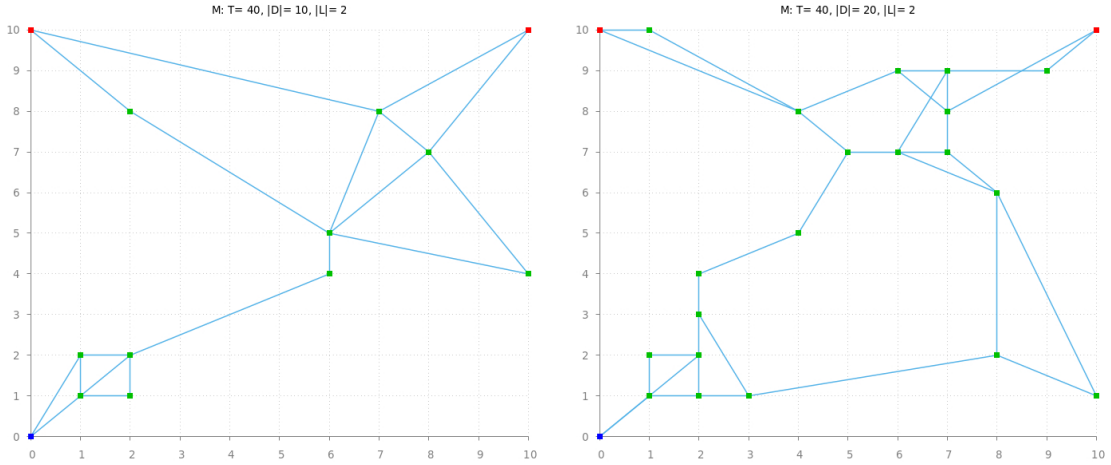


(d)



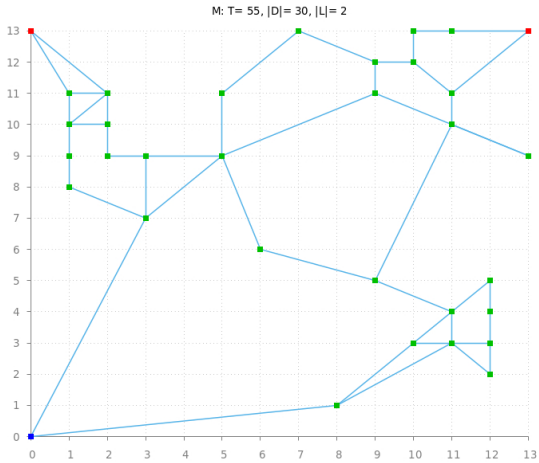
(e)

Figure C.2: Graphs used for the random topology instances in S1.

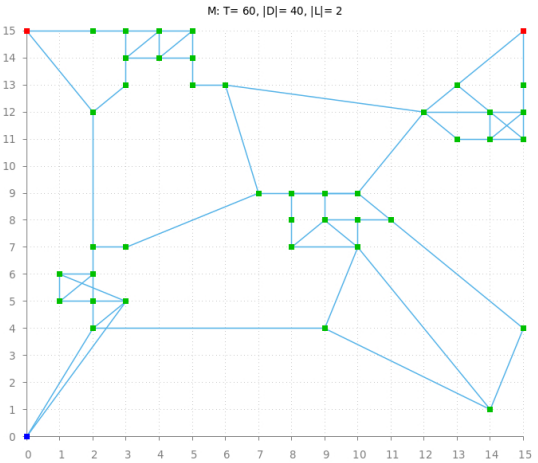


(a)

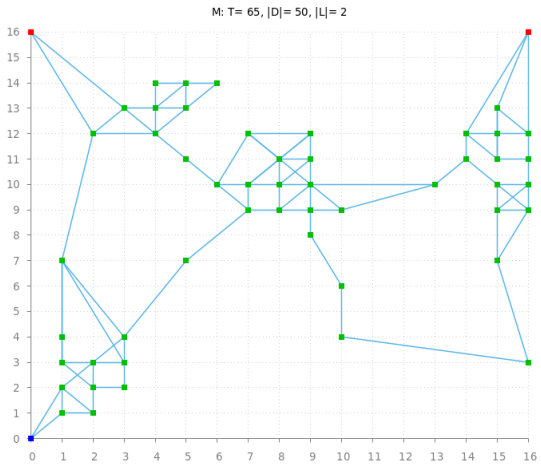
(b)



