DP-TABU: an algorithm to solve single-depot multi-line vehicle scheduling problem

Zhao Xinchao¹ · Sun Hao¹ · Lu Juan² · Li Zhiyu³

Received: 31 March 2021 / Accepted: 10 June 2021 / Published online: 19 June 2021
© The Author(s) 2021

Abstract

A DP-TABU algorithm is proposed which can effectively solve the multi-line scheduling problem of single Depot (SD-ML-VSP). The multi-line regional coordinated dispatch of the single-line depot of the bus is to solve the problems of idle low-peak vehicles and insufficient peak capacity in single-line scheduling. The capacity of multiple lines at the same station is adjusted to realize resource sharing such as timetables, vehicles, and drivers. Shared capacity such as bus departure intervals and bus schedules. Taking the regional scheduling of multiple lines at the same station as the service object, a vehicle operation planning model based on the objective of optimal public transportation resources (minimum bus and driver costs) is established to optimize the vehicle dispatching mode of multiple lines. We applied this algorithm to the three lines S105, S107, and S159 of Zhengzhou Public Transport Corporation, and the results proved that the algorithm is effective. Through comparison with manual scheduling and simulated annealing algorithm, the advantages of DP-TABU algorithm in performance optimization and robustness are further verified.

Keywords DP-TABU · SD-ML-VSP · Resource sharing · Regional scheduling

Introduction

A reasonable operation organization scheduling plan is the basis of operation scheduling. The core of the operation organization scheduling plan is the preparation of bus departure schedules. At the same time, the departure schedule is also a key factor in improving the ride comfort for passengers. SD-ML-VSP refers to the integration of multiple lines in one depot to realize the unified configuration of vehicles and drivers, thereby achieving centralized management of drivers, centralized parking of vehicles, unified planning of driving operations, and unified command of dispatching. The goal is saving manpower and reducing material resources and improving the efficiency of vehicle use.

In China, public transportation companies generally adopt a single-line operation mode. This mode has made the public transportation resource scarce and also has restricted the

application of various new technologies. The SD-ML-VSP is mainly aimed to solve the problems of vehicles idling in low-peak period and insufficient vehicles in peak period of single-line scheduling. Multiple lines at the same station are planned in a unified manner to realize the sharing of timetables, vehicles, and drivers. The regional scheduling of multiple lines at one depot is the service object, and the goal based on the optimal public transportation resources is established (the cost of buses and drivers is the smallest)

Nowadays, Poisson distribution [7] can be used to determine the passengers waiting time and hybrid algorithm [8] can be used to generate departure timetable [1,2] build an optimal model to schedule the departure time of each bus service with the objective of minimizing the average waiting time. Tabu search algorithm [5] and dynamic programming algorithm [6] are good methods for solving two-way public transportation scheduling [3], performing the vehicle scheduling of urban bus lines, but it is very important to reduce the operational cost and improve the quality of public transportation services [4]. The effective scheduling of electric buses for multiple vehicle types is essential for the sustainable practice of public transport, [9] proposes a new methodology for the electric vehicle scheduling problem in public transport. [10] addresses the electric vehicle schedul-

✉ Zhao Xinchao
zxcxinchao@foxmail.com

¹ Zhengzhou Tiamaes Technology Co., Ltd, Zhengzhou, China

² Beijing institute of Petrochemical Technology, Beijing, China

³ China Academy of Space Technology, Beijing, China

Fig. 1 Single line scheduling **Fig. 2** Regional scheduling

ing problem (E-VSP), in which a set of timetabled bus trips, which starts from specific locations and end at specific times, they are carried out by a set of electric buses or vehicles based at a number of depots with limited driving ranges.

At present, most scholars' research on vehicle scheduling problems is basically based on a single line, so the optimization theory of single-line operation and management has been gradually improved, but the existing research mostly focuses on the optimization of single-line and ignores the bus scheduling system, which leads to the inefficiency of optimization algorithms in practical applications.

