Economies of scale in recoverable robust maintenance location routing for rolling stock

D.D. Tönissen[a,*], J.J Arts[b,c]

[a]*School of Business and Economics, Vrije Universiteit Amsterdam, Amsterdam, The Netherlands*
[b]*Luxembourg Centre for Logistics and Supply Chain Management, University of Luxembourg, Luxembourg City, Luxembourg*
[c]*School of Industrial Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands*

Abstract

We consider the problem of locating maintenance facilities in a railway setting. Different facility sizes can be chosen for each candidate location and for each size there is an associated annual facility costs that can capture economies of scale in facility size. Because of the strategic nature of facility location, the opened facilities should be able to handle the current maintenance demand, but also the demand for any of the scenarios that can occur in the future. These scenarios capture changes such as changes to the line plan and the introduction of new rolling stock types. We allow recovery in the form of opening additional facilities, closing facilities, and increasing the facility size for each scenario. We provide a two-stage robust programming formulation. In the first-stage, we decide where to open what size of facility. In the second-stage, we solve a NP-hard maintenance location routing problem. We reformulate the problem as a mixed integer program that can be used to make an efficient column-and-constraint generation algorithm. To show that our algorithm works on practical sized instances, and to gain managerial insights, we perform a case study with instances from the Netherlands Railways. A counter intuitive insight is that economies of scale only play a limited role and that it is more important to reduce the transportation cost by building many small facilities, rather than a few large ones to profit from economies of scale.

Keywords: facility location, maintenance routing, rolling stock, two-stage robust optimization, column-and-constraint generation

1. Introduction

A *line plan* consists of a set of train lines, where each line is a path in the railway network that is operated with a certain frequency by one rolling stock type. The line plan within a railway network changes annually to accommodate changing travel demands. These changes include how lines run, up and down-scaling of service frequencies on any given line, the rolling stock types assigned to the lines, and the introduction of new rolling stock types. We capture these changes with a discrete set of scenarios. A *maintenance facility* is a facility that is responsible for the planned and unplanned maintenance of

---

*Corresponding author
    Email addresses:* `d.d.tonissen@vu.nl` (D.D. Tönissen), `joachim.arts@uni.lu` (J.J Arts)

train units. Because maintenance facilities are used for a long period, any maintenance facility plan should take a wide range of scenarios into account. The size of the opened locations should satisfy the maintenance requirements of the current situation, but recovery against a price is possible for each scenario. Recovery consists of opening additional facilities, closing facilities, and upgrading the size of the existing facilities. Decreasing the size of existing facilities is not allowed. The reason for this is that shrinking the size of an existing maintenance facility is difficult, may not yield any revenue, and can even be costly.

The stations at the two end points of a train line are called *end stations*. When multiple train units of the same rolling stock type are located at the same end station their destinations can be *interchanged*. Whether such an interchange is possible depends on the shunting infrastructure of the end station and the time horizon between arrival and departure of the train units. The favored way for train units to enter a maintenance facility is to interchange from one train line to another until a maintenance facility is reached. In case a maintenance facility cannot be reached by such interchanges, an empty train drive is used to reach a maintenance facility. *Unplanned maintenance* occurs when a failure in the field occurs and in that case the maintenance facility can only be reached by an empty train drive. The routing of the train units to the maintenance facilities is called *maintenance routing* and the problem of finding the most efficient routes in combination with facility location decisions is called the maintenance location routing problem.

In the recoverable robust maintenance location routing problem for rolling stock (RRMLRP), we seek the optimal locations and sizes of maintenance facilities for rolling stock in a railway network. The objective consists of minimizing the annual cost of the facilities and the worst-case annual transportation and recovery costs, given a discrete set of scenarios. The annual cost of a facility depends on its location and size. The size of a facility must be chosen from a discrete set for each location. This discrete set allows us to model economies of scale: a facility which is twice as large costs less than twice as much. As a consequence, it is possible to open a few large facilities to profit from economies of scale or to open multiple smaller facilities to limit the transportation cost.

The recoveries, the inclusion of unplanned maintenance, and the multiple facility sizes that include economies of scale are new compared to the literature. Unplanned maintenance is generally not considered in the maintenance (location) routing literature. With our case study in Section 7, we show that it is important to include unplanned maintenance, as it has a large influence on the number and location of the maintenance facilities but also on the cost. Furthermore, we demonstrate with our case study that the number and location of the opened facilities depends heavily on the allowed facility sizes and the associated cost. Although facility location problems that consider different sizes exist (see the references in Melo et al. (2009)), they are often not considered in the facility location literature. In addition, we could only find one paper that mentions economies of scale in facility size (Melo et al., 2006). As a consequence, including multiple facility sizes that include economies of scale can potentially play an important role in many other settings. We also investigate the trade-off between large facilities with

economies of scale versus many smaller facilities to reduce the transportation cost. Our case study demonstrates that even with increased economies of scale, economies of scale only play a limited role, and that it is more important to reduce transportation cost by building many small facilities. Economies of scale were thought to be more important and as a consequence this result may change the maintenance location strategy of the Netherlands Railways (NS).

We formulate the RRMLRP as a recoverable robust optimization problem, a two-stage robust optimization problem where the first-stage decisions can be modified in a limited way. The first-stage decision for the RRMLRP is to open facilities with a certain size, given candidate locations, and a discrete set of sizes for each candidate location and the associated facility costs. The first-stage decision has to be feasible for the current workload; the opened facilities have to be large enough to handle all maintenance visits occurring in the current situation. The second-stage problem is a maintenance location routing problem where, for each scenario, we can recover the first-stage decision and have to find the optimal routing to the maintenance facilities for the rolling stock. This second-stage problem is NP-hard.

We show that the two-stage model of the RRMLRP can be reformulated to a mixed integer programming model and we use this mixed integer programming formulation (MIP) to develop a column-and-constraint algorithm called scenario addition (SA). SA adds the constraints and variables associated with the scenario with the highest second-stage cost iteratively to the MIP until an optimal solution is found. SA has been applied successfully to a two-stage robust maintenance location routing problem (RMLRP) (Tönissen et al., In press.), a problem similar to the RRMLRP that does not include the aforementioned recovery of location decisions, unplanned maintenance, and multiple facility sizes. In Tönissen et al. (In press.) SA improved the solution time with two orders of magnitude compared to Benders decomposition.

SA finds the scenario with the highest solution value by solving the second-stage problem for each scenario. The RMLRP has a polynomial second-stage problem, while the second-stage problem is NP-hard for the RRMLRP, increasing the solution time of SA significantly. However, the scenario with the highest solution value can be found by solving the NP-hard second-stage problem for a limited number of scenarios. Many scenarios can be eliminated by a procedure that is similar to the pruning of nodes in branch-and-bound algorithms. This procedure uses an upper bound for each scenario and one lower bound. The upper bound can be found by a heuristic and the lower bound is equal to the highest exact second-stage solution value found so far. A scenario can now be eliminated when its upper bound is lower than the current lower bound. When a scenario cannot be eliminated, the second-stage problem for this scenario is solved to optimality, and the lower bound is updated if necessary. This procedure is described in detail in Section 5.2. Computational experiments show that this works very efficiently and that the NP-hard second-stage problem has to be solved only for a few scenarios.

The main contributions of this paper are:

1. We develop an SA algorithm that deals efficiently with the NP-hard second-stage problem by only solving the NP-hard problem for a limited number of scenarios. As a consequence this algorithm

significantly reduces the solution time for the RRMLRP. Moreover, our SA algorithm can potentially reduce the solution time of many other recoverable robust optimization problems.

2. We perform an extensive case study for the NS that shows that economies of scale only play a limited role, and that it is more important to reduce transportation cost by building many small facilities. This is an unexpected result and as a consequence may change the maintenance location strategy of the NS.

3. Our case study shows that the number and location of the opened facilities depends heavily on the allowed facility sizes and the economies of scale. This demonstrate the importance of including multiple facility sizes that include economies of scale in facility location problems and can consequently also play an important role in many other settings.

The paper starts with a literature review. In Section 3, we explain the problem setting and give an unplanned and planned maintenance routing model. In Section 4, we formulate the RRMLRP as a two-stage robust optimization problem. In Section 5, we discuss our solution methods (MIP and SA) to solve the RRMLRP. In Section 6, we provide computational results on randomly generated instances. We study the influence of the number of scenarios and candidate locations on the solution time and compare the methods introduced in Section 5. Finally, we perform a case study with data from the NS to provide managerial insights in Section 7.

2. Literature review

Many papers have been published on various types of facility location problems, see for example the facility location reviews from Daskin (1995) and ReVelle and Eiselt (2005). Furthermore, uncertainty often plays a role for facility decisions and it has to be considered whether other supply chain decisions should be taken simultaneously. For facility location problems under uncertainty, we refer to the review of Snyder (2006) and for facility location in combination with supply chain decisions see the review of Melo et al. (2009). However, facility location is not often studied in a maintenance setting. To our best knowledge there are only four papers that consider facility location in a maintenance setting: Lieckens et al. (2013) (remanufacturing network), Rappold and van Roo (2009) (aircraft engines repair network), van Ommeren and Bumb (2006) (truck maintenance), and Xie et al. (2016b) (locomotive maintenance).

Maintenance routing has been studied extensively for both rolling stock (for example Anderegg et al. (2003); Maróti and Kroon (2005, 2007)) and aviation (Liang et al. (2015); Sarac et al. (2006) and many others). The combination of maintenance routing and facility location, maintenance location routing has been introduced by Feo and Bard (1989) for aviation. Feo and Bard (1989) use a two-phase heuristic to minimize the number of maintenance bases for four day maintenance checks for aircraft. The heuristic first solves the routing problem without maintenance requirements and then the requirements are satisfied by choosing the best maintenance locations by solving a set-covering problem by a greedy procedure.

