厦門大學

Operation and research program case report

Large-scale Multi-vehicle path with Linear Programming

Group Members： 赵哲冕 17420182200850

戚培霖 17420182200773

王业辉 17420182200798

杨 桃 17420182200826

郑嘉慧 34520182201778

Tutor：Weifeng Zhuang

2020. 11. 10

Catalogue

1. Problem Description

1.1 Background description

NHG is a chain store whose business scope covers 123 stores in six states in the northeast. Due to the increase in distribution demand, we need to re-estimate its internal transportation mileage and require the planned route to be the shortest.

Among them, the information we have obtained is the matrix distance between 123 stores, the number of deliveries corresponding to the number of stores that need to be delivered each week, and the number of goods that need to be delivered every day.

The case requires that all vehicles must depart from the distribution center and must return to the distribution center within two days at the latest, and store orders on different dates cannot be grouped on the same day's route.

At the same time, the case gives us quite a lot of constraints. For example, the driver's duty time should be less than 14 hours or the driver's maximum driving time is 11 hours. The duty time includes the driver's driving time, waiting time, and unloading time; the vehicle load must be less than 3200 ft3; the unloading rate is 0.03 minutes/ft3, and the unloading time of less than 30 minutes is also counted as 30 minutes; the vehicle speed is set at 40 mph; and the service time is 8:00-18:00

1.2 Questions raised

When we abstract the case as an operational research problem, the problem can be specifically expressed as: for a series of geographically dispersed customers, under certain constraints, organize the optimal driving route so that the vehicle departs from the distribution center

Deliver the goods to each store in an orderly manner and achieve a certain goal. In this question, this goal is set as the shortest total mileage of the route.

With the development of the commodity economy, stores have higher and higher requirements for efficient distribution. At the same time, there are also considerations for the integrity and timeliness of the goods in the logistics distribution.

These goals are often in conflict with each other, which brings new challenges to our route planning.

2. Problem Analysis

2.1 Content analysis of issues

(1) How to determine variables and establish objective functions

From the description of this question, we get a lot of information to consider. First of all, when determining a variable, what needs to be satisfied is the route planning of the order of the distribution store. This condition can be expressed by defining the variable as whether to pass through one store to another, so the variable should be a binary variable0. The second is the specific date of the week and the batches of different cars, so we need to consider adding other elements to distinguish the variables.

In determining the objective function, the direction considered is to minimize the total path of all vehicles. Therefore, the variable itself and its corresponding path value should be processed and finally added to establish the objective function.

(2) How to use variables to describe the vehicle path

The most difficult and important problem in determining variables is how to use variables to describe the path of a vehicle. By combining the experience of solving problems in the past and the characteristics of this case, we add the starting point information and the arrival point information to the description of variables, and combine the form of matrix to carry out practical operation.

(3) How to achieve cross-day delivery restrictions

Considering the assumptions in the case, The overnight route cannot span more than two days, we decided to introduce the idea of global optimization in two days as a unit of time. For example, vehicles on the first day can be delivered to stores with demand on the first day and to stores with demand on the next day. In such a case, the first variable matrix we get is a 2*2 matrix. The next day's vehicles can be delivered to both the next day's shops and the third day's shops. Therefore, the first 2*2 variable matrix will include the matrix of the first day, the fourth day and the fifth day, and so on.

(4) How can delivery requirements be ensured for each store

According to the relevant description in the case, to meet the delivery demand of each store, the following constraints should be met in the final optimal distribution line: each store as the starting point can only have one destination. Each store as a destination can only have one starting point.

(5) How to ensure that there are as many cars as possible and how many are in use

In defining delivery vehicles, we first need to consider what the average daily demand for goods is for all stores, and then get the average demand for delivery vehicles by dividing the demand by the maximum load of the vehicle. But we still need to consider the maximum demand quantity, so as to define the number of delivery vehicles.

(6) How to introduce time elements to describe travel/duty/business time-related restrictions