Heuristic algorithms are often used to solve combinatorial optimization problems, but conventional heuristic algorithms have disadvantages such as premature convergence and easy to fall into local optimal solutions. This article mainly focuses on the regional scheduling problem of single depot and multiple lines, and build the vehicle scheduling model. Since this model involves a large number of variables, the solution process is very complicated, and considering that the combination of the problem also makes calculation difficult, it is difficult to solve with traditional heuristic algorithms. For this reason, this paper proposes a DP-TABU algorithm to solve the problem of bus scheduling with multiple lines at a single depot.

Problem definition

In the SD-ML-VSP problem, all lines belonging to the same depot can be regarded as a subsystem, and the regional scheduling for all lines in the depot can be the overall system. All regional scheduling can also be described as the aggregation of multiple subsystems into a central system with specific goals. These subsystems are mutually independent and inter-related, restricting and influencing each other.

Regional scheduling is not a new concept in areas where public transportation management is more advanced, but an organization and operation model that has been successfully applied to the public transportation field. The bus operation mode mainly refers to the organization method of the resources of the bus enterprise and the form of bus scheduling and command. At present, there are two main operating modes for China's bus companies: single-line operation scheduling mode and regional scheduling operation mode, while China currently mainly adopts single-line operation scheduling mode.

The single-line operation scheduling mode means that the resources such as buses and drivers owned by the bus line are fixed and unique, and other auxiliary operation resources

Fig. 3 Single-depot multi-line regional scheduling

such as stations and on-site dispatchers are also configured according to the line, as shown in Fig. 1.

As shown in Fig. 2, the operation mode of bus regional scheduling refers to the allocation of resources based on an operating area. An operating area is either a city or a part of a city. The area of each area is mainly determined based on the operating efficiency of the entire area as the highest standard. In regional scheduling, the route has only the concept of service and has no impact on passengers. Public transport is not restricted in resource allocation and schedule.

It can be seen that the focus of single-line operation scheduling and regional scheduling is on a single line and an area composed of multiple lines. In regional scheduling, multiple lines are combined into one area. In single-line scheduling, the driver and the route are unique, but in the regional scheduling, the driver and the route are uniformly allocated as a whole.

Model establishment

In this article, the regional scheduling of public transportation vehicles refers to unified dispatch of vehicles to execute the schedule of multiple routes with the goal of reducing the resources of public vehicles. First, the blockchain can only be executed by specific vehicles and drivers; second, reduce public transportation resources as much as possible to reduce the cost of public transportation vehicles and drivers; third, it is necessary to consider the labor efficiency of drivers.

In the SD-ML-VSP problem, due to the imbalance in the direction and time of the bus passenger flow, the dynamic scheduling mode is used to maximize the turnover efficiency of vehicles and dynamically combine different lines to achieve the greatest possible saving of public transportation resources (bus, driver). Figure 3 details the basic process of SD-ML-VSP.

SD-ML-VSP means that the operating vehicles in the same dispatch area belong to the same depot, that is, the operating vehicles can only be put into operation from the depot, and after the vehicles complete a given shift on a certain route, they will perform other operations according to the scheduling arrangement. Bus line schedule work, after completing a number of tasks, continue to perform other bus line schedule tasks, until the final task is over and return to the depot. In the entire regional scheduling process, all vehicles and drivers are dispatched uniformly.

Symbol definition

Define the symbols and parameters required for modeling as follows: L is the total number of regional scheduling routes in one depot, D is Depot, i means block, s_i means the start time of block i , m_i is the mileage of block i , e_i is the end time of block i , k is the running bus, x_j^{Di} is the 0–1 decision variable, If the shift i is driven by the vehicle k from the depot to the shift i , the value is 1, otherwise it is 0, y_k^{ij} is the 0–1 decision variable. If the next shift to be executed after vehicle k executes shift i is j , the value is 1, otherwise it is 0, δ_k^{ij} is the 0–1 decision variable. If there is an empty driving between shift i and j in vehicle k , the value is 1, otherwise it is 0, $t_{i,j}$ is the empty driving time of block i and block j , $t_{D,i}$ is the empty driving time from depot to block i , $t_{i,D}$ is the empty driving time from block i to depot. The detailed information is shown in Table 1

Problem summary

Single-depot multi-line regional scheduling generally adopts a unified resource centralized management mode. Generally speaking, the regional scheduling center has only one dispatching room. The dispatching room has a dispatching director and several dispatchers. There are multiple dispatch-

