RESEARCH ARTICLE OPEN ACCESS

An Exact Algorithm for the Hazardous Orienteering Problem

Roberto Montemanni[1] | Derek H. Smith[2]

[1]Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Reggio Emilia, Italy | [2]Computing and Mathematics, University of South Wales, Pontypridd, UK

Correspondence: Roberto Montemanni (roberto.montemanni@unimore.it)

Received: 12 November 2024 | **Revised:** 20 March 2025 | **Accepted:** 21 March 2025

Keywords: hazardous applications | mixed integer linear programming | model-based algorithms | orienteering problems

ABSTRACT

The hazardous orienteering problem is the topic of this study. It is a variant of the well-studied orienteering problem, where a vehicle, given a maximum mission time, has to select and visit customers out of a set of requests, aiming at maximizing the total profit associated with the customers selected. In the hazardous version of the problem we consider, the customers are associated with parcels that have to be collected, and some of them might explode during the transportation, after having been picked up. The probability of explosion depends on the characteristics of the parcel and on the time spent by the dangerous parcel itself on the vehicle. If an explosion happens, the collected profit is totally lost. The target becomes then to select the tour that maximizes the expected profit, taking into account the probability of catastrophic events happening. In this article, we propose an exact solving approach based on a mixed integer linear programming model which is dynamically modified by adding new constraints. The computational results substantially improve the state-of-the-art for the problem.

1 | Introduction

The orienteering problem (OP) is a classic optimization problem modeling several real applications, mainly in logistics. It was first introduced in [1] and [2] in the 1980s. There is a single vehicle, leaving from and returning to a depot, that has to plan its mission to visit a set of customers, each one characterized by a profit collected upon visiting the location. A maximum time for the mission is provided, therefore it is not possible to visit all the customers, and the combination with the total highest profit has to be selected. We refer the interested reader to the surveys [3–6] for an exhaustive review of the vast literature available for the problem.

The hazardous orienteering problem (HOP) was first introduced by Santini and Archetti in [7]. In this problem, some of the customers are associated with parcels that have to be collected by the vehicle, and some of these parcels might explode during the transportation. The probability of explosion depends on the time spent by the dangerous parcel on the vehicle, and by the characteristics of the parcel. The probability of explosion of a parcel is independent of the probability of explosion of the other parcels. In case of an explosion, the profit of the mission is neutralized. The target becomes then to select the tour that maximizes the expected profit, taking into account the probability of catastrophic events happening.

Real-world applications of the HOP can be found, for example, in the context of cash-in-transit problems [8], to model the viewpoint of a security transport company that has to decide which requests to accept and which to reject. In these problems, one or more vehicles are used to replenish ATM machines, or as a reversed application, to collect the daily cash incomes at stores. In the past, these problems has been treated as special OP problems [9], although the main focus was purely operational and rarely on the possibility that the vehicles were assaulted for a robbery. Exceptions are multi-period cash-in-transit problems, where such a risk is considered and mitigated by trying to have varying routes for each period [10], and the Risk-Constrained Cash-in-Transit Vehicle Routing Problem (RCTVRP) [11] where the design of the minimal-cost routes is subject to a budget constraint on the risk of robbery. Another application is in the context of hazardous material transportation, where the risk of an accident (e.g., batteries can catch fire due to poor packaging and mechanical stress) is low but might lead to catastrophic events [12].

To position the HOP in the existing literature, it is possible to observe that it falls in that category of stochastic problems in which some low-probability event has a large negative impact. A first example of these problems can be traced back to Sherali et al. [13], where routing of hazardous materials is taken into account. The target of the work is about the impact of the catastrophic event into the system, and the objective is to minimize the expected total impact. More recently two related variants of the 0-1 Knapsack Problem has been introduced: the Distributionally Robust Stochastic Knapsack Problem [14] and the 0-1 Time-Bomb Knapsack Problem [15, 16]. In the first problem, items are presented sequentially to the decision maker, who can stop and consider the knapsack as final, or accept the newly presented object. In case of acceptance, the previously unknown true weight of the object is revealed and if its weight is larger than the residual capacity, all the profit accumulated is lost. In the second problem, a subset of the available items are time-bombs that can explode according to a given probability. If one of these items is selected and explodes, the total profit is lost. The objective is to maximize the expected profit. Previously in the literature other forms of stochastic OPs had been treated, although they are intrinsically different from the HOP. The OP with stochastic profits was first introduced by İlhan et al. in [17]. A random distribution associated with the profit is associated with each customer, and the target is to maximize the probability that the total collected profit is at least equal to a given threshold. In the probabilistic OP (POP) (Archetti et al. [18]), the presence of customers is associated with probabilities, and the objective is to devise an a priori tour that maximizes the difference between the total expected profit and the expected traveling cost, both affected by the probabilistic presence of the customers. Once the solution is applied, customers not requiring a visit are revealed and they are skipped from the tour. Solution techniques for the POP are discussed in [19, 20]. The OP with Stochastic Travel and Service Times (Campbell et al. [21]) is finally characterized by nondeterministic times, and the target is again to maximize the expected total profit while taking probabilities into account. Specialized solving approaches are found in [22, 23].

