Case Study

Large-scale Multi-vehicle path with Linear Programming

王业辉 戚培霖 郑嘉慧 杨桃 赵哲冕

CONTENTS

01

Problem Description

Background description
Problem description

Background description

NHG is a chain store with **123 stores** in six northeastern states. Due to the increase in distribution demand, we need to re-estimate its **internal transportation mileage** and require **the shortest planned route mileage**.

Problem description

The known conditions

- 123 stores in total
- The distance between 123 stores
- The number of stores to be supplied per week and the corresponding times
- Quantity of goods to be delivered per day

Problem description

The relevant constraints

- The driver is on duty less than 14 hours or driving less than 11 hours
- Vehicles are limited to 3200 cubic feet of cargo
- The unloading rate is 0.03min/FT3, and the unloading time is at least 30 minutes
- The speed is 40 miles per hour
- The store opening hours are from 8:00 to 18:00
- The vehicle must start from DC and must finish the journey back to DC at least two days later
- Store orders of different dates cannot be grouped on the same route

02

Problem Analysis

Content analysis of the problem
Problem-solving ideas

Content analysis of the problems

Problems to solve

- How to determine variables and establish objective functions
- How to use variables to describe the vehicle path
- How to achieve cross-day delivery restrictions
- How can delivery requirements be ensured for each store
- How to ensure that there are as many cars as possible
- How to introduce time elements to describe travel/duty/business time-related restrictions

Content analysis of the problems

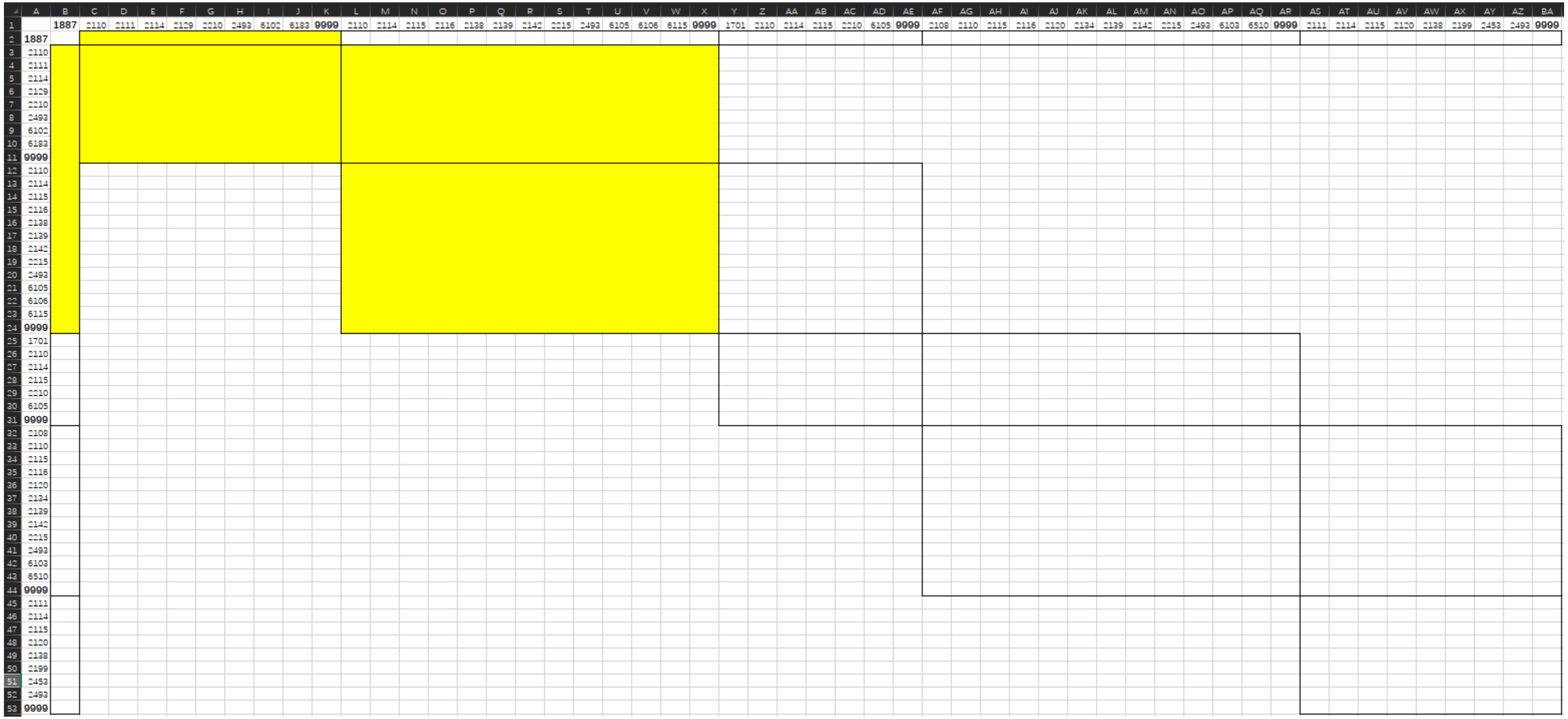
Problems to solve

- How to determine variables and establish objective functions
- How to use variables to describe the vehicle path
- How to achieve cross-day delivery restrictions
- How can delivery requirements be ensured for each store
- How to ensure that there are as many cars as possible
- How to introduce time elements to describe travel/duty/business time-related restrictions