Table 1 Symbol definition

Symbol	Significance
L	Total number of bus lines
s_i	Start time of block i
e_i	End time of block i
m_i	Mileage of block i
$x_k^{D_i}$	0–1 Decision variable
$t_{i,j}$	Empty driving from block i to block j
δ_k^{ij}	0–1 Decision variable
y_k^{ij}	0–1 Decision variable
l	Bus line
D	Depot
N	Total bus blocks of line
i	Bus block
$t_{D,i}$	The empty driving time from depot to block i
$t_{i,D}$	The empty driving time from block i to depot

ing stations in the dispatching room. Each dispatching station can dispatch one line or multiple lines, among which the main tasks of the dispatcher are to prepare the schedule of driving operations, participate in on-site dispatch, vehicle maintenance, arrange drivers and spare vehicles

In the process of model building, it is assumed that the following conditions are true

- The driver unconditionally executes the driving plan;
- The power of the bus is enough to support all-day work;
- The bus has made all the preparations before the current schedule.

Bound for objective function1

When establishing a SD-ML-VSP model, here are mainly four constraints: the uniqueness of the shift, the empty driving time between shifts, the maximum operating mileage of the vehicle, and the waiting time between shifts. The constraint of the uniqueness of the shift ensures that each shift is only executed by one bus, the empty driving time between shifts ensures that the vehicle will try to avoid empty driving time when performing tasks, the maximum operating mileage of the vehicle ensures that the driver's labor intensity is within a reasonable range, and the waiting time between shifts ensures the turnover efficiency of the vehicle.

The constraint of the uniqueness of the shift:

For any line of shifts, it can only be executed by one vehicle and can only be executed once:

$$\sum_{i=1}^N x_k^i = 1 \quad (1)$$

The empty driving time between shifts:

There is and only when the terminal station of shift j and the starting station of adjacent shift i are not the same, the empty driving situation will occur, otherwise there is no empty driving in the adjacent shift:

$$\delta_k^{ij} = \begin{cases} 1 & S(i) = E(j) \\ 0 & S(i) \neq E(j) \end{cases} \quad (2)$$

the maximum mileage of the vehicle:

To ensure the safety of regional scheduling operations, each driver has a limit on the maximum number of tasks. When making a vehicle operation plan, it is necessary to consider the driver's full-day maximum mileage, m_{max} means the maximum mileage, kn means the number of shifts allowed by vehicle k :

$$\sum_{i=1}^{kn} m_i \leq m_{max} \quad (3)$$

The waiting time between shifts:

To improve the efficiency of vehicle turnover, it is necessary to reduce the waiting time between shifts as much as possible. At the same time, the reduction of the waiting time between shifts is also conducive to reducing the use of public transportation vehicles:

$$W_t = \sum_{l=1}^L \sum_{i=2}^N e_{l_i} - s_{l_{i-1}} \quad (4)$$

Objection function

When establishing the SD-ML-VSP model, the initial solution of the shift chain is first solved by the dynamic programming algorithm [11–13]. Four factors are mainly considered here the uniqueness of the shift, the empty driving time between shifts, the maximum mileage of the vehicle, the waiting time between shifts. Where d_{ki} is the empty driving cost of the vehicle from the depot to the starting station of the line, d_{ij} is the empty driving cost between adjacent shifts, w_{ij} is the waiting time between shifts, and d_{jk} is the empty driving cost of the line's terminal station to the depot.

Figure 4 shows the overall process of SD-ML-VSP. First, the schedule is formulated according to the passenger flow of the relevant lines in the entire area. Based on the schedule, it is adopted according to factors such as the number of vehicles allocated to the parking lot, the area of the parking lot, and the number of dispatchers. The dynamic programming algorithm allocates vehicles to each shift to form a shift chain, and then constructs a loss function based on factors such as the driver's labor efficiency and vehicle turnover efficiency. The

Fig. 4 SD-ML-VSP flowchart

tabu search algorithm [5,14–16] is used to score each shift chain to evaluate the quality of the shift chain.

Compared with single-line scheduling, regional scheduling has a great benefit advantage, but the preparation of regional scheduling plans is extremely complicated. The solution to this problem is usually based on a bus network that allows vehicles to be dispatched across lines.

