Solving the integrated multi-period scheduling routing problem for cleaning debris in the aftermath of disasters

Guilherme de Castro Pena[a,b,*], Andréa Cynthia Santos[c], Christian Prins[a]

[a] *Université de Technologie de Troyes, LIST3N, 12, rue Marie Curie, CS 42060, 10004, Troyes CEDEX, France.*
[b] *Universidade Federal de Viçosa - CRP-IEP, Km 7, MG-230, Rodoviário, 38810-000, Rio Paranaíba/MG, Brazil.*
[c] *Normandie Université, UNIHAVRE, UNIROUEN, INSA Rouen, LITIS, 25 Rue Philippe Lebon, 76600 Le Havre, France.*

Abstract

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 and, considering complex issues such as two levels of synchronization between work-troops and dump trucks. 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 vehicles routes in the operational level. A new mathematical model based in a dynamic multi-flow formulation, constructive heuristics and Large Neighborhood Search (LNS)-based metaheuristics 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 aspects addressed here.

Keywords: (O) OR in disaster relief, (O) Combinatorial optimization, (O) Metaheuristics, (O) Humanitarian logistics

1. Introduction

Several situations in big public engineering works and after major disasters (earthquakes, hurricanes, floods, etc) require an effort to clean the target urban area. The former can be planned in advance, while the latter, can be very complex since an unexpected and large urban area can be affected. However, cleaning urban areas after major disasters is very relevant for inhabitants to recover from their effects.

The world has been recently affected by several natural major disasters, such as India floods (2020), the rupture of dams in Brazil (2019), hurricanes in USA (2018), earthquakes in Nepal (2015), tsunamis and floods in Japan (2011), nuclear explosion in Japan (2011), earthquakes in Haiti (2010). When such disasters hit inhabited regions, they strongly impact population, environment and urban infrastructures.

*Corresponding author
Email address: guilherme.pena@ufv.br (Guilherme de Castro Pena)

In general, a disaster can be divided in four phases: mitigation, preparedness, response, and recovery (Altay & Green III, 2006).

In this study, we investigate the integrated multi-period Scheduling Routing Problem to clean debris (SRP-CD) in an urban area after major disasters. This problem appears in the recovery phase, which can take months or years. More precisely, we investigate the optimization of the cleaning operations in an urban area the fast as possible, considering limited resources such as work-troops (WT) and a fleet of dump trucks responsible to transfer debris to landfills.

The SRP-CD is related to three classical optimization problems: a close related problem to the Resource-Constrained Project Scheduling Problem (RCPSP), where within a project, tasks with resources requirements and processing times need to be completed in such a way the project completion date is minimal (Brucker et al., 1999; Hartmann & Briskorn, 2010). According to the amount of debris, two vehicle routing problems can address the SRP-CD : the Full Truckload Vehicle Routing Problem (FTVRP) (Desrosiers et al., 1988) and the Split Delivery Vehicle Routing Problem (SDVRP) (Archetti & Speranza, 2008; Golden et al., 2008). The former consists in designing routes for trucks in order to make a number of fully load trips. The latter aims at designing routes for vehicles such that customers 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 operational level relies on the FTVRP. In this case, the possible gains of using the SDVRP are 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 report of UNDP (2013) presented that, after two years, approximately 1 million of 10 million cubic meters had been removed.

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

(3) the last case occurs when the amount of debris are smaller than the vehicle capacity. Thus, there is no full truck trips and the SDVRP can be used. 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).

All the cases mentioned above rely on NP-Hard problems (Blazewicz et al., 1983; Archetti et al., 2011). Thus, the SRP-CD is 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, 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. We focus in this study, the first case (1), which allows us to produce the first integrated models and algorithms for this case, and taken into account the context of disasters.

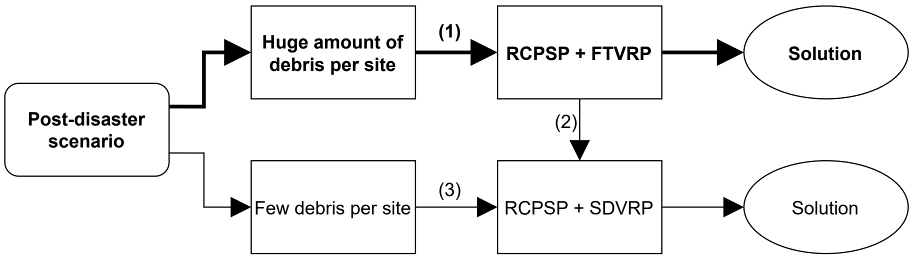


Figure 1: Classification of SRP-CD according to the amount of debris.

The integration of different decision levels in optimization problems is a growing trend in the scientific literature. Interesting entry points on the integration of vehicle routing problems are found in Coelho et al. (2014) for the inventory routing problem, and in Prodhon & Prins (2014) for the location-routing problems. Integrated problems are a challenging field since the global complexity of the integration can increases according to the complexity of each single problem. However, as showed in Prodhon & Prins (2014), the gains of addressing a problem in an integrated way are about 15%. Furthermore, 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, with two levels of synchronization between the assigned WTs and dump trucks and between dump trucks.

The main contributions of this study are the integration of the strategical (scheduling) and operational levels (vehicles routing) of the SRP-CD optimization problem, motivated by removing debris after major disasters, a new mathematical model based on a dynamic flow formulation, including two levels of synchronizations between work-troops and between dump trucks, and between dump trucks; two time discretizations, working days and time horizon; constructive heuristics and Large Neighborhood Search (LNS)-based metaheuristics. In addition, a data benchmark of interesting instances for SRP-CD is provided in this study.

The remaining of the document is organized as follows: a bibliographical review is given in Section 2. In the sequel, the problem is formally defined in Section 3, followed by a detailed description of the proposed methods in Section 4. The results and additional analysis are presented in Section 5. Then, concluding remarks and perspectives are provided in Section 7.

2. Related works

Studies related to the SRP-CD are found in classical optimization problems (see Section 2.1), applications in disaster relief, public works, and mining fields (see Section 2.2). Section 2.3 provides the position of our research from the existing studies from the literature.

2.1. Classical optimization problems related to SRP-CD

The SRP-CD considers aspects of different known classical optimization problems both in the strategical and operational levels. For the former, we can mention the RCPSP. For the latter, the vehicle routing problem is closely related to the FTVRP and the SDVRP.

As described in Brucker et al. (1999); Hartmann & Briskorn (2010), the RCPSP can be defined as a project containing a set of tasks which must be concluded without interruption after starting. Each task has resource requirements to be performed, a duration, and some of them may have predecessors. Objective functions like minimizing the completion time, the makespan, the total lateness for all tasks, among others can be used. The models look for finishing all tasks of the project such that resources and precedence constraints are satisfied. Compared to SRP-CD, the tasks in RCPSP could be seen as the debris nodes to be cleaned by the set of WTs and dump trucks. The resources could be the set of WTs together with dump trucks. Some differences exist between RCPSP and SRP-CD such as the integration with vehicle routing problem in SRP-CD, and the fact that SRP-CD is modeled on a graph due to the road network. One may note that RCPSP is not modeled on a graph, even if the precedence constraints could be defined as a graph.

The FTVRP related to the operational level of SRP-CD is defined as follows. Given a set of vehicles, the problem consists in defining routes for identical vehicles, which travel fully loaded from origins to destinations, starting and ending at the depot without exceed a time limit (Desrosiers et al., 1988). 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 to state a maximum route length, time windows and vehicle capacities. Another problem that can be used in the SRP-CD operational level is the SDVRP (Archetti & Speranza, 2008). It consists in serving a set of customers using a fleet of vehicles, which start and end their route at the depot, but the goods can be delivered (resp. 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. Both FTVRP and SDVRP can be independently integrated to the RCPSP, according to the amount of debris considered in the SRP-CD. Due to that, vehicles can only visit debris nodes with a WT assigned. Moreover, service times, synchronization between WT and dumps trucks, and between dump trucks are also take into account. The SRP-CD minimizes the total travel time for dump trucks as a second optimization level.

The study of Lacomme et al. (2019) addressed the RCPSP coupled to a Pickup and Delivery problem (PDP), where in the RCPSP, each activity of a project requires an amount of resources to be performed. Moreover, vehicles perform pickups and deliveries of resources between activities in order to minimize the makespan. The activities could be served by different vehicles and after starting an activity, the vehicles are allowed to visit others activities. A Mixed Integer Linear Programming (MILP) model, a constraint programming formulation, and a hybrid heuristic which couples a Greedy Randomized Adaptive Search Procedure and an evolutionary local search were proposed. The hybrid heuristic was able to find solutions on an average gap of 3.3% compared to optimal solutions for small instances. Compared to SRP-CD, this problem does not consider WTs-like entities for the activities, a minimization of vehicles travel time, services times and landfills.