After Feo and Bard (1989), no new papers about maintenance location routing were published until Gopalan (2014). Gopalan (2014) locates a minimal number of maintenance facilities on an Euler tour,

while ensuring that all required aircraft maintenance activities can be performed in time. He provides four different heuristics to solve this problem and also consider two robust variants of the problem. Xie et al. (2016a) solve a maintenance location routing problem (MLRP) for the maintenance of locomotives. Besides maintenance location routing, their paper also includes decisions about the capabilities and sizes of the facilities and the assignment of work demands to facilities. They use a heuristic approach that is based on decomposition. The problem is split in among others a facility location and a routing problem and these components are solved iteratively and sequently. The difference between our paper and these papers is that we use exact algorithms and that we include uncertainty by using two-stage robust optimization. In this paper we show that such an algorithm combined with a smart heuristic can also be used for two-stage robust optimization problems that have a NP-hard second-stage problem.

The most recent paper, Tönissen et al. (In press.), includes the maintenance routing for rolling stock in an aggregate way into a facility location model. They model the MLRP as two-stage robust optimization and stochastic programming problems and provide a Benders decomposition and a scenario addition algorithm (also known as column-and-constraint generation or row-and-column generation) algorithm to solve the models to optimality. The difference between Tönissen et al. (In press.) and our paper is the inclusion of unplanned maintenance, different facility sizes, and recoveries in the form of upgrading, closing and opening additional facilities. Furthermore, the second-stage problem of Tönissen et al. (In press.) is polynomial, while the RRMLPR has a NP-hard second-stage problem making it more difficult to solve.

Variations to Benders decomposition or column-and-constraint generation procedures are often used to solve two-stage robust optimization problems. Adaptations of Benders decomposition have been applied to a non-linear convex two-stage robust location transportation problem (Gabrel et al., 2014), a recoverable robust facility location problem (Álvarez-Miranda et al., 2015) and many other problems. Column-and-constraint generation procedures have been applied among others to a reliable p-median facility location problem (An et al., 2014) and a robust facility location problem with uncertain demand locations (Chan et al., 2017). Recently, it was shown that for some problems, column-and-constraint generation algorithms outperform Benders decomposition. Zeng and Zhao (2013) show that a column-and-constraint generation algorithm performs an order of magnitude faster than Benders decomposition for a two-stage robust location transportation problem with demand levels in a polyhedral uncertainty set. Tönissen et al. (In press.) show that their column-and-constraint generation algorithm outperforms Benders decomposition with more than two orders of magnitude for the robust MLRP where uncertainty takes values in a discrete set of scenarios.

A drawback of most two-stage robust optimization models is that the objective value can be high, because the first-stage solution has to be feasible for all scenarios including extreme scenarios. However, when recovery is allowed, the first-stage solution only has to be feasible for the current situation, and for each scenario after recovery. This reduces the cost, improves the solution for most or even all scenarios, while the worst case scenario is still taken into account. *Recoverable robustness*, two-stage

robust optimization where the first-stage variables can be recovered was introduced by Liebchen et al. (2009). Since then, it has been used for many problems such as the shortest path problem (van den Akker et al., 2016; Büsing, 2012), network design (Álvarez-Miranda et al., 2014), rolling stock planning (Cacchiani et al., 2012), evacuation planning (Goerigk et al., 2015), and maintenance routing (Maher et al., 2014).

Recoverable robustness has also been used for facility location (Álvarez-Miranda et al., 2015). The main difference between the RRMLRP and the recoverable robust facility location problem from Álvarez-Miranda et al. (2015) is the inclusion of the routing of rolling stock and multiple facility sizes. Furthermore, Álvarez-Miranda et al. (2015) allocate customers in the first-stage and allows reallocation in the second-stage, while allocation of maintenance visits only happens in the second-stage for the RRMLRP.

To summarize, we use exact algorithms instead of heuristics to solve our maintenance location routing problem. We are the first to combine recoverable robustness with maintenance location routing. Furthermore, we include several new features such as unplanned maintenance and different facilities sizes that includes economies of scale in facility size to the MLRP. In addition the scenario addition algorithm presented in this paper could potentially reduce the solution time for other recoverable robust problems.

3. Unplanned and planned maintenance routing

Unplanned maintenance occurs when a train unit fails in the field. The failed train unit then has to drive, or is towed, to a maintenance facility to be repaired. Driving a train without passengers is called *deadheading*. The deadheading cost consists of driving costs (train driver, fuel etc.) and disservice costs because the train is not available for public transport, resulting in passenger discomfort because of shorter trains. Considerations such as waiting time at stations before deadheading due to track unavailability are implicitly modelled with the disservice cost.

Planned maintenance typically occurs once every half year up to every month. The planned maintenance is not scheduled within the long term rolling stock schedule. The maintenance routing is done operationally and starts 3 to 5 days before a train unit requires maintenance due to mileage. From that point, maintenance routers try to route the train with the help of *interchanges* to an end station close to the maintenance facility. Whether such an interchange is possible depends on the operational rolling stock schedule and the shunting infrastructure of the end stations. Deadheading is used when an interchange is not possible. Deadheading is less expensive for planned maintenance than it is for unplanned maintenance because unplanned maintenance may involve towing and other inconveniences due to the unavailability of tracks on short notice.

The final trip to the maintenance facility is generally done at a pre-arranged time such that there is sufficient space/capacity in the maintenance facility. Consider, for example, a train unit of type a that serves the line with end stations A and B (denoted as line (A, B, a)) on Monday. Suppose further that there is a maintenance slot available at maintenance facility C on Friday at 20:00. In such a case the maintenance routers will try to move the train unit with interchanges to line (B, C, a). (Deadheading

will be used if this is not possible by using interchanges.) The train unit will serve passengers at line (B, C, a) until it is time to drive to the maintenance facility that is located at or close to end station C. A more detailed description and operational maintenance routing model for the NS can be found in Maróti and Kroon (2005, 2007).

In this paper, we do not model the maintenance routing at an individual train unit level for each day in a long planning horizon. Rather, we solve an aggregated version of this operational problem that allows us to determine the expected annual cost of routing train units to maintenance facilities. On a daily operational planning basis, one may use more detailed maintenance routing models. However, the main purpose here is to determine routing costs associated with a specific facility location decision, not to include daily operational planning into a strategic model. Thus, our aggregated maintenance routing model enables us to combine the maintenance routing with facility location decisions and recoveries in Section 4.

3.1. Problem description and model

The goal of the unplanned and planned maintenance routing problem is to determine the annual routing costs for train units to enter given maintenance facilities with given capacities. A *maintenance facility* is a facility that is responsible for the planned inspections and maintenance of rolling stock. The frequency of inspections and maintenance depends on the rolling stock type and typically occurs once every half year up to every month. Given is a physical rail network $G_P = (N_P, E_P)$, consisting of rails E_P, all stations N_P, and a line plan. A line plan consists of a set of lines L, with for each line, two end stations, the type of rolling stock, and the *unplanned and planned maintenance frequency* per year. The planned or unplanned maintenance frequency of a line, is the total number of expected planned or unplanned maintenance visits originating from that train line in one year. Furthermore, a line plan specifies the unplanned and planned deadheading cost for each line to each facility, and the set of possible interchanges with a coordination cost for each interchange. Finally, we are given a set of opened maintenance locations O with for each location a size q_o. q_o is the maximum workload that opened facility $o \in O$ can process annually. The workload of a planned maintenance visit is set at 1 and that of an unplanned maintenance visit at $u \in \mathbb{R}^+$.

Figure 1 shows on the left-hand side an example of a physical rail network graph containing stations and in the middle and right-hand side two line plans for two different scenarios. There are two train types in the example shown in figure 1. The first, denoted by a, is a regional train, stopping at every station, while train type b is an intercity train that skips the small stations. An example of an interchange in the right-hand side of Figure 1 is line (Y, Z, b) to line (Z, V, b), while an interchange from (Y, Z, b) to line (Z, U, a) is not possible because the rolling stock types do not match. Furthermore, the number of end stations is different between the line plans. For example, station X is an end station in the middle of Figure 1, while it is an in-between station of the line (W, Y, b) on the right-hand side.

The maintenance routing graph is a directed flow graph, $G_F = (N_F, A_F)$ that is constructed for a given line plan and scenario by the following steps:
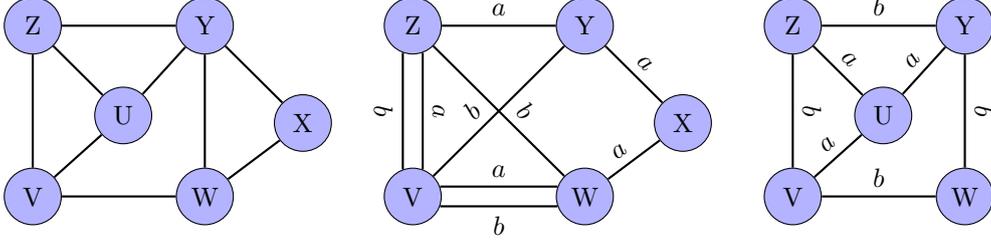
Figure 1: The physical rail network on the left and two possible line plans.

- We make two nodes for every line, one for planned maintenance and one for unplanned maintenance. The set of planned maintenance nodes is denoted by N_{PL} and for the unplanned maintenance nodes we have N_{UL}. We define $N_L = N_{\mathrm{PL}} \cup N_{\mathrm{UL}}$.

- We create one source $\mathcal{S}$ that is connected with a directed arc to each node in N_L.

- We create arcs between the line nodes from N_{PL} where an interchange is possible, with as cost the interchange coordination cost. The set of these interchange arcs is denoted by A_I.

- We make a node for every candidate facility; this set is denoted as N_C. Each node in N_C is connected with an arc to the sink $\mathcal{T}$.

- For each node $n \in N_L$, we create an arc to each facility. The cost of this arc is the deadheading cost of the line to the facility. The cost of the arc can be 0 when deadheading is not necessary because the facility is located at an end station, and the line associated with the node is connected to the facility. The set of all incoming facility arcs is denoted as A_F.

A $\mathcal{S}$-$\mathcal{T}$ path can now be seen as a route (interchanges and deadheading) for an annual maintenance frequency originating from a line to a maintenance facility. In Figure 2, we demonstrate how to create a maintenance routing flow graph for the small example line plan depicted on the left-hand side. In this example, we again show the rolling stock type by using letters and we numbered the lines such that the associated line node can easily be found in the maintenance routing flow graph. We assume that an interchange is possible between line 0 and 2, and $O = \{A,C,D\}$, where A and C are facilities located at end stations A and C, while D is a candidate facility that is located elsewhere. The figure on the right-hand side depicts the flow network with unplanned (U) and planned (P) line nodes.