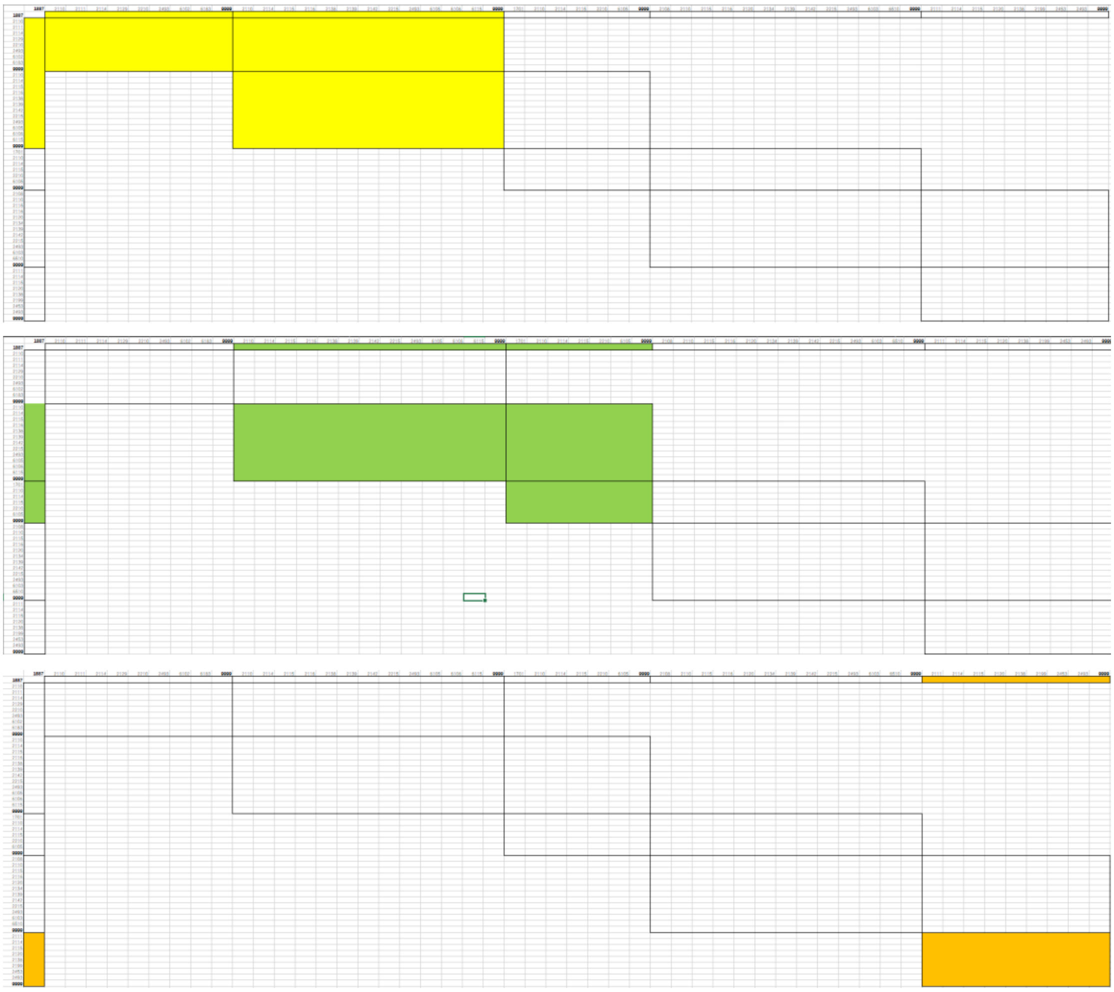
According to the case description, we need to consider three time limits: driving time, duty time and business time. Since the assumption of cross-day delivery in the case must be satisfied, we assume that when the goods are delivered for the last store on the first day, we take a direct rest in place and then continue to the first store on the next day. At the same time, the remaining driving time and duty time of the first day are given to the upper limit of driving time and duty time of the next day.

2.2 Problem-solving ideas

(1) Data processing

First, we select the store that needs to be delivered every day and sort them according to the order time and zip serial number from small to large, then we establish the path matrix with store number as row and column. Taking the store corresponding to the column coordinate as the starting point and the store corresponding to the row coordinate as the destination, a series of (0,1) variables are set in it to indicate whether to pass through this path, and a table represents the driving path of a car.

Since each vehicle runs continuously for up to two days, the possible route of a vehicle is a matrix of 2×2(starting point number × destination number). Because the vehicle starting the next day can not be filled with the goods of the first day, there will be no variables in the cells of the third quadrant in the lower left corner. Yellow area for Monday departure, running on Monday and Tuesday vehicles. The first day is Monday and the next day is Tuesday. Green area for Tuesday departure, running on Tuesday and Wednesday. The first day is Tuesday and the next day is Wednesday. Similar treatment on Thursday. The orange area indicates vehicles leaving on Friday and running only on Friday. The division of the day of the week makes it easier to distinguish the order of which day a car travels in a certain path, and makes it easier to limit the time of the two-day path.



(2) Model implementation

The final driving distance of all vehicles is obtained by multiplying and adding all the vehicle path matrix and distance matrix by the SUMPRODUCT() function in the Excel software, which is the objective function. 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.

Set up enough path matrix to indicate that there are enough cars. If you give priority to driving used vehicles, the vehicles that have not been used for five days are redundant vehicles, and the rest are necessary vehicles.