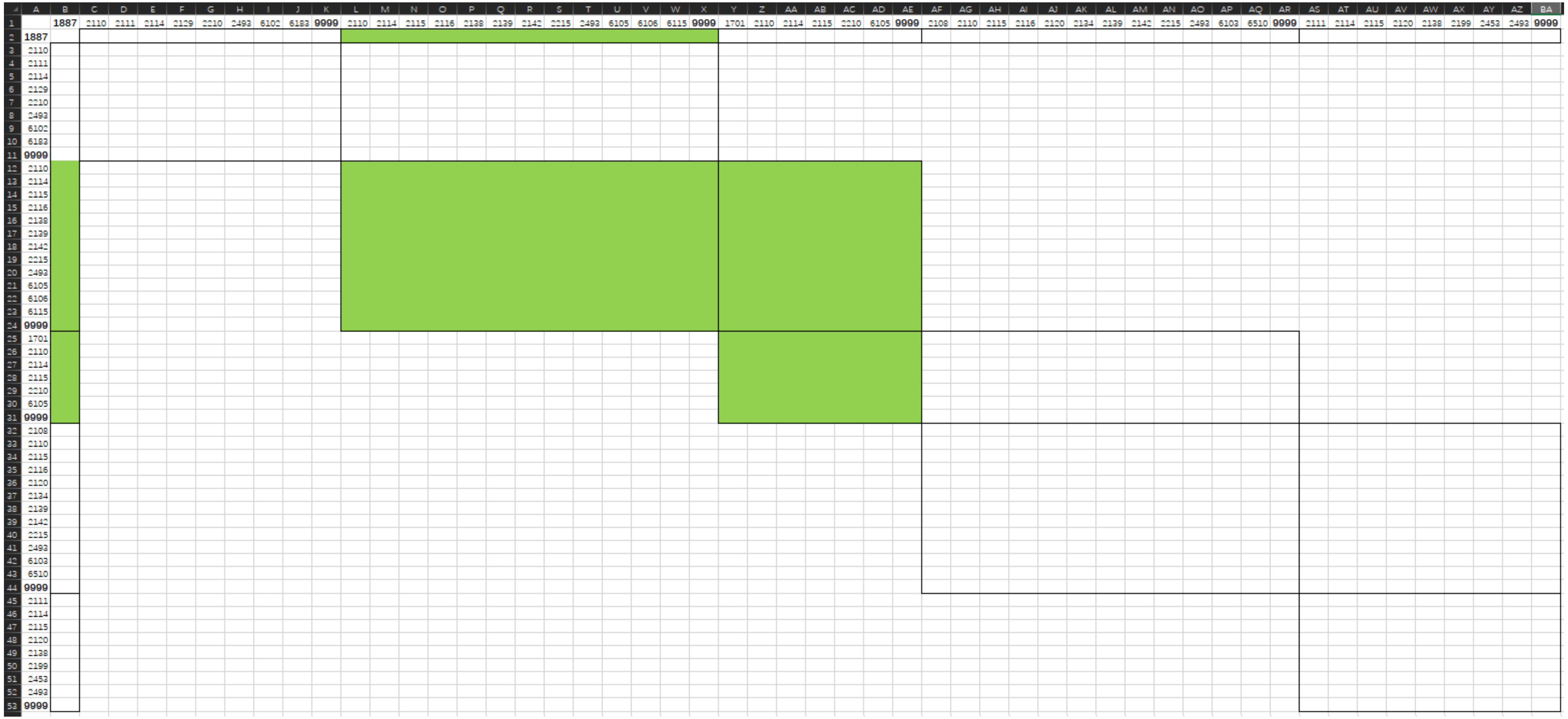
Content analysis of the problems

Problem	Approach
How to determine variables and establish objective functions	Defined as whether the road goes from one store to another and to distinguish between different times and number of trips
How to use variables to describe the vehicle path	The starting point information and the arrival point information are added to the variable description
How to achieve cross-day delivery restrictions	Take two days as a unit of time for overall optimization
How can delivery requirements be ensured for each store	Each store as a starting point can only have one destination, and each store as a destination can only have one starting point
How to ensure that there are as many cars as possible	The average demand of delivery vehicles is obtained by dividing the demand of goods by the maximum carrying capacity of vehicles
How to introduce time elements to describe travel/duty/business time-related restrictions	Assumed that the first day is the last store to deliver goods

