Chapter 10
Vehicle Routing Problem

Vehicle routing problem (VRP) is concerned with finding efficient routes, beginning and ending at a central depot, for a fleet of vehicles to serve a number of customers. See Figure 10.1.

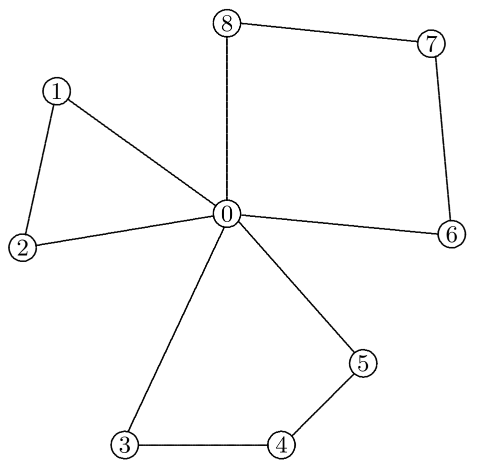

Fig. 10.1 A Vehicle Routing Graph

Due to its wide applicability and economic importance, VRP has been extensively studied. Practically, there are uncertain factors in VRP, such as demands of customers, travel times between customers, customers to be visited, locations of customers, capacities of vehicles, and number of vehicles available. This fact provides a motivation to study uncertain VRP. This chapter introduces some typical models for VRP.

10.1 Problem Description

We assume that: (a) a vehicle will be assigned for only one route on which there may be more than one customer; (b) a customer will be visited by one and only one vehicle; (c) each route begins and ends at the depot; and (d) each

B. Liu: Theory and Practice of Uncertain Programming, STUDFUZZ 239, pp. 147–155.
springerlink.com

customer specifies its time window within which the delivery is permitted or preferred to start.

Let us first introduce the following indices and model parameters:

$i = 0$: depot;

$i = 1, 2, \cdots, n$: customers;

$k = 1, 2, \cdots, m$: vehicles;

D_{ij}: the travel distance from customers i to j, $i, j = 0, 1, 2, \cdots, n$;

T_{ij}: the uncertain travel time from customers i to j, $i, j = 0, 1, 2, \cdots, n$;

S_i: the unloading time at customer i, $i = 1, 2, \cdots, n$;

$[a_i, b_i]$: the time window of customer i, where a_i and b_i are the beginning and end of the time window, $i = 1, 2, \cdots, n$, respectively.

In this book, the operational plan is represented by Liu's formulation [181] via three decision vectors $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{t}$, where

$\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$: integer decision vector representing n customers with $1 \le x_i \le n$ and $x_i \ne x_j$ for all $i \ne j$, $i, j = 1, 2, \cdots, n$. That is, the sequence $\{x_1, x_2, \cdots, x_n\}$ is a rearrangement of $\{1, 2, \cdots, n\}$;

$\boldsymbol{y} = (y_1, y_2, \cdots, y_{m-1})$: integer decision vector with $y_0 \equiv 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \equiv y_m$;

$\boldsymbol{t} = (t_1, t_2, \cdots, t_m)$: each t_k represents the starting time of vehicle k at the depot, $k = 1, 2, \cdots, m$.

We note that the operational plan is fully determined by the decision vectors $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{t}$ in the following way. For each k $(1 \le k \le m)$, if $y_k = y_{k-1}$, then vehicle k is not used; if $y_k > y_{k-1}$, then vehicle k is used and starts from the depot at time t_k, and the tour of vehicle k is $0 \to x_{y_{k-1}+1} \to x_{y_{k-1}+2} \to \cdots \to x_{y_k} \to 0$. Thus the tours of all vehicles are as follows:

$$
\begin{aligned}
&\text{Vehicle 1: } 0 \to x_{y_0+1} \to x_{y_0+2} \to \cdots \to x_{y_1} \to 0;\\
&\text{Vehicle 2: } 0 \to x_{y_1+1} \to x_{y_1+2} \to \cdots \to x_{y_2} \to 0;\\
&\qquad \cdots\\
&\text{Vehicle } m\text{: } 0 \to x_{y_{m-1}+1} \to x_{y_{m-1}+2} \to \cdots \to x_{y_m} \to 0.
\end{aligned}
\tag{10.1}
$$

It is clear that this type of representation is intuitive, and the total number of decision variables is $n + 2m - 1$. We also note that the above decision variables $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{t}$ ensure that: (a) each vehicle will be used at most one time; (b) all tours begin and end at the depot; (c) each customer will be visited by one and only one vehicle; and (d) there is no subtour.