(c)

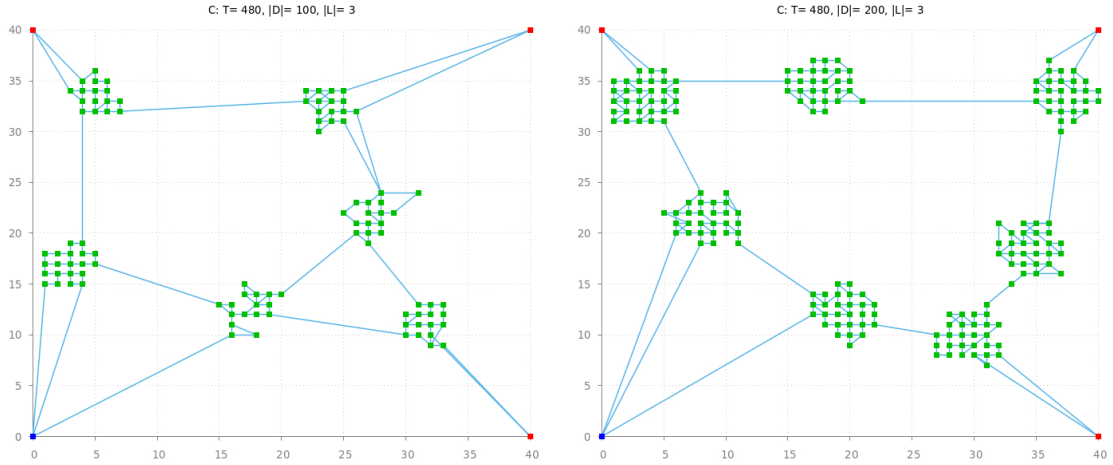


(d)



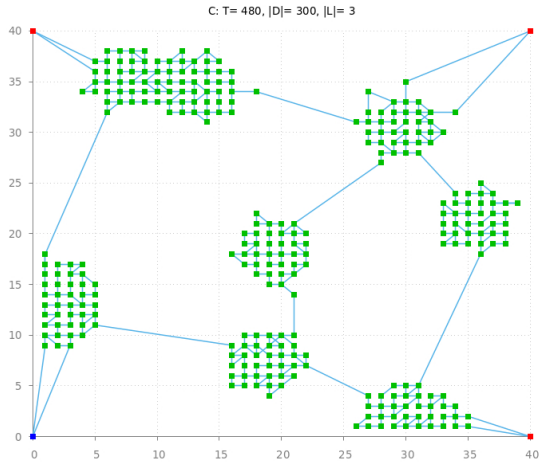
(e)

Figure C.3: Graphs used for the mix topology instances in S1.

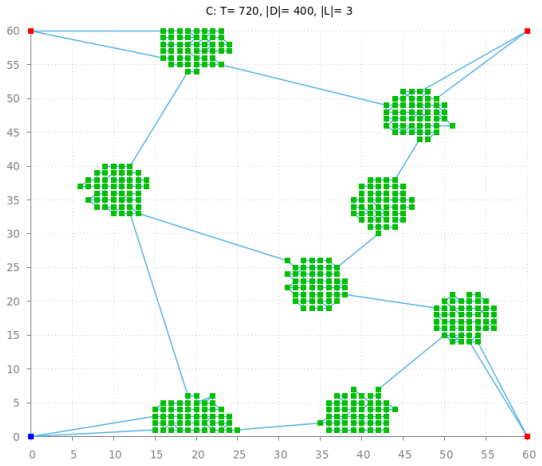


(a)

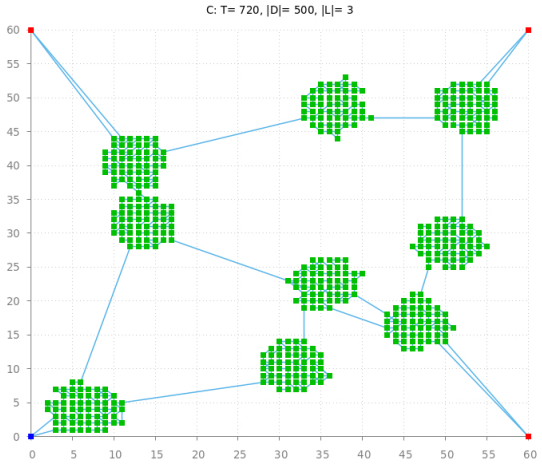
(b)



(c)



(d)



(e)

Figure C.4: Graphs used for the cluster topology instances in S2.