3. Formulation of VRP model

3.1 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 m-th day from Monday to Friday passes through arc $< v_i, v_j >$ after departure	$i, j \in \{0,1,\ldots,N_5\}$ $k \in \{1,\ldots,K\}$
I^m	The collection of coordinate i of the store to be delivered on the m-th day from Monday to Friday E.g, $I^2 = \{N_1+1, N_1+2,\ldots,N_2\}$	$m \in \{1,\ldots,5\}$

3.2 Formulation of vehicle routing problem model

(1) How to establish objective function

Objective function: total path minimization for all vehicles

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

(2) How to ensure continuous vehicle travel

① Vehicle exit and return restrictions: each vehicle needs to start from the starting point once and return to the starting point once

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

② Continuous vehicle routing restriction: the number of times any store serves as a destination equals the number of times it serves as a starting point

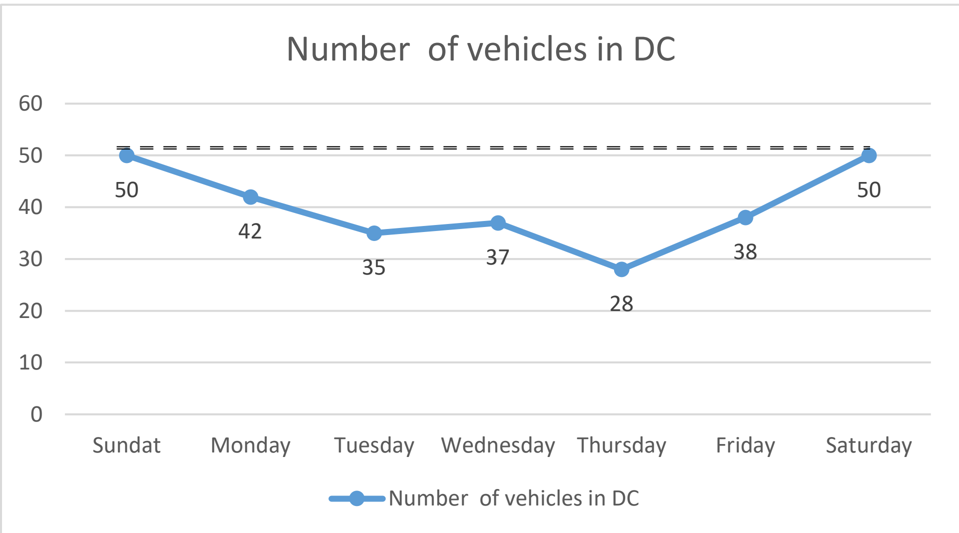
$$\sum_{j=N_{m-1}+1}^{N_m} x_{ejk}^m = \sum_{i=N_{m-1}+1}^{N_m} x_{iek}^m, \forall e \in I^m, k \in \{1,2,\ldots,K\} \tag{3}$$

③ About the itinerary plan supplement: set up the virtual shop, as the place where the surplus vehicles go

Since we don't know the number of vehicles required by the optimal solution, we can only set K to the largest value as possible in the solution process, but we need to know which cars need to be used and which cars are redundant. Therefore, we have added a virtual store to the list of stores I^m that need to deliver goods every day.

$$\begin{aligned} & d_{oN_m} = d_{N_m o} = 0 \\ & d_{N_m j} = Big\ M = 10000 \\ & d_{iN_m} = Big\ M = 10000 \\ & \forall i \in \{1,2,\ldots,N_5\},\ \ j \in \{1,2,\ldots,N_5\} \end{aligned} \tag{4}$$

The distance between the virtual store and the original place is set to zero, and the distance between the virtual store and other stores is set to big M. these two restrictions together make the redundant vehicles must go to the virtual store. In other words, when there is at least one redundant vehicle departure (to the virtual store), the remaining vehicles belong to the vehicles necessary to achieve the optimal solution.



The red line in the graph represents the maximum change in the number of vehicles in a week, which is also the number of vehicles required for the optimal solution.

(3) How to meet the delivery needs of all stores

All the vehicle schedule matrices are summed up to draw a vehicle weekly total travel matrix, which needs to meet the following requirements: each store has been used as the starting point once, and each store has been used as the target point once

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

(4) How to describe the capacity limits of freight vehicles

The total amount of goods required by all stores passing by each vehicle is less than the cargo capacity

$$\sum_{i=0}^{N_5}\sum_{j=0}^{N_5} q_{ij} x_{ijk}^{m} \le 3200, \forall m \in \{1, 2, 3, 4, 5\}, k \in \{1, 2..., K\} \tag{6}$$