The cost of entering/exiting the depot depends on the empty driving cost of the first and last shifts performed by the vehicle throughout the day. Corresponding to the k -th car, the empty driving cost of entering/exiting the depot is calculated as follows:

$$S_1 = (M + d_{ki}) \cdot \left(\sum_{k \in D} \sum_{j \in V} x_k^i \cdot t_{i,D} + \sum_{k \in D} \sum_{j \in V} x_{jk} \cdot t_{D,i} \right) \quad (5)$$

For the k -th vehicle, the total empty driving cost between shifts can be calculated by the product of the inter-shift empty driving time and the unit empty driving cost:

$$S_2 = t_{ij} \cdot \sum_{i=1}^N \sum_{j=1}^N x_{ij} \cdot \delta_{ij} \cdot e_{ij} \quad (6)$$

The waiting time between shifts refers to the time difference between the end time after the current vehicle completes the task of shift i and the start time of the next shift j . If there is an empty driving between shifts i and j , the idle time needs to be subtracted:

$$S_3 = w_{ij} \cdot \sum_{i=1}^N \sum_{j=1}^N x_{ij} \cdot \delta_{ij} \cdot (d_j - e_i - e_{ij}) \quad (7)$$

According to the above requirements, the object function of the SD-ML VSP model can be constructed:

$$\text{Min} = S_1 + S_2 + S_3 \quad (8)$$

$$\text{s.t.} = \begin{cases} \delta_k^{ij} = \begin{cases} 1 & S(i) = E(j) \\ 0 & S(i) \neq E(j) \end{cases} \\ x_k^{Di} = \begin{cases} 0 & y_k^{di} = 0 \\ 1 & y_k^{di} = 1 \end{cases} \\ y_k^{ij} = \begin{cases} 0 & \text{next}(k_i) \neq j \\ 1 & \text{next}(k_i) = j \end{cases} \\ \sum_{k \in D} \sum_{i \in V} x_{ki} + \sum_{i=1}^N \sum_{j=1}^N x_{ij} = 1 \\ \sum_{k \in D} \sum_{j \in V} x_{jk} + \sum_{j=1}^N \sum_{i=1}^N x_{ji} = 1 \end{cases}$$

Among them, the first constraint is to screen out dead heading blocks, the fourth constraint is to ensure that each block has and can only be executed once, and the fifth constraint is to ensure that the executed block can only return to the depot or continue to execute the next block.

Fig. 5 Solution case

Algorithm design

The DP-TABU algorithm first considers four factors: the empty driving cost of the vehicle from the depot to the starting station of the line, the empty driving cost between adjacent shifts, the waiting time between shifts, and the empty driving cost from the terminal station of the line to the depot is solved by the dynamic programming algorithm to solve the initial solution of the shift chain, and then the loss function is created by the tabu search algorithm. Tabu search algorithm originated from the imitation of human memory function. It is a meta-heuristics algorithm [17–19]. It starts from an initial feasible solution, tries a series of specific search directions (movements), and selects the movement that maximizes the specific objective function. To avoid falling into the local optimal solution, the tabu search algorithm records the search process information that has been experienced so as to know the next search direction.

Solution standard

The solution to the vehicle scheduling problem of single depot and multiple lines can be encoded with a set of two-dimensional arrays, where the first column is the execution vehicle and the second column is the execution shift. Assuming that five vehicles execute three routes and a total of fifteen shifts (shifts 1–5 belong to the first route, shifts 6–10 belong to the second route, and shifts 11–15 belong to the third route), the solution to this problem as Fig. 5.

The decoding of this problem is mainly based on the encoding of the two-dimensional array. Taking Fig. 5 as an example, since all the shifts of the three lines can be executed by the four vehicles in the depot, the execution shift of the first vehicle is shift 1-shift 4-shift 9, the first vehicle gen-

erates cross-line dispatch (line 1–line 2), the second vehicle executes shift 2–shift 6–shift 7–shift 11–shift 13, and the second vehicle generates Cross-line dispatch (line 1–line 2–line 3), the execution of the third vehicle is shift 3–shift 8–shift 10–shift 12, and the third vehicle generates cross-line dispatch (line 1–line 2–line 3) The execution shift of the 4th vehicle is shift 5–shift 13–shift 15, and the fourth vehicle generates cross-line dispatch (line 1–line 3).