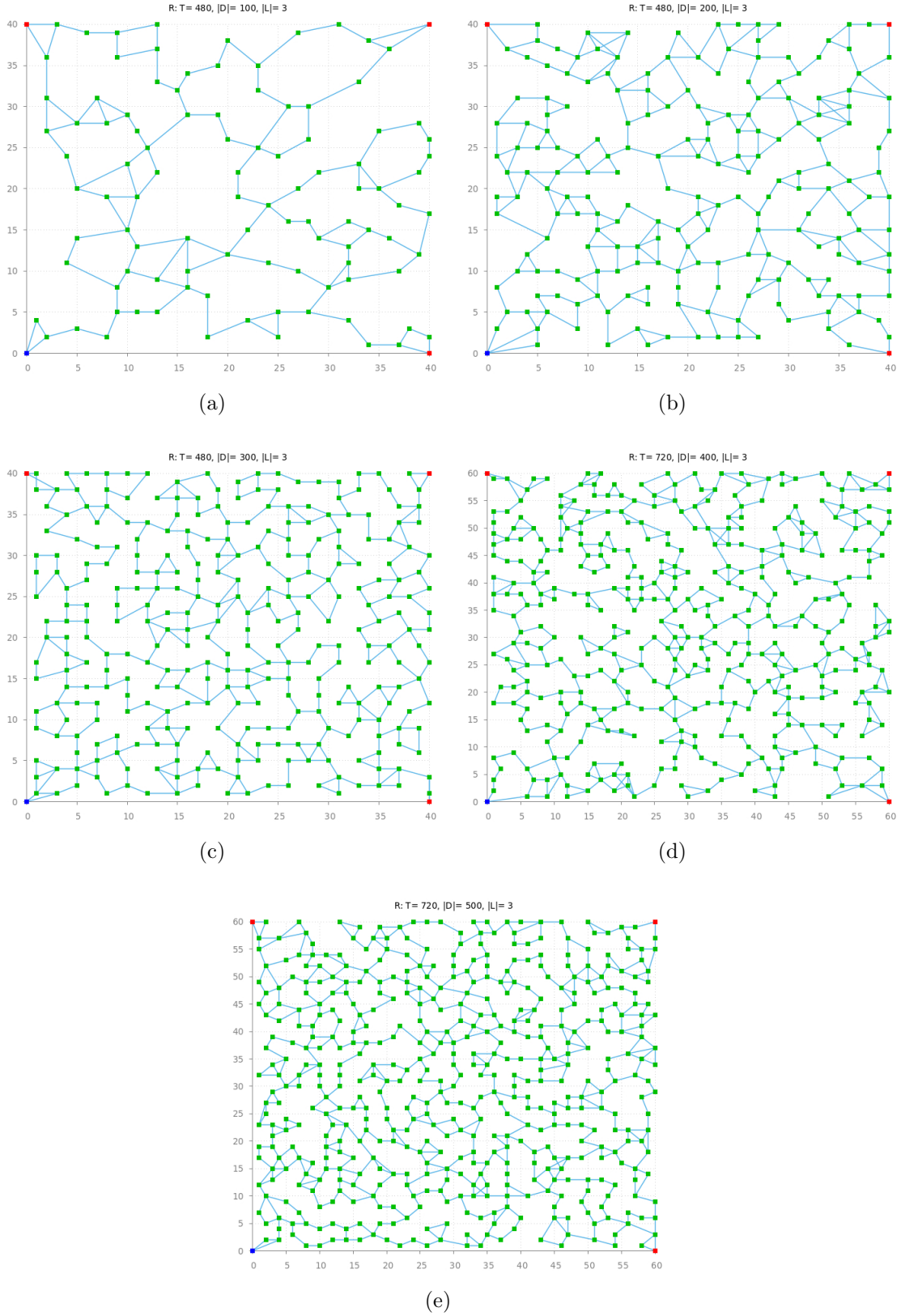


Figure C.5: Graphs used for the random topology instances in S2.