An aggregate version of the example that we have seen at the beginning of this section can be modelled with help of graph G_F. Assume that we have 10 train units serving line (A, B, a) that are expected to require maintenance 3.8 times per year. The number of annual maintenance visits originating from line (A, B, a) is consequently 38. In that case the flow from S to node 2P would be 38. The flow can interchange to Node 0P from node 2P, and go via arc $(0P, C)$ without any costs to maintenance facility C. An alternative route is to deadhead directly from Node 2P to maintenance facility C via arc $(2P, C)$ and incur the deadheading cost. Furthermore, we assume that at end station B on average only 20 interchanges can be done annually due to operational constraints. In that case, 20 maintenance visits
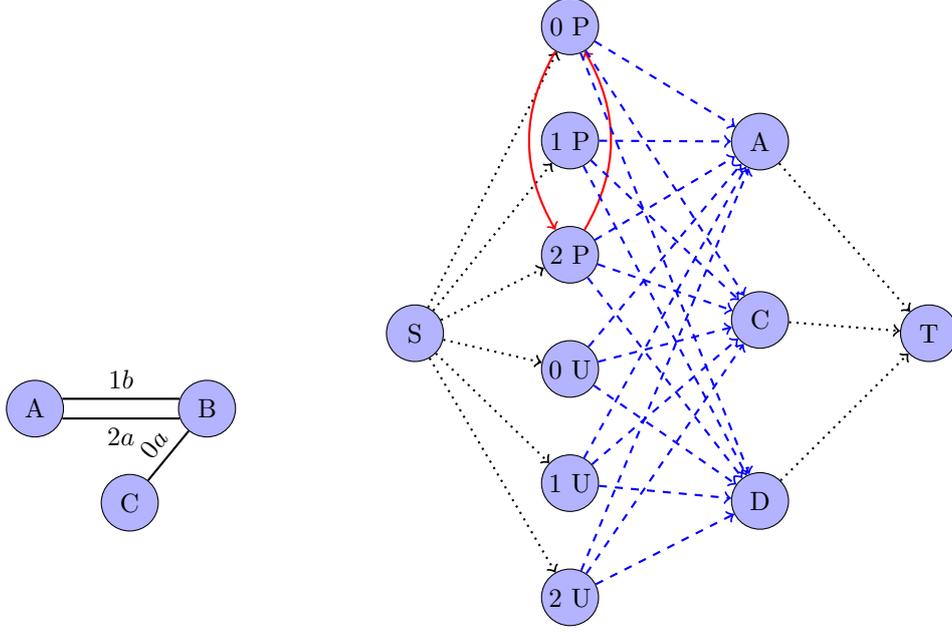
Figure 2: Left a line planning possibility and right the resulting flow graph ($G_F = (N_F, A_F)$). The arcs from and to the source and sink are dotted black, the interchange arcs (A_I^d) solid red and the arcs to the facilities (A_C) are dashed blue.

will reach maintenance facility C via interchanges and the other 18 maintenance visits by deadheading. In addition, note that in the graph G_F only interchanges followed by deadheading directly to the maintenance facility are possible. Deadheading followed by interchanges can easily be allowed in the graph by creating an arc from every planned maintenance line node to each other. That arc represents the deadheading from one line to another, that can be followed by any combination of interchanges and deadheading until the maintenance facility is reached. The reason that we exclude these kind of routes is that they are very expensive because they cause imbalances in the number of train units per line, which need to be solved. As a consequence, these kind of routes are not often used in practice.

Proposition 1. *The number of nodes and arcs in the flow graph G_F is polynomial in the number of lines and end stations.*

Proof. The number of nodes in the flow graph is equal to $|N_L| + |N_C| + 2$. The number of arcs is equal to $|N_L| + |N_L||N_C| + |A_I| + |N_C|$, where $|A_I|$ is bounded by $|N_{\mathrm{PL}}|(|N_{\mathrm{PL}}| - 1)$. □

The maintenance frequency for line l is defined by the parameter m_l, and n_l is the node associated with line l. The set of end stations is given by S, and $g \in \mathbb{R}_0^+$ is a restriction on the annual number of interchanges at end station $s \in S$. The number of annual interchanges over the whole network is restricted by the parameter $G \in \mathbb{R}_0^+$. The flow through arc a associated with the yearly maintenance frequency from line $l \in N_L$, is represented by the decision variable $z_l(a) \in \mathbb{R}_0^+$. For example, $z_3(1, 7)$ represents the frequency of interchanges from line 1 to line 7 for a maintenance visit originating from line 3, and $z_2(8, x)$ represents the frequency of maintenance visits originating from line 2, that reach maintenance facility x via line 8. We define $\delta_{\mathrm{in}}(n)$ and $\delta_{\mathrm{out}}(n)$ as the set of ingoing and outgoing arcs

of node n in graph G_F. $A_I \subset A_F$ is the set of interchange arcs, $A_C = \bigcup_{n \in N_C} \delta_{\text{in}}(n) \subset A_F$, the set of incoming candidate facilities arcs, and $A_s \subset A_F$ as the set of arcs representing the interchanges at end station s. The cost of arc a of type l is $c_l(a)$, which is only defined for arcs in the set $A_I \cup A_C$.

The unplanned and planned maintenance routing model can now be formulated as the following flow model:

$$\text{UMRP} = \min \sum_{l \in N_L} \sum_{a \in A_I \cup A_C} c_l(a) z_l(a)$$

$$\text{s.t.} \sum_{a \in \delta_{\text{in}}(n)} \left(\sum_{l \in N_{\text{PL}}d} z_l(a) + u \sum_{l \in N_{\text{UL}}} z_l(a) \right) \leq q_o \qquad \forall o \in O, \tag{1}$$

$$\sum_{a \in \delta_{\text{in}}(n)} z_l(a) = \sum_{a \in \delta_{\text{out}}(n)} z_l(a) \qquad \forall n \in N_F \setminus \{\mathcal{S}, \mathcal{T}\}, \quad \forall l \in N_L^d, \tag{2}$$

$$z_l(a) = m_l \qquad \forall l \in N_L, \quad a \in \delta_{\text{in}}(n_l) \setminus A_I, \tag{3}$$

$$\sum_{l \in N_L} \sum_{a \in \delta_{\text{in}}(\mathcal{T})} z_l(a) = \sum_{l \in N_L} m_l, \tag{4}$$

$$\sum_{l \in N_{\text{PL}}} \sum_{a \in A_s} z_l(a) \leq g_s \qquad \forall s \in S, \tag{5}$$

$$\sum_{l \in N_{\text{PL}}} \sum_{a \in A_I} z_l(a) \leq G, \tag{6}$$

$$z_l(a) \geq 0 \qquad \forall a \in A_F \quad \forall l \in N_L.$$

Constraints (1) restrict the number of yearly unplanned and planned maintenance visits that can be assigned to opened facility o with size q_o. Constraints (2) are the flow conservation constraints, and Constraints (3) and (4) guarantee that every maintenance visit is assigned to a facility. Constraint (4) is necessary to exclude solutions with more flow than the number of yearly maintenance visits. These solutions are possible because some of the routes from the lines to the candidate facility locations have zero costs. Constraints (5) and (6) are the end station interchange and network interchange constraints.

3.2. Improved unplanned and planned maintenance routing model

Recall, that we are only interested in estimating the annual maintenance cost and not in the routing decision themselves. Furthermore, we assume that train units can only be interchanged when they have the same rolling stock type and that $c_m(a) = c_n(a) \; \forall m, n \in L$, when the rolling stock type assigned to line m and n are the same. With this in mind its possible to eliminate many of the variables by replacing $z_l(a)$ by $z(a)$ and $c_l(a)$ by $c(a)$ without changing the cost of the optimal solution.

For each $z_l(a) \; l \in L, a \in A_F$, we have the following possibilities:

1. $z_l(a) = 0 \quad \forall l \in L$.

2. $z_l(a) > 0$ for exactly one line $l \in L$.

3. $z_l(a) > 0$ for multiple lines $l \in L$.

For possibilities 1) and 2) we can trivially replace $z_l(a)$ by $z(a)$ and $c_l(a)$ by $c(a)$. When we remove the interchange arcs of graph G_F only possibilities 1) and 2) are possible. Consequently, situation 3) can only be reached after an interchange. Because only interchanges are allowed between the same rolling stock types we know that when $z_m(a) > 0$ and $z_n(a) > 0$ that $c_m(a) = c_n(a)$. Consequently, we can also safely replace $z_l(a)$ by $z(a)$ and $c_l(a)$ by $c(a)$ in possibility 3).

We can deal with the different workload and cost between the unplanned and planned lines by making two sets with incoming arcs for each facility node $o \in O$. The first set $\delta_{\text{inP}}(o)$ contains the incoming arcs belonging to the planned maintenance flows and the second set $\delta_{\text{inU}}(o)$ the incoming arcs for unplanned maintenance. These changes decreases the number of variables by a factor L. Furthermore, Constraint (4) can be dropped, because without the l indices, all flow is already restricted by Constraints (3). We can now formulate the following improved unplanned and planned maintenance routing model:

$$\text{IUMRP} = \min \sum_{a \in A_I^d \cup A_C^d} c(a)z(a)$$

$$\text{s.t.} \quad \sum_{a \in \delta_{\text{inP}}^d(n)} z(a) + u \sum_{a \in \delta_{\text{inU}}^d(n)} z(a) \leq q_0 \qquad \forall o \in O \tag{7}$$

$$\sum_{a \in \delta_{\text{in}}^d(n)} z(a) = \sum_{a \in \delta_{\text{out}}^d(n)} z(a) \qquad \forall n \in N_F \setminus \{\mathcal{S}, \mathcal{T}\}, \tag{8}$$

$$z(a) = m_l \qquad \forall l \in N_L, \quad a \in \delta_{\text{in}}(n_l) \setminus A_I, \tag{9}$$

$$\sum_{a \in A_s} z(a) \leq g_s \qquad \forall s \in S, \tag{10}$$

$$\sum_{a \in A_I} z(a) \leq G, \tag{11}$$

$$z(a) \geq 0 \qquad \forall a \in A_F.$$

To summarize, the IUMRP simplifies a multi-commodity flow model to a single-commodity flow model and reduces the number of variables by a factor L. As a result, the solution time and required memory are greatly reduced.