Let $f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ be the arrival time function of some vehicles at customers i for $i = 1, 2, \cdots, n$. We remind readers that $f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ are determined by the decision variables $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{t}$, $i = 1, 2, \cdots, n$. Since unloading can start either immediately, or later, when a vehicle arrives at a customer, the calculation of $f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ is heavily dependent on the operational strategy. Here we assume that the customer does not permit a delivery earlier than the time window. That is, the vehicle will wait to unload until the beginning of the time window if it arrives before the time window. If a vehicle arrives at a customer after the beginning of the time window, unloading will start immediately. For each k with $1 \le k \le m$, if vehicle k is used (i.e., $y_k > y_{k-1}$), then we have

$$f_{x_{y_{k-1}+1}}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) = t_k + T_{0x_{y_{k-1}+1}} \tag{10.2}$$

and

$$\begin{aligned} f_{x_{y_{k-1}+j}}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) =& f_{x_{y_{k-1}+j-1}}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) \vee a_{x_{y_{k-1}+j-1}} \\ & + S_{x_{y_{k-1}+j-1}} + T_{x_{y_{k-1}+j-1}x_{y_{k-1}+j}} \end{aligned} \tag{10.3}$$

for $2 \le j \le y_k - y_{k-1}$. It follows from the uncertainty of travel times T_{ij}'s that the arrival times $f_i(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t})$, $i = 1, 2, \cdots, n$ are uncertain variables fully determined by (10.2) and (10.3).

Let $g(\boldsymbol{x},\boldsymbol{y})$ be the total travel distance of all vehicles. Then we have

$$g(\boldsymbol{x},\boldsymbol{y}) = \sum_{k=1}^{m} g_k(\boldsymbol{x},\boldsymbol{y}) \tag{10.4}$$

where

$$g_k(\boldsymbol{x},\boldsymbol{y}) = \begin{cases} D_{0x_{y_{k-1}+1}} + \sum\limits_{j=y_{k-1}+1}^{y_k-1} D_{x_j x_{j+1}} + D_{x_{y_k}0}, & \text{if } y_k > y_{k-1} \\ 0, & \text{if } y_k = y_{k-1} \end{cases}$$

for $k = 1, 2, \cdots, m$.

10.2 Stochastic Models

Now we assume that the travel times are random variables, and introduce stochastic distance minimization model and probability maximization model.

Stochastic Distance Minimization Model

If we hope that all customers are visited within their time windows with a confidence level α, then we have the following chance constraint,

$$\Pr\{a_i \le f_i(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\} \ge \alpha. \tag{10.5}$$

If we want to minimize the total travel distance of all vehicles subject to the time window constraint, then we have the following stochastic distance minimization model (Liu and Lai [183]),

$$\begin{cases} \min g(\boldsymbol{x},\boldsymbol{y}) \\ \text{subject to:} \\ \qquad \Pr\{a_i \le f_i(\boldsymbol{x},\boldsymbol{y},\boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\} \ge \alpha \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \ i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases} \tag{10.6}$$

Hybrid Intelligent Algorithm

In order to solve the stochastic models, we may employ the hybrid intelligent algorithm documented in Chapter 4 provided that the representation structure, initialization, crossover and mutation operations are revised as follows.

We represent an operational plan by the chromosome $V = (\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$, where the genes $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}$ are the same as the decision vectors. Without loss of generality, we also assume that the time window at the depot is $[a, b]$. This means that the gene $\boldsymbol{t}$ will be restricted in the hypercube $[a, b]^m$.

Let us show how to initialize a chromosome randomly. For gene $\boldsymbol{x}$, we define a sequence $\{x_1, x_2, \cdots, x_n\}$ with $x_i = i$, $i = 1, 2, \cdots, n$, and repeat the following process from $j = 1$ to n: generating a random position n' between j and n, and exchanging the values of x_j and $x_{n'}$. It is clear that $\{x_1, x_2, \cdots, x_n\}$ is just a random rearrangement of $\{1, 2, \cdots, n\}$. Then we obtain a gene $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)$. For each i with $1 \le i \le m-1$, we set y_i as a random integer between 0 and n. Then we rearrange the sequence $\{y_1, y_2, \cdots, y_{m-1}\}$ from small to large. We thus have a gene $\boldsymbol{y} = (y_1, y_2, \cdots, y_{m-1})$. Finally, for each i with $1 \le i \le m$, we set t_i as a random number on the time window $[a, b]$. Then we get a gene $\boldsymbol{t} = (t_1, t_2, \cdots, t_m)$. If the generated chromosome $V = (\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ is proven to be feasible, then it is accepted as a chromosome; otherwise we repeat the above process until a feasible chromosome is obtained.