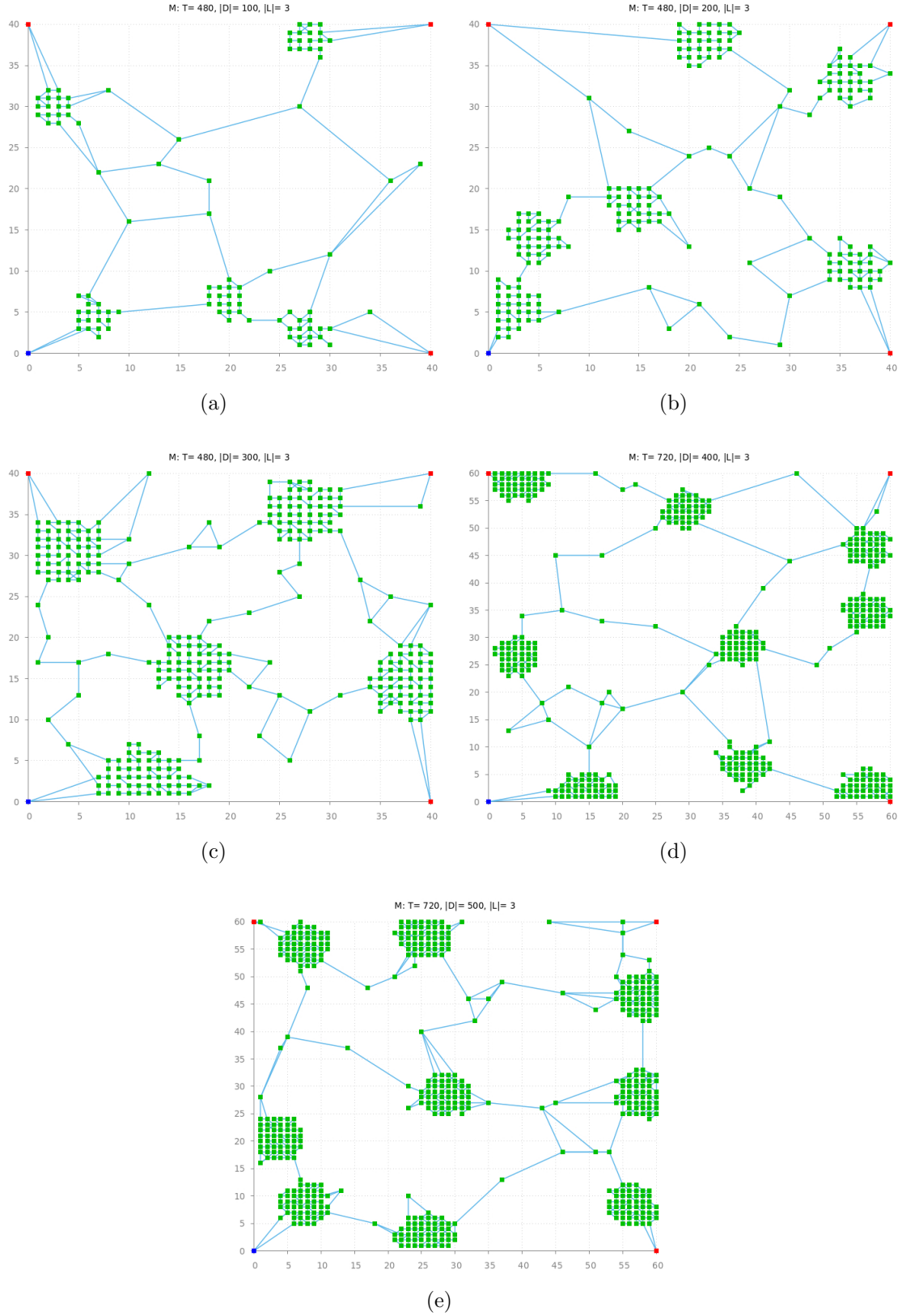


Figure C.6: Graphs used for the mix topology instances in S2.