Neighborhood structure

Neighborhood structure [20–22] refers to the collection of other points near a given point. In combinatorial optimization problems, neighborhood refers to the collection of all solutions that can be obtained by performing an operation on the current solution (this operation can be called a neighborhood action). Then the essential difference of the neighborhood lies in the different actions of the neighborhood.

The neighborhood search algorithm is a very common type of improved algorithm, which finds a better solution by searching the “neighborhood” of the current solution in each iteration. The key to the design of the neighborhood search algorithm is the choice of neighborhood structure, that is, the way the neighborhood is defined.

In view of the SD-ML-VSP problem, three neighborhood operations of block crossing between vehicles, block insertion, and block merging are set here.

Block crossing strategy, that is, two block chains are randomly selected, and two blocks are randomly selected from the selected block chains to replace each other. For example, vehicle 1 (1-4-9) selects block 4, vehicle 3 (3-8-10-12) selects block 10 for block cross replacement to get a new solution: vehicle 1 (1-10-9), vehicle 3 (3-8-4-12).

block insertion strategy refers to randomly selecting two block chains, and randomly selecting one block in one block chain and inserting it into a random position in the other block chain. For example, vehicle 1 (1-4-9) selects block 4 and inserts it into vehicle 3 (3-8-10-12) to obtain a new solution: vehicle 1 (1-9), vehicle 3 (3-8-4-10-12).

The block merging strategy refers to randomly selecting two block chains, and randomly selecting one block chain to merge with another block chain, combine vehicle 1 (1-4-9) selection and vehicle 3 (3-8-10-12)) to get a new solution: Vehicle 3 (3-8-10-12-1-4-9)

In the algorithm iteration process, the block merging strategy, block insertion strategy, and block merging strategy are executed in random order, and the block chain merging operator is called after the execution of the strategy. After each operator is executed, it first judges whether the newly generated block chain meets the requirements, if not, the adjustment will be abandoned, otherwise it will be added to the block chain candidate set

Solution generation

Tabu search algorithm is a manifestation of artificial intelligence and an extension of local search. The tabu search algorithm is based on the local search, by setting a tabu list to stop some operations that have been performed, and using the contempt rule to unban some excellent solutions. The basic steps of the DP-Tabu algorithm:

- Initialization: First, the dynamic planning algorithm is adopted by considering four factors: the empty driving cost from the depot to the starting station of the line, the empty driving cost between adjacent blocks, the waiting time between blocks, and the empty driving cost from the terminal to the depot from the line terminal to solve the initial solution.
- The neighborhood structure generates the candidate solution: through step 1, the initial solution is generated. The neighborhood structure strategy: Block crossing strategy, block insertion strategy, and the block merging strategy are used to generate candidate solutions.
- Tabu search algorithm updates the optimal solution: select the solution with the best fitness value from all the candidate solutions produced in step 2 and compare it with the current optimal solution. If it is better than the current optimal solution, update the current optimal solution and take it as the next one iteration of the current solution, and then add the corresponding operation to the tabu table; if it is not better than the current optimal solution, select the optimal solution that is not in the tabu state from all candidate solutions as the new current solution, and then add the corresponding operation Join the taboo list.
- Judging the termination condition: If the set termination condition is met, the iterative loop will be immediately stopped to output the current optimal solution, otherwise the search operation will continue. The general termination condition is whether the specified number of iterations or iteration time is reached.

Since the optimization scale of multi-line scheduling is relatively large, a plan of “individual adjustment and optimization into zones” was designed in the solution process. According to this plan, the optimization results of single-line scheduling were integrated for regional optimization according to the different routes of vehicles. Take dp-tabu algorithm as the core to solve multiple lines in the entire area cyclically, and determine the final regional scheduling plan that meets the least loss

Fig. 6 Block crossing strategy **Fig. 7** Block insertion strategy **Fig. 8** Block merging strategy

Case study

This paper applies this algorithm to the regional scheduling problem of Zhengzhou bus S105, S107, and S159. Among them, S105 has 11 vehicles, 90 blocks, and the mileage is 11 km, S107 has 13 vehicles, 77 blocks, and the mileage is 15 km, S159 has 15 vehicles, 72 blocks, and a mileage of 17 km. The three routes S105, S107, and S159 are all in the same depot. The three routes are known to have a total of 239 blocks, and the departure timetable of each route is known. Set the maximum mileage allowed throughout the day is 120 km, the length of the tabu table is set to 10, and the maximum number of iterations is set to 2000. The execution plan of the vehicle can be obtained as shown in the Table 2.