Let us illustrate the crossover operator on the pair V_1 and V_2. We denote $V_1 = (\boldsymbol{x}_1, \boldsymbol{y}_1, \boldsymbol{t}_1)$ and $V_2 = (\boldsymbol{x}_2, \boldsymbol{y}_2, \boldsymbol{t}_2)$. First, we generate a random number c from the open interval $(0, 1)$ and define

$$\boldsymbol{t}_1' = c \cdot \boldsymbol{t}_1 + (1-c) \cdot \boldsymbol{t}_2, \quad \boldsymbol{t}_2' = (1-c) \cdot \boldsymbol{t}_1 + c \cdot \boldsymbol{t}_2.$$

The two children V_1' and V_2' are produced by the crossover operation as follows: $V_1' = (\boldsymbol{x}_1, \boldsymbol{y}_2, \boldsymbol{t}_1')$ and $V_2' = (\boldsymbol{x}_2, \boldsymbol{y}_1, \boldsymbol{t}_2')$.

We mutate the chromosome $V = (\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ in the following way. For the gene $\boldsymbol{x}$, we randomly generate two mutation positions n_1 and n_2 between 1 and n, and rearrange the sequence $\{x_{n_1}, x_{n_1+1}, \cdots, x_{n_2}\}$ at random to form a new sequence $\{x'_{n_1}, x'_{n_1+1}, \cdots, x'_{n_2}\}$. We thus obtain a new gene

$$\boldsymbol{x}' = (x_1, \cdots, x_{n_1-1}, x'_{n_1}, x'_{n_1+1}, \cdots, x'_{n_2}, x_{n_2+1}, \cdots, x_n).$$

Similarly, for gene $\boldsymbol{y}$, we generate two random mutation positions n_1 and n_2 between 1 and $m-1$, and set y_i as a random integer number y_i' between 0 and n for $i = n_1, n_1+1, \cdots, n_2$. We then rearrange the sequence

$$\{y_1, \cdots, y_{n_1-1}, y'_{n_1}, y'_{n_1+1}, \cdots, y'_{n_2}, y_{n_2+1}, \cdots, y_{m-1}\}$$

from small to large and obtain a new gene $\boldsymbol{y}'$. For the gene $\boldsymbol{t}$, we choose a mutation direction $\boldsymbol{d}$ in $\Re^m$ randomly. If $\boldsymbol{t} + M \cdot \boldsymbol{d}$ is not in the time window $[a, b]^m$, then we set M as a random number between 0 and M until it is in $[a, b]^m$, where M is a predetermined step length. If the above process cannot

Table 10.1 Travel Distance Matrix

LCTs	0	1	2	3	4	5	6	7
1	18							
2	14	20						
3	14	34	15					
4	21	55	41	28				
5	17	49	43	36	21			
6	21	57	55	51	36	16		
7	18	49	52	51	43	22	13	
8	14	22	35	44	55	41	43	32

Table 10.2 Random Travel Time Matrix (μ, σ^2)

LCTs	0	1	2	3
1	(50,25)			
2	(10,5)	(40,20)		
3	(50,25)	(10,5)	(40,20)	
4	(50,25)	(35,17)	(35,17)	(30,15)
5	(50,25)	(15,7)	(40,20)	(5,2)
6	(15,7)	(40,20)	(10,5)	(45,22)
7	(50,25)	(15,7)	(45,22)	(10,5)
8	(50,25)	(10,5)	(35,17)	(30,15)
LCTs	**4**	**5**	**6**	**7**
5	(30,15)			
6	(35,17)	(40,20)		
7	(30,15)	(10,5)	(40,20)	
8	(10,5)	(30,15)	(35,17)	(35,17)

yield a gene $\boldsymbol{t}$ in $[a, b]^m$ in a predetermined number of iterations, then we set $M = 0$. We replace the parent gene $\boldsymbol{t}$ with its child $\boldsymbol{t}' = \boldsymbol{t} + M \cdot \boldsymbol{d}$.

Example 10.1. We assume that there are 8 customers labeled "$1, 2, \cdots, 8$" in a company and one depot labeled "0". We assume that the travel distances among the depot and customers are listed in Table 10.1.

The travel times among the depot and customers are all normally distributed variables $\mathcal{N}(\mu, \sigma^2)$, which are given in Table 10.2.

The time windows of customers are shown in Table 10.3.