The study of Grimault et al. (2017) investigates the Full Truck Load Pickup and Delivery Problem (FTPDP), with resource synchronization, motivated by an application in the context of road public work. The problem was described in a time horizon of one working day, where a set of heterogeneous trucks serve requests for material transportation (asphalt, gravel or waste). Trucks are assigned to a request only if it is compatible. Besides, service times, trucks fixed costs, travel costs and hourly costs are considered. The objective is 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 realistic 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 results. In the SRP-CD, the time horizon is composed of several working days and the fleet of dump trucks is homogeneous. Another difference lays in the objective function, since SRP-CD has two objective functions: the minimization of working days and the minimization of dump truck routes' costs.

The study of Soares et al. (2019) focused on the FTPDP with a multiple vehicle synchronisation. Three types of vehicles were used (trucks, loaders and lorries) to perform the material transportation within a planning horizon of a single working day. The problem has multi-depots and two types of synchronization: the first happens between lorries 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 minimizes the overall cost that couple transportation, fixed costs for vehicles, distance and time costs. A multi-flow based mathematical formulation and a matheuristic, using a fix-and-optimize with a variable neighbourhood decomposition search were proposed. The methods were tested over 19 instances. The mathematical formulation obtained gaps from 4.4% to 33%. Although, this application contains several close related issues with the operational decisions of SRP-CD, it does not integrate the strategical decisions to the operational ones.

2.2. Applications related to SRP-CD

A first set of applications related to the SRP-CD is the Crew Scheduling Routing Problem (CSRP). The study of Duque et al. (2016) addressed the network repair CSRP in post-disaster relief, focusing on a single repair crew. The goal is to minimize the time in which the blocked nodes become accessible. The idea was to start from a single depot to establish network connectivity and provide humanitarian relief. A

Dynamic Programming 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 Dynamic Programming approach was able to find optimal solutions for 225 out of 300 small instances in up to 24 hours, while the heuristic was able to find optimal solutions for 92,8% out of 225 small instances.

The CSRP investigated in Moreno et al. (2019) takes into account a single crew, with the goal of minimizing the time affected areas remain inaccessible. A mathematical formulation, inspired on the one of Duque et al. (2016) was presented. A branch-and-Benders-cut algorithm was proposed, where the problem was divided into a master problem for determinating the crew schedule, and two subproblems to check the feasibility and compute the schedule total cost. A simple greedy constructive heuristic and two local search heuristics were introduced. Another work focusing on the CSRP is 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, based on Duque et al. (2016) was proposed. The objective is to minimize the time which the last demand node was served. 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. Comparing to SRP-CD, the CSRP does not consider the two levels of synchronization, multi-periods, landfills, neither service times.

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 with 4 million metric tons of debris in a grid of 22 zones of equal area. The stochastic debris clearance problem was investigated in Çelik et al. (2015), which the main goal is to unblock roads in the response phase by moving the debris to the sides of the roads in order to reach other demand nodes. A Markov decision process 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 with 446 nodes and 604 road segments.

Other applications appear to clean road network in disaster relief. Some of them integrate the road network repair to logistics issues, as in Li & Teo (2019), while others integrate the repair road network scheduling to the reachability as in Sakuraba et al. (2016); Akbari & Salman (2017); Barbalho et al. (2020), or either to the network strong connectivity (Coco et al., 2020). In Li & Teo (2019), the problem was divided in two stages: a first stage where a multi-period scheduling problem with limited resources was solved; and a second one where routes for relief materials delivery were defined. The proposed mathematical formulation maximizes the network accessibility in the upper level and minimizes the time in which materials were delivery in the lower level. In addition, a hybrid heuristics based on a steady-state parallel genetic algorithm was also proposed. It was tested on theoretical instances varying from 25 to 1000 nodes, with 5% to 50% of damaged nodes and up to 34 repair crews; and on a case study derived of the Wenchuan earthquake in China.

The problem to unblock roads in order to reconnect the network, while minimizing the operation time, was addressed by Akbari & 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 are assigned to the road by using a waiting time. A MILP formulation, a matheuristic based on an MILP-relaxation, and a local search algorithm were proposed. Tests were done using theoretical and a realistic instance inspired by the Istanbul road network. According to the results, the matheuristic found optimal or near-optimal solutions.

In Sakuraba et al. (2016), the Work-troops Scheduling Problem was addressed. It consists in schedule repairs to improve the access to points where the population gather together. The authors proposed a MILP formulation and three heuristics, a savings, a ranking and a lexicographical one, to solve theoretical and a large-scale realistic instance of Port-au-Prince earthquake, 2010. The lexicographical heuristic performed better than the others and found an average relative deviation of less than 3% from optimal solutions. An extension of this work is found in Barbalho et al. (2020), where the authors proposed a dedicated local search, two metaheuristics, a Greedy Randomized Adaptive Search Procedure and an Iterated Local Search. The methods were applied to solve theoretical instances and the large-scale realistic instance of Haiti earthquake in 2010 from Sakuraba et al. (2016). The network design issues of these studies rely on urban network connectivity. A quite similar application in road public works is found in Coco et al. (2020), where the strong connectivity is required. It integrates a scheduling of planned disruptions with the network strong connectivity. An ϵ-constraint method was proposed and applied to instances of Troyes in France.

The Open Pit Mining (OPM) application consists in defining routes for material transportation from loading to dumping nodes in mining fields, with the goals of improving the amount of material transported by vehicles in order to reduce the total operational costs (Alarie & Gamache, 2002). 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. In the OPM, the differences from SRP-CD rely on the following issues: there are a fixed time to plan the operations, and the objective function stands 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 an unexpected and large area.

Although OPM was mainly solved by means of simulation, a few studies investigated the problem using an optimization approach. For instance, the study of Souza et al. (2010) proposed a MILP formulation to optimize the deviations of the production, quality goals and number of vehicles required for performing operations and a hybrid heuristic. 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 with low variability. In Ercelebi & Bascetin (2009), a queuing network strategy was proposed to optimize the number of truck assignments to shovels, which analyses all possible feasible paths between load and dump nodes. Then, a MILP 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.

2.3. Position of our research

Table 1 presents a summary of the closest related works to SRP-CD regarding the characteristics of the addressed problems found in the literature and the proposed approaches. The works are sorted by year and by applications. The dots in Table 1 means that the corresponding work includes the characteristic and the red dot represents SRP-CD characteristics. 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 SRP-CD characteristics.

3. Problem definition and mathematical model

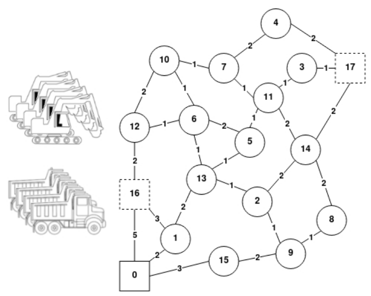
In SRP-CD, the transportation network is modeled on a connected undirected and simple graph $G = (V, E)$, where V is the set of n vertices and E is the set of m edges. Let $O \subset V$, $D \subset V$, $L \subset V$ be respectively, the single depot ($O = \{0\}$), the set of debris nodes and the set of landfills. 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.

		Characteristics																			Approaches			
		Depot		WT/Crew		Fleet		Material collection				Sync.				Working day		Goal						
Applications	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
	SRP-CD	•			•	•			•	•	•		•	•	•		•	•	•		•		•	
RCPSP-PDP	Lacomme et al. (2019)	•				•			•						•			•			•		•	
FTPDP	Soares et al. (2019)		•		•	•			•	•	•		•	•	•	•			•		•		•	
	Grimault et al. (2017)		•		•		•		•	•	•		•			•			•		•		•	
CSRP	Moreno et al. (2019)	•		•				•										•			•	•	•	
	Shin et al. (2019)	•		•				•										•			•		•	
	Duque et al. (2016)	•		•				•										•			•	•	•	
Debris clearance	Lorca et al. (2017)				•				•	•							•		•		•			•
	Çelik et al. (2015)				•			•									•			•	•		•	
Road network repairs	Barbalho et al. (2020)				•			•										•					•	
	Li & Teo (2019)		•		•			•									•			•	•		•	
	Akbari & Salman (2017)		•		•			•				•						•			•		•	
	Sakuraba et al. (2016)				•			•										•			•		•	
OPM	Souza et al. (2010)				•		•		•	•			•		•				•		•		•	
	Ercelebi & Bascetin (2009)				•		•		•	•			•		•				•		•	•		

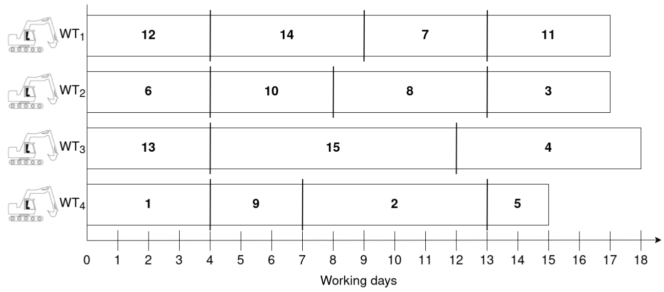
Table 1: Summary of the SRP-CD related works.