(5) How to add time element to the restriction

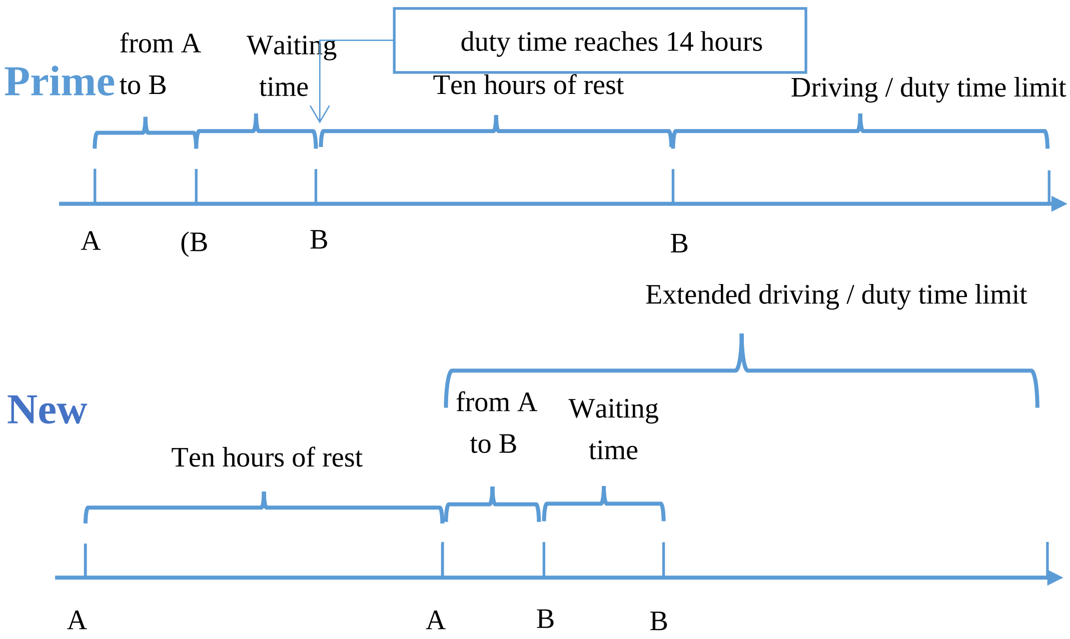
Any vehicle needs to meet three time constraints:

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

We separate the first day and the second day after departure (except for vehicles leaving on Friday); for vehicles that need to deliver goods for two days, assume that the first day is the last store (set as point a), when the goods are delivered, they are directly in place for a 10 hour rest, and then continue to drive to the first store on the second day (set as point B), at the same time, the remaining driving time and duty time of the first day are assigned to the upper limit of the corresponding time of the second day.



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

The size of the added value of the upper limit depends on the remaining driving time, the remaining duty time and the distance from the store A to the store B, which we will discuss in detail in the later part. Next we describe the time limit for the first day:

① First day driving time limit: driving time from DC to a < = 11h

$$drt_1 = \sum_{i=0}^{N_m}\sum_{j=0}^{N_m} d_{ij} x_{ijk}^m / 40 \le 11$$
$$\forall k \in \{1,2,...,K\}, m \in \{1,2,3,4,5\} \tag{7}$$

② Duty time limit of the first day: duty time = from DC to a (travel time + unloading time) < = 14h

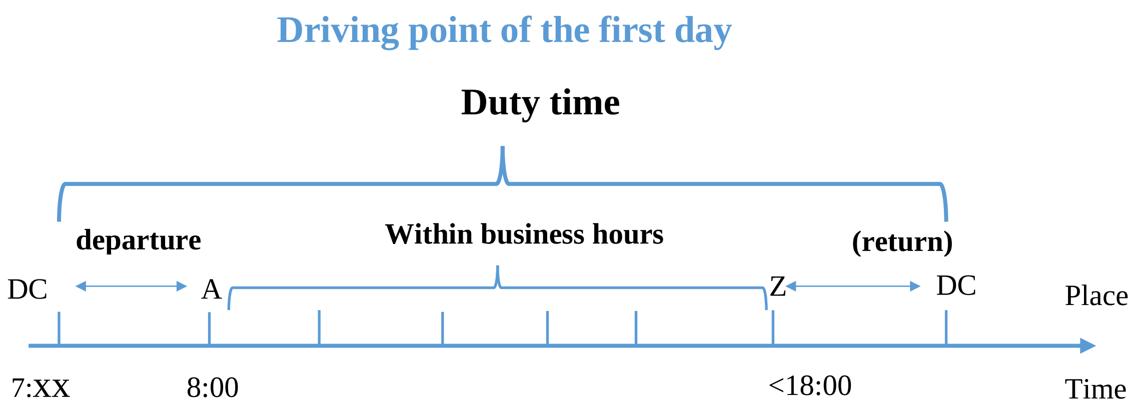
Unloading time less than half an hour shall be calculated as half an hour

$$dut_1 = drt_1 + \sum_{i=0}^{N_m}\sum_{j=0}^{N_m} x_{ijk}^m Max\{0.0005 q_{ij}, 0.5\}) \le 14$$
$$\forall k \in \{1,2,...,K\}, m \in \{1,2,3,4,5\} \tag{8}$$

③ The 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： 00p.m.

$$dut_1 - \sum_{j=0}^{N_m} d_{oj} x_{ojk}^m / 40 - \sum_{i=0}^{N_m} d_{io} x_{iok}^m / 40 \le 18:00 - 8:00 = 10$$
$$\forall k \in \{1,2,...,K\}, m \in \{1,2,3,4,5\} \tag{9}$$



A: the first store to arrive

Z: the last store to arrive

When the vehicle does not return to DC point on the first day, Z->DC travel does not exist

For the next day's time limit:

For convenience, use the following letter to indicate the relevant time

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

$$S_{dut} = surplus\ duty\ time\ on\ the\ 1th\ day = 14 - dut_1 \tag{10}$$

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

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

① Driving time limit for the following day:

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

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

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

Case 1: after arriving at the store B, start waiting until the duty time reaches the upper limit

Case 2: before arriving at the store B, the duty time is the first to reach the upper limit

Case 3: Before reaching the store B, the driving time is the first to reach the upper limit

② Duty time limit for the next day:

$$dut_2 = dut_{1\text{-}2} - dut_1 \le 14 + N, \forall k \in \{1,2,...,K\}, m \in \{1,2,3,4\} \tag{12}$$

$$dut_{1\text{-}2} = drt_{1\text{-}2} + \sum_{i=0}^{N_{m+1}} \sum_{j=0}^{N_{m+1}} x_{ijk}^{m} Max\{0.0005q_{ij}, 0.5\},$$
$$\forall k \in \{1,2,...,K\}, m \in \{1,2,3,4\}$$

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

Case 1: after arriving at the store B, start waiting until the duty time reaches the upper limit

Case 2: before arriving at the store B, the duty time is the first to reach the upper limit

Case 3: Before reaching the store B, the driving time is the first to reach the upper limit

③ Restriction of shop opening on the second day

In the same way, the vehicle will begin unloading the first store at 8 a.m., using the formula of the same meaning as (9)

Among them, the next day on duty = two days on duty-the first day on duty

$$dut_2 - \sum_{i=N_{m-1}}^{N_m} \sum_{j=N_m+1}^{N_{m+1}} d_{oj} x_{ojk}^{m} / 40 - \sum_{i=N_m+1}^{N_{m+1}} d_{io} x_{iok}^{m} / 40 \le 10$$
$$\forall k \in \{1,2,...,K\}, m \in \{1,2,3,4\} \tag{13}$$

4. Data sources & Processing method

4.1 Data source and processing

In the process of solving and simulating our model, we will use the table “total”. The source, processing and function of this table will be explained in detail below.

First of all, for this table, our data sources are the additional tables “distance” and “deliveries” in the case, but we have made some improvements on the basis of the original table

① In “distances1”, we keep the original distance data completely, and form a contrast distance matrix according to the ascending order of rows and columns of zip. The first column represents all starting points, the first row represents all destinations, and the NO.i and NO.j rows represent the distance from the i-1 distribution point to the j-1 distribution point. And we set the distance between the distribution point and itself to 10000 (large enough, similar to the utility of big M) to prevent vehicles from being delivered in place. For example, row 4 of column C in the figure represents the distance between a store with a zip value of 1420 and a store with a zip value of 1101.

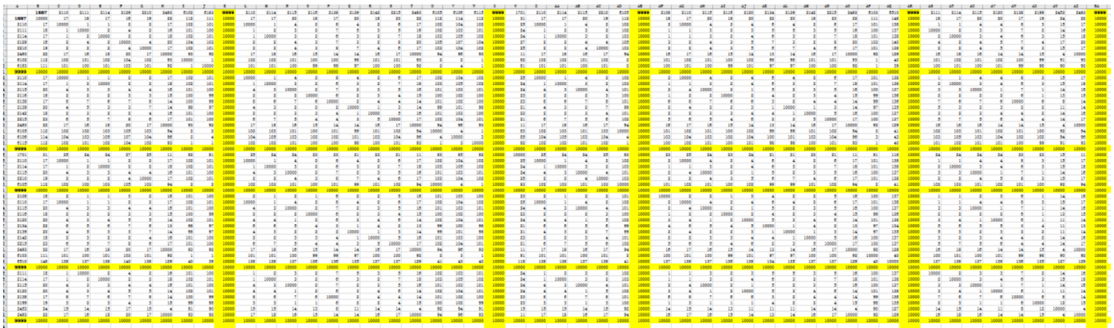
A	B	C	D	E	F
Zip	1060	1101	1420	1510	1570
1060	10000	19	68	82	70
1101	19	10000	81	68	55
1420	68	81	10000	17	45
1510	82	68	17	10000	32
1570	70	55	45	32	10000
1581	74	60	41	18	26
1606	69	55	25	11	20
1701	84	70	43	20	38
1730	104	90	34	35	58

② In the “distanceNo”, we select the stores whose “ST required “is "no" in the “deliveries” table, and according to the processing in “distance1”, we put the stores of each day in the same small matrix block, and draw their corresponding distance matrix. And after the distance of each day, we set a zip value of 9999 "virtual store", and set the distance between the virtual store and the origin (WDC) to 0. The reason for this is that if a certain vehicle does not need to leave under the optimal solution (i.e. the total number of vehicles used to start each day is less than 10), then the vehicles that do not need to start will "go" to this virtual shop and return, and there will be no impact on the total distance.

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

However, because the number of stores with “ST required no” is too large and the processing is more complex, we do not take this as an example to show.

③ In the “distance yes”Among them, we screened out the stores whose” ST required “is "yes" in the deliveries table, and according to the processing in distance1, we put the stores of each day in the same small matrix block, and draw their corresponding distance matrix. In this way, a matrix of about 5 * 5 will be obtained, as shown in the figure. In addition, because there are fewer stores with “ST required yes” and the processing is relatively simple, we will take the store with “ST required yes” as an example to demonstrate.