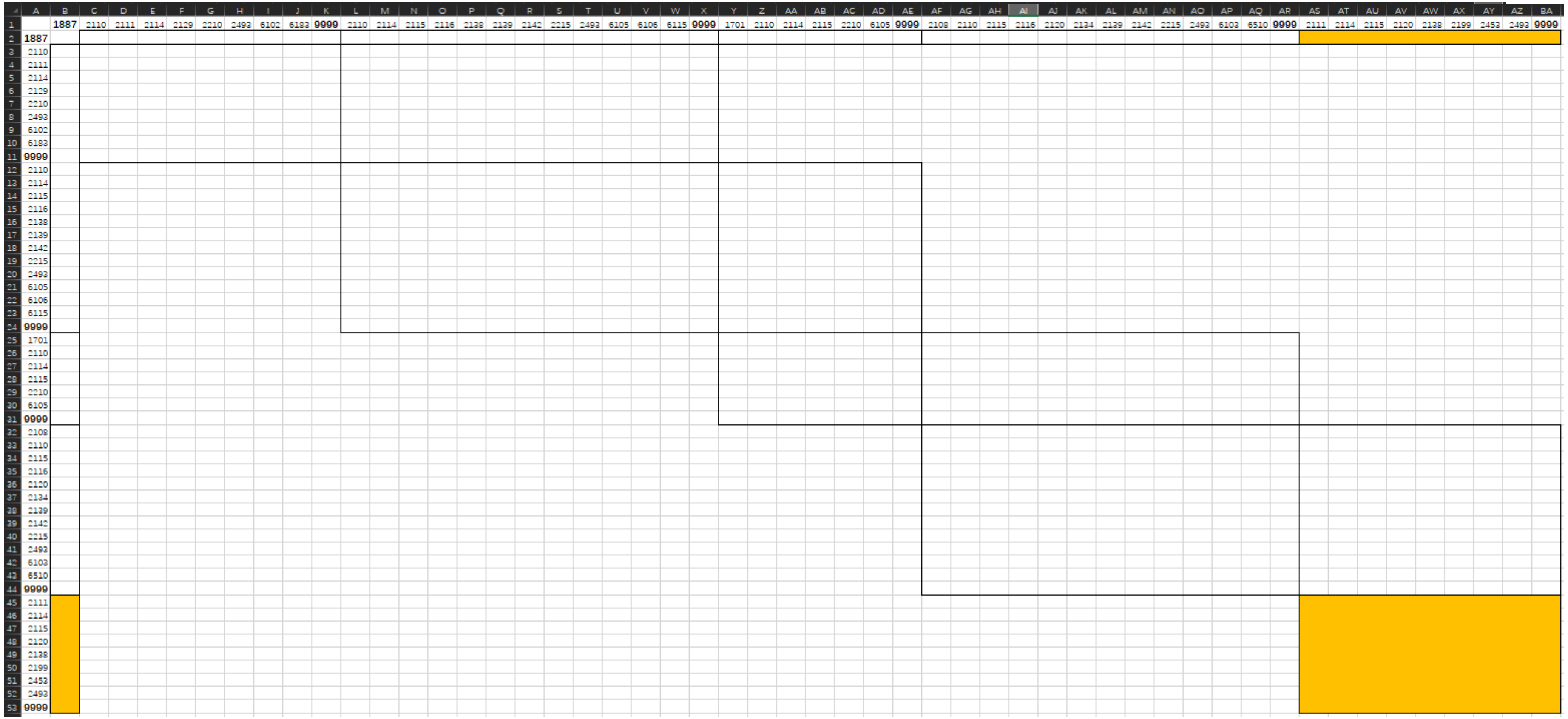
Problem-solving ideas : Data processing



Problem-solving ideas : Data processing



Problem-solving ideas : Data processing



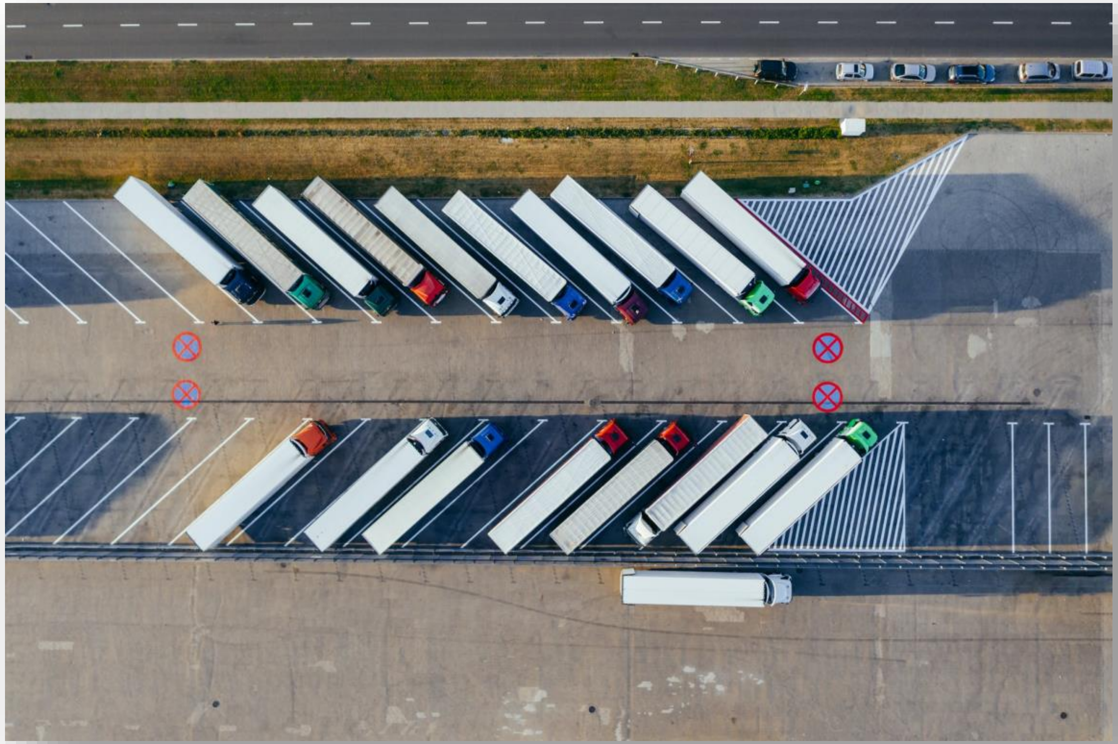
Problem-solving ideas : Model implementation

By **multiplying and adding** the path matrix and distance matrix of all vehicles, the final driving distance of all vehicles is obtained, which is **the objective function**;

Problem-solving ideas : Model implementation

The driving paths of all vehicles are separated according to the number of days, and then they are summarized into a matrix of total vehicle paths, and each shop with demand on the day is required to be passed, so that **the delivery demand** can be met

Problem-solving ideas : Model implementation



Set up enough driving path matrix to show that **there are enough vehicles**. If the used vehicles are preferred to drive, the vehicles that have not been used for five days are redundant vehicles, and the remaining vehicles are necessary vehicles;

03

Formulation of VRP model

Description of symbols

Formulation of vehicle routing problem model

Description of symbols

	Description	Domain
v_i	The i-th vertex, where v_0 is the starting point, and is sorted from small to large according to the order time and zip sequence number. Stores that need orders for many days can appear repeatedly	$i \in \{0,1,\ldots,N_5\}$
d_{ij}	The distance between v_i and v_j, $d_{ii} = Big\ M = 10000$, $d_{ij} = d_{ji}$	$i,j \in \{0,1,\ldots,N_5\}$
q_{ij}	The amount of goods required by the store at the destination v_j for the day	$i,j \in \{0,1,\ldots,N_5\}$
K	Maximum number of distribution vehicles available daily (adequate vehicle supply)	
N_m	The total number of stores that need to be delivered in the first m days within a week is $N_m - 1$ (the stores that require multiple orders are considered multiple times), and the N_m -th point is the virtual address inserted manually, $N_0 = 0$	$m \in \{1,\ldots,5\}$
x_{ijk}^m	Whether the k -th vehicle that departs on the day m moves from v_i to v_j.	$i,j \in \{0,1,\ldots,N_5\}$ $k \in \{1,\ldots,K\}$
I^m	The collection of coordinate i of the store to be delivered on the m -th day from Monday to Friday . Eg. $I^2 = \{N_1+1, N_1+2, \ldots, N_2\}$	$m \in \{1,\ldots,5\}$

Description of symbols

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA
1		1887	2110	2111	2114	2129	2210	2493	6102	6183	9999	2110	2114	2115	2116	2138	2139	2142	2215	2493	6105	6106	6115	9999	1701	2110	2114	2115	2210	6105	9999	2108	2110	2115	2116	2120	2134	2139	2142	2215	2493	6103	6510	9999	2111	2114	2115	2120	2138	2199	2453	2493	9999
2	1887																																																			
3	2110																																																			
4	2111																																																			
5	2114																																																			
6	2129																																																			
7	2210																																																			
8	2493																																																			
9	6102																																																			
10	6183																																																			
11	9999																																																			
12	2110																																																			
13	2114																																																			
14	2115																																																			
15	2116																																																			
16	2138																																																			
17	2139																																																			
18	2142																																																			
19	2215																																																			
20	2493																																																			
21	6105																																																			
22	6106																																																			
23	6115																																																			
24	9999																																																			
25	1701																																																			
26	2110																																																			
27	2114																																																			
28	2115																																																			
29	2210																																																			
30	6105																																																			
31	9999																																																			
32	2108																																																			
33	2110																																																			
34	2115																																																			
35	2116																																																			
36	2120																																																			
37	2134																																																			
38	2139																																																			
39	2142																																																			
40	2215																																																			
41	2493																																																			
42	6103																																																			
43	6510																																																			
44	9999																																																			
45	2111																																																			
46	2114																																																			
47	2115																																																			
48	2120																																																			
49	2138																																																			
50	2199																																																			
51	2453																																																			
52	2493																																																			
53	9999																																																			

Formulation of vehicle routing problem model

1.How to establish objective function

$$\min F(x) = \sum_{k=1}^{K} \sum_{m=1}^{5} \sum_{i=0}^{N_m} \sum_{j=0}^{N_m} d_{ij} x_{ijk}^{m}$$

Formulation of vehicle routing problem model

2. How to ensure continuous vehicle travel

①Vehicle exit and return restrictions:

