Research Article

An Inverse Robust Optimisation Approach for a Class of Vehicle Routing Problems under Uncertainty

Liang Sun\textsuperscript{1,2} and Bing Wang\textsuperscript{1}

\textsuperscript{1}School of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China
\textsuperscript{2}School of Transportation and Vehicle Engineering, Shandong University of Technology, Zibo 250049, China

Correspondence should be addressed to Bing Wang; 1441380568@qq.com

Received 29 October 2015; Accepted 14 December 2015

Academic Editor: Seenith Sivasundaram

Copyright © 2016 L. Sun and B. Wang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There is a trade-off between the total penalty paid to customers (TPC) and the total transportation cost (TTC) in depot for vehicle routing problems under uncertainty (VRPU). The trade-off refers to the fact that the TTC in depot inevitably increases when the TPC decreases and vice versa. With respect to this issue, the vehicle routing problem (VRP) with uncertain customer demand and travel time was studied to optimise the TPC and the TTC in depot. In addition, an inverse robust optimisation approach was proposed to solve this kind of VRPU by combining the ideas of inverse optimisation and robust optimisation so as to improve both the TPC and the TTC in depot. The method aimed to improve the corresponding TTC of the robust optimisation solution under the minimum TPC through minimising the adjustment of benchmark road transportation cost. According to the characteristics of the inverse robust optimisation model, a genetic algorithm (GA) and column generation algorithm are combined to solve the problem. Moreover, 39 test problems are solved by using an inverse robust optimisation approach: the results show that both the TPC and TTC obtained by using the inverse robust optimisation approach are less than those calculated using a robust optimisation approach.

1. Introduction

Vehicle routing problems (VRP) are a crucial issue in industrial and system engineering, and involve routing a fleet of vehicles from a depot to service a set of customers. If one or both of demand and edge costs, including transportation cost and travel time are uncertain, the variant VRP becomes VRPU. It can be considered to be one of the more important problems—both in theory and practice—in the fields of transportation, distribution, and logistics. Hence, the search for an optimisation approach for VRPU to improve the operation efficiency of distribution centres (DC) and perfect the quality of customer service is significant.

At present, the optimisation approach for solving VRPU includes stochastic [1–6] and robust optimisation approaches. The former focuses on minimising the expected cost or maximising the expected revenue, but the expected values may be not the actual utility function influencing decision-makers. This is because the risk preference of decision-makers is neglected. Besides, it is difficult to reduce the fluctuations in a solution and their sensitivity to constraint conditions. Unlike the stochastic optimisation approach, the risk preference of decision-makers is taken into account in the robust optimisation approach. Meanwhile, a solution to robust optimisation explicitly incorporates conflicting objectives of solution robustness (the optimal solution is close to optimal for all realisation of the input data) and model robustness (the optimal solution is almost feasible for all realisation of the input data).

As for the situations above, the robust optimisation approach to solving VRPU has become a new research hotspot as its use to solve this kind of problem has increased in recent years. According to the current theory on robust optimisation [7–9], the robust optimisation approach for solving VRPU can be classified into two types.

1.1. The Robust Optimisation Approach for the Optimal Solution. This approach mainly focuses on the feasibility of the optimal solution from a set of scenarios. Sungur et al. introduced a robust optimisation approach to solve the
VRP with uncertain demand [10]. Gounaris et al. derived the robust optimisation counterparts of several deterministic capacitated vehicle routing problem (CVRP) formulations. Meanwhile, they developed robust rounded capacity inequalities for two broad classes of demand supports and it is shown how they are efficiently separated [11]. Lee et al. investigated VRP with deadlines, where the objective is to satisfy the requirements of specific customers with minimum travel distances while regarding both customer deadlines and vehicle capacities [12]. Agra et al. proposed two new formulations for robust VRP: the first extended the well-known resource inequalities by using adjustable robust optimisation, while the other generalised a path inequalities formulation to cater for the uncertain context [13]. Owing to the transportation cost changing in a given interval, Montemanni et al. established a robust optimisation model with fluctuation minimisation of the TTC as the optimisation objective. Besides, the branch and bound approach is applied to solve the model [14]. With the consideration of the uncertain demands in the event of a disaster, a robust optimisation model is established by Zhang et al. to establish emergency resource allocation in the event of a natural disaster. In addition, with the interval data representing the demand in each disaster area, the adjustable robust optimisation idea is used to build a robust optimisation model considering the uncertainty of demand [15, 16]. Jafari-Eskandari et al. solved the milk-run issue under the condition of demand uncertainty by using robust optimisation approach. Besides, particle swarm optimisation approach is also applied [17]. With regard to transporting the wounded and relief materials after an earthquake, Najafi et al. solved the problem by adopting a hierarchical multiobjective optimisation approach to obtain the robust optimisation solution with some optimisation objectives, including minimising waiting time for treatments of the wounded, minimising the response time in the demand for relief materials, and minimising the number of vehicles used [18].