We suppose that the unloading times $(S_i, i = 1, 2, \cdots, 8)$ at locations are all 15 minutes.

We assign a confidence level $\alpha = 0.80$ at which all customers are visited within their time windows. If we want to minimize the total travel distance of all vehicles subject to the chance constraint, then we have a stochastic distance minimization model. A run of the hybrid intelligent algorithm (10000

Table 10.3 Time Windows of Customers

i	$[a_i, b_i]$	i	$[a_i, b_i]$	i	$[a_i, b_i]$
1	$[09:30, 14:10]$	2	$[09:20, 11:00]$	3	$[09:40, 11:10]$
4	$[09:20, 13:00]$	5	$[09:10, 15:20]$	6	$[08:20, 10:00]$
7	$[09:40, 12:10]$	8	$[09:20, 10:00]$		

cycles in simulation, 5000 generations in GA) shows that the best operational plan is:

Vehicle 1: depot$\rightarrow$ 6 $\rightarrow$ 7 $\rightarrow$depot, starting time = 8:45;
Vehicle 2: depot$\rightarrow$ 3 $\rightarrow$depot, starting time = 9:17;
Vehicle 3: depot$\rightarrow$ 8 $\rightarrow$ 1 $\rightarrow$ 2 $\rightarrow$ 5 $\rightarrow$ 4 $\rightarrow$depot, starting time = 8:35.

The total travel distance of the three vehicles is 221. Furthermore, when the obtained operational plan is performed, we have

$$\Pr\{a_i \le f_i(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{t}^*) \le b_i, i = 1, 2, \cdots, 8\} = 0.85.$$

Probability Maximization Model

If we hope that total travel distance does not exceed a fixed number $\overline{g}$, then we have a distance constraint $g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g}$. If we want to maximize the probability that all customers are visited within their time windows subject to the distance constraint, then we have the following probability maximization model,

$$\begin{cases} \max \Pr\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\} \\ \text{subject to:} \\ \qquad g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g} \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \ i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases} \tag{10.7}$$

Example 10.2. We set $\overline{g} = 240$. A run of the hybrid intelligent algorithm (10000 cycles in simulation, 5000 generations in GA) shows that the best operational plan is:

Vehicle 1: depot$\rightarrow$ 1 $\rightarrow$ 3 $\rightarrow$ 2 $\rightarrow$depot, starting time = 8:51;
Vehicle 2: depot$\rightarrow$ 6 $\rightarrow$ 5 $\rightarrow$ 7 $\rightarrow$ 4 $\rightarrow$depot, starting time = 9:04;
Vehicle 3: depot$\rightarrow$ 8 $\rightarrow$depot, starting time = 8:58.

When the obtained operational plan is performed, the total travel distance is 232, and

$$\Pr\{a_i \le f_i(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{t}^*) \le b_i, i = 1, 2, \cdots, 8\} = 0.88.$$

10.3 Fuzzy Models

Here we assume that the travel times are fuzzy variables instead of stochastic variables. Since the travel times are fuzzy variables, every customer will be visited at a fuzzy time.

Fuzzy Distance Minimization Model

If we hope that all customers are visited within their time windows with a confidence level α, then we have the following chance constraint,

$$\mathrm{Cr}\left\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\right\} \ge \alpha. \tag{10.8}$$

If we want to minimize the total distance traveled of all vehicles subject to time window constraints, then we have the following fuzzy distance minimization model (Zheng and Liu [336]),

$$\begin{cases} \min g(\boldsymbol{x}, \boldsymbol{y}) \\ \text{subject to:} \\ \qquad \mathrm{Cr}\left\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\right\} \ge \alpha \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \; i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases}$$

Example 10.3. Let us consider a fuzzy vehicle routing problem shown in Figure 10.1. We assume that the distance matrix is listed in Table 10.1 and the time windows customers are given in Table 10.3. We also assume that

Table 10.4 Fuzzy Travel Time Matrix

LCTs	0	1	2	3
1	(25,50,75)			
2	(5,10,15)	(20,40,60)		
3	(25,50,75)	(5,10,15)	(20,40,60)	
4	(25,50,75)	(17,35,53)	(17,35,53)	(15,30,45)
5	(25,50,75)	(7,15,23)	(20,40,60)	(2,5,8)
6	(7,15,23)	(20,40,60)	(5,10,15)	(22,45,68)
7	(25,50,75)	(7,15,23)	(22,45,68)	(5,10,15)
8	(25,50,75)	(5,10,15)	(17,35,53)	(15,30,45)
LCTs	4	5	6	7
5	(15,30,45)			
6	(17,35,53)	(20,40,60)		
7	(15,30,45)	(5,10,15)	(20,40,60)	
8	(5,10,15)	(15,30,45)	(17,35,53)	(17,35,53)

the travel times among the deport and customers are all triangular fuzzy variables as shown in Table 10.4. Finally, we suppose that the unloading times at the 8 locations are all 15 minutes.