Bibliography

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1989). Chapter IV network flows. *Handbooks in Operations Research and Management Science*, Vol. 1: *Optimization* (G. L. Nemhauser, A. H. Rinnooy Kan, and M. J. Todd, eds.), North-Holland, Amsterdam:211–369.

Aiex, R. M., Resende, M. G., and Ribeiro, C. C. (2002). Probability distribution of solution time in GRASP: An experimental investigation. *Journal of Heuristics*, 8(3):343–373.

Aiex, R. M., Resende, M. G., and Ribeiro, C. C. (2007). TTT plots: a perl program to create time-to-target plots. *Optimization Letters*, 1(4):355–366.

Akbari, V. and Salman, F. S. (2017). Multi-vehicle synchronized arc routing problem to restore post-disaster network connectivity. *European Journal of Operational Research*, 257(2):625–640.

Alarie, S. and Gamache, M. (2002). Overview of solution strategies used in truck dispatching systems for open pit mines. *International Journal of Surface Mining, Reclamation and Environment*, 16(1):59–76.

Altay, N. and Green III, W. G. (2006). OR/MS research in disaster operations management. *European Journal of Operational Research*, 175(1):475–493.

Angelelli, E. and Speranza, M. G. (2002). The periodic vehicle routing problem with intermediate facilities. *European Journal of Operational Research*, 137(2):233–247.

Archetti, C., Feillet, D., Gendreau, M., and Speranza, M. G. (2011). Complexity

of the vrp and sdvrp. *Transportation Research Part C: Emerging Technologies*, 19(5):741–750.

Archetti, C. and Speranza, M. G. (2008). The split delivery vehicle routing problem: a survey. In *The vehicle routing problem: Latest advances and new challenges* (Golden, Bruce L and Raghavan, Subramanian and Wasil, Edward A, eds.), pages 103–122. Springer.

Aronson, J. (1989). A survey of dynamic network flows. *Annals of Operations Research*, 20:1–66.

Barbalho, T. J., Santos, A. C., and Aloise, D. J. (2020). Metaheuristics for the work-troops scheduling problem. *International Transactions in Operational Research*.

Ben-Ameur, W. (2004). Computing the initial temperature of simulated annealing. *Computational Optimization and Applications*, 29(3):369–385.

Blazewicz, J., Lenstra, J. K., and Kan, A. R. (1983). Scheduling subject to resource constraints: classification and complexity. *Discrete Applied Mathematics*, 5(1):11–24.

Boonmee, C., Arimura, M., and Asada, T. (2018). Location and allocation optimization for integrated decisions on post-disaster waste supply chain management: On-site and off-site separation for recyclable materials. *International Journal of Disaster Risk Reduction*, 31:902–917.

Brucker, P., Drexl, A., Möhring, R., Neumann, K., and Pesch, E. (1999). Resource-constrained project scheduling: Notation, classification, models, and methods. *European Journal of Operational Research*, 112(1):3–41.

Çelik, M., Ergun, Ö., and Keskinocak, P. (2015). The post-disaster debris clearance problem under incomplete information. *Operations Research*, 63(1):65–85.

Coco, A. A., Duhamel, C., and Santos, A. C. (2020). Modeling and solving the multi-period disruptions scheduling problem on urban networks. *Annals of Operations Research*, 285(1-2):427–443.

Coelho, L. C., Cordeau, J.-F., and Laporte, G. (2014). Thirty years of inventory routing. *Transportation Science*, 48(1):1–19.

Coelho, L. C., Renaud, J., and Laporte, G. (2016). Road-based goods transportation: a survey of real-world logistics applications from 2000 to 2015. *INFOR: Information Systems and Operational Research*, 54(2):79–96.

Crevier, B., Cordeau, J.-F., and Laporte, G. (2007). The multi-depot vehicle routing problem with inter-depot routes. *European Journal of Operational Research*, 176(2):756–773.

Demir, E., Bektaş, T., and Laporte, G. (2012). An adaptive large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, 223(2):346–359.

Desrosiers, J., Laporte, G., Sauve, M., Soumis, F., and Taillefer, S. (1988). Vehicle routing with full loads. *Computers & Operations Research*, 15(3):219–226.

Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Dorigo, M. and Stützle, T. (2003). The ant colony optimization metaheuristic: Algorithms, applications, and advances. In *Handbook of Metaheuristics* (Gendreau, Michel and Potvin, Jean-Yves, eds.), pages 250–285. Springer.