1.2. The Robust Optimisation Approach for the Optimal Objective Function Value. The objective function of the model is a utility function that embodied a trade-off between the optimisation objective and variability therein. According to the expectation-variance theory proposed by Markowitz [19], List et al. established a formulation and a solution procedure for fleet sizing under uncertainty in future demand and operating conditions. Their formulation concentrated on robust optimisation by using a partial moment measure of risk [20–25]. A robust optimisation model to minimise the risk of each route for the haulage of dangerous goods is proposed by Qin et al. which includes the influence of various meteorological conditions on accident frequency and the severity thereof [26]. As for the VRP under uncertain demand and cost, Sun and Wang proposed a robust optimisation model with expectation-combination deviation and analysed the related properties of the robust optimisation model considering the different strength of preference among decision-makers for deviation expectations [27]. The robust optimisation model with expectation-deviated expectation is mainly used to obtain the robust optimisation solution meeting the preference of decision-making risk for customers on the whole, whereas the model cannot be used directly to formulate a transport plan for a particular kind of customer. In response to this problem, Sun and Wang proposed a robust optimisation model with conditional expectation-deviated conditional expectation to handle special customers among the whole, so as to determine the robust optimisation model for the optimal transport scheme. Besides, they also analysed the properties of the model [28, 29].

Although the robust optimisation approach can overcome some shortcomings found during the optimal solution of VRPU by using stochastic optimisation approach, both approaches just seek an eclectic optimal solution among several optimisation objectives if these objectives present a trade-off phenomenon. Unlike these two approaches, DC often adjusts the prices of enterprise resources, including products or services to simultaneously optimise these objectives with trade-off as far as possible in practical business operations management. The price adjustment here refers to the fact that different transport prices affecting all realisation of input data are adopted according to the various service requirements of customers to maintain a low total transportation cost (TTC).

Based on the analysis above, an inverse robust optimisation approach is proposed to solve such problems by combining inverse optimisation and robust optimisation according to the actual operation process of DC. The main idea of inverse robust optimisation is to establish a two-stage model, where the first stage is to solve a robust optimisation solution which minimises the total penalty by using a robust optimisation approach. While the second-stage model aims to ensure that the TTC under the robust optimisation solution is not more than the minimum TTC obtained from the first-stage model by minimising the adjustment of benchmark road transportation costs. Based on the characteristics of the robust optimisation solution in the first-stage model, the crossover and mutation operators are redesigned to solve the first-stage model by GA. While based on the properties of the second-stage model, a column generation approach is adopted to solve the second-stage model. Owing to the inverse robust optimisation approach proposed in this study being aimed at solving the VRPU under a single-stage decision, the research status of the robust optimisation approach for the VRPU under multistage decisions [30] is not discussed here.

The differences between the inverse robust optimisation approach proposed in this research and the robust optimisation approach reported in the literature are as follows.

(1) The Optimisation Principle Is Different. The main idea of the robust optimisation approach is to find an optimal solution in a feasible region to keep the solution feasible for the data in an uncertain dataset almost everywhere or the optimal objective function value obtained optimal to the uncertain dataset almost everywhere. It reflects a kind of optimisation mechanism, while the inverse robust optimisation approach proposed here indicates a regulatory mechanism. It aims to explore how to adjust the benchmark road transportation cost to minimise the corresponding TTC.
of the robust optimisation solution which has obtained the minimum total penalty.

(2) The Assumptions Are Different. The upper change limit of each customer demand is the maximum load of vehicles in some robust optimisation approaches. The customer demand is often more than the maximum load of vehicles in the actual operation process of DC. Thus, the assumption is loosened that the demand produced by customers could be more than the maximum load of vehicles in this research. If the demand of customers is more than the maximum load of vehicles, the demand is met by transporting the payload in several cycles or stages.

The rest of this paper is organised as follows.

Section 2 provides the model used in the inverse robust optimisation approach. Section 3 provides the solution method used in the inverse robust optimisation approach. Section 4 summarises the results of the theoretical analysis and further verifies them by analysing several examples. Finally, the authors conclude and make recommendations for future research directions in Section 5.

2. Problem Description and Formulation

2.1. Basic Definitions

Definition 1 (benchmark road transportation cost). This refers to a reference for setting the road transportation cost of different DCs or depots. It can be formulated by government or formed by the market over long-term competition.

Definition 2 (customer penalty). It is the penalty paid by the DC to the customers based on their contract if the delivery vehicle arrives to the customers later than the allowed latest time of arrival. The greater the customer penalty, the lower the distribution service level provided by the DC; hence, it is more likely to induce customer loss.

Definition 3 (model of demand uncertainty set $U_D$). It is assumed that $C = \{1, 2, \ldots, n\}$ is a set of customers and each customer can meet the condition of $i \in C$. Besides, the value of demand is in the range $[\bar{r}_i, \bar{r}_i + o_i]$, where $o_i$ represents the maximum deviation from the nominal demand value $\bar{r}_i$. Moreover, a nonnegative integer $\Gamma$ is introduced as a parameter to control the degree of robustness for the demand uncertainties; then the uncertainty set of demand data is given as follows:

$$U_D = \left\{ \bar{r} \in R^{|C|} \mid \bar{r}_i = \bar{r}_i + o_i, \sum_{k \in C} w_i \leq \Gamma, 0 \leq w_i \leq 1, \forall i \in C \right\}.$$  

Definition 4 (model of travel time uncertainty $U_T$). Vertex 0 represents a depot and $V = C \cup \{0\}$.