Each vehicle needs to start from the starting point once and return to the starting point once

$$\sum_{i=N_{m-1}+1}^{N_m} x_{iok}^m = 1, \forall k \in \{1, 2, ..., K\}$$

$$\sum_{j=N_{m-1}+1}^{N_{m+1}} x_{ojk}^m = 1, \forall k \in \{1, 2, ..., K\}$$

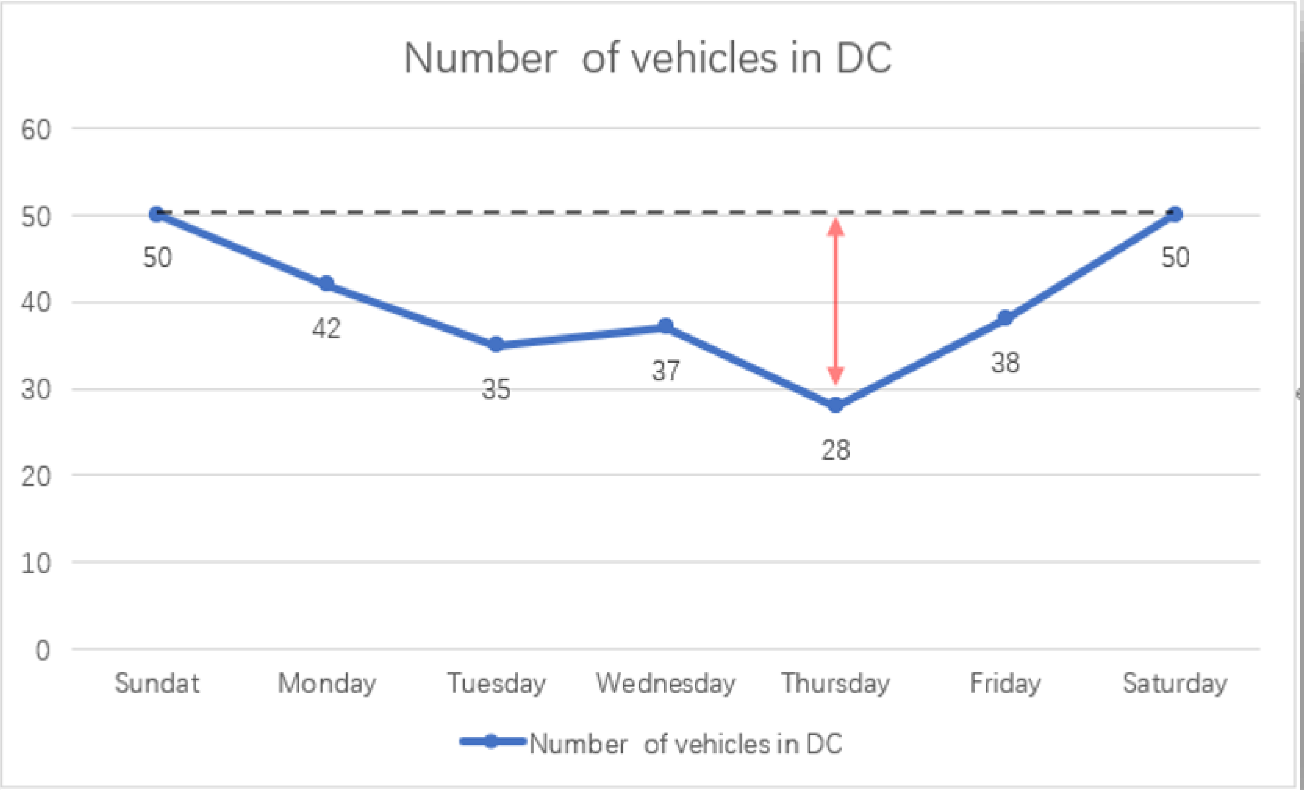
②Continuous vehicle routing restriction:

The number of times any store serves as a destination equals the number of times it serves as a starting point

$$\sum_{j=N_{m-1}+1}^{N_m} x_{ejk}^m = \sum_{i=N_{m-1}+1}^{N_m} x_{iek}^m, \forall e \in I^m, k \in \{1, 2, ..., K\}$$

Formulation of vehicle routing problem model

3. How to determine the actual number of vehicles used



$$d_{oN_m} = d_{N_m o} = 0$$

$$d_{N_m j} = Big\ M = 10000$$

$$d_{iN_m} = Big\ M = 10000$$

$$\forall i \in \{1, 2, \ldots, N_5\},\quad j \in \{1, 2, \ldots, N_5\}$$

Formulation of vehicle routing problem model

4. How to meet the delivery needs of all stores

$$\sum_{k=1}^{K}\sum_{i=0}^{N_5}\sum_{m=1}^{5} x_{ijk}^{m} = 1,\ j \in \{1, 2, \ldots, N_5\}$$

$$\sum_{k=1}^{K}\sum_{j=0}^{N_5}\sum_{m=1}^{5} x_{ijk}^{m} = 1, i \in \{1, 2, \ldots, N_5\}$$

Every store has been used as a starting point, and every store has been used as a target point once

Formulation of vehicle routing problem model

5. How to describe the capacity limits of freight vehicles

$$\sum_{i=0}^{N_5}\sum_{j=0}^{N_5} q_{ij} x_{ijk}^{m} \le 3200, \forall m \in \{1,2,3,4,5\}, k \in \{1,2...,K\}$$

Formulation of vehicle routing problem model

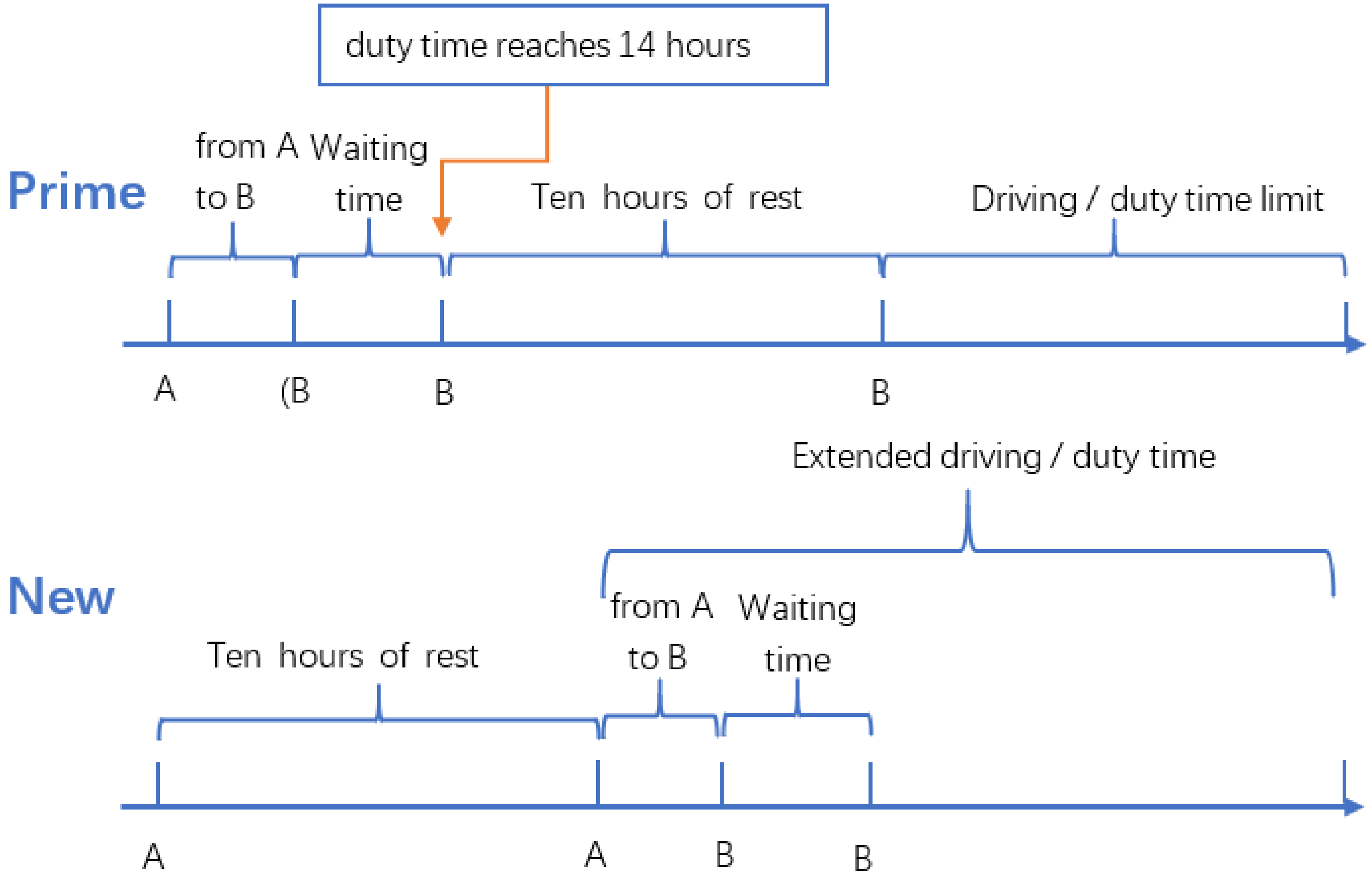
6. How to add time element to the restriction