It can be seen from Table 2 that 20 vehicles are all responsible for regional scheduling tasks, but there are also 10 vehicles performing single-line scheduling tasks. According to the workload of the day, it can be seen that 30 vehicles are needed to meet operational requirements.

Through the scheduling plan, it is known that 39 vehicles are needed to arrange the full-day operation plan. Table 3 compares the difference between this algorithm and the manual scheduling scheme in detail.

According to Table 3, it can be concluded that for the same station area scheduling problem for the three lines of S105, S107, and S159, 39 vehicles are manually required to complete the scheduling plan, while the optimal scheduling scheme of the DP-TS algorithm only needs 26 vehicles to meet the constraints. Under the premise of the conditions, the regional dispatch of vehicles is realized, and the effective reduction of the parking and rest time of the vehicles greatly improves the turnover efficiency of the vehicles.

Simulated annealing algorithm has advantages that traditional optimization algorithms can hardly compare in dealing with difficult problems such as global optimization and discrete variable optimization. Therefore, many scholars often use this algorithm to solve NP-hard problems. In this paper, the SA algorithm is used to solve the SD-ML-VSP problem, and the initial temperature is set to 5000 degrees, the cooling coefficient is 0.94, and the number of iterations is 1000.

Table 4 compares the DP-TS, SA algorithm, and the cost of the scheduling scheme for single station area scheduling calculated by manual scheduling. Obviously, the turnover efficiency of the scheduling scheme obtained by the DP-Tabu algorithm is much lower than the other two schemes. This is because the algorithm takes into account the maximum rest time between shifts and the maximum working mileage constraints of the vehicle during the design process, and takes into account the impact of different constraints on the target value, but it can be found that the calculation time of the SA algorithm is lower than that of the DP-Tabu algorithm, this is because the SA algorithm does not consider the influence of the neighborhood operator when calculating the scheduling plan, while the DP-Tabu algorithm considers the influence of the neighborhood operator in the design process. Therefore, the calculation time is slightly longer than that of SA algorithm. Therefore, the DP-Tabu algorithm can further optimize the single station multi-line regional scheduling schedule, improve the quality of the solution, and verify the effectiveness of the algorithm.

It can be seen from the Fig. 9 that the number of bus vehicles used by different scheduling schemes are different. The manual scheme requires 39 vehicles, and different scheduling results can be obtained by changing the parameters of the dp-tabu algorithm. This article mainly compares the 26