$E = \{(i, j) \mid i, j \in V \text{ and } i \neq j\}$ is a set of arcs, and each arc meets the condition whereby $(i, j) \in E$. In addition, the travel time takes values from $[\bar{t}_{ij}, \bar{t}_{ij} + \theta_{ij}]$, where $\theta_{ij}$ represents the maximum deviation from the nominal travel time $\bar{t}_{ij}$. Moreover, a nonnegative integer $\Lambda$ is introduced as a parameter to control the degree of robustness of the travel time uncertainties. Then the uncertainty set of travel time data can be expressed as follows:

$$U_T = \left\{ t \in R^{|E|} \mid \bar{t}_{ij} = \bar{t}_{ij} + \theta_{ij} \rho_{ij}, \sum_{i \in C} \sum_{j \in C} \rho_{ij} \leq \Lambda, \ 0 \leq \rho_{ij} \leq 1, \forall (i, j) \in E \right\}.$$  

2.2. Problem Description. $G = (V, E)$ is a graph, where $V = \{0, 1, 2, \ldots, n\}$ is the set of vertices and $E = \{(i, j) \mid i \in V, j \in V, i \neq j\}$ is the set of edges. Besides, vertex 0 represents a depot without demand, where a fleet of $K$ identical vehicles with the capacity of $Q$ is located. Each edge $(i, j)$ has a nonnegative benchmark transportation cost $c_{ij}$.

It is assumed that the travel time between each pair of customers and the demands thereof are uncertain. Two types of uncertainty sets with adjustable parameters are considered for the possible realisation of travel time and demand. The problem satisfies the following conditions:

(a) Each customer is assigned to at least one route.

(b) The demand of every customer is totally satisfied and it can be serviced by more than one vehicle.

(c) Each route must begin and end at the depot and it has to visit at least one customer.

(d) The total demand serviced by each vehicle is not more than its capacity.

(e) The vehicle can arrive before the earliest service time allowed by customers, and it can also arrive after the latest service time; however, it has to pay a penalty to the customer if it arrives later than the latest service time.

The inverse robust optimisation proposed in this paper has two goals: (1) to find an optimal solution to minimise the TTC by making the solution feasible for each travel time and demand defined in the uncertainty sets and (2) to find a preferred transportation cost to ensure that the corresponding TTC of the optimal solution obtained above is not more than the minimum TTC in the first-stage model. At the same time, the values of transportation cost differ from the benchmark transportation cost as little as possible.

2.2.1. Parameters and Decision Variables

(1) Parameters. Parameters in the first-stage model are as follows:

- $c_{li}$ is the unit penalty paid to the customers if the vehicle arrives after $u_i$.
- $[e_{ij}, u_{ij}]$ is the earliest, and latest, service time allowed by customer $i$.
- $s_{eri}$ is the unloading time of a vehicle at customer $i$. 

$S$ is a subset of $V$.
$ar{t}_{ij}$ is the nominal travel time between customer $i$ and customer $j$.
$M$ is a large number.
$F$ is a subset of $U_D$.
$B$ is a subset of $U_T$.
One has $(u_i)^F = \{1, i \in F; 0, \text{ otherwise}\}$.
One has $(m_{ij})^B = \{1, (i, j) \in B; 0, \text{ otherwise}\}$.

Parameters in the second-stage model are as follows:
$c_{ij}$ is the benchmark transportation cost between customer $i$ and customer $j$;
$X^1$ is the robust optimisation solution for the MTPC in the first-stage model;
$x_{ij}^{k1} = \{1, x_{ij}^k = 1 \text{ in } X^1; 0, x_{ij}^k = 0 \text{ in } X^1\}$;
$x_{ij}^{kq} = \{1, x_{ij}^k = 1 \text{ in the } q\text{th feasible solution}; 0, x_{ij}^k = 1 \text{ in the } q\text{th feasible solution}\}$;
$X^*$ stands for the corresponding robust optimisation solution to the minimum TTC in the first-stage model;
$x_{ij}^k = \{1, x_{ij}^k = 1 \text{ in } X^*; 0, x_{ij}^k = 0 \text{ in } X^*\}$.

(2) Decision Variables. The following are decision variables in the first-stage model:
$x_{ij}^k = \{1, \text{ if vehicle } k \text{ travels from customer } i \text{ to customer } j; 0, \text{ otherwise}\}$;
$s_{ik}$ is the arrival time of vehicle $k$ at customer $i$, and $s_{ik} \in \mathbb{R}^+$;
$(s_{ik})^B$ represents $s_{ik}$ in $B$;
$d_{ik}$ is the proportion of the demand of customer $i$ serviced by vehicle $k$, where $d_{ik} \geq 0$;
$(d_{ik})^F$ denotes $d_{ik}$ in $F$.

Decision variables in the second-stage model are as follows:
$a_{ij}$ represents a decision variable. It is the increased amount by which the corresponding TTC of $X^1$ is no more than that minimising TTC, where $a_{ij} \geq 0$.
$b_{ij}$ represents a decision variable. It is the increased amount by which the corresponding TTC of $X^1$ is no more than that minimising TTC, where $0 \leq b_{ij} \leq c_{ij}$.

2.3. Inverse Robust Optimisation Formulation for VRPU

2.3.1. First-Stage Model. One has

\[ \text{(Stage 1) min } C_2 \ast \sum_{i=0}^{n} \sum_{k=1}^{m} \max(s_{ik} - u_i, 0) \] (3)

Constraints (4) to (6) are classical routing constraints. Constraint (4) ensures that each vertex is visited at least once. Besides, constraint (5) is a flow conservation constraint, while constraint (6) is a subtour-elimination constraint, constraint (7) implies that customer $i$ can be served by vehicle $k$ only if $k$ passes through $i$, constraints (8) and (9) make all solutions feasible at each travel time and for each demand defined in the uncertainty sets, and constraint (10) means that the quantity delivered by each vehicle is not more than its capacity.