The methods currently available for the HOP are those discussed in [7]. The authors presented a mixed-integer nonlinear programming model (it will be described in Section 3) with some valid inequalities and several techniques to obtain heuristic solutions or upper bounds for the value of the optimal solution. In particular, nontrivial upper bounds were obtained by solving linear approximations, piecewise linear approximation, continuous relaxation and elementary constraint relaxation of the nonlinear model. Heuristic solutions were instead obtained by attacking the nonlinear model or some manipulations of it with black-box solvers, or by a dynamic programming approach carefully designed for the problem.

The main contribution of the article is represented by a new model-based exact algorithm relying on a logarithmic transformation.

The article is organized as follows. In Section 2, the HOP is formally introduced, whereas a nonlinear model is described in Section 3. The new model-based exact algorithm is presented and analyzed in Section 4. Computational experiments, covering all the instances available from the previous literature, are presented and commented in Section 5. Conclusions and a proposal for future work are found in Section 6.

2 | Problem Definition

The classic OP can be defined as follows. Let $V = \{1, \dots, n\}$ represents the customer locations and let $V' = V \cup \{0\}$, where 0 represents the depot, be the set of vertices of a complete digraph $G = (V', A)$, with A being the set of arcs. Each vertex $i \in V$ has a profit $p_i \in \mathbb{R}^+$, which is collected upon visit, and each arc $(i, j) \in A$ is characterized by a deterministic travel time $t_{ij} \in \mathbb{R}^+$. Let $T \in \mathbb{R}^+$ represent the maximum mission time, that is, the tour travel time cannot exceed T. The goal is to select a subset of the customers and to devise an elementary tour with duration less than or equal to T that starts and end at the depot and visits the selected customers, while maximizing the total profit collected.

Starting from the definition of the OP, the HOP is characterized by $H \subseteq V$ containing the customers with hazardous goods, and for each vertex $i \in H$ a parameter $\lambda_i \in \mathbb{R}^+$ is given. The parameter λ_i represents the rate of an exponential random variable, and is used to model the probability of explosion along the route for the parcel eventually collected at customer i. The parameter is used to define as $F_i(\tau) = 1 - e^{-\lambda_i \tau}$ $(\tau \geq 0)$ the cumulative distribution function of the exponential random variable associated with i. It represents the probability that the catastrophic event will happen within a time $\tau \geq 0$ starting at the moment the vehicle visits vertex i and loads the parcel. In case of explosion, all the profit associated with the vehicle is lost, generating a null profit (the material costs associated with the loss of the vehicle and collateral damages is not considered). The target of the optimization is to maximize the *expected* total profit, that takes into account explosion probabilities. Given a feasible tour visiting the customers in $K \subseteq V$, the expected profit can be expressed as follows:

$$\mathbb{E}(K) = \left(\sum_{i \in K} p_i \right) \left(\prod_{i \in K \cap H} e^{-\lambda_i \tau_i} \right) \qquad (1)$$

where τ_i is the time the parcel of customer i spends on the vehicle in an optimized tour visiting the customers of K. Notice that formula (1) exploits the independence of the probability of explosion of a parcel from the probability of explosion of the other parcels.

An example of a HOP instance is found in Figure 1.