If the confidence level α is 0.80, then a run of the hybrid intelligent algorithm (10000 cycles in simulation, 5000 generations in GA) shows that the best operational plan is:

Vehicle 1: depot$\to$ 6 $\to$ 7 $\to$ 4 $\to$ 5 $\to$depot, starting time = 9:44;
Vehicle 2: depot$\to$ 8 $\to$depot, starting time = 8:48;
Vehicle 3: depot$\to$ 2 $\to$ 3 $\to$ 1 $\to$depot, starting time = 9:21.

The total distance travelled by the three vehicles is 224. Furthermore, when the operational plan is performed, we have

$$\mathrm{Cr}\{a_i \le f_i(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{t}^*) \le b_i, i = 1, 2, \cdots, 8\} = 0.87.$$

Credibility Maximization Model

If we hope that total travel distance does not exceed a fixed number $\overline{g}$, then we have a distance constraint $g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g}$. If we want to maximize the credibility that all customers are visited within their time windows subject to the distance constraint, then we have the following credibility maximization model,

$$\begin{cases} \max \mathrm{Cr}\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\} \\ \text{subject to:} \\ \qquad g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g} \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \ i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases} \tag{10.9}$$

Example 10.4. We set $\overline{g} = 240$. A run of the hybrid intelligent algorithm (10000 cycles in simulation, 5000 generations in GA) shows that the best operational plan is

Vehicle 1: depot$\to$ 8 $\to$ 1 $\to$ 3 $\to$depot, starting time = 8:55;
Vehicle 2: depot$\to$ 6 $\to$ 5 $\to$ 4 $\to$ 7 $\to$depot, starting time = 8:51;
Vehicle 3: depot$\to$ 2 $\to$depot, starting time = 9:21.

When the optimal operational plan is performed, the total travel distance is 231, and

$$\mathrm{Cr}\{a_i \le f_i(\boldsymbol{x}^*, \boldsymbol{y}^*, \boldsymbol{t}^*) \le b_i, i = 1, 2, \cdots, 8\} = 0.96.$$

10.4 Hybrid Models

Now we suppose that the travel times are hybrid variables, and introduce hybrid distance minimization model and chance maximization model.

Hybrid Distance Minimization Model

If we hope that all customers are visited within their time windows with confidence level α, then we have the following chance constraint,

$$\mathrm{Ch}\left\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\right\} \ge \alpha. \tag{10.10}$$

If we want to minimize the total travel distance of all vehicles subject to the time window constraint, then we have the following hybrid distance minimization model,

$$\begin{cases} \min g(\boldsymbol{x}, \boldsymbol{y}) \\ \text{subject to:} \\ \qquad \mathrm{Ch}\left\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\right\} \ge \alpha \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \ i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases} \tag{10.11}$$

Chance Maximization Model

If we hope that total travel distance does not exceed a fixed number $\overline{g}$, then we have a distance constraint $g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g}$. If we want to maximize the chance that all customers are visited within their time windows subject to the distance constraint, then we have the following chance maximization model,

$$\begin{cases} \max \mathrm{Ch}\left\{a_i \le f_i(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t}) \le b_i, i = 1, 2, \cdots, n\right\} \\ \text{subject to:} \\ \qquad g(\boldsymbol{x}, \boldsymbol{y}) \le \overline{g} \\ \qquad 1 \le x_i \le n, \quad i = 1, 2, \cdots, n \\ \qquad x_i \ne x_j, \quad i \ne j, \ i, j = 1, 2, \cdots, n \\ \qquad 0 \le y_1 \le y_2 \le \cdots \le y_{m-1} \le n \\ \qquad x_i, y_j, \quad i = 1, 2, \cdots, n, \quad j = 1, 2, \cdots, m-1, \quad \text{integers.} \end{cases} \tag{10.12}$$

10.5 Exercises

Problem 10.1. Design a hybrid intelligent algorithm to solve hybrid models for vehicle routing problem (for example, the travel times are random and travel distances are fuzzy).

Problem 10.2. Build uncertain models for vehicle routing problem (for example, the travel times are uncertain variables with identification function (λ, ρ)), and design a hybrid intelligent algorithm to solve them.