2.3.2. Second Stage Model. There are two possible situations if the benchmark road transportation cost is adjusted too much. One situation is that the enterprise fails to be sustained, while the other is that customers cannot afford the high freight charges so that they are forced to abandon the services provided by the DC. Thus, the second-stage model focuses on minimising the adjustment range of benchmark road transportation cost:

\[ \text{(Stage 2) min } \sum_{i \in C} \sum_{j \in C} a_{ij} + \sum_{i \in C} \sum_{j \in C} b_{ij} \] (14)
\[ \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} \geq \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1}, \] 
\[ q = 1, \ldots, p, \]
\[ \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \leq \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} c_{ij} x_{ij}^{k1}, \]
\[ \beta_{ij} \leq c_{ij}, \ i = 0, \ldots, n; \ j = 0, \ldots, n, \]
\[ \alpha_{ij} \geq 0, \ i = 0, \ldots, n; \ j = 0, \ldots, n, \]
\[ \beta_{ij} \geq 0, \ i = 0, \ldots, n; \ j = 0, \ldots, n, \]

where \( p \) is the number of feasible solutions in the first-stage model. It is stipulated that the first feasible solution is the robust optimisation solution, namely, \( X^1 \) in the first-stage model. Optimisation objective (14) entails minimising the total adjustment of benchmark road transportation cost. Constraints (15) and (16) are used to ensure that the corresponding TTC to the robust optimisation solution \( X^1 \) is not more than the minimum TTC in the first-stage model. Constraint (17) ensures that the decreased amount of adjustment of benchmark road transportation costs cannot exceed \( c_{ij} \).

2.3.3. The Properties of the Inverse Robust Optimisation Model

Property 1. If \( T = \{(i, j) \mid x_{ij}^{k1} = 1\} \), then \( \sum_{i \in T} \sum_{j \in T} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} = \sum_{i \in E} \sum_{j \in E} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \).

Proof. One has

\[ \sum_{i \in E} \sum_{j \in E} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} = \sum_{i \in T} \sum_{j \in T} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} \]
\[ + \sum_{i \in E-T} \sum_{j \in E-T} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} \]

owing to

\[ \sum_{i \in E-T} \sum_{j \in E-T} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} = 0, \]
\[ \sum_{i \in T} \sum_{j \in E-T} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1} \]
\[ = \sum_{i \in E} \sum_{j \in E} \left( c_{ij} + \alpha_{ij} - \beta_{ij} \right) x_{ij}^{k1}. \]

It is indicated by Property 1 that the optimal values of TTC obtained by adjusting only the corresponding \( c_{ij} \) in \( T \) and adjusting all \( c_{ij} \) are the same. Property 1 can be used to reduce the number of columns of the coefficient matrix in the second-stage model during calculation.

Property 2. The \( q \) feasible solutions in the first-stage model are divided into \( M \) groups, where the minimum value of the corresponding TTC of the feasible solution in the \( r \)th group is \( \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \); then constraint (15) can be modified to

\[ \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \geq \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1}, \]
\[ t = 1, \ldots, M. \]

Proof. For the \( r \)th group of constraints,

\[ \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \geq \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1}, \]
\[ r = 1, \ldots, p, \]

where \( p_r \) is the number of constraints in Group \( r \) and \( s_r \) is the minimum value of the corresponding TTC to the feasible solution in the \( r \)th group of constraints. So it is true that

\[ \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1} \geq \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{k=1}^{m} (c_{ij} + \alpha_{ij} - \beta_{ij}) x_{ij}^{k1}, \]
\[ q = 1, \ldots, p. \]

Property 2 explains that, after appropriately grouping the constraints in constraint (15), the corresponding constraints for the minimum TTC of the feasible solutions in each group are selected to construct new constraints. The number of rows of the coefficient matrix in the second-stage model can be reduced during computation by using Property 2.

2.3.4. Complexity. The size of coefficient matrix with regard to constraint (15) is \( 2 \times q \times |V|^2 \). By virtue of Properties 1 and 2, the size of the coefficient matrix with regard to constraint (15) can be reduced to \( 2 \times M \times (|V| + |K| - 1) \). The computational
complexity of this inverse robust optimisation problem can be reduced significantly using Properties 1 and 2. It is interesting to note that we cannot find an optimal solution for first-stage model using a heuristic algorithm in most cases, whereas we can find an optimal solution for second-stage model using Property 2 in certain situations.

3. Solution Methods

The process of selecting vehicle routes allows the selection of any combination of customers in determining the delivery route for each vehicle. Therefore, the robust counterpart of split delivery vehicle routing problem is also a combinatorial optimisation problem where the number of feasible solutions for the problem increases exponentially with the number of customers to be serviced. In addition, the robust counterpart of the problem is closely related to the travelling salesman problem where an out and back tour from a central location is determined for each vehicle. Since there is no known polynomial algorithm that will find the optimal solution in every instance, the robust counterpart of the problem is considered NP-hard. For such problems, the use of heuristics is considered a reasonable approach in finding solutions.

3.1. The Concept behind the Algorithm

(1) The VRPU is converted into the corresponding robust problem which must be solved.

(2) A GA is applied to solve the first-stage model.

(3) Properties 1 and 2 are used to simplify the coefficient matrix of the second-stage model, and then a column generation algorithm [31] is used to solve the second-stage model.