4. Recoverable robust maintenance location routing

In this section, we introduce the recoverable robust maintenance location routing problem. We formulate the RRMLRP as a recoverable robust two-stage problem. The first-stage decision consists of opening facilities at different locations of different sizes. The second-stage decision is taken for every scenario, and corresponds to the IUMRP in combination with the option to open additional facilities, close facilities, or upgrade the size of existing facilities. For the unplanned and planned maintenance routing we use the notation introduced in Section 3, but an index d will be added to denote the scenario to which the variable or parameter belongs to. An overview of the notation that will be used for the two-stage problem can be found in Table 1.

Recall that the size of a facility is the maximum workload that it can process annually. The workload generated by a planned maintenance visit is set at 1 and that of an unplanned maintenance visit as u. The total annual workload of the entire line plan for the current situation is denoted by M. The sizes for a facility at location $n \in N_C$ are denoted by the set Q^n. A tuple $i \in Q^n$, consists of a size q_{ni} that represents the annual workload that a facility can handle and the annual facility cost c_{ni} for location n and size i. The first-stage decision is represented by Y, that contains the binary decision variables $y_{ni} \in \{0,1\} \ \forall n \in N_F, i \in Q^n$. The binary decision variable y_{ni} is 1 when a facility of size i is opened at location n and 0 otherwise. This first-stage decision has to be feasible for the current annual workload and recovery is allowed for the scenarios $d \in D$ in the second-stage.

Recovery can be done by upgrading the facility sizes, opening additional facilities, and closing facilities. Recall that decreasing the facility size is not allowed. We introduce the recovery factors k^d, f^d, and $h^d \in \mathbb{R}$ that scale the annual facility cost when executing these actions for scenario d. If the current facility size is $j \in Q^n$ at location n, then the cost of upgrading from size j to size i is given by $k^d(c_{ni} - c_{nj})$ for a scenario d. The cost of opening additional facility n with size i is $f^d c_{ni}$ and the profit of closing that facility is $h_i^d c_{ni}$ for scenario $d \in D$. We use annual costs for the facilities and recoveries to be able to combine them with the annual maintenance routing cost. These annual costs can be found by dividing the total cost of constructing a facility or performing a recovery by its depreciation period. For the recoveries, we define the binary decision variables v_{ni}^d, x_{ni}^d, and w_{ni}^d that represent upgrading, additionally opening, and closing facility n, with size i for scenario d, respectively.

We formulate the following two stage model:

$$\text{(RRMLRP- 2SRO)} \quad \min \sum_{n \in N_C} \sum_{i \in Q^n} c_{ni} y_{ni} + \max_{d \in D} \text{IUMLRP}_d(Y)$$

Subject to:

$$\sum_{n \in N_C} \sum_{i \in Q^n} y_{ni} q_{ni} \geq M \tag{12}$$

$$\sum_{i \in Q^n} y_{ni} \leq 1 \qquad\qquad \forall n \in N_C, \tag{13}$$

$$y_{ni} \in \{0,1\} \qquad\qquad \forall n \in N_F, \quad \forall i \in Q^n,$$

where

$$\text{IUMLRP}_d(Y) = \min \sum_{a \in A_I^d \cup A_C^d} c^d(a) z^d(a) + \sum_{n \in N_C} \sum_{i \in Q^n} (f^d c_{ni} x_{ni}^d - h^d c_{ni} w_{ni}^d + k^d c_{ni} (v_{ni}^d + w_{ni}^d - y_{ni}))$$

$$\text{s.t.} \sum_{i \in Q^n} y_{ni} + \sum_{i \in Q^n} x_{ni}^d \leq 1 \qquad\qquad \forall n \in N_C, \tag{14}$$

$$w_{ni}^d \leq y_{ni} \qquad\qquad \forall n \in N_C, \quad \forall i \in Q^n, \tag{15}$$

$$\sum_{i \in Q^n} v_{ni}^d = \sum_{i \in Q^n} (y_{ni} - w_{ni}^d) \qquad\qquad \forall n \in N_C, \tag{16}$$

$$\sum_{i \in Q^n} (y_{ni} q_{ni} - w_{ni}^d q_{ni}) \leq \sum_{i \in Q^n} v_{ni}^d q_{ni} \qquad \forall n \in N_C, \tag{17}$$

$$\sum_{a \in \delta_{\mathrm{inP}}^d(n)} z^d(a) + u \sum_{a \in \delta_{\mathrm{inU}}^d(n)} z^d(a) \leq \sum_{i \in Q^n} (v_{ni}^d q_{ni} + x_{ni}^d q_{ni}) \qquad \forall n \in N_C, \tag{18}$$

$$\sum_{a \in \delta_{\mathrm{in}}^d(n)} z^d(a) = \sum_{a \in \delta_{\mathrm{out}}^d(n)} z^d(a) \qquad \forall n \in N_F^d \setminus \{\mathcal{S}, \mathcal{T}\}, \tag{19}$$

$$z^d(a) = m_l^d \qquad \forall l \in N_L^d, \quad a \in \delta_{\mathrm{in}}^d(n_l^d) \setminus A_I, \tag{20}$$

$$\sum_{a \in A_s^d} z^d(a) \leq g_s^d \qquad \forall s \in S^d, \tag{21}$$

$$\sum_{a \in A_I^d} z^d(a) \leq G^d, \tag{22}$$

$$z^d(a) \geq 0 \qquad \forall a \in A_F^d \quad \forall l \in N_L^d,$$

$$x_{ni}^d, w_{ni}^d, v_{ni}^d \in \{0,1\} \qquad \forall n \in N_C, \quad \forall i \in Q^n.$$

The first-stage objective minimizes the cost of the opened maintenance facilities and the worst-case cost of the maintenance location routing problem. This objective is of special interest to risk averse decision makers. Constraint (12) guarantees that the sizes of the opened maintenance facilities are sufficient for the current situation and Constraints (13) that we can open every location with at most one size.

The objective of the IUMLRP consists of two parts. The first part represents the maintenance routing costs and the second part the recovery costs. The cost of opening an additional facility is $f^d c_{ni}$ and the gain of closing a facility is $h^d c_{ni}$. To keep the objective linear, we also "upgrade" a facility when it has the same size as the current facility. The cost for an upgrade is given by $\sum_{i \in Q^n} k^d c_{ni}(v_{ni}^d + w_{ni}^d - y_{ni})$. This is 0 when the facility is closed or when the facility has the same size as the already opened facility. When an opened facility at location n with size j is upgraded to size i, then the cost is $k^d(c_{ni} - c_{nj})$.

Constraints (14) guarantee that only one facility is opened at each location. Constraints (15) guarantee that a facility can only be closed when it has been opened. Constraints (16) make sure that a facility that has been opened, and not closed, is kept open and Constraints (17) guarantee that the size of that facility (when not closed) is larger than or equal to the previous size. Constraints (18) restrict the number of yearly unplanned and planned maintenance visits that can be assigned to opened facility (n, i) with size q_{ni}. Constraints (19) are the flow conservation constraints, and Constraints (20) guarantee that every maintenance visit is assigned to a facility. Constraints (21) and (22) are the end station interchange and network interchange constraints.

Different from most two-stage problems is that the IUMLRP$_d(Y)$ is NP-hard and that the size of the recourse matrix is not fixed. The values and dimensions of the recourse matrix change for each scenario because the line plan of each IUMLRP$_d(Y)$ is different. Furthermore, we have relative complete recourse because of the recoveries, when we assume that the total facility size of all candidate facilities is enough to maintain all train units.

Graph and set notation:

$G_F^d(Y) = (N_F^d, A_F^d)$	maintenance routing flow graph consisting of nodes N_F^d and arcs A_F^d
$\mathcal{S}$	source node
$\mathcal{T}$	sink node
N_C	set of nodes associated with the candidate facilities C ($N_C \subset N_F^d$)
Q^n	set of sizes for facility n
D	set with discrete scenarios
S^d	set with all end stations for scenario d
L^d	set with all train lines for scenario d
N_L^d	set of nodes associated with the lines L^d ($N_L^d \subset N_F^d$)
n_l^d	the node associated with line l and scenario d
A_C^d	set of arcs belong to scenario d that go to the candidate facility nodes ($A_C^d \subset A_F^d$)
A_I^d	set of interchange arcs for scenario d ($A_I^d \subset A_F^d$)
A_s	set of interchange arcs for end station s and scenario d ($A_s^d \subset A_I^d$)
$\delta_{\mathrm{in}}^d(n)$	set containing the incoming arcs of node n for scenario d ($\delta_{\mathrm{in}}^d(n) \subset A_F^d$)
	we can use $\delta_{\mathrm{inP}}^d(n)$ or $\delta_{\mathrm{inU}}^d(n)$ to only take the planned or unplanned maintenance visits
$\delta_{\mathrm{out}}^d(n)$	set containing the outcoming arcs of node n for scenario d ($\delta_{\mathrm{out}}^d(n) \subset A_F^d$)
	we can use $\delta_{\mathrm{outP}}^d(n)$ or $\delta_{\mathrm{outU}}^d(n)$ to only take the planned or unplanned maintenance visits

Parameters:

m_l^d	annual maintenance frequency of train units originating from line l in scenario d
M	annual workload of the entire line plan for the current situation
$c^d(a)$	cost for using arc a for scenario d
q_{ni}	annual workload for location n and size i
c_{ni}	annual facility cost for location n and size i
g_s^d	annual interchange capacity of end station s for scenario d
G^d	annual network interchange capacity for scenario d
u	increased workload factor for unplanned maintenance
k^d	recovery scale factor for upgrading a facility
f^d	recovery scale factor for opening an additional facility
h^d	recovery scale factor for closing a facility

Decision variables:

v_{ni}	binary recovery decision variable for upgrading candidate facility n to size i
w_{ni}	binary recovery decision variable for closing candidate facility n with size i
x_{ni}	binary recovery decision variable for additionally opening candidate facility n with size i
y_{ni}	binary first-stage decision variable for opening candidate facility n with size i
$z^d(a)$	annual maintenance frequency decision variable for arc a and scenario d

Table 1: Graphs, sets, parameters, and decision variables for the RRMLRP.