④ We set up a series of variable tables, variable ab, 1 < = a < = 10, 1 < = b < = 5, to represent the NO.a car leaving on the b-th day from Monday to Friday. The reason why a is set to [1,10] is that we observe that 10 vehicles per day should be sufficient.

And in each "variable" table, like the distance processing, we stack by date and sort by zip code in each day, thus forming a large matrix. And from a macro point of view, it is divided into 5 * 5, a total of 25 large matrices, respectively representing the "line" from day NO.i to day NO.j (1 < = i < = 5, 1 < = j< = 5), and the limit set by these cells is "only 0 or 1", where "0" represents that the vehicle does not pass the line, and "1" represents the vehicle passing through the line.

Obviously, I must be greater than or equal to j. And in variable ab, only the four line matrices starting from day NO.b, b + 1 and ending at day b and b + 1 will be set as variables, and the other parts will not be set as variables. As shown in the figure, variable 11 represents the store matrix that the first car departs on the first day. Because each car can stay overnight, the vehicles on the first day can be delivered to the shops with demand on the first day and those with demand the next day. In such a case, the effective data matrix we get is 2 * 2 matrix.

	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
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
2	1887	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
3	2110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
4	2111	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
5	2114	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
6	2129	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
7	2210	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
8	2493	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
9	6102	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
10	6183	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
11	9999	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
12	2110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
13	2114	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
14	2115	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
15	2116	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
16	2138	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
17	2139	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
18	2142	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
19	2215	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
20	2493	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
21	6105	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
22	6106	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
23	6115	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
24	9999	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
25	1701																													
26	2110																													
27	2114																													
28	2115																													
29	2210																													
30	6105																													
31	9999																													
32	2108																													
33	2110																													
34	2115																													
35	2116																													
36	2120																													

distances1 | distancesNo | distancesYes | distances yes | varibles11 | varibles21 | varibl ...

⑤ We then group the locations by day based on the data in the table "deliveries" and sort the locations by zip value Like "distances," we form a "start destination" matrix. The cell values in this matrix represent the volume of cargo to be carried from the starting point in row I to the destination in column j. Obviously, no matter where you start from, as long as the destination is the same, the volume of goods to be carried is certain. So the values in each column must be the same. Moreover, a virtual store with a zip value of 9999 does not need a load capacity and cannot start from a virtual store, so the row and column cell values of a zip value of 9999 are all 0. We name this table "cube yes".

	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

⑥ On the basis of "cube yes", we will calculate the unloading time. According to the description in the title, the unloading rate is 0.03min/ft3, and each unloading is no less than 30min.If the time unit is converted into hour, the unloading rate is 0.0005h/ft3, and each time is no less than 0.5h.

With this, we can set our time function to $\max\{0.0005 * cube，0.5\}$. In this way, you can calculate the unloading time of the car to each store every day—— regardless of the vehicle label. By adding this time into the corresponding matrix of cube and replacing the demand of goods, we get a new matrix unloading time matrix: "unload time".

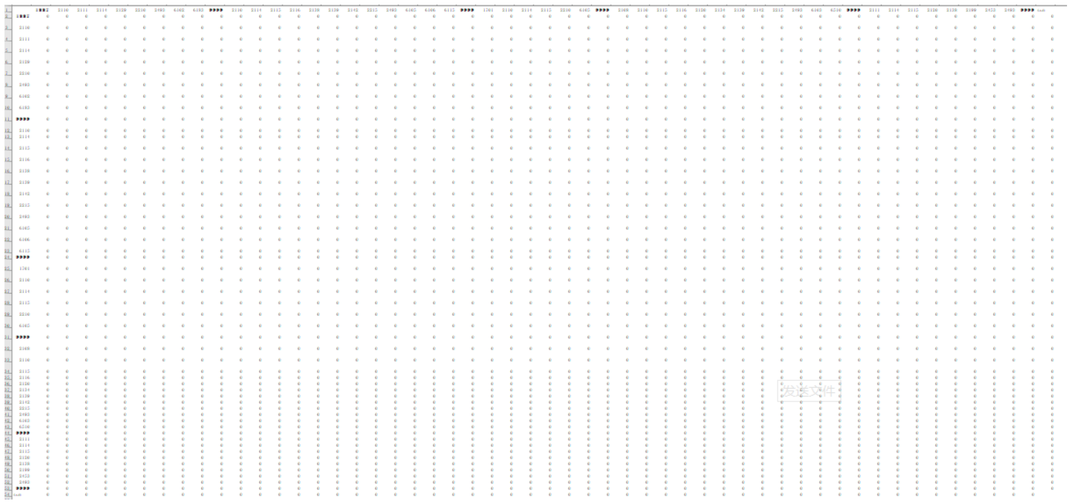
It can be seen that when the cargo is small, due to the minimum unloading time limit, the unloading time in many places is 0.5h.