3.2. The Algorithm and Its Steps. The algorithm is operated in the following steps.

Step 1. A VRPU is transformed into the robust counterpart of VRPU ready for solution.

Step 2 (initialising the parameters). To select an appropriate population \( N_p \), crossover probability \( p_c \), and mutation probability \( p_m \), a stochastic approach is applied to generate the initial population. Besides, the optimal individual is labelled as the current optimal solution set, and the evolutionary generation is set to \( G = 0 \).

Step 3 (selection operator). Let \( G = G + 1 \); the mechanism combining a roulette and elitist strategy is adopted to select the individual according to its adaptive value.

Step 4 (crossover operator). The crossover operator (see Section 3.3.3) proposed in this study is used.

Step 5 (mutation operator). The mutation operator (see Section 3.3.4) proposed in this study is used.

Step 6. Update the current optimal solutions.

Step 7 (terminating conditional judgment). If the fixed number of generations is reached, the evolution is terminated; otherwise, the operation returns to Step 3 until this is satisfied.

Step 8. All individuals produced by solving the first-stage model by GA are recorded. Each component of \( X^* \), namely, \( x_{ij}^k \), which corresponded to the minimum TTC in the first-stage model, and each component of \( X^1 \), namely, \( x_{ij}^{k1} \), are recorded.

Step 9. First, \( x_{ij}^{k1} \), \( x_{ij}^{kq} \) and \( x_{ij}^{*} \) are substituted into the coefficient matrix in the second-stage model. Then the coefficient matrix is simplified by using Properties 1 and 2. Finally, a column generation algorithm (see Section 3.3.4) is applied to solve the second-stage model.

3.3. Detailed Description of the Steps in the Algorithm

3.3.1. Chromosome Representation. It is assumed that the number of vehicles is fixed, and the customer-based coding approach is applied to each chromosome. There are three characters representing each gene: the first is the serial number of the customer served by the vehicle and the number of the depot. If the character in first row is 0, it is a depot identifier; otherwise, it is the customer identifier. Besides, the second and third characters represented the earliest start time and the latest start time allowed by each customer. The advantage of this encoding method is that the subtours can be avoided during encoding, thus saving time during calculation.

3.3.2. Crossover Operator. The common crossover operators include ordered crossover (OX), partial mapped crossover (PMX), circulation crossover (CX), conventional single-point or two-point crossover, and crossover with edge recombination. All of these crossover operators fail to account for the characteristics of the objective function in Stage 1. In terms of the objective function in the first-stage model, it is necessary to reduce the time in which a vehicle returns to the depot after finishing its last service, so as to reduce the total penalty. Thereby, if the subroute with the minimum total penalty among individuals can be the subroute with the maximum total penalty among the optimal individuals, then the individual is more likely to be the optimal individual. With respect to this analysis, the new crossover operator of the GA is found as follows:

(1) Two individuals, A and B, are selected from the current population at random.

(2) The subroute with the minimum total penalty in A and B, as the reserve section thereof, is written to intermediate individuals \( A' \) and \( B' \).

(3) Crossover is performed. The gene in the reserve section of \( A' \) or \( B' \) is copied to its offspring individual, namely, New\( A'(B) \), while the genes on other gene locus are written in New\( B(A) \) according to their order of appearance on \( B(A) \).
(4) The individuals, after crossover, are decoded again according to constraints (4) to (13) to adjust the position of gene 0 so as to resolve the start and end times for arrival of each served customer and vehicle.

As shown in the example (see Figure 2), the shortest subroute satisfying the constraints on individual A is $0 \rightarrow 2 \rightarrow 4 \rightarrow 5 \rightarrow 0$, which serves as the reserve section of A, while the subroute meeting the conditions on individual B is $0 \rightarrow 3 \rightarrow 4 \rightarrow 0$, which is the reserve section of B.

3.3.3. **Mutation Operator.** The subroute which has highest total penalty in any individual determined whether or not the individual was optimal. The gene of the other subroutes in an individual can be exchanged with that of the subroute with the highest total penalty. In this way, the subroute with highest total penalty can be changed to maintain population diversity. Based on two-point mutation, the mutation operator ran as follows:

1. An individual A is selected at random.
2. A gene locus chosen arbitrarily from the subroutes with highest total penalty and another gene is selected arbitrarily from the other subroute.
3. The two genes are exchanged.
4. The individuals, after mutation, are decoded again in accordance with constraints (4) to (13) to adjust the position of gene 0 so as to recalculate the start and end time of arrival of the vehicles at each served customer.

Figure 3 shows that $0 \rightarrow 2 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 0$ is the subroute with the maximum total customer penalty for any individual, while $0 \rightarrow 1 \rightarrow 4 \rightarrow 0$ is the subroute with the minimum total customer penalty.

3.3.4. **Column Generation Algorithm.** The column generation algorithm [31] is applied to solve the second-stage model.

4. **Computational Results**

4.1. **Comparison between the Robust, Inverse, and Optimisation Approaches**

4.1.1. **Selecting Test Problems and Setting Parameters.** These computational experiments are performed on two sets of problems: the first is taken from the well-known Solomon problems [32]. In this case, we used problems R, C, and RC; the second is taken from http://www.uv.es/~belengue/sdvrp.html and used the Eil and $S^*D^*$ problems. Since Eil and $S^*D^*$ problems were originally designed for the SDVRP without earliest and latest start times, we assigned a latest start time to the maximum load of their vehicles so that the minimum TTC of RC201 could not be more than that of RC108. Besides, with the effect of constraint (16), the dec. value for TTC of various test problems are induced by real differences between the minimum TTC and the corresponding TTC to MTPC in various test problems.