14

Theorem 1. *The IUMLRP$_d(Y)$ is NP-hard.*

Proof. The IUMLRP$_d(Y)$ is NP-hard because the capacitated facility location problem (CFLP) is a special case of this problem. In the CFLP, a set of demand points, a set of facilities, the sizes of the locations and their cost, and the cost of assigning a demand point to a facility are given. The objective is to open a set of facilities and to assign each demand point to a facility, while not exceeding the size and minimizing the cost. The IUMLRP$_d(Y)$ extends the CFLP by including maintenance routing, already opened facilities, and the possibility of upgrading or closing a facility. When we take the special case where none of the facilities are opened and set our budget G^d at 0, then we can only open facilities and only deadheading from the lines to the locations is possible. In that case we can interpret the lines as demand points, with as demand the required maintenance visit frequency, then the deadheading cost is exactly the same as the assignment cost, and the IUMLRP$_d(Y)$ can be used to solve the CFLP. □

5. Solution Methods

In this section, we explain our solution methods: the mixed integer formulation and the improved scenario addition method.

5.1. Mixed integer programming formulation

To solve the two-stage formulation of Section 4, we reformulate the problem to a deterministic equivalent that can be solved by standard solvers such as CPLEX. The deterministic equivalent of the RRMLRP- 2SRO is a mixed integer programming formulation, that uses one large maintenance routing flow graph, instead of one graph for every scenario. The steps used in Section 3 are adapted to define the graph $G_M = (N_M, A_M)$:

- We make two nodes for every line and scenario, one for planned maintenance and one for unplanned maintenance. The set with nodes representing the planned maintenance is denoted as N_{PL}^d and N_{UL}^d represents the unplanned maintenance belonging to scenario $d \in D$. Again $N_L^d = N_{\mathrm{PL}}^d \cup N_{\mathrm{UL}}^d$.

- We create one source $\mathcal{S}$ that is connected with a directed arc to each node in $\bigcup_{d \in D} N_L^d$.

- We create arcs between the line nodes from N_{PL}^d $\forall d \in D$ where an interchange is possible, with as cost the interchange coordination cost. The set of these interchange arcs is denoted by A_I^d.

- We make a node for every candidate facility; this set is denoted as N_C. Each node in N_C is connected with an arc to the sink $\mathcal{T}$.

- For each node $n \in \bigcup_{d \in D} N_L^d$, we create an arc to each facility. The cost of this arc is the deadheading cost of the line to the facility. The cost of the arc can be 0, when deadheading is not necessary because the facility is located at an end station, and the line associated with the node is connected to the facility. The set of these incoming facility arcs is denoted as A_F^d.

15

Note that the flow from different scenarios can never reach each other in the graph G_M. Consequently, we can drop the index d from $c^d(a)$ and $z^d(a)$. In addition, when we drop the index d for a set, it is short for taking the union of the sets for all scenarios, e.g. $\delta_{\text{in}}(n) = \bigcup_{d \in D} \delta_{\text{in}}^d(n)$ for any $n \in N_M$. This gives the following mixed integer programming (MIP) model:

$$(\text{RRMLRP-MIP}) \quad \min \sum_{n \in N_C} \sum_{i \in Q^n} c_n y_{ni} + z_{\max}$$

$$\text{s.t.} \quad \sum_{i \in Q^n} y_{ni} + \sum_{i \in Q^n} x_{ni}^d \leq 1 \qquad \forall n \in N_C, \quad \forall d \in D, \tag{23}$$

$$w_{ni}^d \leq y_{ni} \qquad \forall n \in N_C, \quad \forall i \in Q^n, \quad \forall d \in D, \tag{24}$$

$$\sum_{i \in Q^n} v_{ni}^d = \sum_{i \in Q^n} (y_{ni} - w_{ni}^d) \qquad \forall n \in N_C, \quad \forall d \in D, \tag{25}$$

$$\sum_{i \in Q^n} (y_{ni} q_{ni} - w_{ni}^d q_{ni}) \leq \sum_{i \in Q^n} v_{ni}^d q_{ni} \qquad \forall n \in N_C, \quad \forall d \in D, \tag{26}$$

$$\sum_{n \in N_C} \sum_{i \in Q^n} y_{ni} q_{ni} \geq M \tag{27}$$

$$\sum_{a \in \delta_{\text{inP}}^d(n)} z(a) + u \sum_{a \in \delta_{\text{inU}}^d(n)} z(a) \leq \sum_{i \in Q^n} (v_{ni}^d q_{ni} + x_{ni}^d q_{ni}) \qquad \forall n \in N_C, \quad \forall d \in D, \tag{28}$$

$$\sum_{a \in \delta_{\text{in}}(n)} z(a) = \sum_{a \in \delta_{\text{out}}(n)} z(a) \qquad \forall n \in N_M \setminus \{\mathcal{S}, \mathcal{T}\}, \tag{29}$$

$$z(a) = m_l^d \qquad \forall d \in D, \quad \forall l \in N_L^d, \quad a \in \delta_{\text{in}}^d(n_l^d) \setminus A_I^d, \tag{30}$$

$$\sum_{a \in A_s^d} z(a) \leq g_s^d \qquad \forall d \in D, \quad \forall s \in S^d, \tag{31}$$

$$\sum_{a \in A_I^d} z(a) \leq G^d \qquad \forall d \in D, \tag{32}$$

$$z_{\max} - \sum_{a \in A_I^d \cup A_C^d} c(a) z(a)$$

$$\geq \sum_{n \in N_C} \sum_{i \in Q^n} (f^d c_{ni} x_{ni}^d - h^d c_{ni} w_{ni}^d + k^d c_{ni}(v_{ni}^d + w_{ni}^d - y_{ni})) \quad \forall d \in D, \tag{33}$$

$$v_{ni}^d, w_{ni}^d, x_{ni}^d \in \{0,1\} \qquad \forall n \in N_C, \quad \forall i \in Q^n, \quad \forall d \in D,$$

$$y_{ni} \in \{0,1\} \qquad \forall n \in N_C, \quad \forall i \in Q^n,$$

$$z(a) \geq 0 \qquad \forall a \in A_M.$$

The constraints are similar to those of the $\text{IUMLRP}_d(Y)$ in Section 4. Additionally, Constraint (27) guarantees that the opened facilities have a sufficiently large size for the current situation. Constraints (33) guarantee that $z_{\max}$ equals the worst-case second-stage costs.

5.2. Scenario addition

For ease of explanation, we first explain the SA method as introduced in Tönissen et al. (In press.), apply it to the RRMLRP, and continue with our improved more complex version of the SA method.

5.2.1. The original method

We set the iteration counter j at 1 and let D^j denote the scenario set belonging to iteration j. The set D^1 contains one randomly chosen scenario from $d \in D$. The SA method consists of the following steps:

1. Compute the solution to the RRMLRP-MIP with D replaced by D^j. Denote the optimal objective of this problem as LB^j and the solution of iteration j as Y^j.

2. Calculate for each scenario $d \in D$, $\text{IUMLRP}_d(Y^j)$ separately. Set $\text{UB}^j := \sum_{n \in N_C} \sum_{i \in Q^n} c_{ni} y_{ni}^j + \max_{d \in D} \text{IUMLRP}_d(Y^j)$ and let $w^j := \arg\max_{d \in D} \text{IUMLRP}_d(Y^j)$. When $w^j \in D^j$ stop (because $\text{UB}^j = \text{LB}^j$) and return Y^j as the optimal solution and UB^j as the optimal objective value. Otherwise ($w^j \notin D^j$) the algorithm proceeds to the next step.

3. Let $D^{j+1} := \{w^j\} \cup D^j$. Update $j := j + 1$ and go back to Step 1.

These steps give a very diverse set of scenarios as it always adds the worst-case scenario for the current first-stage solution. Because of this, the algorithm generally converges quickly, with only a small number of scenarios in D^j.

5.2.2. Improved Scenario Addition

Because the $\text{IUMLRP}_d(Y^j)$ is NP-hard, we improve the algorithm by aiming to avoid solving the $\text{IUMLRP}_d(Y^j)$ for every scenario $d \in D$ in each iteration. When we know that a scenario does not have the worst-case cost for a given first-stage decision Y, we do not need to compute the optimal solution for this scenario. Assume that we can determine an upper bound HUB_d^j to the $\text{IUMLRP}_d(Y^j)$ by a heuristic. In that case it is not necessary to solve $\text{IUMLRP}_d(Y^j)$ to optimality for a scenario $d \in D$, when there is a scenario $s \in D \setminus \{d\}$ for which $\text{IUMLRP}_s(Y^j) \geq \text{HUB}_d$. The steps to calculate HUB_d^j will be explained directly below the algorithm. The improved scenario addition (ISA) algorithm consists of the following steps:

1. Compute the solution to the RRMLRP-MIP with D replaced by D^j. Denote the optimal objective of this problem as LB^j and the solution of iteration j as Y^j.

2. Calculate an upper bound HUB_d^j with a heuristic for each scenario $d \in D$ and sort the scenarios in descending HUB_d^j order.

3. Set $\text{LBM}^j := \text{IUMLRP}_d(Y^j)$, and $w^j := d$, where d is the first scenario from the list (the scenario with the highest HUB_d^j).

4. Go through the list with scenarios $d \in D$. When $\text{HUB}_d^j \leq \text{LBM}^j$, go to Step 5. Otherwise solve $\text{IUMLRP}_d(Y^j)$ and when $\text{IUMLRP}_d(Y^j) > \text{LBM}^j$ set $\text{LBM}^j := \text{IUMLRP}_d(Y^j)$ and $w^j := d$.

5. Set $\text{UB}^j := \sum_{n \in N_C} \sum_{i \in Q^n} c_{ni} y_{ni}^j + \text{LBM}^j$. When $w^j \in D^j$ stop (because $\text{UB}^j = \text{LB}^j$) and return Y^j as the optimal solution and UB^j as the optimal objective value. Otherwise ($w^j \notin D^j$) the algorithm proceeds to the next step.