Drexl, M. (2012). Synchronization in vehicle routing—a survey of vrps with multiple synchronization constraints. *Transportation Science*, 46(3):297–316.

Duque, P. A. M., Dolinskaya, I. S., and Sörensen, K. (2016). Network repair crew scheduling and routing for emergency relief distribution problem. *European Journal of Operational Research*, 248(1):272–285.

El Hachemi, N., Gendreau, M., and Rousseau, L.-M. (2011). A hybrid constraint programming approach to the log-truck scheduling problem. *Annals of Operations Research*, 184(1):163–178.

EM-DAT (2014). EM-DAT. The international disaster database. [Online]. Available from: `https://www.emdat.be/guidelines`. Accessed in 2021.

Ercelebi, S. G. and Bascetin, A. (2009). Optimization of shovel-truck system for surface mining. *Journal of the Southern African Institute of Mining and Metallurgy*, 109(7):433–439.

Fioroni, M. M., Franzese, L. A. G., Bianchi, T. J., Ezawa, L., Pinto, L. R., and de Miranda, G. (2008). Concurrent simulation and optimization models for mining planning. In *Simulation Conference, 2008. WSC 2008. Winter*, pages 759–767. IEEE, Miami, FL, USA.

Gendreau, M., Nossack, J., and Pesch, E. (2015). Mathematical formulations for a 1-full-truckload pickup-and-delivery problem. *European Journal of Operational Research*, 242(3):1008–1016.

Golden, B. L., Raghavan, S., and Wasil, E. A. (2008). *The vehicle routing problem: latest advances and new challenges*, volume 43. Springer Science & Business Media.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., and Moore, R. (2017). Google earth engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment.* Elsevier.

Grimault, A., Bostel, N., and Lehuédé, F. (2017). An adaptive large neighborhood search for the full truckload pickup and delivery problem with resource synchronization. *Computers & Operations Research*, 88:1–14.

Guastaroba, G., Speranza, M. G., and Vigo, D. (2016). Intermediate facilities in freight transportation planning: a survey. *Transportation Science*, 50(3):763–789.

Hartmann, S. and Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, 207(1):1–14.

Henderson, D., Jacobson, S. H., and Johnson, A. W. (2003). The theory and practice of simulated annealing. In *Handbook of Metaheuristics* (Gendreau, Michel and Potvin, Jean-Yves, eds.), pages 287–319. Springer.

Huang, Y., Santos, A. C., and Duhamel, C. (2020a). Bi-objective methods for road network problems with disruptions and connecting requirements. *Journal of the Operational Research Society*, 71(12):1959–1971.

Huang, Y., Santos, A. C., and Duhamel, C. (2020b). Model and methods to address urban road network problems with disruptions. *International Transactions in Operational Research*, 27(6):2715–2739.

ICSMD (2021). International Charter on Space and Major Disasters. [Online]. Available from: `https://www.disasterscharter.org`. Accessed in 2021.

Júnior, A. d. C. G., Souza, M. J. F., and Martins, A. X. (2005). Simulated annealing aplicado à resolução do problema de roteamento de veículos com janela de tempo. *TRANSPORTES*, 13(2):5–20.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.

Lacomme, P., Moukrim, A., Quilliot, A., and Vinot, M. (2019). Integration of routing into a resource-constrained project scheduling problem. *EURO Journal on Computational Optimization*, 7(4):421–464.

Li, S. and Teo, K. L. (2019). Post-disaster multi-period road network repair: Work scheduling and relief logistics optimization. *Annals of Operations Research*, 283(1):1345–1385.

López-Ibánez, M., Dubois-Lacoste, J., Cáceres, L. P., Birattari, M., and Stützle, T. (2016). The IRACE package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58.

Lorca, Á., Çelik, M., Ergun, Ö., and Keskinocak, P. (2017). An optimization-based decision-support tool for post-disaster debris operations. *Production and Operations Management*, 26(6):1076–1091.

Mancini, S. (2016). A real-life multi depot multi period vehicle routing problem with a heterogeneous fleet: Formulation and adaptive large neighborhood search

based matheuristic. *Transportation Research Part C: Emerging Technologies*, 70:100–112.