The TTC of eil76 is not improved: owing to the difference between the corresponding TTC to MTPC and the minimum TTC being small, the TTC found with these two optimisation approaches is improved to an insignificant extent. Taking RC108 and RC201, for example, there is a fivefold difference in the maximum loads of their vehicles so that the minimum TTC of RC201 could not be more than that of RC108. Besides, with the effect of constraint (16), the dec. value for the TTC of RC201 is much less than that of RC108 in the second-stage model. The TTC of S76D1 and S76D4 is improved significantly owing to the reordering frequency of S76D1; that is to say, conveying the goods to customers directly from the depot, is less than that of S76D4. It is suggested that the MTPC of S76D1 and the corresponding TTC differed greatly. Thus, the maximum offsets of demand and travel time are 100 and 300, respectively. The levels of uncertainty of demand are the same (i.e., $\omega = 0.5$); meanwhile, the levels of uncertainty in the travel time are also the same (i.e., $\rho = 0.5$). In addition, the values of these parameters are as follows: population size $N_p = 100$; crossover probability $P_c = 0.6$; mutation probability $P_m = 0.05$; $\Gamma = 20$, $\Delta = 300$, and $C_t = 0.2$. All of the computational tests presented here were performed on a Pentium CPU B950 with 4 GB of RAM. The solution was implemented in MATLAB R2013a.
Table 1: Computational results: I-RO and S-RO.

<table>
<thead>
<tr>
<th></th>
<th>S-RO</th>
<th>I-RO</th>
<th>Dec. (%)</th>
<th>S-RO</th>
<th>I-RO</th>
<th>Dec. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>eil30</td>
<td>MTPC</td>
<td>802.6</td>
<td>802.6</td>
<td>—</td>
<td>S76D1</td>
<td>MTPC 1779.8</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>1856</td>
<td>1543.3</td>
<td>16.8%</td>
<td>TTC</td>
<td>2539</td>
</tr>
<tr>
<td>eilA76</td>
<td>MTPC</td>
<td>1788.4</td>
<td>1788.4</td>
<td>—</td>
<td>S76D4</td>
<td>MTPC</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3223</td>
<td>2925.5</td>
<td>9.2%</td>
<td>TTC</td>
<td>4414</td>
</tr>
<tr>
<td>eilB76</td>
<td>MTPC</td>
<td>857.7</td>
<td>857.7</td>
<td>—</td>
<td>S101D1</td>
<td>MTPC 550.8</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3422</td>
<td>3289.8</td>
<td>3.8%</td>
<td>TTC</td>
<td>3856</td>
</tr>
<tr>
<td>eilC76</td>
<td>MTPC</td>
<td>914.1</td>
<td>914.1</td>
<td>—</td>
<td>S101D3</td>
<td>MTPC 613.7</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>2924</td>
<td>2574</td>
<td>12%</td>
<td>TTC</td>
<td>5172</td>
</tr>
<tr>
<td>eilA101</td>
<td>MTPC</td>
<td>1571</td>
<td>1157.1</td>
<td>—</td>
<td>RC102</td>
<td>MTPC 1859.8</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3980</td>
<td>3473.3</td>
<td>12.7%</td>
<td>TTC</td>
<td>5090</td>
</tr>
<tr>
<td>eilB101</td>
<td>MTPC</td>
<td>809.2</td>
<td>809.2</td>
<td>—</td>
<td>RC104</td>
<td>MTPC 1701</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>4020</td>
<td>3203.7</td>
<td>20.3%</td>
<td>TTC</td>
<td>5296</td>
</tr>
<tr>
<td>R101</td>
<td>MTPC</td>
<td>2382.9</td>
<td>2382.9</td>
<td>—</td>
<td>RC108</td>
<td>MTPC 1718</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3927</td>
<td>3642.1</td>
<td>7.3%</td>
<td>TTC</td>
<td>5277</td>
</tr>
<tr>
<td>R103</td>
<td>MTPC</td>
<td>1894.4</td>
<td>1894.4</td>
<td>—</td>
<td>RC201</td>
<td>MTPC 1791.9</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3915</td>
<td>3514.2</td>
<td>10.2%</td>
<td>TTC</td>
<td>4172</td>
</tr>
<tr>
<td>R105</td>
<td>MTPC</td>
<td>2168.1</td>
<td>2168.1</td>
<td>—</td>
<td>RC203</td>
<td>MTPC 882.7</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3808</td>
<td>3320.7</td>
<td>12.8%</td>
<td>TTC</td>
<td>4456</td>
</tr>
<tr>
<td>C102</td>
<td>MTPC</td>
<td>1620.2</td>
<td>1620.2</td>
<td>—</td>
<td>RC205</td>
<td>MTPC 1285.4</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>4477</td>
<td>3849.4</td>
<td>14%</td>
<td>TTC</td>
<td>4631</td>
</tr>
<tr>
<td>C106</td>
<td>MTPC</td>
<td>2142.4</td>
<td>2142.4</td>
<td>—</td>
<td>C108</td>
<td>MTPC 1670</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>4439</td>
<td>3781.5</td>
<td>14.8%</td>
<td>TTC</td>
<td>4560</td>
</tr>
</tbody>
</table>

Table 2: Calculated results: IRO and MRO.