6. Let $D^{j+1} := \{w^j\} \cup D^j$. Update $j := j + 1$ and go back to Step 1.

To estimate an upper bound to the $\text{IUMLRP}_d(Y^j)$ we solve the $\text{IUMRP}_d(Y^j)$. The $\text{IUMRP}_d(Y^j)$ is the IUMRP defined in Section 3.2 where we use scenario $d \in D$ as input for the parameters and the graph G_F, and the opened facilities of Y^j as O. When the size of the facilities in O is feasible for scenario d, we directly solve the $\text{IUMRP}_d(Y^j)$. When it is infeasible we make the solution feasible by upgrading and opening additional facilities by a heuristic rule, before solving the $\text{IUMRP}_d(Y^j)$. Our heuristic rule first upgrades the size of the already opened facilities until the solution is feasible or upgrading is not possible. When a solution is still not feasible after upgrading, additional facilities are opened until a feasible solution is found. Assume that the facility sizes $i \in Q^n$ for each facility $n \in N_C$ are sorted in ascending order of facility size. We use the following steps to get a good upper bound (HUB_d) for the $\text{IUMLRP}_d(Y^j)$:

1. When $\sum_{n \in N_C} \sum_{i \in Q^n} y^j_{ni} q_{ni} \geq \sum_{l \in N^d_L} m^d_l$, set $\text{HUB}_d := \text{IUMRP}(Y^j, d)$ and stop. Otherwise proceed to the next step.

2. Define the variables Φ^j, that have exactly the same values as Y^j.

3. Upgrade the sizes of existing facilities. For each $\phi^j_{ni} \in \Phi^j$ do:

   - If facility ϕ^j_{ni} is opened, close it and open facility $\phi^j_{n,i+1}$ if it exists.
   - When $\sum_{n \in N_C} \sum_{i \in Q^n} \phi^j_{ni} q_{ni} \geq \sum_{l \in N^d_L} m^d_l$, go to Step 5.

4. Open additional facilities. Go through all locations $n \in N_C$:

   - When $\sum_{i \in Q^n} \phi^j_{ni} = 0$, open facility n with the lowest size such that $\sum_{n \in N_C} \sum_{i \in Q^n} \phi^j_{ni} q_{ni} \geq \sum_{l \in L^d} m^d_l$. When opening facility n cannot satisfy the needed workload, open facility n with its highest size.
   - When $\sum_{n \in N_C} \sum_{i \in Q^n} \phi^j_{ni} q_{ni} \geq \sum_{l \in L^d} m^d_l$, go to Step 5.

5. Set $\text{HUB}_d := \text{IUMRP}(\Phi^j, d) + \sum_{n \in N_C} \sum_{i \in Q^n} r^{dj}_{ni}$. The recovery value r^{dj}_{ni} for location n with size i for scenario d and iteration j is:

   - $k^d(c_{ni} - c_{nm})$ if $\phi^j_{ni} = 1$ and $\exists m \in Q^n, y^j_{nm} = 1$, with $c_{nm} < c_{ni}$ (facility (n,m) has been upgraded to (n,i)).
   - $f^d c_{ni}$ if $\phi^j_{ni} = 1$ and $y^j_{ni} = 0$ (additionally opened).
   - Otherwise the value is 0.

6. Computational experiments

In this section, we report computational experiments on randomly generated instances to test the computational performance of the algorithms we developed. We compare the size of instances that can be solved by the MIP, SA, and ISA. Although we generate instances randomly, the fixed and random parameters are based on those found in practice to create reasonable instances. All experiments are programmed in Java with the CPLEX library version 12.6.3, and run on a laptop with an Intel Core i7-4710MQ Quad Core 2.5 GHz processor with 32 GB of RAM. We use CPLEX standard settings and the reported solution times include the time required for building the model.

18

6.1. Test bed generation

We adapt the test bed from Tönissen et al. (In press.), such that it includes unplanned maintenance visits, the recovery parameters f, k, h, and the facility sizes. We only describe the adaptations and refer to Section 7 of Tönissen et al. (In press.) for the full details. The number of unplanned maintenance visits for each line is the number of planned maintenance visits times a number that is uniformly distributed between 1 and 2 ($\mathcal{U}(1,2)$). In many practical settings, the recovery factors are restricted to $k^d > 1$ (upgrading capacity is more expensive than buying it immediately), $f^d > 1$ (buying a facility now is better than buying it later), $f^d > k^d$ (upgrading size is less expensive than building a new facility with the same size as the upgrade), and $0 \leq h^d < 1$ (selling facilities is not profitable). Consequently, we generate k^d from $\mathcal{U}(1,2)$, f^d from $k^d \cdot \mathcal{U}(1,2)$, and h^d from $\mathcal{U}(0,1)$ for each $d \in D$.

We generate the set Q with 1, 3, 5, and 9 facility sizes elements and set for each instance $Q_n = Q \ \forall n \in N_c$. The cost for the sizes are estimated with the square root safety staffing rule (Halfin and Whitt, 1981). When the size of a facility is increased by a factor x, the needed safety size C to deal with uncertainty is only increased by a factor $\sqrt{x}$. Consequently, we assume that the cost increases by a factor $\frac{x + C\sqrt{x}}{1 + C} < x$. We estimate that $C \approx 0.21$ for a standard location of size 1/3 M and the annual cost of this location is $\mathcal{U}(0.7, 1.6)$ million euros per year. When the cost of location n of size 1/3 M is determined, we calculate the cost of the other sizes for location n by multiplying the cost with the factors shown in Table 2.

Size	1/12 M	1/8 M	1/6 M	1/4 M	1/3 M	1/2 M	2/3 M	M	4/3 M
Factor	0.29	0.42	0.54	0.77	1.00	1.45	1.90	2.78	3.65

Table 2: Cost increases for locations compared to a facility with size $1/3M$.

The set with one facility size contains the size 1/3 M, the set with three sizes also includes 1/6 M and 2/3 M, the set of 5 sizes also includes 1/12 M and 4/3 M, and the set with 9 sizes includes all mentioned sizes. Our test set consists of 225 instances for each combination of 10, 25, and 50 facility locations and 1, 3, 5, and 9 sizes. We start with 15 instances with 1 scenario: 5 instances for each of the three rectangles discussed in Section 7.7.1 of Tönissen et al. (In press.). The next 15 instances have 2 scenarios, and we double the number of scenarios until we reach 32768 scenarios. We refer to these sets of 15 instances as scenario subsets.

6.2. The experiments

We test the computational performance of the MIP, SA, and ISA. When more than 80% (13 or more out of 15) of the instances for a scenario subset can be solved within an hour, we continue to the next scenario subset, where the number of scenarios is doubled. In Table 3, we denote the last scenario subset for which at least 80% of the instances could be solved within an hour for 10, 25, and 50 candidate facilities for 1 and 9 sizes. The results of the sets with 3 and 5 sizes are automatically captured within this interval. We see in Table 3 that SA improves the MIP and ISA improves it further.

	# Candidate locations		
Algorithm	10	25	50
MIP	256-2048	512-1024	32-128
SA	16384-32768	4096-8192	1022-2048
ISA	32768	32768	32768

Table 3: Last scenario set interval solved by the algorithms for 10, 25, and 50 candidate locations for 9 and 1 discrete sizes.

The difference in number of scenarios for the different sizes that the MIP can solve can be explained by the fact that the number of nodes in the branch and bound tree for the MIP increases with a factor 4.6, 165.0, and 2.4 for the 10, 25, and 50 facility sets with 1 and 9 sizes respectively. The average number of iterations varies between 1.9 and 2.1 for the location and size combinations for the SA and ISA algorithm, which is much less than 32768.

In Table 4 we show for each scenario subset with 10, 25, and 50 candidate facilities and 9 sizes, the average solution time in seconds when solved by ISA. Furthermore, we present the number of necessary iterations and the average number of times the IUMLRP had to be solved for each iteration. We can see that only few iterations are necessary and that the IUMLRP only has to be solved for a limited number of scenarios. However, we also see that the railway graph and the defined basic line planning possibilities influences this, as the IUMLRP is solved more often for the instances with 10 candidate locations than for the instances with 25 and 50 locations.

7. Case study

We test our ISA algorithm on instances from the NS. We use a green field approach, where existing facilities of the NS are kept out of scope. The goal of this section is to obtain managerial insights. In Section 7.1, we describe the NS instances, followed by our experiments in Section 7.2.

7.1. NS instances

We assume that the candidate locations are always located at the end stations. We have 59 end stations of which 55 are also a candidate location. The facility costs are an estimation of the average annual cost of land, the necessary infrastructure and the maintenance facility itself including all side buildings. Furthermore, we either decrease or increase the facility cost based on the average land price of the province that a location is in. To create the cost for the different sizes, we multiply the cost estimation for each location with the factors depicted in Table 2 from Section 6. The network interchange budget G^d is $\mathcal{U}(0.25M, M)$ $\forall d \in D$ and the unplanned maintenance factor u is set to 0.25. The upgrade factor k^d is $\mathcal{U}(1, 2)$, additionally opening factor f^d is $k^d \cdot \mathcal{U}(1, 1.5)$ and the closing factor h^d is $\mathcal{U}(0, 1)$ for each $d \in D$.

	# Candidate locations								
	10			25			50		
Scenarios	time	iterations	IUMLRP	time	iterations	IUMLRP	time	iterations	IUMLRP
1	0.2	1	1	0.8	1	1	4.2	1	1
2	0.7	1.7	1.1	5.9	1.5	1.1	17.1	1.5	1.2
4	0.5	1.8	1.1	2.9	1.9	1.1	30.1	1.9	1.6
8	0.6	1.9	1.4	4.7	2	1.1	7.4	1.9	1.6
16	1	2.2	1.9	2.2	2	1	119.8	2.3	3.4
32	1.1	2.1	2.3	2.2	2	1.1	82	2.3	2.5
64	0.9	2	2.7	1.8	1.9	1.2	95.8	2.2	3.5
128	1.5	2.2	3.3	4.3	2.1	1.4	35.3	2.1	9.6
256	1.6	2.2	6.3	2.8	2	1.2	29.6	2.1	8.6
512	7.3	2.5	9.8	4	2	1.3	7.3	2	2
1024	4	2.1	10.9	7.2	2.1	1.4	17.7	2.1	4.9
2048	8.6	2.4	14	12.9	2.1	1.5	23.1	2	2.4
4096	17.7	2.5	35.3	20.5	2	1.4	73.4	2	22.4
8192	23.9	2.3	26.4	46.1	2.1	1.5	75.1	2	3.7
16384	53	2.3	79.4	86.5	2	2.5	147.1	2	4.3
32768	131	2.3	160	231	2.1	2.2	595.1	2.1	5.6

Table 4: Solution time in seconds, iterations, and average number of times the IUMLRP is solved with ISA.