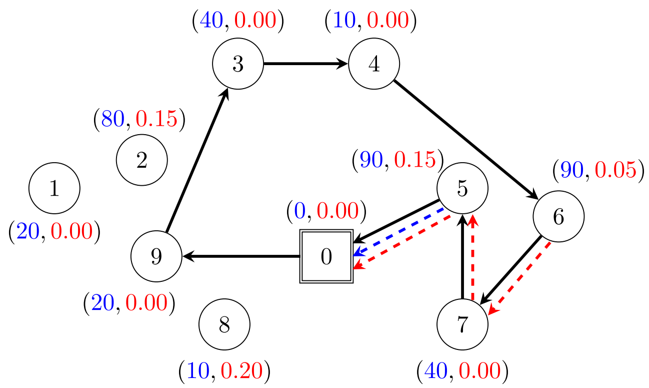

FIGURE 1 | Example of a HOP instance, where prizes are represented in blue and λs (to calculate propensities to explode) in red. The values of travel times and of the maximum mission time T are omitted to keep the figure easy to read. The selected route is depicted in black, whereas in dashed blue and red represent the times spent on the vehicle by the potentially exploding items collected at customer 5 and 6, respectively. These times will contribute to the expected profit of a solution. Notice that some customers have been left out either for time constraints or because they were considered too risky due to their propensity to explode or their location within the route.

3 | A Nonlinear Model

A nonlinear model, originally proposed in [7], can be devised starting from the definition of expected profit provided in (1). The following decision variables are used by the model:

- $x_{ij} \in \{0, 1\}$ for $(i, j) \in A$, taking value 1 iff the vehicle traverses arc (i, j);
- $y_i \in \{0, 1\}$ for $i \in V$, taking value 1 iff the vehicle visits vertex i;
- $\tau_i \in \mathbb{R}^+$ for $i \in V$, representing the travel time along the path that leads from i back to the depot, if i is visited, or 0 otherwise.

The resulting model that will be referred to as NL in the remainder of the article is the following one:

$$(NL) \quad \max \left(\sum_{i \in V} p_i y_i \right) \left(\prod_{i \in H} e^{-\lambda_i \tau_i} \right) \tag{2}$$

$$\text{s.t.} \quad y_0 = 1 \tag{3}$$

$$\sum_{j:(i,j) \in A} x_{ij} = y_i \qquad i \in V' \tag{4}$$

$$\sum_{j:(j,i) \in A} x_{ij} = y_i \qquad i \in V' \tag{5}$$

$$\sum_{(i,j) \in A} t_{ij} x_{ij} \leq T \tag{6}$$

$$\tau_i \geq \tau_j + t_{ij} - M(1 - x_{ij}) \qquad (i, j) \in A, i \neq j \tag{7}$$

$$\tau_i \geq t_{i0} x_{i0} \qquad i \in V \tag{8}$$

$$x_{ij} \in \{0, 1\} \qquad (i, j) \in A \tag{9}$$

$$y_i \in \{0, 1\} \qquad i \in V' \tag{10}$$

$$\tau_i \in [0, T - t_{0i}] \qquad i \in V \tag{11}$$

The objective function (2) maximizes the expected profit, by implementing directly the definition provided in (1). Constraint (3) is a technical constraint forcing the visit of the depot. Equalities (4) and (5) are flow-balance constraints and impose that if a location is entered by the vehicle, it also has to be exited, and vice-versa. Constraint (6) is a budget constraint ensuring that the duration of the tour does not exceed the maximum allowed mission time T. Constraints (7) bound τ_i to be at least $\tau_j + t_{ij}$ once customer j is visited right after customer i by the vehicle (the constant M can be set to $T - t_{j0} + t_{ij}$ as suggested in [7]). Inequalities (8) bound τ_i in case customer i is the last customer visited by the vehicle before it goes back to the depot. Constraints (9–11) finally define the domain of the variables.

4 | Solving the Nonlinear Model NL With a Linear Solver

The method we propose is iterative and is based on the logarithmic transformation, already used in the past to linearize concepts within linear programming optimization [16, 24], and—in the context of the HOP—in [7] to derive a relaxation for the nonlinear model NL:

$$\prod_{k \in K} \alpha_k = \beta \Leftrightarrow \sum_{k \in K} \log(\alpha_k) = \log(\beta) \tag{12}$$

This property allows the solution of the HOP as a mixed integer linear programming model which is iteratively evolved by adding new constraints (see inequality (17)).

We can transform the model NL into a new mixed integer linear program—we will refer to the latter as L—by changing the objective function (2) to

$$\max \sum_{i \in V} p_i y_i \tag{13}$$

In such a transformation, the probabilistic component of the original objective function is omitted, obtaining therefore a linear model for a deterministic OP with side constraints to calculate the value of τ variables. To exploit model L to solve the original problem NL, an iterative mechanism is introduced, where an upper and a lower bound for the optimal solution of the HOP are evolved by iteratively solving model L, which is enriched by new constraints restricting the search space at each iteration.