In the edge-set E, edge $[i, j]$ models a shortest path in terms of time linking nodes i and j in the real network, with pre-computed travel time $c_{ij} \in \mathbb{R}_+^*$, 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 are discretized by an one time unit, with a time limit $B = \{1...T\}$. Let W and $K = \{1, ..., k\}$ be respectively the number of WTs (each one is composed of bulldozers, excavators and the human resources) and a fleet of homogeneous dump trucks available at the depot, with a capacity Q each. The goal of the SRP-CD strategical level is to assign a set of WTs to debris nodes every time period, in the time horizon; while the operational level aims at defining synchronized multiple-trips for the fleet of dump trucks during each working day, such that dump trucks capacity is satisfied. Each WT is assigned to a vertex $i \in D$ if and only if there is no WT at node i, and a WT remains at node i until all debris is removed. Moreover, a WT starts a working day moving to a debris node and removing debris or removing debris if it is already assigned to a debris node. Concerning the dump trucks, they visit nodes $i \in D$ if and only if there is a WT at node i. In addition, loading and unloading operations have a fixed service time, which are represented respectively by $t_l, t_u \in B$. Dump trucks start and finish a working day at the depot. The objectives of SRP-CD are twofold: the primary objective is to minimize the total number of working days to clean all sites containing debris, and the second one looks for minimizing the dump trucks' total travel time.

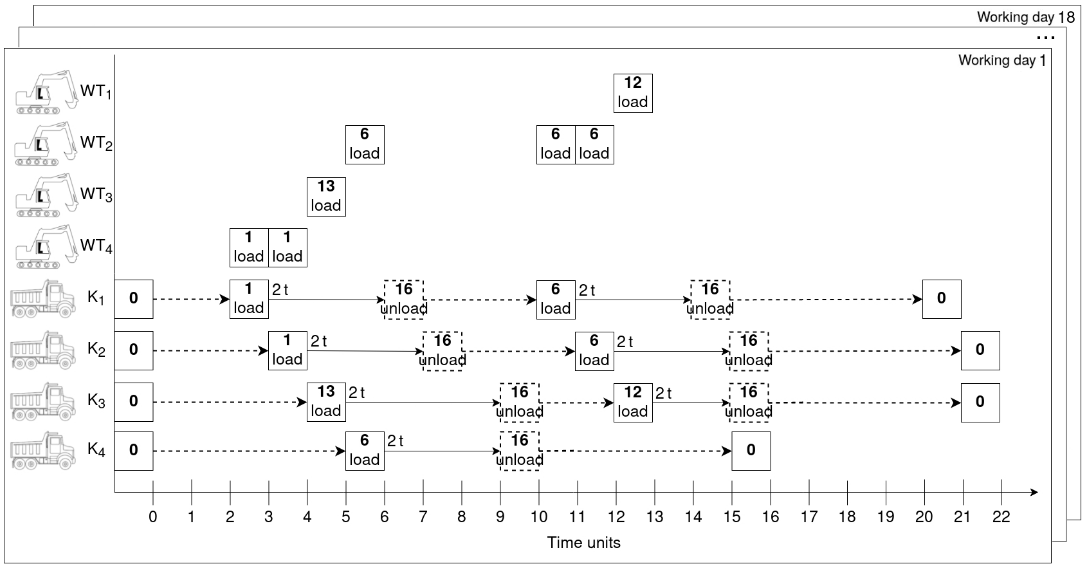
Figure 2 depicts an example of a scenario containing the following characteristics: $O = \{0\}$, $D = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15\}$ and $L = \{16, 17\}$, $W = 4$, $K = \{K_1, K_2, K_3, K_4\}$, $Q = 2$t, $T = 22$, $t_l = 1, t_u = 1$. The graph is presented in the Figure 2-(a), where the depot, each debris node and each landfill are identified respectively by a solid square, a solid circle and a dotted square. Moreover, travel times in time units are associated with the edges. The overall scheduling is detailed in Figure 2-(b), where the solution presents a total of 18 working days to clean all debris nodes. Figure 2-(c) shows the details of routing for the 4 dump trucks in "the first working day", where routes were assigned to debris nodes 1, 6, 12 and 13. Empty dump truck trips and full truck trips are shown, respectively, by dotted arrows and solid arrows. The solid square represents the depot or the loading operation in a debris node and the dotted square indicates the unloading operation in the landfill. The loaded debris, in tons, are shown in trucks' routes. In the working day, the dump trucks K_1 and K_2 were able to visit nodes 1 and 6 with synchronized routes, since two trucks cannot be loaded at same time unit. The dump truck K_3 visited nodes 13 and 12, and the dump truck K_4 visited the debris node 6. The times of each route can also be noticed in the figure including load and unload times. The dump truck K_1 has a route of 21 time units of which 16 time units for travel. The dump truck 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 2: Example for SRP-CD with (a) the graph G, (b) the scheduling of WTs and (c) the routing of dump trucks in the first working day.

3.1. Mathematical formulation

The proposed mathematical formulation is given in Equations (1) to (20). 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, and the more 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 time units. 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 dynamic multi-flow formulation described in this section is the first one found in the literature for SRP-CD. It has two fleets of homogeneous vehicles: one of WTs and another of dump trucks. The binary variables used in the mathematical formulation are defined as follows:

- x_i^h establishes if a WT is assigned to the node i in working day h ($x_i^h = 1$), otherwise ($x_i^h = 0$).
- y_{ij}^h determines if a WT visits the node j after i in the beginning of working day h ($y_{ij}^h = 1$),

otherwise ($y^h_{ij} = 0$).

- f^{th}_{ijk} states if the dump truck k arrives at node j from i in time period t of working day h ($f^{th}_{ijk} = 1$), otherwise ($f^{th}_{ijk} = 0$).

In addition, variables $0 \leq l^{th}_{ik} \leq Q$ determine the accumulated load of each dump truck k at node i in time period t and in working day h; while, variables $0 \leq u^{th}_{ik} \leq Q$ state the amount of debris removed at i loaded in the dump truck k, in time period t and in working day h. The variable $\delta \geq 0$ represents a number of working days used in the time horizon. The model is composed of two objectives functions $\mathcal{F}_1$ and $\mathcal{F}_2$ (resp. given in Equations (1) and (2)), which are performed in two optimization phases. The primary one is $\mathcal{F}_1$ which minimizes the number of days to clean all sites, and the second objective function $\mathcal{F}_2$ optimizes the total travel time of dump trucks, satisfying the primary objective. The result of the first execution ($\mathcal{F}_1$) becomes a new time horizon $A' = \{1...H'\}$ for the second optimization phase, where the model is run with the function $\mathcal{F}_2$, without the constraints (3).

Table 2 summarizes all notations and variables of the problem.

The main assumptions for the SRP-CD are:

- Each debris node has at most one WT assigned. This is considered to provide more visit points for dump trucks.

- The WT remains at the debris node until it is totally cleaned. Without such assumption, WTs could be assigned in different debris nodes every day before finishing to clean it completely. This is not a suitable situation in the context of disasters.

- A WT moves to another debris node in a different working day. 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.

- Dump trucks visit only debris nodes with WT. This assumption is trivial since dump trucks are loaded by WTs in the problem.

- Landfill nodes are considered with unlimited capacity.

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

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

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

	Symbol	Description
Sets	$G = (V, E)$	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 $[i, j]$, $(i, j \in V)$
	$K = \{1, ..., k\}$	Fleet of homogeneous dump trucks
	$A = \{1, ..., H\}$	Time horizon containing H working days
	$B = \{1, ..., T\}$	Time discretization for a working day containing T time periods
Parameters	W	Number of WTs available
	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 $[i, j] \in E$, $(c_{ij} \in \mathbb{R}^*_+)$
	t_l	Service time for a loading operation, $(t_l \in B)$
	t_u	Service time for an unloading operation, $(t_u \in B)$
Variables	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
Objectives	$\mathcal{F}_1$	Objective function for the model in the first optimization phase
	$\mathcal{F}_2$	Objective function for the model in the second optimization phase

Table 2: Notations and variables used in the SRP-CD formulation.

Constraints (3) together with the objective function $\mathcal{F}_1$ determine the minimum number δ of working days necessary to clean all debris nodes, where h represents every working day from 1 to day H. 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. As mentioned before, this is not an appropriate situation for the application focused in this study.