Table 2 Final result

Bus	Arrangement	Mileage (km)
1	s105_1-s105_11-s105_22-s105_30-s159_99-s159_102-s105_59-s159_107-s159_109-s105_89	134
2	s105_2-s105_12-s105_23-s105_31-s105_41-s105_50-s105_60-s105_69-s105_79-s105_90	110
3	s105_3-s105_13-s105_24-s105_32-s105_32-s105_42-s105_51-s105_61-s105_70-s105_80	99
4	s105_4-s105_14-s105_25-s105_33-s105_43-s105_52-s159_105-s105_71-s105_81	105
5	s159_91-s105_15-s105_26-s105_35-s159_100-s105_54-s105_63-s105_73-s159_110	117
6	s105_6-s105_16-s159_95-s105_36-s105_45-s105_55-s159_106-s105_74-s105_83	111
7	s105_5-s159_93-s159_94-s105_34-s105_44-s105_53-s105_62-s105_72-s105_82	111
8	s159_92-s105_17-s105_27-s105_37-s105_46-s159_103-s105_64-s105_75-s105_84	111
9	s105_7-s105_18-s105_28-s105_38-s105_47-s105_56-s105_65-s105_76-s105_85	99
10	s105_8-s105_19-s159_96-s159_98-s159_101-s159_104-s105_66-s159_108-s105_86	129
11	s105_9-s105_20-s105_29-s105_39-s105_48-s105_57-s105_67-s105_77-s105_87	99
12	s105_10-s105_21-s159_97-s105_40-s105_49-s105_58-s105_68-s105_78-s105_88	105
13	s107_1-s107_8-s107_18-s107_29-s107_38-s107_48-s107_59-s107_69-s159_99	137
14	s159_78-s159_83-s107_19-s159_86-s107_39-s107_49-s159_92-s159_94-s107_76	145
15	s107_2-s107_9-s107_20-s107_30-s159_89-s107_50-s107_60-s107_70-s159_100	139
16	s159_79-s107_10-s107_21-s107_31-s107_40-s107_51-s107_61-s159_95-s107_77	139
17	s107_3-s107_11-s107_22-s107_32-s107_41-s107_52-s107_62-s107_71-s159_101	137
18	s159_80-s107_12-s107_23-s107_33-s107_42-s107_53-s107_63-s107_72	122
19	s107_4-s107_13-s107_24-s107_34-s107_43-s107_54-s107_64-s159_96	122
20	s159_81-s107_14-s107_25-s107_35-s107_44-s107_55-s107_65-s107_73	122
21	s107_5-s107_15-s107_26-s159_87-s107_45-s159_91-s107_66-s159_97	126
22	s107_6-s159_84-s107_27-s107_36-s107_46-s107_56-s107_67-s107_74	122
23	s107_7-s107_16-s107_28-s107_37-s159_90-s107_57-s159_93-s159_98	126
24	s159_82-s107_17-s159_85-s159_88-s107_47-s107_58-s107_68-s107_75	126
25	s159_1-s159_7-s159_13-s159_19-s159_25	85
26	s159_2-s159_8-s159_14-s159_20-s159_26	85
27	s159_3-s159_9-s159_15-s159_21-s159_27	85
28	s159_4-s159_10-s159_16-s159_22-s159_28	85
29	s159_5-s159_11-s159_17-s159_23	68
30	s159_6-s159_12-s159_18-s159_24	68

Table 3 Scheme comparison

Total bus	Manual	30 bus plan	Difference	26 Bus plan	Difference
Operating vehicles	39	26	13	30	9
S105	11	10	1	12	−1
S107	13	11	2	12	1
S159	15	5	10	6	9
Working time	272h10m	272h10m	0	272h10m	0
Resting time	156h42m	90h55m	None	108h	None
Running time	428h52m	363h5m	65h47m	382h35m	46h17m
Turnover efficiency	63.46%	74.96%	None	70.52%	None

Table 4 Comparison of computing results using DP-TS and SA algorithm

Algorithms	Vehicles	Resting time	Efficiency	Calculating time(min)
DP-TS	26	90h55m	74.96%	5.36
SA	31	110h30m	69.53%	2.47
Manual	39	156h42m	63.46%

Fig. 9 Result of SD-ML-VSP

vehicles plan and the 30 vehicles plan. Since the adopted timetable is the same, the working hours of the three lines are the same, but the vehicle resources are saved due to the improvement of vehicle turnover efficiency.

For any search algorithm for optimization, its convergence has important theoretical significance. The algorithm designed in this paper has good convergence in the solution process. The figure below shows the convergence of the algorithm after a certain regional dispatch line is called. From the convergence of the solution courseware, the DP-Tabu algorithm has better convergence and can quickly find

the optimal solution. The iterative convergence process is shown in Fig. 10.

Conclusion

DP-TABU algorithm can be widely applied in the field of public transport service requirements of single parking area multi-line scheduling of the scene. The algorithm can effectively solve the traditional drivers such as labor efficiency in rotation type scheduling scheme is too low, the execution plan cost disadvantage of vehicles, drivers and other resources is

Fig. 10 Iteration progress

too high, the bus companies and the jobs program match the vehicle lines more efficient, more intelligent, more accurate to further enhance public enterprise information technology, digital and intelligent level.