The overall solving procedure is summarized in Algorithm 1 and described as follows. The two variables LB and UB, containing a lower and an upper bound for the optimal cost of the original HOP are initialized at lines 1 and 2 together with the container for the best solution retrieved (line 3). The linear model L is then built starting from model NL and changing the objective function to (13), which neglects the probability aspects of the problem (line 4).

Model L is fed to a solver and by exploiting callback functions (line 5), actions are undertaken each time a new feasible solution

ALGORITHM 1 | HOP Solver.

1 LB=0;
2 UB=$+\infty$;
3 $BestSol = \emptyset$;
4 Build model L as (13) s.t. (3)–(11);
5 **while** *solving model L (with a given maximum computation time)* **do**
6   **if** *a feasible solution $\hat{y}$ is retrieved* **then**
7   Build model I as (15) s.t. (3)-(11), (14);
8     $\hat{\tau}$ = Assignment in an optimal solution of model I;
9   **if** $\left(\sum_{i\in V} p_i\hat{y}_i\right)\left(\prod_{i\in H} e^{-\lambda_i\hat{\tau}_i}\right) > LB$ **then**
10       $BestSol = Sol$;
11       $LB = \left(\sum_{i\in V} p_i\hat{y}_i\right)\left(\prod_{i\in H} e^{-\lambda_i\hat{\tau}_i}\right)$;
12   **end**
13   **if** $\sum_{i\in V} p_i\hat{y}_i < UB$ **then**
14       $UB = \sum_{i\in V} p_i\hat{y}_i$;
15   **end**
16   Add (16) and (17) to model L;
17   **end**
18 **end**
19 **return** $BestSol, LB, UB$;

$\hat{y}$ is retrieved. Namely, a new integer linear program I is created and solved to find a sequencing of the customers identified by $\hat{y}$ that minimizes explosion probability. This is achieved starting from model NL, adding the constraints

$$y_i = \hat{y}_i \quad i \in V \tag{14}$$

to lock the customer assignment, and optimizing the new objective function

$$\max \sum_{i\in H} -\lambda_i\tau_i \tag{15}$$

that aims at minimizing explosion probabilities (line 7). The timing variables $\hat{\tau}$ associated with an optimal sequencing are stored (line 8) and if the solution $\hat{y}, \hat{\tau}$ improves the best-known lower bound once evaluated according to the original HOP objective function, both $BestSol$ and LB are updated (lines 9–12). If the cost of solution $\hat{y}$ evaluated according to the objective function neglecting probabilities is lower than UB, then UB is updated to the new function (lines 13–15). Later in Proposition 1, it will be shown why the cost calculated is a valid upper bound for the HOP.

Because an optimal sequencing for the customers of $\hat{y}$ was found, the following inequality can be added to L to avoid the same combination of y variables in the future (line 16):

$$\sum_{i\in V, \hat{y}_i=0} y_i + \sum_{i\in V, \hat{y}_i=1} (1-y_i) \geq 1 \tag{16}$$

Moreover, the following inequality is added to model L to constrain the probability of an explosion to take only values that can eventually lead to solution with a better value, according to the HOP objective function (line 16):

$$\sum_{i\in H} -\lambda_i\tau_i > \log_e \frac{LB}{UB} \tag{17}$$

The formal validity of inequality (17) will be demonstrated in Proposition 2.

The algorithm terminates either when the search space is exhausted or when the maximum allowed computation time is hit, and the relevant information about solutions are returned at line 19.

The following theoretical results are necessary to demonstrate the formal correctness of the approach we propose.

Model L is a relaxation of model NL, and therefore it provides a valid upper bound UB for NL:

Proposition 1. *Given an HOP instance, the following inequality holds*:

$$\max \sum_{i\in V} p_i y_i \text{ s.t. } (3)-(11)$$
$$\geq \max \left(\sum_{i\in V} p_i y_i\right)\left(\prod_{i\in H} e^{-\lambda_i\tau_i}\right) \text{ s.t. } (3)-(11)$$

Proof. The inequality follows from the observation that $\prod_{i\in H} e^{-\lambda_i\tau_i} \leq 1$ by definition. □

Proposition 1 validates the upper bound appearing at lines 13 and 14 of Algorithm 1.