About the time limit

① The single driving time is less than 11 hours

② The single duty time is less than 14 hours

③ Unloading time belongs to store business hours



When the vehicle is only running for one day, A is equal to DC, B does not exist.

Formulation of vehicle routing problem model

7. Time limits for the first day

①Driving time limit of the first day:

Driving time from DC to a < = 11h:

$$drt_1 = \sum_{i=0}^{N_m} \sum_{j=0}^{N_m} d_{ij} x_{ijk}^m / 40 \le 11$$

$$\forall k \in \{1, 2, \ldots, K\}, m \in \{1, 2, 3, 4, 5\}$$

②Duty time limit of the first day:

Duty time = from DC to a (travel time + unloading time) < = 14h
Unloading time less than half an hour shall be calculated as half an hour

$$dut_1 = drt_1 + \sum_{i=0}^{N_m} \sum_{j=0}^{N_m} x_{ijk}^m Max\{0.0005 q_{ij}, 0.5\}) \le 14$$

$$\forall k \in \{1, 2, \ldots, K\}, m \in \{1, 2, 3, 4, 5\}$$

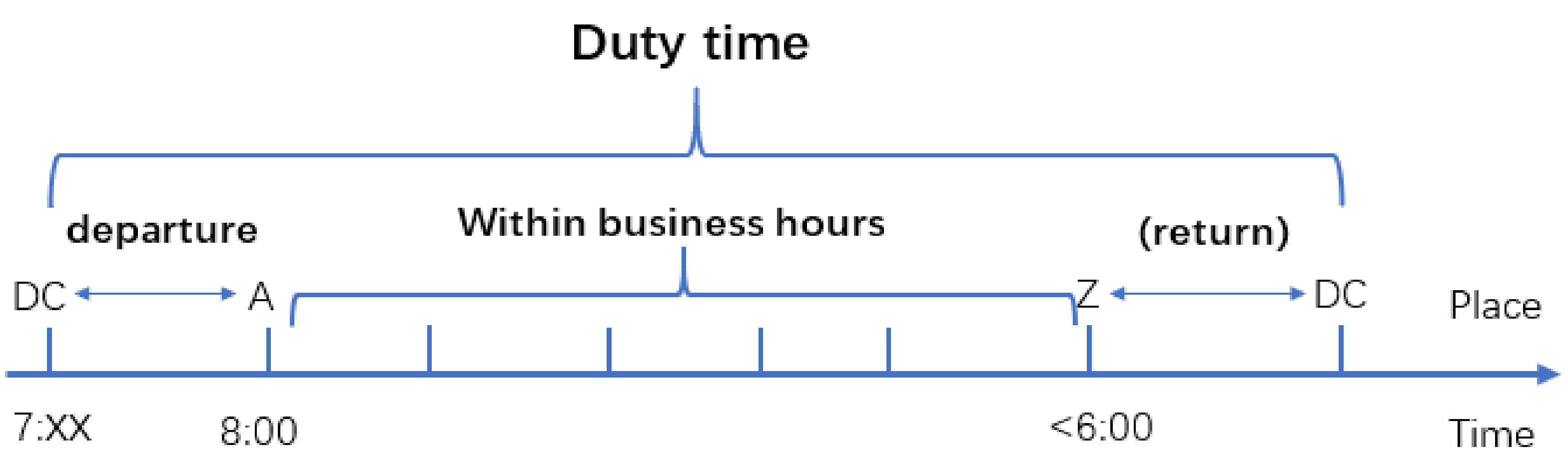
Formulation of vehicle routing problem model

7. Time limits for the first day

③The first day shop business hours limit:

Duty time - journey time from origin - journey time to return to origin < = 10 Unloading starts at 8 a.m. for the first store, and delivery to the last store must be completed by 6 p.m

Driving point of the first day



$$dut_1 - \sum_{j=0}^{N_m} d_{oj} x_{ojk}^m / 40 - \sum_{i=0}^{N_m} d_{io} x_{iok}^m / 40 \le 18:00 - 8:00 = 10$$

$$\forall k \in \{1, 2, ..., K\}, m \in \{1, 2, 3, 4, 5\}$$

Formulation of vehicle routing problem model

8. Time limits for the second day

①Driving time limit of the Second day:

$$S_{drt} = surplus\ driving\ time\ on\ the\ 1th\ day = 11 - drt_1$$

$$S_{dut} = surplus\ duty\ time\ on\ the\ 1th\ day = 14 - dut_1$$

$$S_{dist} = Time\ from\ a\ to\ B = \sum_{i=N_m+1}^{N_{m+1}} \sum_{j=N_{m-1}+1}^{N_m} d_{ij} x_{ijk}^{m} / 40$$

$$\forall k \in \{1, 2, \ldots, K\}, m \in \{1, 2, 3, 4\}$$

Formulation of vehicle routing problem model

8. Time limits for the second day

①Driving time limit of the Second day:

$$drt_2 = drt_{1\text{-}2} - drt_1 =\leq 11 + M, \forall k \in \{1,2,\ldots,K\}, m \in \{1,2,3,4\}$$

$$drt_{1\text{-}2} = \sum_{i=0}^{N_{m+1}} \sum_{j=0}^{N_{m+1}} d_{ij} x_{ijk}^{m} / 40, \quad \forall k \in \{1,2,\ldots,K\}, m \in \{1,2,3,4\}$$

$$M = \begin{cases} S_{dist} & S_{dist} \leq S_{drt} \leq S_{dut} or S_{dist} \leq S_{dut} \leq S_{drt} \\ S_{dut} & S_{dut} \leq S_{drt} \leq S_{dist} or S_{dut} \leq S_{dist} \leq S_{drt} \\ S_{drt} & S_{drt} \leq S_{dut} \leq S_{dist} or S_{drt} \leq S_{dist} \leq S_{dut} \end{cases}$$