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

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

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

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

Inequalities (4) guarantee that W WTs leave the depot only once in all the time horizon. Equalities (5) link variables x to y and ensure the assignment of WTs to a debris node i if there is a flow entering node i. Such WTs can come from the depot or from a debris node. Constraints (6) ensure the flow conservation for WTs. The general idea remains similar to a classical flow conservation constraints, where all incoming flow to a debris node has to leave such a debris node. The main difference relies on the indexation over the time. Precisely, the outgoing flow has a date superior to the date of the incoming flow by considering the next working day. Constraints (7) determine that exactly one WT is assigned to each debris node at some working day.

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

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

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

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

Inequalities (8) establish that in each working day, every dump truck can be used or not, the decision of using dump trucks is left for the model. Constraints (9) state the flow conservation for dump trucks in each working day. As for the WTs, the flow is indexed over the time. The value s corresponds to the sum of the traveling time (c_{ij}) and the service time (resp. t_l time to load at a debris node and time t_u to unload at a landfill). This allows controlling the date of incoming and outgoing flows accordingly.

Inequalities (10) link variables f_{ijk}^{th} and x_i^h. They ensure that a flow enters a debris node i if and only if there is a WT assigned to i. Constraints (11) guarantee the synchronization between dump trucks. For instance, if a truck (k) is loading at a debris node, 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 and outgoing flows.

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

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

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

Inequalities (12) determine the accumulated amount of debris, loaded at a dump truck. Such constraints are inspired on the well known Miller, Tucker and Zemlin constraints for the travelling salesman problem (Miller et al., 1960). One may note that the variables are indexed by the period time t in a working day h. Thus, the load l in the period $t+s$ of h will be greater than the actual l charge plus the new load u, if $f_{jik}^{(t+c_{ji})h} = 1$.

Inequalities (13) establish the dump trucks capacity is satisfied. Constraints (14) guarantee that all debris node have to be cleaned until the last working day used.

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

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

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

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

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

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

The domain of the variables are given from (15) to (20).

4. Heuristic methods

We propose heuristics and LNS-based metaheuristics to solve the SRP-CD, detailed respectively in Sections 4.1 and 4.2.

4.1. Constructive heuristics

The proposed constructive heuristics are composed of four main steps: (a) the assignment of WTs to debris nodes, (b) the routing of dump trucks to debris nodes already with WTs, (c) the loading of dump trucks at the debris nodes and (d) the unloading of dump trucks at the landfills. The step (a) focuses on the strategical decisions of the WTs scheduling, while the steps (b), (c) and (d) are related to operational decisions of dump trucks' trips. The steps are performed by a procedure called *buildingSolution()*.

The heuristics perform a greedy or a random construction in the steps (a) and (b), according to a given criterion. The step (c) performs load at dump trucks in each node, and in the step (d), full-loaded dump trucks are routed to the nearest landfill from the corresponding debris node. Six criteria are proposed for the strategic level of decision and seven for the operational one, on a total of $6 \times 7 = 42$ criteria combinations. In addition, a random criterion is also used for the two levels. All of these combinations were considered in the numerical experiments.

4.1.1. Greedy and random constructive heuristics

The six greedy criteria proposed are as follows:

- Less Debris First (LDF) : 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) : WTs are assigned in non increasing order of the higher amount of debris.
- Smaller Travel Time First (STTF) : 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) : WTs are assigned to the most distant debris nodes from the depot, considering a non increasing order of travel time.
- Smaller Ratio (Debris/Travel Time) First (SDTTF) : a ratio between the amount of debris and the travel time is done, and then WTs are assigned in increasing order of debris nodes with the smaller ratio.
- Greater Ratio (Debris/Travel Time) First (GDTTF) : a ratio between the amount of debris and the travel time is done, and then 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. In addition, the Less Trucks First (LTF) criterion is also used in the operational level and consists in routing dump trucks to debris nodes with less trucks already allocated. This is done in increasing order of the dump trucks number, working in a debris node. The algorithms using the aforementioned criteria have a similar structure, which is provided in Algorithm 1. The best solution s, the incumbent solution s', and the criteria sets C1 and C2 are initialized in lines 1-3. The loops of lines 4-8 and 5-8 vary the set of criteria and build the corresponding deterministic solution for every combination of criteria. The criterion for WTs is referred to as cWT, and for the dump trucks is referred to as cK. In lines 7-8, according to 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 9. This heuristic is referred to as Greedy Constructive Heuristic (GCH).

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. Furthermore, RCH includes a stopping-criterion, that can be considered for instance, a fixed number of iterations, a fixed execution time or a fixed number of iterations without any improvements. The algorithm slight differs from the one of GCH in the following points. Lines 2 and 3 are removed, the loops are replaced for the stopping-criterion loop and in line 6, cWT and cK are set to a random choice of debris nodes.

Algorithm 1: Greedy constructive heuristics structure

Data: G=(V,E)

Result: s

1 $s \leftarrow \infty$, $s' \leftarrow \emptyset$

2 $C1 \leftarrow \{LDF, MDF, LDF, GDF, SDTTF, GDTTF\}$

3 $C2 \leftarrow \{LDF, MDF, LDF, GDF, SDTTF, GDTTF, LTF\}$

4 **forall** $cWT \in C1$ **do**

5   **forall** $cK \in C2$ **do**

6         $s' \leftarrow buildingSolution(G, cWT, cK)$

7   **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**

8             $s \leftarrow s'$

9 return s

4.2. LNS-based metaheuristics for SRP-CD

LNS metaheuristics proposed by Shaw (1998), has been successful applied for difficult NP-problems (Pisinger & Ropke, 2010). It moves the solution in the search space by performing removals and insertions. The use of LNS for SRP-CD is motivated by its simplicity, and according to the literature, by the high quality of solutions obtained in solving complex problems and in addressing difficult constraints. 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 LNS proposed here for the SRP-CD is as follows. The initial solution applies a greedy or a random heuristic described in Section 4.1. Moreover, LNS makes use of a removal procedure that takes away a random debris node of the current solution at a time. It is worth mentioning that removing a debris implies that a set of vehicles' routes need to be renewal. Two strategies are used to insert a debris node into a solution: First Improvement (FI) and Best Improvement (BI). In the former, when a first insertion improves the current solution, it is updated; while, in the latter, all possible insertions are checked and the best improvement one are used to update the current solution. The strategy, either *BI* or *FI*, is a parameter in the LNS algorithm, referred to as LS. Furthermore, after a remove-insertion operations, a solution is accepted if and only if it improves the incumbent one. For this reason, LNS for SRP-CD has also a cut procedure to optimize the construction of solution after an insertion. During the construction, the procedure rejects the new partial solution if the scheduling cost is greater than the one of the incumbent solution.

A general LNS scheme is given in the Algorithm 2. Parameters are initialized in line 1 (resp. the best solution, the incumbent solution and an auxiliary one), then a feasible initial solution is generated in line 2. In the sequel, the best known solution is set in line 3. The loop 4 to 11 is repeated until the predefined stopping-criterion is met. An auxiliary solution is initialized in line 5. The algorithm performs removals in line 6 and insertions in line 7. Condition, in lines 8 and 9, updates the best solution found so far. In the following, the acceptance criterion is used to update the incumbent solution whenever it accepts only improved ones.

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.

5. Computational experiments

The computational experiments 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 and the instances generator 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.

Algorithm 2: LNS-based metaheuristic structure

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

Result: s

$s \leftarrow \emptyset$, $s' \leftarrow \emptyset$, $s'' \leftarrow \emptyset$

$s' \leftarrow buildingSolution(G, cWT, cK)$

$s \leftarrow s'$

while stopping-criterion not met **do**

  $s'' \leftarrow s'$

  $x \leftarrow remove(s'')$

  $insert(s'', x, cK', LS)$

  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**

    $s \leftarrow s''$

  if $accept(s'', s')$ **then**

    $s' \leftarrow s''$

return s

5.1. Data generation

We developed an instance generator, 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). This results in three network topologies, referred here as (C)luster, (R)andom, and (M)ix. The Euclidean distance is associated with each edge representing 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 to 7 areas, and the axis represent time units. Figure 3 shows three examples of instances generated by the algorithm with the depot in blue, the landfills in red and the debris nodes in green. Figure 3-(a) presents an instance with a cluster topology containing 80 debris nodes and 4 landfills, Figure 3-(b) depicts an instance with a random topology containing 30 debris nodes and 4 landfills, and Figure 3(c) shows an instance with the mix topology containing 180 debris nodes and 7 landfills.

Two set of instances, S1 and S2, were created from medium- to large-size instances. For each set, 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 3, 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 load (t_l), unload (t_u) and the working day (T).