Proposition 2. *Any feasible solution y of NL such that*

$$\left(\sum_{i\in V} p_i y_i\right)\left(\prod_{i\in H} e^{-\lambda_i\tau_i}\right) > LB$$

fulfills the following inequality:

$$\sum_{i\in H} -\lambda_i\tau_i > \log_e \frac{LB}{UB}$$

Proof.

$$\left(\sum_{i\in V} p_i y_i\right)\left(\prod_{i\in H} e^{-\lambda_i\tau_i}\right) > LB \Rightarrow$$
$$\Rightarrow \log_e\left(\sum_{i\in V} p_i y_i\right) + \sum_{i\in H} -\lambda_i\tau_i > \log_e LB \Rightarrow$$
$$\Rightarrow \sum_{i\in H} -\lambda_i\tau_i > \log_e LB - \log_e\left(\sum_{i\in V} p_i y_i\right) \Rightarrow$$
$$\Rightarrow \sum_{i\in H} -\lambda_i\tau_i > \log_e \frac{LB}{\sum_{i\in V} p_i y_i} \Rightarrow$$
$$\Rightarrow \sum_{i\in H} -\lambda_i\tau_i > \log_e \frac{LB}{UB}$$

□

Proposition 2 validates the inequality (17) appearing at line 16 of Algorithm 1.

To enhance the linear relaxation of the model and speed up the original solving schema described in Algorithm 1, the generalized subtour elimination constraints (as presented in [7]) are added to our implementation:

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} \geq y_k \ \forall S \subseteq V, k \in S$$

It is notoriously impractical to enumerate and add all of these inequalities, as they are exponential in number. Following the general approach in the literature, we generate them upon violation, as described in [7]. Given a fractional solution (x^*, y^*), an auxiliary graph $G^* = (V^*, A^*)$ is created, with $V^* = \{i \in V' : y_i^* > 0\}$ and $A^* = \{(i,j) : i,j \in V^*, i \neq j\}$. A capacity of value x_{ij}^* is associated with each arc (i,j). The vertices of $V^* \setminus \{0\}$ are sorted by nondecreasing order of y_i^* and examined in this order. For each vertex i, a 0-i max-flow/min-cut problem is solved. If the maximum flow is lower than y_i^*, then a new inequality is added to the model according to the corresponding cut S_i ($i \in S_i, 0 \in V^* \setminus S_I$). The capacity of arc $(0,i)$ is increased by the quantity $1 - \sum_{j \notin S_i} \sum_{k \in S_i} x_{jk}^*$ after the calculations for vertex i, to avoid the generation of the same cut more than once.

5 | Computational Experiments

In this section, some computational experiments are conducted to compare the new algorithm we propose with the methods available in the literature. We first briefly introduce the benchmark set adopted in Section 5.1 and then (Section 5.2) we compare the new method introduced in Section 4 with the approaches available in the literature.

5.1 | Benchmark Instances

Some HOP benchmark instances were generated in [7] starting from the instances originally proposed in Tsiligirides [2] for the classic OP. Distances and maximum mission times are inherited from these classic OP instances. The following approach was followed to generate the new instances (starting from 49 OP instances with 19, 30, or 31 customers). Hazardous customers are selected at random and are $\lfloor \alpha \cdot m \rfloor$, where m is the number of customers in the original instance and $\alpha \in \{0.1, 0.2, 0.3, 0.4\}$ is a parameter. For each hazardous customer i, the value λ_i is chosen uniformly from the interval $[0.05, 0.1]$, whereas the profit p_i is calculated as $\beta \cdot q_i$, where q_i is the original profit and $\beta \in \{2, 3, 4, 5\}$. This makes these customers more attractive, because they are penalized by their hazardous nature. In this way, a total of 784 instances is available and can be downloaded from the repository [25].

5.2 | Experimental Results

The algorithms discussed in Section 4 were implemented in Python, and Gurobi 11.0 [26] was used to solve all the integer linear programs faced during the computation, making use of callbacks. The experiments were run on a computer equipped with an Intel Core i7 12700F running at 2.1GHz and 32 GB of RAM. The methods discussed in [7], which are the baseline for our comparison, were instead run on an Intel Xeon CPU running at 2.4 GHz and equipped with 4 GB of RAM, using Gurobi 9.0 as a linear solver and Baron 22.1 [27] as a nonlinear solver. All the experiments reported are based on a maximum computation time of 3600 s for each method/instance combination.