All scenarios are based on four basic line plans. These basic line plans are: the current situation (2015), an estimation of 2018, and two possibilities for approximately 2025. The future line plans are based on the plan "Beter en Meer" (Prorail and NS, 2014) made by the NS and Prorail. These line plans contain all the lines (97, 97, 99, and 100 lines), the rolling stock type serving the line, and an estimate of the number planned yearly maintenance visits per line. Scenarios are made by picking a basic line plan, and slightly altering the planned maintenance frequency and rolling stock types. The new planned maintenance frequency for each line of the line plan is generated from a triangular distribution. The planned maintenance frequency of the basic plan is the mode of this distribution. Furthermore, we assume that the number of maintenance visits can decrease by 32.5% and increase by 75%, due to uncertainty in the number of maintenance visits each train unit requires each year and the number of passengers using a certain line. A maximum of 20% of the rolling stock types of the lines can be swapped with each other. Moreover, the unplanned maintenance frequency for a line is the same as the planned maintenance frequency, as they occur approximately equally often for the NS.

7.2. Experiments

We start with investigating the influence of the number of scenarios on the number of opened facilities, optimal cost, and solution time. We then continue by investigating several important managerial questions.

7.2.1. Scenarios

We start with 10 instances with one scenario, and increase the number of scenarios a fourfold in each next set. In Table 5 we show the minimum and maximum number of opened facilities, followed by the average number of opened facilities between parentheses. Furthermore, we denote the average number of iterations, average number of times the IUMLRP has to be solved per iterations, the average total cost in millions per year and the average solution time. Most facilities are opened with a size of 1/12 M, the largest size opened depends on the instance and lies between 1/8 M and 1/3 M.

Scenarios	opened	iterations	IUMLRP	cost (M/ year)	time (min.)
1	13-16 (14.7)	1.0	1.0	13.0	0.3
4	14-16 (15.4)	1.2	2.0	14.0	3.4
32	15-18 (16.1)	3.4	2.1	15.1	127.0
128	16-18 (16.8)	3.2	2.5	15.6	65.2
512	15-18 (16.3)	3.6	5.1	16.1	96.6
2048	15-18 (16.7)	4.0	5.5	16.6	164.2*
8192	16-19 (17.2)	3.9	9.3	17.0	164.8*

Table 5: Scenario results for the RRMLRP.
*For 2049 one instance and for 8192 three instances could not be solved within 5 hours and a time of 5 hours is used to compute the averages.

The number of facilities, the cost and the solution time seems to increase with the number of scenarios. The solution time increases suddenly for 32 scenarios and then decreases again for 128 scenarios. The reason for this is that some of the instances with 32 scenarios, have several scenarios that have close performance and finding the worst scenarios required adding all of these. When more scenarios are added, there are fewer scenarios with close to worst-case performance.

The NS instances are more difficult than the instances from the computational experiments. One of the reasons for this is that the NS instances have 55 candidate location compared to a maximum of 50 candidate facilities for the computational instances. However, this difference is not enough to explain the increase in solution time of two orders of magnitude. Furthermore, we can observe that the NS instances require more iterations (approximately 4 vs 2) and that solving the MIP (with a subset of the scenarios) goes quickly for the computational instances, while it takes a long time to close the gap for the NS instances. From the RMLRP and SMLRP, we know that the computational instances are not easy, as they are more difficult than the NS instances in Tönissen et al. (In press.). Consequently, the large difference in solution time is caused by some characteristics specific to the RRMLRP. We note also that

most of the scenarios for the case instances are growth scenarios. Growth scenarios are more likely to be important than the randomly generated scenarios in the computational study. We expect further that growth scenarios are more difficult for the RRMLRP, because a choice has to be made between opening more facilities initially or using recovery actions such as upgrading and opening additional facilities.

The expected value of the wait and see solution (WS) is calculated for the ten instances with 512 scenarios. The wait and see cost for the ten instances are on average 16.0 million euros per year, making the value of perfect information (WS - RRMLRP) 0.1 million euros. The maximum difference between the WS and the RRMLP for the ten instances is 1.9% and the average difference is 0.8%. This difference is smaller than the 3.2% found in Tönissen et al. (In press.), we expect that one of the reasons for this is the allowed recovery.

From now on, all experiments are done with ten instances with 512 scenarios and a maximum solution time of 3 hours. Approximately 10% of the instances are not solved within 3 hours, for those instances the best found solution is returned. The relative gap ($\frac{\text{UB}-\text{LB}}{\text{LB}}$) of those instances is most of the time around 0.005 and the maximum found gap is 0.035.

7.2.2. Facility sizes

In this section we answer the question, "Does the set of facility sizes influence the number of opened facilities and optimal worst-case cost?" We answer this question by varying the different facility sets of which the results can be found in Table 6. The set with 5 and 9 sizes have the sizes mentioned in Section 6.1. As we can see, the more sizes there are, the lower the cost is. However, we also see that the cost increase is limited when we only allow a size of 1/12 M. The solutions with the 5 and 9 sizes, consists mainly of facilities with size 1/12 M and on some strategically chosen places facilities with size 1/8 M to 1/3 M. These "strategic" places are end stations where many lines end and/or that can be reached easily by the interchanges. When only one facility size is allowed the cost increases with the facility size. The cost are 16.7 million for 1/12 M, 17.5 million for 1/6 M and they increase up to 27.9 million for facilities of size 4/3 M. Furthermore, the number of opened facilities decreases when the sizes increases from 21.4 opened facilities to only 2.0 facilities. Consequently, the number and the allowed facility sizes heavily influence the number of opened facilities and the cost.

Currently, the NS uses three large maintenance facilities of approximately size 1/3 M. Table 6 indicates that for the same size six facilities should be opened and that when smaller facilities are allowed even more facilities should be opened. These results could potentially change the maintenance facility location strategy of the NS from larger facilities with economies of scale to reducing transportation cost by building many small facilities. Because different facility sizes and facility sizes that includes economies of scale are barely considered in the facility location literature they can potentially also play an important role in other settings.

Sizes	opened	largest size	cost (M/ year)
9 set	16-18 (16.9)	$1/4M$	16.4
5 set	16-19 (17.5)	$1/3M$	16.5
$1/12M, 1/3M, 4/3M$	17-21 (19.5)	$1/3M$	16.6
$1/12M$	20-23 (21.4)	$1/12M$	16.7
$1/6M, 1/3M, 2/3M$	10-11 (10.7)	$1/3M$	17.5
$1/6M$	10-12 (11.0)	$1/6M$	17.5
$1/3M$	6 (6.0)	$1/3M$	19.7
$2/3M$	3-4 (3.6)	$2/3M$	23.4
$4/3M$	2 (2.0)	$4/3M$	27.9

Table 6: Results for different sizes, sorted in ascending cost for the RRMLRP.

7.2.3. Economies of scale

In this section we answer the question, "Does increasing the economies of scale in facility size decrease the number of opened facilities and optimal worst-case cost?". We do that by increasing the overcapacity factor C and as a consequence the economies of scale in capacity sizes play a larger role. Table 7 shows the new cost factors compared to the sizes.

Size / C	0.21	0.45	0.75	1.5	3
1/12 M	0.29	0.33	0.36	0.40	0.44
1/8 M	0.42	0.45	0.48	0.52	0.55
1/6 M	0.54	0.56	0.59	0.62	0.66
1/4 M	0.77	0.79	0.80	0.82	0.84
1/3 M	1.00	1.00	1.00	1.00	1.00
1/2 M	1.45	1.41	1.38	1.33	1.29
2/3 M	1.90	1.82	1.75	1.65	1.56
M	2.78	2.61	2.46	2.24	2.05
4/3 M	3.65	3.38	3.14	2.80	2.50

Table 7: Cost increases for facilities compared to a facility with size 1/3 M.

In Table 8 the results are shown. When the economies of scale increase the number of facilities decreases, however the optimal worst-case cost increases. The optimal worst-case cost increases, because even with economies of scale far larger than that is expected for the NS it is still beneficial to open small facilities. The cost of the mentioned small facilities increases from 0.29 to 0.44 times the cost of a facility of size 1/3 M, resulting in higher optimal worst-case cost. As a consequence, the share of facility cost (fac %) in the optimal worst-case cost also increases from 38.2% to 42.7%.

C	opened	fac %	cost (M/ year)
0.21	15-17 (16.0)	38.2	16.1
0.45	14-15 (14.3)	39.3	16.7
0.75	12-15 (14.0)	40.7	17.1
1.5	11-13 (12.6)	41.3	17.6
3	10-16 (12.4)	42.7	18.6

Table 8: Results when changing the overcapacity factor C for the RRMLRP.

7.2.4. Unplanned maintenance visits

In this section we answer the question, "What influence has the annual unplanned maintenance frequency on the number of opened facilities and the optimal worst-case cost?". We vary the annual unplanned maintenance frequency by multiplying the current annual unplanned maintenance frequency for each line by a factor. The results are shown in Table 9. When there is no unplanned maintenance (factor is 0), only an average of 12.4 facilities are necessary and the cost are reduced to 5.5 million. When the unplanned maintenance is increased by a factor 2, the number of facilities increases to an average of 19.5, and the cost increases to 26.3 million. We can conclude from Table 9 that the inclusion of the unplanned maintenance visits is important as it has a large influence on the cost and the number of opened maintenance facilities. An additional managerial insight is that unplanned maintenance has a large influence on the cost and that consequently large investments can be made to reduce unplanned maintenance.

factor	opened	fac %	cost (M/ year)
0	12-14 (12.4)	83.5	5.5
0.25	14-16 (14.2)	63.4	8.2
0.5	13-16 (14.6)	50.1	10.8
0.75	14-19 (15.5)	44.0	13.4
1.0	14-17 (16.2)	38.6	16.0
2.0	18-21 (19.5)	28.9	26.3
4.0	24-26 (24.5)	21.9	46.4

Table 9: Unplanned maintenance results for the RRMLRP.