②Duty time limit of the Second day:

$$dut_2 = dut_{1\text{-}2} - dut_1 \leq 14 + N, \forall k \in \{1,2,\ldots,K\}, m \in \{1,2,3,4\}$$

$$dut_{1\text{-}2} = drt_{1\text{-}2} + \sum_{i=0}^{N_{m+1}} \sum_{j=0}^{N_{m+1}} x_{ijk}^{m} Max\{0.0005 q_{ij}, 0.5\},$$

$$\forall k \in \{1,2,\ldots,K\}, m \in \{1,2,3,4\}$$

$$N = \begin{cases} S_{dut} & S_{dist} \leq S_{drt} \leq S_{dut} or S_{dist} \leq S_{dut} \leq S_{drt} \\ S_{dut} & S_{dut} \leq S_{drt} \leq S_{dist} or S_{dut} \leq S_{dist} \leq S_{drt} \\ S_{drt} & S_{drt} \leq S_{dut} \leq S_{dist} or S_{drt} \leq S_{dist} \leq S_{dut} \end{cases}$$

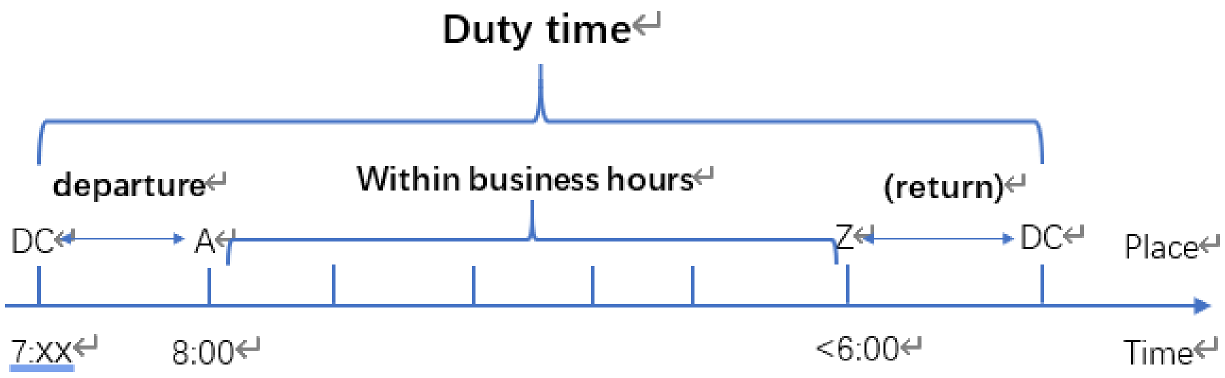
Formulation of vehicle routing problem model

8. Time limits for the second day

③The second day shop business hours limit:

Similarly, the vehicle will start unloading at the first store at 8 a.m., using the same formula as calculating the first day shop opening time limit
Among them, the duty time of the second day = the duty time of two days - the duty time of the first day

Driving point of the first day



$$dut_2 - \sum_{i=N_{m-1}}^{N_m} \sum_{j=N_m+1}^{N_{m+1}} d_{oj} x_{ojk}^m / 40 - \sum_{i=N_m+1}^{N_{m+1}} d_{io} x_{iok}^m / 40 \le 10$$

$$\forall k \in \{1, 2, \ldots, K\}, m \in \{1, 2, 3, 4\}$$

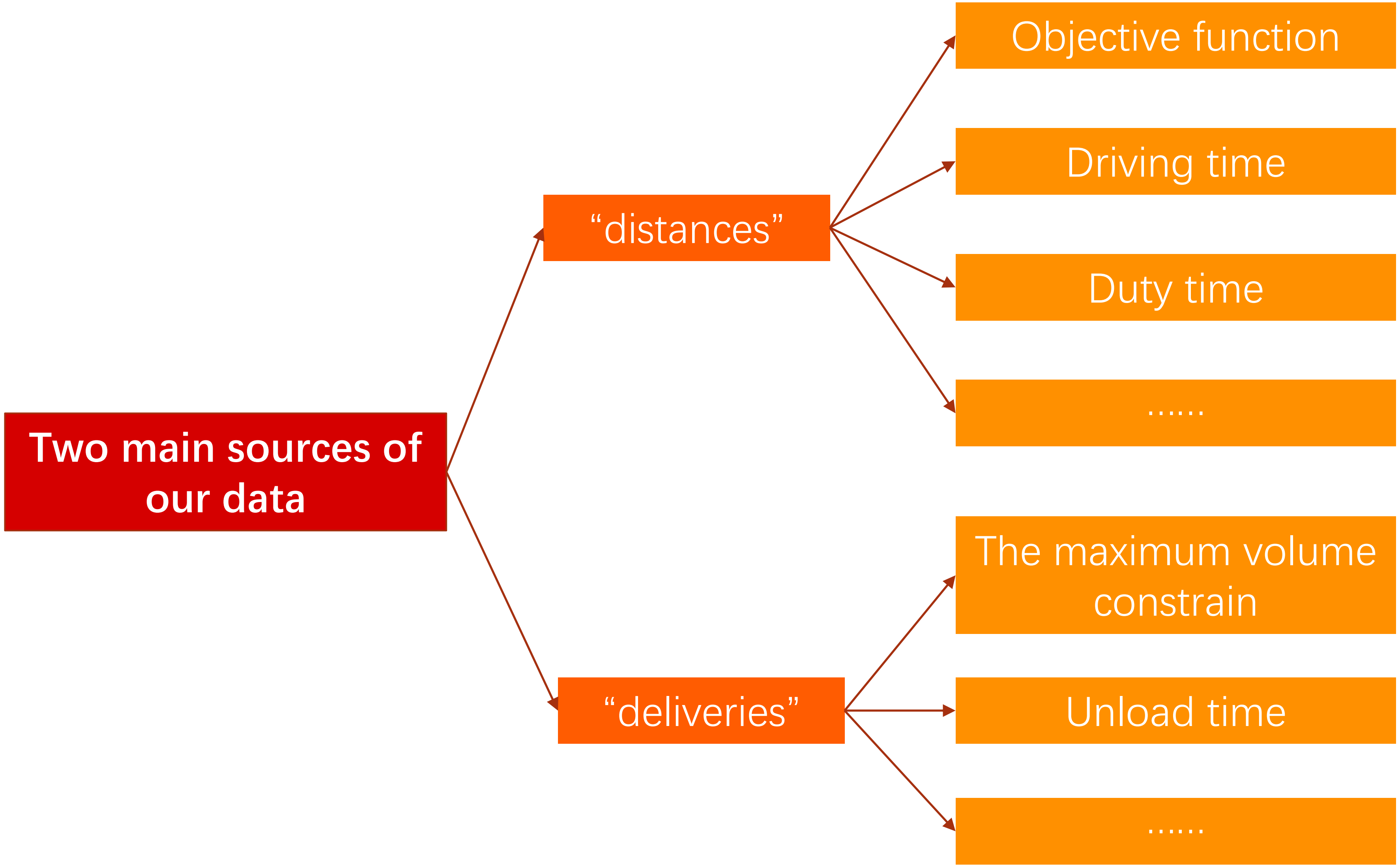
04

Data sources and Processing method

Data sources

Processing method

Data sources



Processing method

6 improvement of the data

1:Rearrange the data in distance by zip order group by 'date order' to get 'distance1' and add dummy zip.