Miller, C., Tucker, A., and Zemlin, R. (1960). Integer programming formulations and traveling salesman problems. *Journal of the ACM*, 7:326–329.

Moreno, A., Munari, P., and Alem, D. (2019). A Branch-and-Benders-Cut algorithm for the crew scheduling and routing problem in road restoration. *European Journal of Operational Research*, 275(1):16–34.

Parragh, S. N., Doerner, K. F., and Hartl, R. F. (2008a). A survey on pickup and delivery models part ii: Transportation between pickup and delivery locations. *Journal für Betriebswirtschaft*, 58(2):81–117.

Parragh, S. N., Doerner, K. F., and Hartl, R. F. (2008b). A survey on pickup and delivery problems, part i: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58(1):21–51.

Pisinger, D. and Ropke, S. (2010). Large neighborhood search. In *Handbook of Metaheuristics* (Gendreau, Michel and Potvin, Jean-Yves, eds.), pages 399–419. Springer.

Potvin, J.-Y. and Rousseau, J.-M. (1993). A parallel route building algorithm for the vehicle routing and scheduling problem with time windows. *European Journal of Operational Research*, 66(3):331–340.

Pramudita, A. and Taniguchi, E. (2014). Model of debris collection operation after disasters and its application in urban area. *International Journal of Urban Sciences*, 18(2):218–243.

Pramudita, A., Taniguchi, E., and Qureshi, A. G. (2014). Location and routing problems of debris collection operation after disasters with realistic case study. *Procedia-Social and Behavioral Sciences*, 125:445–458.

Pritsker, A. A. B., Waiters, L. J., and Wolfe, P. M. (1969). Multiproject scheduling with limited resources: A zero-one programming approach. *Management Science*, 16(1):93–108.

Prodhon, C. and Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1):1–17.

Quanjing (2021). Quanjing. [Online]. Available from: `https://quanjing.com/`. Accessed in 2021.

Ribeiro, G. M. and Laporte, G. (2012). An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. *Computers & Operations Research*, 39(3):728–735.

Ropke, S. and Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4):455–472.

Sakuraba, C. S., Santos, A. C., Prins, C., Bouillot, L., Durand, A., and Allenbach, B. (2016). Road network emergency accessibility planning after a major earthquake. *EURO Journal on Computational Optimization*, 4(3-4):381–402.

Santini, A., Ropke, S., and Hvattum, L. M. (2018). A comparison of acceptance criteria for the adaptive large neighbourhood search metaheuristic. *Journal of Heuristics*, 24(5):783–815.

Sayarshad, H. R., Du, X., and Gao, H. O. (2020). Dynamic post-disaster debris clearance problem with re-positioning of clearance equipment items under partially observable information. *Transportation research part B: methodological*, 138:352–372.

SERTIT (2021). Regional Image Processing and Remote Sensing Service. [Online]. Available from: `https://sertit.unistra.fr/en/`. Accessed in 2021.

Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *International conference on principles and practice of constraint programming*, pages 417–431. Springer, Pisa, Italy, October 26-30.

Shin, Y., Kim, S., and Moon, I. (2019). Integrated optimal scheduling of repair crew and relief vehicle after disaster. *Computers & Operations Research*, 105:237–247.

Sinotruk (2021). Front dump truck. [Online]. Available from: `https://www.sinotrucks.net/product/Front-Dump-Truck-31.html`. Accessed in 2021.

Soares, R., Marques, A., Amorim, P., and Rasinmäki, J. (2019). Multiple vehicle synchronisation in a full truck-load pickup and delivery problem: A case-study in the biomass supply chain. *European Journal of Operational Research*, 277(1):174–194.

Solano-Charris, E., Prins, C., and Santos, A. C. (2015). Local search based metaheuristics for the robust vehicle routing problem with discrete scenarios. *Applied Soft Computing*, 32:518–531.

Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2):254–265.

Souza, M. J., Coelho, I. M., Ribas, S., Santos, H. G., and Merschmann, L. H. d. C. (2010). A hybrid heuristic algorithm for the open-pit-mining operational planning problem. *European Journal of Operational Research*, 207(2):1041–1051.