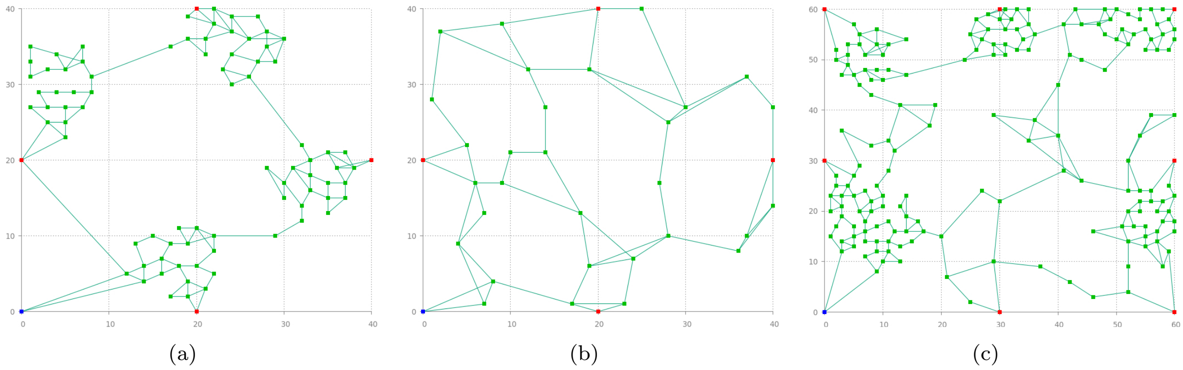


Figure 3: Example of generated instances with different network topologies.

	# Inst.	$\|D\|$	$\|L\|$	W	$\|K\|$	Q	t_l t_u	T
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 3: Characteristics of the instance sets.

5.2. Numerical results

The benchmarks of instances 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. The greedy constructive heuristics are compared in Section 5.2.2. In the following, the tuning experiments for the metaheuristics' parameters are presented in Section 5.2.3. In Section 5.2.4, the experiments and analysis using all instances are detailed.

5.2.1. Mathematical model results

For the sake of clarity, the value of H in the model is computed for each instance using the Equation (21), where γ is pre-computed as follows. Given all shortest paths between origins and destinations in G, γ is set to the value of the biggest one.

$$H = \sum_{i \in D} v_i / [(|K| \times Q \times T)/\gamma] \tag{21}$$

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 mathematical formulation has been validated with 15 small instances (resp. 5 instances for each topology, C, R and M) with $|D| = 5$, $W = 2$ and $|K| = 3$, for which it was able to prove optimally for the two objective functions. But, the most relevant is to show the limits of the model. For this purpose, we took the 18 small instances (resp. 6 for each topology) of S1 with $|D| = 10$ and provide the results in Table 4. The first three columns present the instances characteristics: topology type, number W of WTs and number $|K|$ of dump trucks. 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.

The mathematical formulation was able to prove optimality for the first objective function for 7 out of 18 instances. Obviously, whenever 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. Columns without results indicates that the model did not find any feasible solution.

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 4: Results for the mathematical model in the instances with 10 debris nodes.

Sensitivity analysis have been performed to investigate the impact of some parameters on solving SRP-CD. For such a purpose, one parameter is varied at a time, such as the number of dump trucks, the number of WT, and the volume of debris; while the other parameters are fixed. Table 5 presents the results by varying the number of dump trucks. As $|K|$ increases, the value of $\mathcal{F}_1$ decreases due the fact that a higher capacity (more dump trucks) is available to remove debris in a day. In addition, the value of $\mathcal{F}_2$ tends to increase due to more trips involved in leaving and returning to the depot every day.

Table 6 shows the results by varying the number of WTs. As W increases, the values of $\mathcal{F}_1$ and $\mathcal{F}_2$ tends to decrease. Since more critical resources are available, $t(s)$, GAP and Nb_nodes tends to be reduced with some exceptions as the instance with Mixed topology, $W = 4$ and $|K| = 8$. A possible explanation is that other resources as dump trucks have limited the general resolution for such exceptions.

Table 7 depicts the results by varying the volume of debris. As expected, this parameter makes the problem difficult to be solved. This can be observed by the higher GAP values of $\mathcal{F}_1$, accordingly with the increase on the volume of debris. The main reasons are that under limited resources as dump trucks and WTs, more debris might improve the combinatorial choices, and require a bigger time horizon. This latter aspect increases the number of variables and constraints for a mathematical formulation indexed over the time. A particular situation has also been raised in this experiment, where $\mathcal{F}_2$ meets an optimal cost, but not $\mathcal{F}_1$. Naturally, the corresponding solution is not optimal.

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.
	2	2	19	7200	10.53	102621	160421	238441	1052	7200	4.28	117100	372580	238251
C	2	4	10	7200	10.00	196977	317461	476081	1072	7201	4.12	53730	741780	475981
	2	8	6	7200	16.67	96676	631541	951361	1096	7201	2.99	87849	1480180	951301
	2	2	21	7200	9.52	203639	154501	236021	1184	7201	4.10	773956	358500	235811
R	2	4	11	7200	9.09	327932	305751	471271	1184	7202	4.10	395484	713750	471161
	2	8	**6**	337	**0.00**	658	608251	941771	1220	7200	1.72	280250	1424250	941711
	2	2	18	7200	16.67	129199	130327	198597	964	7201	5.29	804928	301686	198417
M	2	4	10	7200	20.00	137853	257923	396543	1028	7200	11.19	107320	600642	396443
	2	8	6	7200	16.67	174056	513115	792435	1029	7200	4.81	87297	1198554	792375

Table 5: Evaluation of varying $|K|$ in the instances with 10 debris nodes.

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.
	2	8	6	7200	16.67	96676	631541	951361	1096	7201	2.99	87849	1480180	951301
C	4	8	**5**	250	**0.00**	0	631541	951361	1060	7201	4.51	249279	1480180	951311
	8	8	**5**	167	**0.00**	0	631541	951361	1052	7201	1.14	852970	1480180	951311
	2	8	**6**	337	**0.00**	658	608251	941771	1220	7200	1.72	280250	1424250	941711
R	4	8	**5**	227	**0.00**	2018	608251	941771	-	-	-	-	-	-
	8	8	**5**	72	**0.00**	0	608251	941771	1184	7201	0.76	903430	1424250	941721
	2	8	6	7200	16.67	174056	513115	792435	1029	7200	4.81	87297	1198554	792375
M	4	8	5	7208	20.00	1264228	513115	792435	964	7201	3.79	1131294	1198554	792385
	8	8	**4**	124	**0.00**	0	513115	792435	**964**	22	**0.00**	1948	1198554	792395

Table 6: Evaluation of varying W in the instances with 10 debris nodes.

5.2.2. Greedy constructive heuristics comparison

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

- The Abs-Rank gives the 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

Instances		$\mathcal{F}_1$						$\mathcal{F}_2$					
	Volume	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.	UB	t(s)	GAP	Nb_nodes	Nb_var.	Nb_const.
C	200	37	7200	53.06	15580	320841	476861	2030	7200	3.82	21517	745160	476491
	400	81	7200	91.92	1176	641681	953701	3846	7201	1.14	6549	1490320	952891
	600	132	7200	94.62	0	962521	1430541	5763	7200	1.12	1352	2235480	1429221
R	200	40	7200	30.00	12137	309001	472021	**2271**	177	**0.00**	0	717000	471621
	400	88	7200	90.91	2828	618001	944021	4542	7201	2.40	19479	1434000	943141
	600	-	-	-	-	-	-	-	-	-	-	-	-
M	200	37	7200	38.42	15617	260653	397173	1826	7200	4.96	51150	603372	396803
	400	80	7204	92.50	3098	521305	794325	3600	7201	4.00	2271	1206744	793525
	600	-	-	-	-	-	-	-	-	-	-	-	-

Table 7: Evaluation of varying the volume of debris in the instances with 10 debris nodes, considering $W = 2$ and $|K| = 2$.

more heuristics finding the same best solution value, both will be rewarded. The sum of all ranking for all instances provide how many times a heuristic found the best solution value among all the others for a set of instances.

- 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 heuristics find a 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, both will have 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 ranking analysis for each set of instances are presented in Table 8, where the best results are in bold. In the cell K\W, W indicates the criteria for work-troops (scheduling), and K indicates the criteria for dump trucks (routing).

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.

To summarize, the highlighted heuristics of the four matches LDF, MDF, STTF and GDTTF are used in the metaheuristics tuning. This allowed us to reduce the number of possible combinations.

	K \ W	Abs-Rank LDF	MDF	STTF	GTTF	SDTTF	GDTTF	Avg-Rank 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 8: Ranking analysis for greedy criteria in both sets.

5.2.3. Metaheuristics tuning

The metaheuristics tuning were done using the Iterated racing for automatic algorithm configuration (IRACE) package (López-Ibánez et al., 2016), which works with a machine learning mechanism. The set of instances S1 was used to tune the following parameters:

(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.