2:Split the data in distance1 into two parts by ST required or not.

3:Create variable tables each of which stands for a truck in a specific day and add starting-destination matrix.

4:Create 'cube yes' table to show the required volume of each destination.

5:Use 'MAX' to calculate the unload time of each destination and put the in 'unload time' table in matrix from

6:Create 'Objective function' table and add the constrains and calculate the result out

Processing method

1.Rearrange the data

A	B	C	D	E	F
Zip	1060	1101	1420	1510	1570
1060	10000	19	68	82	70
1101	19	10000	81	68	55
1420	68	81	10000	17	45
1510	82	68	17	10000	32
1570	70	55	45	32	10000
1581	74	60	41	18	26
1606	69	55	25	11	20
1701	84	70	43	20	38
1730	104	90	34	35	58

For example, the numbers circled in the diagram represent the two stores of 1420 and 1101 located at a distance of 81. And the distance between the same ‘zip’ is 10000 which is the same as the Big M.

We group distances by date and reorder them in zip order, where the intersection of a row and a column represents the distance between two stores at the beginning of the row and the beginning of the column.

Processing method

2.Split the data in distance1

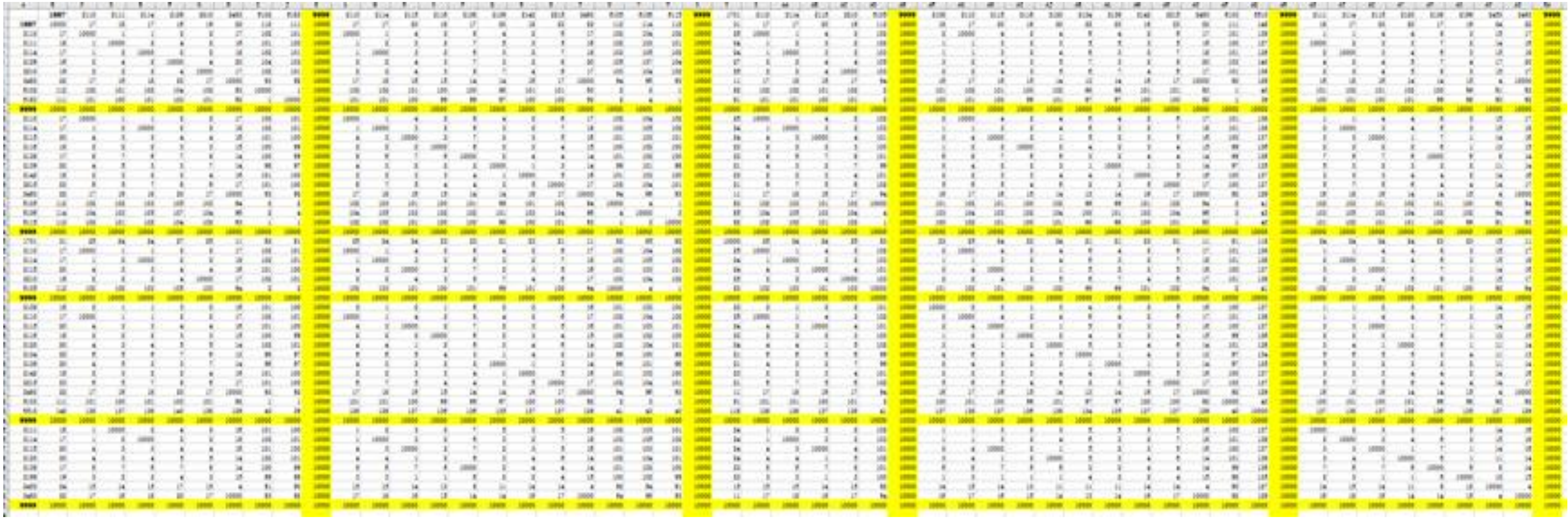
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
	1887	1581	1752	1821	2129	2132	2135	2466	2493	3906	5401	6032	6040	6043	6095	6096
1887	10000	49	43	8	16	30	23	24	22	71	201	121	110	105	115	115
1581	49	10000	10	41	36	29	27	22	25	105	233	81	70	64	75	75
1752	43	10	10000	36	33	27	24	20	13	100	228	90	79	74	84	85
1821	8	41	36	10000	18	25	24	19	17	77	205	116	105	100	110	110
2129	16	36	33	18	10000	10	7	14	20	79	215	113	102	97	108	108
2132	30	29	27	25	10	10000	6	9	13	99	229	108	97	91	102	102
2135	23	27	24	24	7	6	10000	6	11	85	223	105	94	89	99	99
2466	24	22	20	19	14	9	6	10000	5	91	221	101	90	84	95	95
2493	22	25	13	17	20	13	11	5	10000	90	220	103	91	86	97	97
3906	71	105	100	77	79	99	85	91	90	10000	211	190	179	174	184	185
5401	201	233	228	205	215	229	223	221	220	211	10000	245	243	247	228	224
6032	121	81	90	116	113	108	105	101	103	190	245	10000	20	24	20	25
6040	110	70	79	105	102	97	94	90	91	179	243	20	10000	6	15	21
6043	105	64	74	100	97	91	89	84	86	174	247	24	6	10000	21	26
6095	115	75	84	110	108	102	99	95	97	184	228	20	15	21	10000	5
6096	115	75	85	110	108	102	99	95	97	185	224	25	21	26	5	10000
6103	111	71	80	106	103	97	95	90	92	180	235	10	8	14	9	14
6108	108	68	77	103	100	94	92	87	89	177	236	14	6	12	11	16
6156	112	72	81	107	105	99	96	92	94	181	236	10	10	16	10	15
6183	111	71	80	106	103	97	95	90	92	180	236	11	8	14	10	15
6241	83	42	51	77	75	69	67	62	64	150	268	60	41	38	54	59
6269	94	53	63	89	86	80	77	73	75	163	256	36	17	14	31	36
6320	124	82	91	117	109	96	103	102	105	190	285	60	48	35	59	64
6340	121	87	95	116	106	93	100	99	102	194	289	64	52	39	63	68
6415	124	84	92	118	116	110	108	103	105	191	261	37	23	18	35	40
6457	127	86	96	122	119	113	110	106	108	196	252	22	24	30	27	31
6524	146	106	115	141	139	133	130	126	128	215	270	30	44	50	44	49
6825	166	126	135	161	159	153	150	146	148	235	292	57	64	70	66	71
6897	177	137	147	172	170	164	161	157	159	247	303	68	75	81	77	82
9999	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000	10000

We split the data in distance1 into two tables, divided by ST required: distance Yes and distance No.