Subtil, R. F., Silva, D. M., and Alves, J. C. (2011). A practical approach to truck dispatch for open pit mines. In *35Th APCOM symposium*, pages 24–30.

Tan, Y. and Takakuwa, S. (2016). A practical simulation approach for an effective truck dispatching system of open pit mines using VBA. In *Winter Simulation Conference (WSC), 2016*, pages 2394–2405. IEEE, Arlington, VA, USA, December 11-14.

Tillman, F. A. and Cain, T. M. (1972). An upperbound algorithm for the single and multiple terminal delivery problem. *Management Science*, 18(11):664–682.

UNDP (2013). Technical guide for debris management - the Haitian experience 2010 - 2012. Technical report, United Nations Development Programme.

United Nations (2018). United Nations. World population in urban areas. [Online]. Available from: `https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html`. Accessed in 2021.

Wang, I.-L. (2018). Multicommodity network flows: A survey, part i: Applications and formulations. *International Journal of Operations Research*, 15(4):145–153.

Guilherme DE CASTRO PENA

Doctorat : Optimisation et Sûreté des Systèmes

Année 2021

Modèles et méthodes pour le problème intégré de planification et tournées appliqué aux catastrophes majeures

Le nettoyage des débris dans les zones urbaines après des catastrophes majeures est très important pour permettre aux habitants de se remettre de leurs effets. Lors de catastrophes, une zone inattendue et étendue peut être touchée. De plus, le temps et les coûts pour effectuer les opérations de nettoyage peuvent être très élevés. Cette thèse aborde le problème intégré multi-période de planification et de tournées de véhicules pour nettoyer les débris (SRP-CD) après des catastrophes majeures. Le problème comprend des décisions stratégiques (planification) et opérationnelles (tournées) et prend en compte des questions complexes telles que deux niveaux de synchronisation : entre les équipes de travail et les camions-bennes, et entre les camions-bennes nécessaires pour charger, transporter et décharger les débris pendant les journées de travail. L'objectif du SRP-CD est double : minimiser le nombre de jours pour le nettoyage global, au niveau stratégique, et minimiser le coût total de routage des véhicules, au niveau opérationnel. Un nouveau modèle mathématique basé sur une formulation dynamique multi-flot, des heuristiques constructives, des métaheuristiques basées sur la recherche par grand voisinage (LNS) et des métaheuristiques hybrides basées sur la LNS et le recuit simulé sont proposés. Des expérimentations pour le modèle et les approches sont réalisées, afin de mesurer la performance et la robustesse des méthodes proposées. A notre connaissance, ce sont les premières contributions dans la littérature pour le SRP-CD, incluant tous les aspects abordés dans cette thèse.

Mots clés : optimisation combinatoire – gestion des crises – logistique (gestion) – métaheuristiques – aide humanitaire.

Models and Methods for the Integrated Scheduling Routing Problem Applied to Major Disasters

Cleaning debris in urban areas after major disasters is very relevant to inhabitants to recover from their effects. In natural disasters, an unexpected and large area can be affected. Moreover, the time and the costs to perform the cleaning operations can be very high. In this work, the integrated multi-period scheduling routing problem to clean debris (SRP-CD) after major disasters is investigated. The problem includes strategical (scheduling) and operational (routing) decisions, considering complex issues such as two levels of synchronization between work-troops and dump trucks, and between dump trucks necessary to load, transport and unload debris during limited working days. The goal of SRP-CD is twofold: minimizing the number of days for the overall cleaning, in the strategical level; and minimizing the total costs of vehicle routes in the operational level. A new mathematical model based on a dynamic multi-flow formulation, constructive heuristics, Large Neighborhood Search (LNS)-based metaheuristics and hybrid metaheuristics based on LNS and Simulated Annealing are proposed. Comparison experiments for the model and the approaches are carried out to measure performance and robustness of the proposed methods. To the best of our knowledge, these are the first contributions in the literature for SRP-CD, including all the aspects addressed in this thesis.

Keywords: OR in disaster relief – crisis management – combinatorial optimization – metaheuristics – humanitarian logistics.

Thèse réalisée en partenariat entre :

Grand Est – Alsace Champagne-Ardenne Lorraine

Aube en Champagne – Le Département

utt – Université de Technologie Troyes

Ecole Doctorale "Sciences pour l'Ingénieur"