(b) two types of local search: BI and FI.

(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. Results are presented in Table 9.

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 9: Parameters calibrated by 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.

5.2.4. Heuristics comparison

The tuning results described in Section 5.2.3 are used in these experiments, where the goals are to analyse performance of the proposed methods: *GCH*, *RCH* (using a fixed execution time of 20 seconds as stopping-criterion), *LNS-GG* and *LNS-RG*. For the methods *RCH*, *LNS-GG* and *LNS-RG*, the results consider the best outcome of 10 different runs for each instance, using different seeds.

The results for S1 and S2 are showed in the Tables 10, 11 and 12 respectively for instances with cluster, random and mix topologies. As mentioned in Section 5.1, S1 and S2 contain 90 instances each, resulting in a total of 180 instances. In Tables 10, 11 and 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 best results are highlighted in boldface.

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). A deviation and a relative deviation are computed between the obtained solution and the best known one respectively for the first and the second objectives, as indicated in Equations (22) and (23).

$$Dev = \mathcal{F}_1^m - \mathcal{F}_1^b \tag{22}$$

$$Dev(\%) = \frac{\mathcal{F}_2^m - \mathcal{F}_2^b}{\mathcal{F}_2^b} \times 100 \tag{23}$$

The two first columns in Tables 10, 11 and 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; α shows the *Dev* average for 6 instances of each line; β provides the *Dev(%)* average of 6 instances for each line, and *t(s)* gives the average running time in seconds. A method is considered better according to a lexicographical order of *Nb_best*, α and β.

Table 10 presents the results for the cluster topology. According to the results, *LNS-RG* found 70% of the b.k. solutions, with an average deviation for the scheduling and routing of respectively 0.40 working days and 0.53% for the total travel time of dump trucks, and with an average execution time of 300.06 seconds. *LNS-GG* reached about 38% of the b.k. solutions, with an average deviation of 0.48 working days and 1.67% for the total travel time, and an average execution time of 346.78 seconds. The method *RCH* obtained about 7% of the b.k. solutions and the *GCH* did not find any b.k. solution.

In Table 11, the results for random topology are depicted. Regarding the results, *LNS-RG* achieved about 67% of the b.k. solutions, with an average deviation of 0.55 for the working days. It obtained 1.13% for the total travel time of the dump trucks, consumed an average execution time of 259.90 seconds. *LNS-GG* obtained 35% of the b.k. solutions, with an average deviation of 0.68 for the working days and 2.60% for the total travel time, and with an average execution time of 273.20 seconds. *RCH* found 10% of the b.k. solutions and *GCH* did not find any b.k. solution.

Instances	$\|D\|$	GCH Nb_best	GCH α	GCH β	GCH t(s)	RCH Nb_best	RCH α	RCH β	RCH t(s)	LNS-GG Nb_best	LNS-GG α	LNS-GG β	LNS-GG t(s)	LNS-RG Nb_best	LNS-RG α	LNS-RG β	LNS-RG t(s)
S1	10	0	0.33	1.72	~ 0	4	0.00	0.06	20	1	0.33	0.59	~ 0	5	0.00	0.04	20
	20	0	2.17	7.39	~ 0	0	0.83	2.78	20	6	0.00	0.00	0.03	3	0.17	0.40	20
	30	0	1.17	2.74	~ 0	0	0.33	1.38	20	6	0.00	0.00	0.15	0	0.33	1.38	20
	40	0	4.67	7.64	~ 0	0	3.50	5.71	20	4	0.33	0.07	0.30	4	0.17	0.22	20
	50	0	6.67	6.94	~ 0	0	3.67	3.79	20	6	0.00	0.00	0.63	0	3.33	3.27	20
S1: Avg. [Total]		[0]	3.00	5.29	~ 0	[4]	1.67	2.75	20	**[23]**	**0.13**	**0.10**	0.22	[12]	0.80	1.06	20
S2	100	0	0.50	7.74	~ 0	0	0.83	16.56	20	0	0.17	1.46	8.35	6	0.00	0.00	27.12
	200	0	1.00	9.77	~ 0	0	2.50	24.09	20	0	0.33	3.03	69.38	6	0.00	0.00	88.47
	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
	400	0	2.67	8.59	0.17	0	7.50	23.39	20	0	1.17	3.04	920.88	6	0.00	0.00	799.30
	500	0	3.33	8.15	~ 0	0	8.50	23.26	20	0	1.33	4.34	2136.78	6	0.00	0.00	1727.65
S2: Avg [Total]		[0]	1.93	9.25	0.07	[0]	4.70	21.92	20	[0]	0.83	3.23	693.34	**[30]**	**0.00**	**0.00**	580.12
C: Avg. [Total]		[0]	2.47	7.27	0.03	[4]	3.18	12.34	20	[23]	0.48	1.67	346.78	**[42]**	**0.40**	**0.53**	300.06

Table 10: Results for the approaches in the instances with cluster topology.

Instances	$\|D\|$	GCH Nb_best	GCH α	GCH β	GCH t(s)	RCH Nb_best	RCH α	RCH β	RCH t(s)	LNS-GG Nb_best	LNS-GG α	LNS-GG β	LNS-GG t(s)	LNS-RG Nb_best	LNS-RG α	LNS-RG β	LNS-RG 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
	20	0	2.67	8.98	~ 0	0	1.17	4.36	20	2	0.17	0.56	0.07	4	0.00	0.49	20
	30	0	3.50	10.29	~ 0	0	2.17	7.06	20	6	0.00	0.00	0.15	0	0.50	1.63	20
	40	0	5.50	7.21	~ 0	0	3.50	4.72	20	6	0.00	0.00	0.28	0	2.17	3.82	20
	50	0	7.50	11.09	~ 0	0	4.33	6.53	20	5	0.17	0.07	0.55	1	2.83	5.23	20
S1: Avg. [Total]		[0]	4.03	8.23	~ 0	[6]	2.23	4.54	20	**[20]**	**0.13**	**0.39**	0.22	[11]	1.10	2.23	20
S2	100	0	0.67	9.35	~ 0	0	1.00	18.45	20	1	0.17	2.08	8.12	5	0.00	0.10	26.35
	200	0	1.33	9.91	~ 0	0	2.17	19.91	20	0	1.33	6.60	58.78	6	0.00	0.00	75.20
	300	0	2.33	11.69	0.17	0	4.17	22.17	20	0	1.83	5.94	266.47	6	0.00	0.00	185.58
	400	0	2.00	6.92	~ 0	0	6.00	19.80	20	0	1.17	4.25	701.35	6	0.00	0.00	648.50
	500	0	2.83	8.64	~ 0	0	8.50	23.26	20	0	1.67	5.13	1696.15	6	0.00	0.00	1563.38
S2: Avg. [Total]		[0]	1.83	9.30	0.03	[0]	4.37	20.72	20	[1]	1.23	4.80	546.17	**[29]**	**0.00**	**0.02**	499.80
R: Avg. [Total]		[0]	2.93	8.77	0.02	[6]	3.30	12.63	20	[21]	0.68	2.60	273.20	**[40]**	**0.55**	**1.13**	259.90

Table 11: Results for the approaches in the instances with random topology.

Table 12 shows the results for the mix topology. Concerning the results, *LNS-RG* obtained 75% of the b.k. solutions an average deviation of 0.30 for working days and 1.32%. In addition, it obtained an average execution time of 303.85 seconds. *LNS-GG* achieved about 33% of the b.k. solutions, with an average deviation of 1.23 working days and 2.84% for the total travel time, in an average execution time of 344.25 seconds. *RCH* obtained about 13% of the b.k. solutions and *GCH* did not find any b.k. solution.

We noticed that the improvements obtained using the *LNS-RG* and *LNS-GG* methods when compared with the other methods become very relevant whenever the instances increase and in particular for S2. For instance, on average, the solutions obtained by the *LNS-RG* are for the cluster topology up to 4.70 better that the others considering the working days, and about 21.92% of total travel time. A quite similar result is found for the other topologies: improvements of up to 4.37 working days and 20.72% of total travel time for the random topology, and improvements of up to 4.73 working days and 22.70% of total travel time for mix topology. Not surprisingly, the metaheuristics *LNS-RG* and *LNS-GG* require more execution time. Their average time increases with the instance size, and the number of iterations in the stopping-criterion. The more time consuming instances were the ones with $|D| = 500$, with up to 2241.28 seconds for the instances with mix topology.