	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

⑥ The last table is our objective function table. In this table, we mainly have five parts: result matrix, objective function, driving time calculation block, duty time calculation block and time window calculation block. Here are five parts

(1) Result matrix

The result matrix is a matrix of 53 * 53, which represents the superposition result of the final operation line of each day. At the edge of the result matrix, there is a sum column to calculate the sum of the results of each row and column. Ideally, the sum of each row and column, except for the first row and the first column, is only 1.(As shown in the figure)



(2) Objective function

The calculation method of our objective function is to calculate our result matrix and "distances yes" matrix by SUMPRODUCT() function, and get the sum of the corresponding position after multiplying. In this way, the total distance of all vehicles in five days can be calculated, and we have to find the minimum value for this distance.

Formula 1 Calculation formula of objective function

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

(3) Driving time calculation block

This section is used to calculate the driving time of the car. If you want to calculate the driving time of a car, you must calculate the driving distance of the car. We use the variable matrix and distances yes matrix of each car to do the "SUMPRODUCT calculation", and we can calculate the distance that each car needs to travel. Then divide this distance by speed 40 to calculate the driving time of the car. The driving time should be less than or equal to 11 hours.

Table 1 driving time calculation block

	day 2	day 1		day 2	day 1		day 2	day 1		day 2	day 1		day 1
dt of v11	0	0	dt of v12	0	0	dt of v13	0	0	dt of v14	0	0	dt of v15	0
dt of v21	0	0	dt of v22	0	0	dt of v23	0	0	dt of v24	0	0	dt of v25	0
dt of v31	0	0	dt of v32	0	0	dt of v33	0	0	dt of v34	0	0	dt of v35	0
dt of v41	0	0	dt of v42	0	0	dt of v43	0	0	dt of v44	0	0	dt of v45	0
dt of v51	0	0	dt of v52	0	0	dt of v53	0	0	dt of v54	0	0	dt of v55	0
dt of v61	0	0	dt of v62	0	0	dt of v63	0	0	dt of v64	0	0	dt of v65	0
dt of v71	0	0	dt of v72	0	0	dt of v73	0	0	dt of v74	0	0	dt of v75	0
dt of v81	0	0	dt of v82	0	0	dt of v83	0	0	dt of v84	0	0	dt of v85	0
dt of v91	0	0	dt of v92	0	0	dt of v93	0	0	dt of v94	0	0	dt of v95	0
dt of v101	0	0	dt of v102	0	0	dt of v103	0	0	dt of v104	0	0	dt of v105	0

Formula 2 calculation formula of vehicle travel time

```
=SUMPRODUCT('distances yes'!B2:X24,varibles11!B2:X24)/40
```

(4) On duty time calculation block

This part is used to calculate the duty time of drivers. Besides the driving time, there is also unloading time, so add the two parts together. The driver's duty time should be less than or equal to 14 hours.

Table 2 Calculation of duty time

dut of v11	0	0	dut of v12	0	0	dut of v13	0	0	dut of v14	0	0	dut of v15	0
dut of v21	0	0	dut of v22	0	0	dut of v23	0	0	dut of v24	0	0	dut of v25	0
dut of v31	0	0	dut of v32	0	0	dut of v33	0	0	dut of v34	0	0	dut of v35	0
dut of v41	0	0	dut of v42	0	0	dut of v43	0	0	dut of v44	0	0	dut of v45	0
dut of v51	0	0	dut of v52	0	0	dut of v53	0	0	dut of v54	0	0	dut of v55	0
dut of v61	0	0	dut of v62	0	0	dut of v63	0	0	dut of v64	0	0	dut of v65	0
dut of v71	0	0	dut of v72	0	0	dut of v73	0	0	dut of v74	0	0	dut of v75	0
dut of v81	0	0	dut of v82	0	0	dut of v83	0	0	dut of v84	0	0	dut of v85	0
dut of v91	0	0	dut of v92	0	0	dut of v93	0	0	dut of v94	0	0	dut of v95	0
dut of v101	0	0	dut of v102	0	0	dut of v103	0	0	dut of v104	0	0	dut of v105	0

Formula 3 calculation formula of duty time

```
=BI4+SUMPRODUCT('unload time'!B2:K11,varibles21!B2:K11)
```

(5) Time window calculation block

The store has a fixed opening time, from 8:00 to 18:00, a total of 10 hours. The vehicle must be driven and unloaded within this period of time. However, it should be noted that the driving time before arriving at the first store and leaving from the last store do not need to be counted in this time window. So the business hours limit can add up the driving time and the unloading time, subtract the time to go to the first store and the time to leave the last store, and make the total time less than 10 hours.