Yes means a straight-through vehicle is required, while No means a straight-through vehicle is not required.

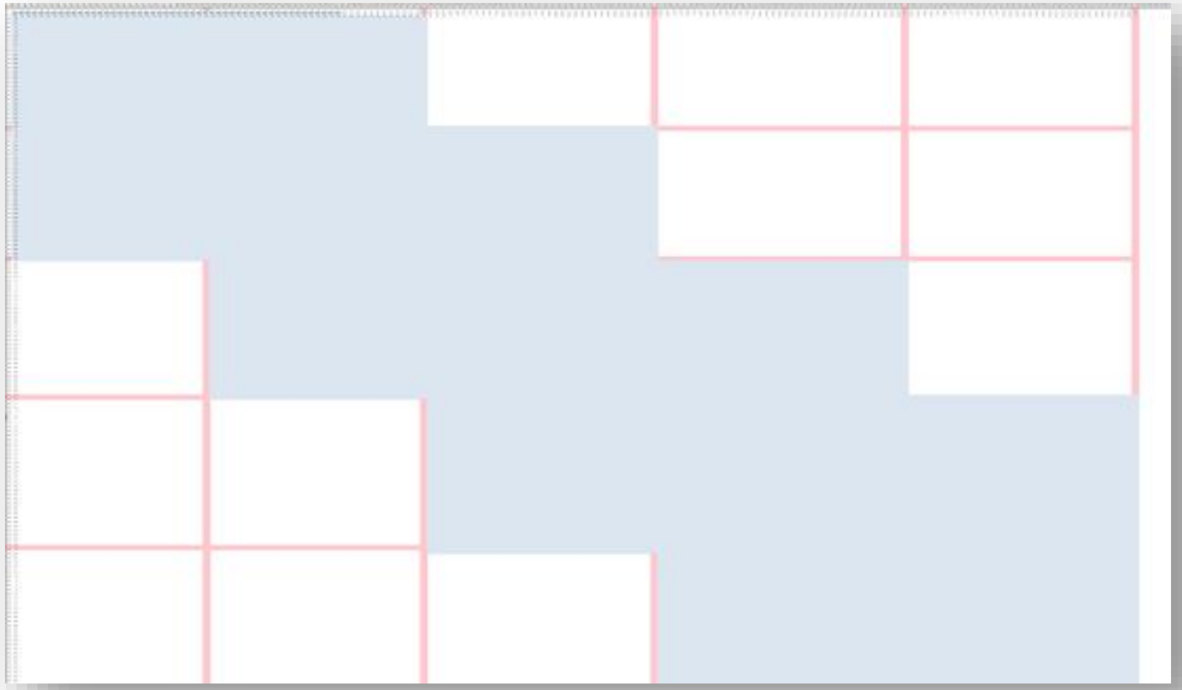
Processing method

2.Split the data in distance1



Processing method

3. Create variable tables



We created a series of ‘variable mn’ tables to represent the mth cars that departed on the nth day. And the intersection of the cell number is set to 0 or 1, indicating whether the vehicle will get through this path or not.

As each vehicle can only run for a maximum of two days in a row, so the final range of valid cells as shown in the figure.

Processing method

4. Create 'cube yes' table

	A	B	C	D	E	F	G	H	I	J	K
1		1887	2110	2111	2114	2129	2210	2493	6102	6183	9999
2	1887	0	285	352	106	120	163	152	110	183	0
3	2110	0	285	352	106	120	163	152	110	183	0
4	2111	0	285	352	106	120	163	152	110	183	0
5	2114	0	285	352	106	120	163	152	110	183	0
6	2129	0	285	352	106	120	163	152	110	183	0
7	2210	0	285	352	106	120	163	152	110	183	0
8	2493	0	285	352	106	120	163	152	110	183	0
9	6102	0	285	352	106	120	163	152	110	183	0
10	6183	0	285	352	106	120	163	152	110	183	0
11	9999	0	0	0	0	0	0	0	0	0	0
12	2110	0	285	352	106	120	163	152	110	183	0
13	2114	0	285	352	106	120	163	152	110	183	0
14	2115	0	285	352	106	120	163	152	110	183	0
15	2116	0	285	352	106	120	163	152	110	183	0
16	2138	0	285	352	106	120	163	152	110	183	0
17	2139	0	285	352	106	120	163	152	110	183	0
18	2142	0	285	352	106	120	163	152	110	183	0
19	2215	0	285	352	106	120	163	152	110	183	0
20	2493	0	285	352	106	120	163	152	110	183	0
21	6105	0	285	352	106	120	163	152	110	183	0
22	6106	0	285	352	106	120	163	152	110	183	0
23	6115	0	285	352	106	120	163	152	110	183	0
24	9999	0	0	0	0	0	0	0	0	0	0
25	1701	0	285	352	106	120	163	152	110	183	0

We produce a matrix to represent the daily demand for goods for each store, for each day of the week, in zip order.

And zip 9999 represent dummy stores which don't have goods need.

Processing method

5. Create ‘unload time’ table

	A	B	C	D	E	F	G	H	I	J	K
1		1887	2110	2111	2114	2129	2210	2493	6102	6183	9999
2	1887	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
3	2110	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
4	2111	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
5	2114	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
6	2129	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
7	2210	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
8	2493	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
9	6102	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
10	6183	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
11	9999	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
12	2110	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
13	2114	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
14	2115	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
15	2116	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5
16	2138	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5	0. 5

We take each value in the 'cube yes' table, calculate the result using ‘MAX’ function to get our required unload time, and put them all into a matrix

All unload time will be less than 0.5hour(30minutes), the out put is as follows:

Processing method

6. Create objective function table

OF table
- The result matrix
- The drive time calculation block
- The duty time calculation block
- The time window calculation block
- The objective function

Processing method

6. Create objective function table

```
=SUMPRODUCT(B2:BA53,'distances yes'!B2:BA53)
```

This formula shows us multiplying the final result matrix with the distance matrix and using the SUMPRODUCT function to compute the final result and adding the constraints we discussed earlier to solve for linear programming.

Conclusion

Conclusion

Data processing

↓

Make reasonable assumptions

↓

Establish the optimization objectives of the problem

↓

Abstract the relevant constraints of the problem

↓

Introduce the linear programming of large-scale data to solve the model.

Conclusion

01 Advantages of the model

The construction of the model is intuitive and easy to understand

The implementation of the model is creative

The result must be the optimal solution

Conclusion

02 Disadvantages of the model

Large amount of data processing

A lot of restrictions, need professional tools to calculate

Specific structures need to be designed to implement the model

Conclusion

03 Improvement of the model

Taking into account some factors such as road linearity, road obstacle, traffic light and traffic load change

Using other data processing tools such as Matlab

Thank you for watching