<table>
<thead>
<tr>
<th></th>
<th>M-RO</th>
<th>I-RO</th>
<th>Dec. (%)</th>
<th>M-RO</th>
<th>I-RO</th>
<th>Dec. (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>eil51</td>
<td>MTPC</td>
<td>102.8</td>
<td>73.9</td>
<td>28.1%</td>
<td>S51D2</td>
<td>MTPC 547</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>2035</td>
<td>1780</td>
<td>12.5%</td>
<td>TTC</td>
<td>2140</td>
</tr>
<tr>
<td>eilD76</td>
<td>MTPC</td>
<td>1005.2</td>
<td>1002.6</td>
<td>0.2%</td>
<td>S51D6</td>
<td>MTPC 585.1</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>3064</td>
<td>2773.5</td>
<td>9.9%</td>
<td>TTC</td>
<td>3668</td>
</tr>
<tr>
<td>C104</td>
<td>MTPC</td>
<td>555.6</td>
<td>505.6</td>
<td>9%</td>
<td>S76D3</td>
<td>MTPC 158</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>4521</td>
<td>4380.1</td>
<td>3.1%</td>
<td>TTC</td>
<td>3981</td>
</tr>
<tr>
<td>RC106</td>
<td>MTPC</td>
<td>2100.3</td>
<td>2064.8</td>
<td>1%</td>
<td>S101D5</td>
<td>MTPC 554</td>
</tr>
<tr>
<td></td>
<td>TTC</td>
<td>5173</td>
<td>4758.1</td>
<td>12.6%</td>
<td>TTC</td>
<td>5944</td>
</tr>
</tbody>
</table>

Note: Dec. denotes the percentage decrement when minimising TTC due to the introduction of the inverse robust optimisation approach.

Letting $\rho = 0$, it can be seen from Figure 4 that TTC increased as $\Gamma$ increased from 20 to 60. To analyse the reasons for the change in TTC, the concepts behind two types of routes were first proposed. The routes in this VRPU can be divided into two categories: one is the route directly connecting customer and vehicle, while the other is that where the customers are directly linked to each other (see Figure 6).

As the first-stage model aims to minimise the total penalty, route A increased and route B decreased with increasing $\Gamma$ with the same maximum load of vehicles. Owing to the TTC increase in route A exceeding the reduction along route B, the value of TTC increased continuously. In addition, the adjusted TTC is not more than the minimum for the corresponding value of $\Gamma$ in the second-stage model: the TTC obtained by I-RO fails to exceed that acquired by the S-RO.
Table 3: Relationships between solution and control algorithms.

<table>
<thead>
<tr>
<th>Solution algorithm</th>
<th>First-stage model</th>
<th>Second-stage model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Improved GA</td>
<td>Improved GA</td>
<td>Column generation algorithm with simplified decision variables and constraints</td>
</tr>
<tr>
<td>Control algorithm</td>
<td>Standard GA</td>
<td>Column generation algorithm without simplifying the decision variables and constraints</td>
</tr>
</tbody>
</table>

Table 4: Comparison of the calculated results in the Solomon standard test library.

<table>
<thead>
<tr>
<th>R102</th>
<th>S</th>
<th>I</th>
<th>C101</th>
<th>S</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTPC</td>
<td>68.28</td>
<td>63.34</td>
<td>MTPC</td>
<td>1002.7</td>
<td>802.9673</td>
</tr>
<tr>
<td>TTC</td>
<td>4053</td>
<td>3923</td>
<td>TTC</td>
<td>4478.4</td>
<td>4374.4</td>
</tr>
<tr>
<td>CPU(s)</td>
<td>82788</td>
<td>234.37</td>
<td>CPU(s)</td>
<td>482.866</td>
<td>249.273</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R104</th>
<th>S</th>
<th>I</th>
<th>C103</th>
<th>S</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTPC</td>
<td>24.65</td>
<td>16</td>
<td>MTPC</td>
<td>457.4016</td>
<td>373.9236</td>
</tr>
<tr>
<td>TTC</td>
<td>3940</td>
<td>3939</td>
<td>TTC</td>
<td>4496.9</td>
<td>4424.3</td>
</tr>
<tr>
<td>CPU(s)</td>
<td>972.59</td>
<td>205.077</td>
<td>CPU(s)</td>
<td>340.981</td>
<td>271.66</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R106</th>
<th>S</th>
<th>I</th>
<th>C105</th>
<th>S</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTPC</td>
<td>49.692</td>
<td>38.8</td>
<td>MTPC</td>
<td>950.3895</td>
<td>914.0186</td>
</tr>
<tr>
<td>TTC</td>
<td>3642.4</td>
<td>3529.8</td>
<td>TTC</td>
<td>4189</td>
<td>4224.9</td>
</tr>
<tr>
<td>CPU(s)</td>
<td>650.467</td>
<td>174.02</td>
<td>CPU(s)</td>
<td>265.825</td>
<td>251.258</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>R108</th>
<th>S</th>
<th>I</th>
<th>C107</th>
<th>S</th>
<th>I</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTPC</td>
<td>14.4779</td>
<td>9.0696</td>
<td>MTPC</td>
<td>1086.4</td>
<td>791.4168</td>
</tr>
<tr>
<td>TTC</td>
<td>3708.4</td>
<td>3435.2</td>
<td>TTC</td>
<td>4419.2</td>
<td>4505.5</td>
</tr>
<tr>
<td>CPU(s)</td>
<td>895.246</td>
<td>317.996</td>
<td>CPU(s)</td>
<td>266.218</td>
<td>168.98</td>
</tr>
</tbody>
</table>

Note. Solution algorithm is denoted by I, while the control algorithm is denoted by S.