Table 3 time window calculation block

opent of v11	0	0	opent of v12	0	0	opent of v13	0	0	opent of v14	0	0	opent of v15	0
opent of v21	0	0	opent of v22	0	0	opent of v23	0	0	opent of v24	0	0	opent of v25	0
opent of v31	0	0	opent of v32	0	0	opent of v33	0	0	opent of v34	0	0	opent of v35	0
opent of v41	0	0	opent of v42	0	0	opent of v43	0	0	opent of v44	0	0	opent of v45	0
opent of v51	0	0	opent of v52	0	0	opent of v53	0	0	opent of v54	0	0	opent of v55	0
opent of v61	0	0	opent of v62	0	0	opent of v63	0	0	opent of v64	0	0	opent of v65	0
opent of v71	0	0	opent of v72	0	0	opent of v73	0	0	opent of v74	0	0	opent of v75	0
opent of v81	0	0	opent of v82	0	0	opent of v83	0	0	opent of v84	0	0	opent of v85	0
opent of v91	0	0	opent of v92	0	0	opent of v93	0	0	opent of v94	0	0	opent of v95	0
opent of v101	0	0	opent of v102	0	0	opent of v103	0	0	opent of v104	0	0	opent of v105	0

Formula 4 calculation formula of business hours

```
=SUMPRODUCT('unload time'!$L$2:$X$24,varibles11!$L$2:$X$24)+'objective function'!BH3-SUMPRODUCT('distances yes'!$B$12:$B$24,varibles11!$B$12:$B$24)/40-SUMPRODUCT('distances yes'!$L$2:$X$2,
varibles11!$L$2:$X$2)/40
```

4.2 Experimental platform

The experimental platform (tool) we use is EXCEL. Although our problem is multithreaded and appears to be a nonlinear programming problem at first sight, we can take the "number of vehicles" in the multithreading problem to a large enough value and allow the vehicle vacancy, that is, not to start. Then, the multithreading problem can be transformed into a single thread problem to solve the "multiple vehicles" constraint.

Moreover, we can use matrix multiplication and SUMPRODUCT() function to transform the variables in nonlinear programming problems into linear solvable degree, and use linear programming problems to achieve the optimal solution.

In the process of solving linear programming problems in EXCEL, because the original plug-in of EXCEL solver can not solve such a huge problem (whether the number of variable cells or the limited number is far beyond the scope that EXCEL solver can solve), so we use the expansion plug-in of EXCEL to solve 8000 variable cell linear programming problems. But even so, this plug-in is far from the effect we want. So in EXCEL, we can only simplify the problem, to solve a simple model, in order to achieve the effect of peeping in the tube.

5. Conclusion

In this paper, based on the division of the distribution categories of NHG demand points, the large-scale multi-vehicle path problem is studied in detail. First of all, we deal with the data, and then according to the actual situation, we make reasonable assumptions on the problem, and establish the optimization objectives of the problem. Finally, we abstract the relevant constraints of the problem, so as to establish a large-scale multi-vehicle path planning model, and introduce the linear programming of large-scale data to solve the model. In view of the problems and limitations in the process of modeling, we improved the model and algorithm, so that the feasibility of the model has been improved.

5.1 Advantages of the model

(1) The construction of the model is intuitive and easy to understand

Based on the analysis of the case, we first define whether the vehicles which is in the k-th batch of vehicles and start on the M day of Monday to Friday pass through the arc $< v_i, v_j >$ after departure as decision variable, and the value is 0 or 1. Then, the objective function is established by minimizing the sum of the total path lengths passing through the site. Finally, it defines 10 restrictions: the limit of distribution demand, the limit of vehicle path must be continuous, the limit of virtual store and the distance of original place, the limit of vehicle departure and return, the limit of vehicle capacity, the limit of two-day driving time, two-day duty time and two-day shop operation time. Finally, it fully meets the needs mentioned in the case.

(2) The implementation of the model is creative

In order to realize the model and solve the problem under the existing conditions, we have made a very creative design for the implementation of the model, such as: the idea of overall optimization in two days as a unit of time to achieve a week of overall optimization, the introduction of time elements to meet the assumption of cross-day delivery, special table design to achieve data processing, and so on. As a result, the results can be obtained efficiently through Excel Solver tools.

（3）The result must be the optimal solution

Because this problem is still a linear programming problem, although it takes time and effort in the process of solving, the optimal solution can be obtained in the end.

5.2 Disadvantages of the model

(1) Large amount of data processing

The problem of vehicle path planning is a very complex problem. There are many alterable variables provided in this case. There are too many variables to be dealt with in the form of matrix, so it is very difficult for general tools to deal with.

（2）A lot of restrictions, need professional tools to calculate

There are many restrictions in this case, such as vehicle capacity, driver's duty time, store delivery time and so on. We need to add all kinds of constraints to the model, and it is very difficult to build the model and solve the model by using professional tools.

（3）Specific structures need to be designed to implement the model

Due to the limitation of the data processing tools we have mastered, we must design implement the model, so the design of the implementation method takes a lot of time.

5.3 Improvement of the model

The optimal path is the basis of the path planning problem. However, how to construct a dynamic model of road network to take into account some factors such as road linearity, road obstacle, traffic light and traffic load change can make the conclusion of the model more practical and realistic.

Limited and Excel data processing capacity, using other data processing tools such as Matlab can better solve large-scale multi-vehicle path planning problems.