The method discussed in Section 4 is compared with those available in the literature, and presented in [7]: under *Previous methods* we report only the best among the lower bounds obtained by solving the nonlinear model NL, by solving the nonlinear model obtained from NL by changing (2) into its logarithm with and without some valid inequalities, or by running a dynamic programming approach, and the best among the upper bounds obtained by solving linear approximations, piecewise linear approximation, continuous relaxation, and elementary constraints relaxation of the nonlinear model NL. Lower bounds (LB), upper bounds (UB), and percentage optimality gaps (*Gap %*) calculated as $\frac{UB-LB}{UB}$, and the percentage of optimal solutions retrieved (*Opt %*), are reported. Under *New method* we summarized the results obtained by the approach described in Section 4, for which we report the same information plus the computation time in seconds (*Sec*). The results are summarized in Table 1, where they are aggregated according to different criteria, and averaged consequently. In detail, we aggregate according to the number of customers n, to the mission time limit T, and to the values of α and β (see Section 5.1). Overall results are also presented in the last row of the table. Instance-wise results can be retrieved at [28].

The results presented in Table 1 suggest that the new method can successfully cope with the instances considered. In the given time, the large majority of the instances was solved to optimality and the optimality gap remains relatively low even for the instances that were not closed by the approach. From the table, it emerges that the new factors introduced to describe the HOP, highly affect the performance of the solver: higher percentages of hazardous customers (α) and the increase of profits for the same customers (β) makes these instances more difficult. Both these factors make the role of uncertainty more prominent in the optimization, and the expected profit itself becomes more dependent on probability, and in turn is more affected by very profitable customers associated with a high risk. The number of customers n and the mission time T—that are inherited from the classic OP—also have a role in the difficulty of the instances, as expected. It is especially clear from the table how longer mission times offer more possible sequencings of the customers, and in turn more difficult instances. Overall results indicate however very marginal optimality gaps and the vast majority of the instances solved to optimality in the given time of one hour. The improvements over the previous methods available are twofold: the method we propose is the first method fully based on linear programming able to systematically provide high-quality lower and upper bounds. On the other hand, the results achieved indicate that the method is superior to several methods proposed before, those being either independent heuristics or upper bounding techniques.

6 | Conclusion

A new approach for the HOP has been discussed in this article. The problem is a variant of the well-studied OP, where a vehicle, given a maximum mission time, has to select and visit

TABLE 1 | Comparison among previous methods for the HOP and the new approach we propose.

Instances			Previous methods [7]				New method				
Par	Val	# Inst	LB	UB	Gap %	Opt %	LB	UB	Gap %	Opt %	Sec
n	19	176	249.42	324.69	22.65	0.57	249.42	249.42	0.00	100.00	30.82
	30	288	140.20	163.51	15.85	8.33	140.20	143.49	1.52	95.14	320.44
	31	320	472.54	532.61	12.26	4.69	472.54	479.36	1.02	95.00	314.60
T	5	16	14.10	16.58	9.50	81.25	14.10	14.10	0.00	100.00	0.01
	10	16	28.00	35.96	22.45	0.00	28.00	28.00	0.00	100.00	0.04
	15	48	121.70	157.83	23.60	0.00	121.70	121.70	0.00	100.00	2.28
	20	48	143.99	197.74	27.15	0.00	143.99	143.99	0.00	100.00	23.09
	23	16	189.65	252.61	24.80	0.00	189.65	189.65	0.00	100.00	22.89
	25	48	176.84	242.38	25.86	0.00	176.85	176.85	0.00	100.00	99.97
	27	16	212.92	310.74	30.06	0.00	212.92	212.92	0.00	100.00	37.13
	30	48	208.81	281.08	24.29	0.00	208.81	208.81	0.00	100.00	195.69
	32	16	269.63	356.50	22.72	0.00	269.63	269.63	0.00	100.00	32.84
	35	48	250.39	306.14	17.64	0.00	250.39	250.39	0.00	100.00	208.57
	38	16	323.83	395.39	17.37	0.00	323.83	323.83	0.00	100.00	14.77
	40	48	287.88	349.76	17.33	0.00	287.88	299.59	3.58	89.58	516.55
	45	32	396.29	467.86	14.55	3.13	396.29	407.86	1.95	93.75	333.91
	46	16	147.64	177.77	16.10	0.00	147.64	161.53	6.95	81.25	812.94
	50	32	299.14	343.86	12.77	0.00	299.14	299.14	0.00	100.00	421.30
	55	32	316.57	369.26	12.97	0.00	316.57	320.05	1.56	93.75	513.34
	60	32	344.11	395.58	11.91	0.00	344.11	353.57	1.47	93.75	336.56
	65	32	365.66	399.81	8.32	3.13	365.66	368.30	1.04	96.88	186.62
	70	32	382.27	418.23	8.57	0.00	382.27	384.48	0.92	96.88	206.05
	73	16	210.99	231.03	8.28	6.25	210.99	210.99	0.00	100.00	40.60
	75	32	399.59	441.78	9.46	0.00	399.59	406.51	1.69	93.75	358.83
	80	32	408.51	438.15	7.20	21.88	408.51	411.00	0.38	96.88	258.88
	85	32	410.31	450.08	8.41	9.38	410.31	414.49	1.83	90.63	548.71
	90	16	621.92	674.05	7.20	18.75	621.92	638.58	2.09	93.75	234.65
	95	16	629.39	687.36	8.23	12.50	629.39	645.09	2.07	87.50	479.21
	100	16	619.06	666.52	6.87	6.25	619.06	626.11	0.96	93.75	265.05
	105	16	619.98	648.67	4.49	31.25	619.98	625.35	0.83	93.75	264.50
	110	16	617.07	659.75	6.42	18.75	617.07	633.14	2.49	81.25	727.47
α	0.1	196	334.74	363.67	9.93	6.63	334.74	336.08	0.21	99.49	39.44
	0.2	196	311.28	356.60	14.38	5.61	311.28	311.64	0.17	99.49	87.41
	0.3	196	288.44	347.54	18.35	4.59	288.44	294.57	1.21	95.41	328.14
	0.4	196	267.01	333.57	21.00	3.57	267.01	275.16	2.31	90.31	557.17
β	2	196	287.82	307.46	8.67	10.20	287.82	287.82	0.00	100.00	14.86
	3	196	294.00	332.45	14.00	5.10	294.01	294.27	0.17	99.49	92.29
	4	196	304.54	364.91	18.63	3.57	304.54	306.46	0.69	96.94	261.56
	5	196	315.10	396.57	22.36	1.53	315.10	328.89	3.04	88.27	643.45
Overall		784	300.37	350.35	15.91	5.10	300.37	304.36	0.98	96.17	253.04