This solution is designed to study the country, “the driver of the vehicle bundled” mode of SD-ML-VSP problem, taking into account the cost of the vehicle and driver, and using scientific model algorithm, namely DP-Tabu algorithm to find and meet the minimum number of vehicles used to dispatch scheduling plan, and verify the effectiveness of the algorithm and applied to the actual production environment.

Compared with the original technology program which can be used with a variety of preparation methods buses driving program, while ensuring the driver has enough time to rest (dwell time) of the case, try to improve labor efficiency and taking into account the empty driving on vehicle scheduling.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Tang Jinjun, Yang Yifan, Qi Yong (2018) A hybrid algorithm for urban transit schedule optimization. *Physica A* 512:745–755
2. Yuan W et al (2017) A data-driven and optimal bus scheduling model with time-dependent traffic and demand. *IEEE Trans Intell Trans Syst* 18(9):2443–2452
3. Zhao X et al (2020) Two-way Vehicle Scheduling Approach in Public Transit Based on Tabu Search and Dynamic Programming Algorithm. In: 2020 IEEE 5th international conference on intelligent transportation engineering (ICITE). IEEE
4. Xingquan Z et al (2014) Vehicle scheduling of an urban bus line via an improved multiobjective genetic algorithm. *IEEE Trans Intell Trans Syst* 16(2):1030–1041
5. Barbarosoglu Gulay, Ozgur Demet (1999) A tabu search algorithm for the vehicle routing problem. *Comput Oper Res* 26(3):255–270
6. Sakoe H, Chiba S (1978) Dynamic programming algorithm optimization for spoken word recognition. *IEEE Trans Acoustics Speech Signal Process* 26(1):43–49
7. Consul PC, Jain GC (1973) A generalization of the Poisson distribution. *Technometrics* 15(4):791–799
8. Mirjalili S, Siti H, Mohd Z (2010) A new hybrid PSOGSA algorithm for function optimization. In: 2010 international conference on computer and information application. IEEE
9. Enjian Yao et al (2020) Optimization of electric vehicle scheduling with multiple vehicle types in public transport. *Sustainable Cities Soc* 52:101862
10. Min W et al (2016) An adaptive large neighborhood search heuristic for the electric vehicle scheduling problem. *Comput Oper Res* 76:73–83
11. Murray JJ et al (2002) Adaptive dynamic programming. In: *IEEE transactions on systems, man, and cybernetics, Part C (Applications and Reviews)* 32(2), 140–153
12. Eddy Sean R (2004) What is dynamic programming? *Nat Biotechnol* 22(7):909–910
13. Olle S, Guzzella L (2009) A generic dynamic programming Matlab function. In: 2009 IEEE control applications (CCA) & intelligent control, (ISIC). IEEE
14. Fiechter C-N (1994) A parallel tabu search algorithm for large traveling salesman problems. *Discrete Applied Mathematics* 51(3):243–267
15. Costa D (1994) A tabu search algorithm for computing an operational timetable. *Euro J Oper Res* 76(1):98–110
16. Schermer Daniel, Moeini Mahdi, Wendt Oliver (2019) A hybrid VNS/Tabu search algorithm for solving the vehicle routing problem with drones and en route operations. *Comput Oper Res* 109:134–158
17. Osman Ibrahim H, Kelly James P (1997) Meta-heuristics theory and applications. *J Oper Res Soc* 48(6):657–657
18. Voß S et al (2012) eds. *Meta-heuristics: advances and trends in local search paradigms for optimization*. Springer Science & Business Media
19. Jones Dylan F, Mirrazavi S. Keyvan, Tamiz Mehrdad (2002) Multi-objective meta-heuristics: an overview of the current state-of-the-art. *Euro J Oper Res* 137(1):1–9
20. Beyers JM et al (2003) Neighborhood structure, parenting processes, and the development of youths externalizing behaviors: a multilevel analysis. *Am J Community Psychol* 31(1):35–53
21. Zhang CY et al (2007) A tabu search algorithm with a new neighborhood structure for the job shop scheduling problem. *Comput Oper Res* 34(11):3229–3242
22. Li Jun-Qing et al (2011) A hybrid tabu search algorithm with an efficient neighborhood structure for the flexible job shop scheduling problem. *Int J Adv Manuf Technol* 52(5):683–697

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.