Figure 1: Representation of each chromosome.

method. When $\Gamma = 50$, the TTC from the S-RO shows a greater decrease than the TTC from I-RO, which suggests that the optimising effect of I-RO is related to $\Gamma$ (e.g., in problem S76D2, the effect of using I-RO is optimal at $\Gamma = 50$).

Figure 5 shows that, at $\Gamma = 20$, the TTC obtained by I-RO is still not more than that from S-RO when $\omega$ changed from 0.2 to 1. Compared with Figure 1, the difference between the TTCs obtained by I-RO and S-RO is less obvious, because the variation of customer demand induced the changes in routes A and B so as to change the minimum TTC in the first-stage model. Figures 1 and 2 show that the optimal performance of TTC found by I-RO is high when the values of $\Gamma$ and $\omega$ are large on the condition that the travel time is certain.

On the assumption that $\omega = 0$, it can be seen from Figure 7 that when $\Lambda \leq 2500$, the vehicles detoured to avoid routes with delayed travel time to send the goods to customers while minimising the total penalty. For this reason, the number of route A and route B changed to induce a change in TTC. Owing to the adjusted TTC not being more than the minimum for the corresponding $A$ value in the second-stage model, the TTC obtained by I-RO is not more than that acquired by S-RO.

In Figure 8, setting $\Lambda = 600$, when $\rho$ changed from 0.2 to 1, there are many routes with no delay. In this period, vehicles sent goods to customers as early as possible by detouring to reduce the total penalty: inevitably, TTC presented an increasing trend. Although the TTC obtained by I-RO is not more than that acquired by S-RO during the change in $\rho$ from 0.2 to 1, the former fluctuated to some extent, which was brought about by the changing number of routes A and B. The effect on TTC was improved most obviously by I-RO when $\rho$ was 0.6.

4.3. The Performance of the Solution Algorithms. To verify the validity of the proposed solution algorithm, eight examples are run and compared to data from a control algorithm. It was proved that the proposed solution algorithm was effective by analysing the output and operational values. The relevant parameters were set as follows: the maximum offsets of uncertain demand were 100, while those of uncertain travel time were 300, the population size $N_p = 100$, crossover probability $P_c = 0.5$, mutation probability $P_m = 0.05$, $C_r = 0.1$, $\Gamma = 20$, $\Lambda = 600$, $\omega = 0.5$, and $\rho = 0.5$. The relationships between the solution algorithm proposed in this research and a control algorithm are summarised in Table 3.

As shown in Table 4, MTPC is significantly improved by I compared with S, while the performance of TTC changed little. Owing to the redesigned crossover and mutation operators needed to search the longest and shortest sub-routes
for the total penalty, the search time was longer than that of standard PMX and two-point mutation. However, the computation time for the second stage was reduced by simplifying the coefficient matrix in the second-stage model to offset the increased time in the first-stage model to some extent. Thus, the computation time for I was less than that when using S.

5. Conclusions

The inverse robust optimisation approach for solving the VRP under uncertain demand and travel time is studied here. It is found that the TPC and TTC could be improved simultaneously by minimising the judgment range of benchmark road transportation cost. Meanwhile, the optimal solution is feasible almost everywhere for all data in the uncertain datasets on demand and travel time. Considering the characteristics of the robust optimisation solution for the minimum total penalty and the properties of inverse robust optimisation model, GA and a column generation algorithm are combined to solve the inverse robust optimisation model. As indicated by theoretical analysis and simulation results, the inverse robust optimisation approach is able to improve two trade-off
optimisation indices at the same time during the optimisation thereof.

In terms of the characteristics of the models, it is inevitable that the TTC increased with increasing degree of conservatism of decision-makers in these solutions found using a single-objective robust optimisation to solve for MTPC. As a result, the operating burdens of businesses are increased. Although multi-objective robust optimisation could reduce the increase of the TTC in the depot induced by the decrease of TPC to some extent, the optimal performance with regard to total penalty was impaired. While an inverse robust optimisation approach could simultaneously improve TPC and TTC in the depot to greatly decrease the operating burden incumbent on a business by adjusting the benchmark road transportation cost, the results of simulations show that the delivery scheme formulated for DC using an inverse robust optimisation approach could be used when customer demand fluctuated or when the road travel time was affected by significant delays. In this way, business operating costs could be reduced, and at the same time, a high-quality service of DC is maintained.

Via robust inverse optimisation approach, decision-makers of depot find a robust optimal solution to minimise TPC, and then they can minimise business operating costs to maintain a high-quality service of DC.

With respect to the characteristics of the inverse robust optimisation model proposed in this research, the following aspects are expected to benefit from further research:

1. Although the proposed inverse robust optimisation approach can improve two contradictory indicators at the same time by adjusting the road transportation cost, it is still necessary to investigate how to optimise multiple conflicting indicators simultaneously to a greater extent.

2. Both inverse robust optimisation and robust optimisation approaches are conducted on the basis of an important assumption under which the number of customers would not change during optimisation. If the assumption is loosened to allow changes in the number of customers during optimisation, it is thought that future research exploring how to utilise the inverse robust optimisation approach in this case would be beneficial.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References