customers out of a set of requests, with the target of maximizing the total profit collected at the selected customers. In the hazardous version considered in this study, parcels are collected at customers, and some of these parcels might explode during the transportation, with catastrophic consequences, and a profit of 0. The probability of explosion is proportional to the time spent by each dangerous parcel on the vehicle. The target becomes then to select the tour that maximizes the expected profit. We proposed an approach based on a partial linearization and a consequent dynamic mixed integer linear program, which is evolved based on new upper and lower bounds for the optimal value of the objective function, and ultimately leads to an optimal solution. The approach only uses a linear programming solver, making the method particularly viable. The experimental results corroborate the potential of the new method, showing a consistent reduction on the best-known optimality gap for the instances considered. Future work is to see the potential of the new approach on other problems with similar characteristics.

Author Contributions

The two authors contributed equally to all the aspects of the article.

Acknowledgments

Open access publishing facilitated by Universita degli Studi di Modena e Reggio Emilia, as part of the Wiley - CRUI-CARE agreement.

Disclosure

The authors have nothing to report.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

The data that support the findings of this study are available in HOP at https://github.com/alberto-santini/hazardous-orienteering-problem/. These data were derived from the following resources available in the public domain: https://github.com/alberto-santini/hazardous-orienteering-pr, https://github.com/alberto-santini/hazardous-orienteering-problem/.

References

1. B. L. Golden, L. Levy, and R. Vohra, "The Orienteering Problem," *Naval Research Logistics* 34, no. 3 (1987): 307–318.

2. T. Tsiligirides, "Heuristic Methods Applied to Orienteering," *Journal of the Operational Research Society* 35 (1984): 797–809.

3. C. Archetti, M. G. Speranza, and D. Vigo, "Vehicle Routing Problems With Profits," in *Vehicle Routing: Problems, Methods, and Applications*, ed. P. Toth and D. Vigo (Society for Industrial and Applied Mathematics, 2014), 273–298.

4. M. Fischetti, J. J. Salazar-Gonzalez, and P. Toth, *The Generalized Traveling Salesman and Orienteering Problems* (Springer, 2007), 609–662.

5. A. Gunawan, H. C. Lau, and P. Vansteenwegen, "Orienteering Problem: A Survey of Recent Variants, Solution Approaches and Applications," *European Journal of Operational Research* 255, no. 2 (2016): 315–332.