Instances		GCH				RCH				LNS-GG				LNS-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)
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
	20	0	1.00	4.06	~ 0	0	0.50	2.79	20	3	0.00	0.28	0.08	5	0.00	0.15	20
	30	0	2.83	9.68	~ 0	0	1.67	7.48	20	5	1.33	2.72	0.17	1	0.67	4.84	20
	40	0	4.50	13.96	~ 0	0	3.00	11.45	20	3	1.00	5.89	0.35	3	0.33	1.95	20
	50	0	5.50	11.17	~ 0	0	2.83	7.65	20	6	0.00	0.00	0.68	0	2.00	6.29	20
S1: Avg. [Total]		[0]	2.93	9.00	~ 0	[6]	1.60	5.87	20	**[20]**	**0.50**	**2.01**	0.26	[15]	0.60	2.65	20
S2	100	0	0.50	11.85	~ 0	0	1.00	21.93	20	0	0.17	2.29	8.72	6	0.00	0.00	26.92
	200	0	1.67	10.14	~ 0	0	2.17	20.11	20	0	1.00	2.80	101.55	6	0.00	0.00	73.58
	300	0	3.50	3.74	~ 0	2	4.00	14.30	20	0	5.00	3.57	213.85	6	0.00	0.00	197.00
	400	0	3.67	11.83	0.17	0	7.67	28.93	20	0	1.83	3.58	875.78	6	0.00	0.00	833.32
	500	0	3.83	12.07	~ 0	0	8.83	28.25	20	0	1.83	6.14	2241.28	6	0.00	0.00	1807.63
S2: Avg. [Total]		[0]	2.63	9.93	0.03	[2]	4.73	22.70	20	[0]	1.97	3.68	688.24	**[30]**	**0.00**	**0.00**	587.69
M: Avg. [Total]		[0]	2.78	9.46	0.02	[8]	3.17	14.29	20	[20]	1.23	2.84	344.25	**[45]**	**0.30**	**1.32**	303.85

Table 12: Results for the approaches in the instances with mix topology.

To summarize, it is noticeable that *LNS-GG* works better for the S1 instances, while the random aspects play a key role for large ones. This is supported by the results obtained for S2 instances considering the three topologies, where the *LNS-RG* results overcome the other heuristics.

For the sake of reproducibility, Table 13 depicts the best known solutions found among the proposed methods for each instance. For S1 and S2, the inner columns refer to the instances characteristics and the topologies. For each topology the left and right values correspond, respectively, the number of working days ($\mathcal{F}_1$) and the total travel time ($\mathcal{F}_2$). The values in boldface represent the optimum for $\mathcal{F}_1$.

S1											S2										
Instances					Topologies						Instances					Topologies					
T	W	$\|K\|$	$\|D\|$	$\|L\|$	C		R		M		T	W	$\|K\|$	$\|D\|$	$\|L\|$	C		R		M	
40	2	2	10	2	20	1128	20	1299	18	1031	480	5	5	100	3	24	34263	26	38982	25	34862
	3	3			13	1122	**13**	1291	12	1012		6	6			20	34876	22	38355	21	34917
	4	4			10	1123	**10**	1291	9	1012		7	7			18	34474	19	38570	18	34593
	2	4			10	1143	11	1327	10	1078		5	10			20	38376	20	41823	20	39154
	3	6			7	1137	**7**	1299	6	1040		6	12			17	38284	17	42224	17	39202
	4	8			**5**	1133	**5**	1291	5	1040		7	14			15	38358	15	42138	15	39179
40	2	2	20	2	36	2102	37	2335	41	2372	480	5	5	200	3	48	66651	52	76275	57	87113
	3	3			24	2102	25	2335	27	2350		6	6			40	66265	44	76925	48	86370
	4	4			18	2102	19	2353	21	2372		7	7			35	67079	38	76867	42	86715
	2	4			19	2180	21	2523	22	2438		5	10			40	77081	40	83771	40	94073
	3	6			13	2130	13	2415	14	2394		6	12			34	76560	34	83810	34	93129
	4	8			10	2130	10	2409	11	2390		7	14			29	76147	29	83245	29	93804
55	2	2	30	2	55	4707	42	3688	37	3284	480	5	5	300	3	80	120301	83	125995	87	133370
	3	3			37	4690	28	3662	24	3220		6	6			68	121922	70	125466	74	133212
	4	4			28	4712	21	3682	19	3252		7	7			59	121077	60	125451	65	134153
	2	4			29	4823	25	3985	22	3545		5	10			60	131179	60	135952	60	146215
	3	6			19	4747	16	3844	14	3450		6	12			50	130958	50	136979	51	174774
	4	8			14	4745	12	3812	11	3420		7	14			43	130308	43	135709	43	176498
60	2	2	40	2	75	6801	79	7981	52	5019	720	5	5	400	3	99	249391	103	261004	98	233048
	3	3			50	6801	54	8019	34	4969		6	6			82	248549	85	260457	82	234334
	4	4			38	6801	40	8005	26	4983		7	7			72	251401	73	258502	70	234814
	2	4			40	7051	43	8263	31	5508		5	10			80	284626	80	286304	80	267413
	3	6			25	6823	28	8093	19	5178		6	12			67	284409	67	285232	67	267634
	4	8			19	6867	21	8115	15	5303		7	14			58	282679	58	284142	58	268467
65	2	2	50	2	100	11074	85	9033	65	6808	720	5	5	500	3	117	289758	122	300937	117	278882
	3	3			67	11083	56	9002	43	6711		6	6			98	290019	101	300270	99	277250
	4	4			51	11111	43	9038	33	6793		7	7			85	288885	88	301550	84	277863
	2	4			53	11220	46	9357	35	7152		5	10			100	328747	100	337103	100	310085
	3	6			34	11196	31	9372	24	7124		6	12			84	327984	84	335356	84	308406
	4	8			26	11167	23	9283	18	7031		7	14			72	326016	72	334462	72	306848

Table 13: Best known solutions among all the methods for each instance.

5.2.5. Robustness of the proposed heuristics

Time-to-target (TTT plots) proposed by (Aiex et al., 2002, 2007) are used in this section to analyse the robustness of the LNS-GG and LNS-RG. TTT plots compute the running time for a heuristic to find a solution at least as good as a target one for a given instance. A total of 100 independent execution for each heuristic were done using different seeds. Then, running times are sorted in an ascending order list. It important to highlight that the LNS-GG has random components in the local search moves.

TTT plots are built using a probability $\rho_i = (i - 1/2)/100, i = \{1...100\}$, associated with every result. The points (τ_i, ρ_i) are plotted for $i = \{1, ..., 100\}$, where τ_i is the running time for the ith solution in the sorted list. Figure 4 depicts TTT plots for six instances: one instance from S1 and another from S2, for each of each topology. The target solution should not be too easy, nor too difficult to reach, hence, for each instance, a deviation in the b.k. solution up to 10% for $\mathcal{F}_1$ and $\mathcal{F}_2$ was fixed to obtain the target values. The axes x and y indicate respectively τ_i and ρ_i to reach the target value. The more straight the plot is, the higher probability to reach the target value is. Moreover, the closer to the y axis the plot is, the faster the method is.

Figures 4-(a), 4-(c) and 4-(e) depict TTT plots for three S1 instances, respectively, for cluster, random and mix topologies. Results indicate *LNS-GG* performed better than *LNS-RG* : TTT plots are closer to the y axis, but the plots have a few disruptions. This indicates the method is sensitive to some seeds and did not found a solution that reached the target value. Figures 4-(b), 4-(d) and 4-(f) show TTT plots for three S2 instances, using cluster, random and mix topologies. It is noticeable that after 20 seconds necessary to generate the initial solution using *LNS-RG*, this method reached target values very fast. *LNS-GG* has limitations to reach target values for large instances, consuming more time and a considerable sensibility to some seeds. For the former aspect, this can be observed by the position of the plots, and for the latter one, it can be seen by a lack of solutions for some seeds, generating some disruptions in the plot. Clearly, results indicate the robustness (fast and not sensible to the seeds) of the *LNS-RG* over the *LNS-GG* for larger instances.

6. Theoretical contributions and insights

In this section, we summarize the main theoretical contributions, results and insights obtained with this research. The formalization of SRP-CD by means of a mathematical model, the development of heuristics and the large set of experiments have allowed us to observe interesting theoretical issues described in the sequel.

The proposed mathematical model is based on dynamic multi-flow and was validated on small instances. To our knowledge, considering all parameters focused in this study, this is the first formulation integrating scheduling and routing for SRP-CD. In spite of its limits on solving medium instances, the flow indexed over the time allowed to further address the two levels of synchronization.