7.2.5. Rolling stock types

In this section we answer the question, "Does decreasing the number of rolling stock types decrease the number of opened facilities and optimal worst-case cost?". Currently, the NS has 5 intercity (IC) and 6 regional (R) rolling stock types. We study what the result would be if the number of rolling stock

types would be limited to 3 intercity and 3 regional rolling stock types and just 1 intercity and 1 regional rolling stock type.

In Table 10 we see that limiting the number of rolling stock types reduces the optimal worst-case cost with 11.8%. The reason for this is that with less rolling stock types, more interchanges are possible for the planned maintenance visits even in the worst-case scenario. Furthermore, the number of opened maintenance visits decreases with approximately 1 when we limit the rolling stock types to 1 intercity and 1 regional. Consequently, it is beneficial for the NS to limit their number of different rolling stock types.

IC/R	opened	cost (M/ year)
5-6	15-18 (16.3)	16.1
3-3	16-18 (16.5)	15.1
1-1	15-16 (15.4)	14.2

Table 10: Rolling stock types results for the RRMLRP.

7.2.6. Interchange budget

In this section we answer the question, "Does increasing the interchange budget decrease the number of opened facilities and optimal worst case cost?". The interchange budget is varied from 0 to 2 M for 10 instances with 513 scenarios. The results can be found in Table 11.

Budget	opened	cost (M/ year)
0	16-19 (17.2)	17.0
$0.25M$	16-19 (16.9)	16.5
$0.5M$	15-17 (16.3)	16.3
$0.75M$	15-17 (16.4)	16.3
M	15-17 (16.5)	16.3
$2M$	16-19 (16.8)	16.3

Table 11: Budget results for the RRMLRP.

The number of opened facilities is similar and the difference can be explained by the fact that there are multiple optimal solutions. The optimal worst-case cost initially decreases, but then stays the same. The reason for this is that in the worst-case the budget can never be fully utilized due to the fact that there is always a scenario where interchanging is hard because lines of the same rolling stock type are not connected to each other. Although the interchange budget barely influences the number of opened facilities and cost, it does influence the location of the opened facilities and their size. For example, with a budget of 0, Maastricht is only opened with a size of 1/12 M, while with a budget of 2M its size is

often 1/4 M, because it can also be reached by interchanges.

8. Conclusion

We formulated the RRMLRP as a two-stage problem, where the second-stage model is NP-hard. We developed an ISA algorithm that deals efficiently with the NP-hard second-stage problem by only solving the NP-hard problem for a limited number of scenarios. Our computational experiments show that ISA works efficiently; the number of iterations is approximately two and the number of NP-hard problems is low (on average 5.6 for the instances with 32768 scenarios). ISA works so efficiently that it can solve computational instances with 50 candidate facilities and 32768 scenarios within an hour. Besides significantly reducing the solution time for the RRMLRP, this algorithm can also potentially reduce the solution time of many other recoverable robust optimization problems.

The case study indicates that it is important to include unplanned maintenance as it has a large influence on the cost and the number of maintenance facilities. Furthermore, we demonstrate with our case study that the number and location of the opened facilities depend heavily on the allowed facility sizes and the associated cost. This result can also play an important role for facility location problems in different settings. An unexpected result for the NS is that economies of scale are less important than reducing the transportation cost. As a consequence, cost can be saved by building more but smaller facilities. In addition, unplanned maintenance has a large influence on the number and location of the maintenance facilities. The total cost can be significantly reduced by reducing the annual unplanned maintenance frequency. Furthermore, reducing the number of rolling stock types reduces the cost by approximately 12% and increasing the interchange budget decreases the cost with 22%.

An interesting future research direction is to include rolling stock dependent resources. In this paper we assumed that every maintenance facility can maintain each rolling stock type, but in some settings a rolling stock type can only be maintained at a facility when matching resources are installed. Installing matching resources for all rolling stock types at every location may be too expensive from a practical perspective. Furthermore, a mechanic has to work sufficiently many hours on a specific rolling stock type, to be allowed to keep maintaining it. The latter is for example not possible when there are many small maintenance facilities that maintain all rolling stock types. Consequently, including these resources may result in different locations of the facilities and it could affect the number of opened maintenance facilities.

E. Álvarez-Miranda, I. Ljubić, S. Raghavan, and P. Toth. The recoverable robust two-level network design problem. *INFORMS Journal on Computing*, 27(1):1–19, 2014.

E. Álvarez-Miranda, E. Fernández, and I. Ljubić. The recoverable robust facility location problem. *Transportation Research Part B: Methodological*, 79:93–120, 2015.

Y. An, B. Zeng, Y. Zhang, and L. Zhao. Reliable p-median facility location problem: two-stage robust models and algorithms. *Transportation Research Part B: Methodological*, 64:54–72, 2014.

L. Anderegg, S. Eidenbenz, M. Gantenbein, C. Stamm, D. S. Taylor, B. Weber, and P. Widmayer. Train routing algorithms: Concepts, design choices, and practical considerations. In *Proceedings of the Fifth Workshop on Algorithm Engineering and Experiments*, volume 111, pages 106–118. SIAM, 2003.

C. Büsing. Recoverable robust shortest path problems. *Networks*, 59(1):181–189, 2012.

V. Cacchiani, A. Caprara, L. Galli, L. Kroon, G. Maróti, and P. Toth. Railway rolling stock planning: Robustness against large disruptions. *Transportation Science*, 46(2):217–232, 2012.

T. C. Chan, Z.-J. M. Shen, and A. Siddiq. Robust defibrillator deployment under cardiac arrest location uncertainty via row-and-column generation. *arXiv preprint arXiv:1507.04397v4*, 2017.

M. S. Daskin. *Network and Discrete Location: Models, Algorithms, and Applications*. Wiley, New York, 1995.

T. A. Feo and J. F. Bard. Flight scheduling and maintenance base planning. *Management Science*, 35 (12):1415–1432, 1989.

V. Gabrel, M. Lacroix, C. Murat, and N. Remli. Robust location transportation problems under uncertain demands. *Discrete Applied Mathematics*, 164:100–111, 2014.

M. Goerigk, K. Deghdak, and V. TKindt. A two-stage robustness approach to evacuation planning with buses. *Transportation Research Part B: Methodological*, 78:66–82, 2015.

R. Gopalan. The aircraft maintenance base location problem. *European Journal of Operational Research*, 236(2):634–642, 2014.

S. Halfin and W. Whitt. Heavy-traffic limits for queues with many exponential servers. *Operations Research*, 29(3):567–588, 1981.

Z. Liang, Y. Feng, X. Zhang, T. Wu, and W. A. Chaovalitwongse. Robust weekly aircraft maintenance routing problem and the extension to the tail assignment problem. *Transportation Research Part B: Methodological*, 78:238–259, 2015.

C. Liebchen, M. Lübbecke, R. Möhring, and S. Stiller. The concept of recoverable robustness, linear programming recovery, and railway applications. In *Robust and Online Large-Scale Optimization*, volume 5686 of *Lecture Notes in Computer Science*, pages 1–27. Springer Berlin Heidelberg, 2009.

K. T. Lieckens, P. J. Colen, and M. R. Lambrecht. Optimization of a stochastic remanufacturing network with an exchange option. *Decision Support Systems*, 54:1548–1557, 2013.

S. J. Maher, G. Desaulniers, and F. Soumis. Recoverable robust single day aircraft maintenance routing problem. *Computers & Operations Research*, 51:130–145, 2014.

G. Maróti and L. Kroon. Maintenance routing for train units: The transition model. *Transportation Science*, 39(4):518–525, 2005.

G. Maróti and L. Kroon. Maintenance routing for train units: The interchange model. *Computers & Operations Research*, 34(4):1121–1140, 2007.

M. T. Melo, S. Nickel, and F. Saldanha-da-Gama. Dynamic multi-commodity capacitated facility location: a mathematical modeling framework for strategic supply chain planning. *Computers & Operations Research*, 33(1):181–208, 2006.

M. T. Melo, S. Nickel, and F. Saldanha-da-Gama. Facility location and supply chain management–a review. *European Journal of Operational Research*, 196(2):401–412, 2009.

Prorail and NS. Beter en meer, 2014. URL `https://www.rijksoverheid.nl/documenten/rapporten/2014/03/28/bijlage-1e-operationele-uitwerking-prorail-en-ns-beter-en-meer`.

J. A. Rappold and B. D. van Roo. Designing multi-echelon service parts networks with finite repair capacity. *European Journal of Operational Research*, 199:781–792, 2009.

C. S. ReVelle and H. A. Eiselt. Location analysis: A synthesis and survey. *European Journal of Operational Research*, 165(1):1–19, 2005.

A. Sarac, R. Batta, and C. M. Rump. A branch-and-price approach for operational aircraft maintenance routing. *European Journal of Operational Research*, 175(3):1850–1869, 2006.

L. V. Snyder. Facility location under uncertainty: a review. *IIE Transactions*, 38(7):547–564, 2006.

D. D. Tönissen, J. J. Arts, and Z.-J. Shen. Maintenance location routing for rolling stock under line and fleet planning uncertainty. *Transportation Science*, In press.

J. M. van den Akker, P. C. Bouman, J. A. Hoogeveen, and D. D. Tönissen. Decomposition approaches for recoverable robust optimization problems. *European Journal of Operational Research*, 251(3):739–750, 2016.

J. C. W. van Ommeren and A. F. Bumb. Locating repair shops in a stochastic environment. *Computers & Operations Research*, 33:1575–1594, 2006.

S. Xie, X. Chen, Z. Wang, Y. Ouyang, K. Somani, and J. Huang. Integrated planning for multiple types of locomotive work facilities under location, routing, and inventory considerations. *Interfaces*, 46(5): 391–408, 2016a.

W. Xie, Y. Ouyang, and K. Somani. Optimizing location and capacity for multiple types of locomotive maintenance shops. *Computer-Aided Civil and Infrastructure Engineering*, 31(3):163–175, 2016b.

B. Zeng and L. Zhao. Solving two-stage robust optimization problems using a column-and-constraint generation method. *Operations Research Letters*, 41(5):457–461, 2013.