6. P. Vansteenwegen, W. Souffriau, and D. van Oudheusden, "The Orienteering Problem: A Survey," *European Journal of Operational Research* 209, no. 1 (2011): 1–10.

7. A. Santini and C. Archetti, "The Hazardous Orienteering Problem," *Networks* 81, no. 2 (2023): 235–252.

8. A. Chiussi, C. Orlis, R. Roberti, and W. Dullaert, "ATM Cash Replenishment Under Varying Population Coverage Requirements," *Journal of the Operational Research Society* 73 (2021): 869–887.

9. C. Orlis, N. Bianchessi, R. Roberti, and W. Dullaert, "The Team Orienteering Problem With Overlaps: An Application in Cash Logistics," *Transportation Science* 54, no. 2 (2020): 470–487.

10. G. Fröhlich, K. Doerner, and M. Gansterer, "Secure and Efficient Routing on Nodes, Edges, and Arcs of Simple-Graphs and of Multi-Graphs," *Networks* 76, no. 4 (2020): 431–450.

11. L. Talarico, K. Sörensen, and J. Springael, "Metaheuristics for the Risk-Constrained Cash-In-Transit Vehicle Routing Problem," *European Journal of Operational Research* 244, no. 2 (2015): 457–470.

12. D. Lisbona and T. Snee, "A Review of Hazards Associated With Primary Lithium and Lithium-Ion Batteries," *Process Safety and Environmental Protection* 89, no. 6 (2011): 434–442.

13. H. Sherali, L. Brizendine, T. Glickman, and S. Subramanian, "Low Probability High Consequence Considerations in Routing Hazardous Material Shipments," *Transportation Science* 31, no. 3 (1997): 237–251.

14. J. Cheng, E. Delage, and A. Lisser, "Distributionally Robust Stochastic Knapsack Problem," *SIAM Journal of Optimization* 24, no. 3 (2014): 1485–1506.

15. M. Monaci, C. Pike-Burke, and A. Santini, "Exact Algorithms for the 0-1 Time-Bomb Knapsack Problem," *Computers and Operations Research* 145 (2022): 105848.

16. R. Montemanni and D. H. Smith, "Model-Based Algorithms for the 0–1 Time-Bomb Knapsack Problem," *Computers and Operations Research* 178 (2025): 107010.

17. T. İlhan, S. Iravani, and M. Daskin, "The Orienteering Problem With Stochastic Profits," *IIE Transactions* 40, no. 4 (2008): 406–421.

18. E. Angelelli, C. Archetti, C. Filippi, and M. Vindigni, "The Probabilistic Orienteering Problem," *Computers and Operations Research* 81 (2017): 269–281.

19. X. Chou, L. M. Gambardella, and R. Montemanni, "A Tabu Search Algorithm for the Probabilistic Orienteering Problem," *Computers and Operations Research* 126 (2021): 105107.

20. R. Montemanni and D. H. Smith, "A Model-Based Algorithm for the Probabilistic Orienteering Problem," *Computers and Operations Research* 176 (2025): 106947.

21. A. M. Campbell, M. Gendreau, and B. W. Thomas, "The Orienteering Problem With Stochastic Travel and Service Times," *Annals of Operations Research* 186 (2011): 61–81.

22. V. Papapanagiotou, R. Montemanni, and L. M. Gambardella, "Objective Function Evaluation Methods for the Orienteering Problem With Stochastic Travel and Service Times," *Journal of Applied Operational Research* 6, no. 1 (2014): 16–29.

23. V. Papapanagiotou, R. Montemanni, and L. M. Gambardella, "Hybrid Sampling-Based Evaluators for the Orienteering Problem With Stochastic Travel and Service Times," *Journal of Traffic and Logistics Engineering* 3, no. 2 (2015): 108–114.

24. R. Montemanni, V. Leggieri, and C. Triki, "Mixed Integer Formulations for the Probabilistic Minimum Energy Broadcast Problem in Wireless Networks," *European Journal of Operational Research* 190, no. 2 (2008): 578–585.

25. A. Santini, "Hazardous Orienteering Problem," 2022, https://github.com/alberto-santini/hazardous-orienteering-problem.

26. Gurobi Optimization, LLC, "Gurobi 11.0," 2024, https://www.gurobi.com.

27. The Optimization Firm, "Baron Solver," 2024, https://minlp.com/baron-solver.

28. R. Montemanni, "HOP - Hazardous Orienteering Problem," 2024, https://github.com/Monte14/HOP/.