Results showed that the mathematical model solved small instances with up to 4 WTs, up to 8 dump

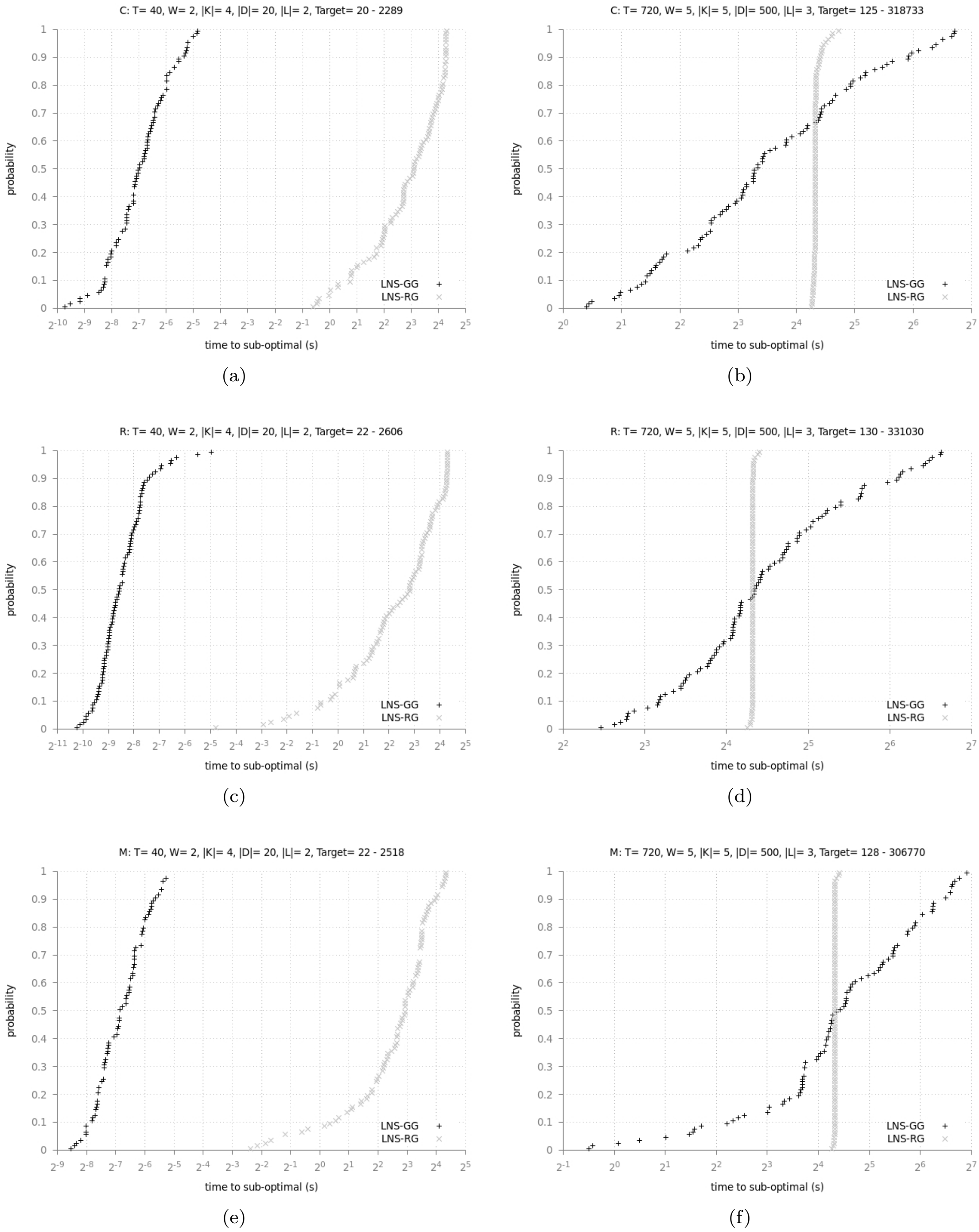


Figure 4: Analyzes of TTT plots of instances with cluster (a,b), random (c, d), and mix (e, f) topologies.

trucks and up to 10 debris nodes. For these cases, the model was able to produce optimal solutions for $\mathcal{F}_1$, while obtained feasible solutions for $\mathcal{F}_2$ within a reasonable computational time. This clearly indicates that $\mathcal{F}_2$ is rather difficult to be solved optimally by means of MILP. In addition, as the sensibility analysis suggest, the volume of debris plays a key role on the ability of the formulation to solve the problem. Instances with more than 10 debris nodes and a higher volume of debris are not solved by the mathematical formulation due to the computational time or memory limitation. This is a consequence of the amount of resources (WT, vehicles and time periods) required to address more debris. This aspect is traduced in the mathematical formulation as more variables and constraints, and in the overall problem as an increase on the combinatorial aspects.

The experimental analysis was applied as a methodology to evaluate the heuristics. This has been done since is still an open question to know why some heuristics perform better than others. Results show that *LNS-RG* overcame the other heuristics in three topologies. For instances with a cluster topology, *LNS-RG* found 70% of the best known solutions, while *LNS-GG* reached about 38% of the best known solutions. For instances with a random topology, *LNS-RG* achieved about 67% of the best known solutions, and *LNS-GG* obtained 35% of the best known solutions. And, for instances with a mix topology, *LNS-RG* found 75% of the best known solutions, while *LNS-GG* achieved about 33% of the best known solutions. *LNS-RG* has been shown to better perform and be more robust for larger instances in relation to the proposed methods.

The aforementioned theoretical issues open several avenues of opportunities in order to address a more efficient resource management and a faster recovery in the context of SRP-CD applications. One may note that using less resources (WT and vehicles) leads to a smaller cost, but increases the number of working days for the overall cleaning. Moreover, the fact that the volume of debris appears as a key point, it might imply in a multi-year governmental budgetary and operational planning.

7. Concluding remarks and perspectives

This study 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 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.

In addition, this study opens several avenues of research. Several 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 SDVP. Other issues like giving a priority to debris nodes can also be taken into account. In case of a high volume of debris, other approaches like sorting and valorizing debris, by direct use or recycling, could also be investigated. In terms of methods, hybrid ones and new local searchers can be proposed. In particular, there is room to improve the routing part of the SRP-CD.

Acknowledgements

The authors would like to thank the anonymous reviewers for their careful reading of our manuscript and their many insightful comments and suggestions. This research was developed in the TRIDE project, funded by the *Région Grand Est - Conseil départemental de l'Aube, France.*

References

Ahuja, R. K., Magnanti, T. L., & 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., & Ribeiro, C. C. (2002). Probability distribution of solution time in GRASP: An experimental investigation. *Journal of Heuristics*, *8*, 343–373.

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

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

Alarie, S., & 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*, 59–76.

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

Archetti, C., Feillet, D., Gendreau, M., & Speranza, M. G. (2011). Complexity of the VRP and SDVRP. *Transportation Research Part C: Emerging Technologies*, *19*, 741–750.

Archetti, C., & 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.) (pp. 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., & Aloise, D. J. (2020). Metaheuristics for the work-troops scheduling problem. *International Transactions in Operational Research*, .

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

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

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

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

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

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

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

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

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

Grimault, A., Bostel, N., & 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.

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

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

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

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

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

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

Moreno, A., Munari, P., & 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*, 16–34.

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

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

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

Sayarshad, H. R., Du, X., & 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.

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* (pp. 417–431). Springer, Pisa, Italy, October 26-30.

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

Soares, R., Marques, A., Amorim, P., & 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*, 174–194.

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

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

UNDP (2013). *Technical Guide for Debris Management - The Haitian experience 2010 - 2012*. Technical Report United Nations Development Programme.

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

Appendix A. Supplementary material

This appendix is devoted to present the acronyms used in the document.

Appendix A.1. Acronyms

BI - Best Improvement

C - Cluster

cK' - Criterion for dump truck in the insertion - LNS

cK - Criterion for dump truck

CSRP - Crew Scheduling Routing Problem

cWT - Criterion for WT

FI - First Improvement

FTPDP - Full Truck Load Pickup and Delivery Problem

FTVRP - Full Truck load Vehicle Routing Problem

GCH - Greedy Constructive Heuristic

GDTTF - Greater Ratio (Debris/Travel Time) First

GTTF - Greater Travel Time First

IRACE - Iterated racing for automatic algorithm configuration

LDF - Less Debris First

LNS-GG - Large Neighborhood Search - Greedy Greedy

LNS-RG - Large Neighborhood Search - Random Greedy

LNS - Large Neighborhood Search

LTF - Less Trucks First

M - Mix

MDF - More Debris First

MILP - Mixed Integer Linear Programming

OPM - Open Pit Mining

PDP - Pickup and Delivery problem

R - Random

RCH - Random Constructive Heuristic

RCPSP - Resource-Constrained Project Scheduling Problem

SDTTF - Smaller Ratio (Debris/Travel Time) First

SDVRP - Split Delivery Vehicle Routing Problem

SRP-CD - Scheduling routing problem to clean debris

STTF - Smaller Travel Time First

TTT - Time-to-target

WT - Work-troops