A turn-back track constraint train scheduling algorithm on a multi-interval rail transit line

Z. B. Jiang[1] & X. Y. Xiao[1,2]
[1]*School of Transportation Engineering, Tongji University, China*
[2]*The Key Laboratory of Road and Traffic Engineering, Ministry of Education, Tongji University, China*

Abstract

With the rapid increase of passenger volume, the interval of some rail transit lines in China has reached a minimum value, which makes turn-back capacity one of the main capacity-limiting factors instead of section headway, particularly during the peak hours on workdays. A general simulation modeling framework and algorithm in which the divide and conquer rule is adopted for train scheduling problems on a multi-interval rail transit line with turn-back track constraint is proposed in the paper, and the objective is to obtain a feasible timetable based on passenger demand at different time periods, line capacity, number of rolling stocks, location of depots, train routing, turn-back type. Lastly, a new version of the TPM software is programmed by the proposed framework and algorithm, and Shanghai Rail Transit Line 2 is presented to demonstrate. The result shows that the proposed algorithm performs very efficiently, and a feasible timetable with a five-interval period, 535 trains and one turn-back track constraint at GLR station is generated in 15 seconds by TPM.
Keywords: rail transit, train scheduling, multi-interval, turn-back track constraint.

1 Introduction

Rail transit management has become an increasingly important issue of the public transport system, one of which is the rail transit scheduling problem (*RTSchPrb*). This problem focuses on determining a feasible timetable for a set of trains that doesn't violate track capacity and satisfies operational constraints. Compared with railway system, the *RTSchPrb* emphasizes that all trains of the

WIT Transactions on The Built Environment, Vol 135, © 2014 WIT Press
www.witpress.com, ISSN 1743-3509 (on-line)
doi:10.2495/CR140121

whole day need be programmed at one time in a relatively short time. Urban rail transit has been developing rapidly in many cities in China, and passenger flow grows significantly. Under the condition of huge passenger volume, the interval of some lines in Shanghai and Beijing has reached a minimum value, making turn-back capacity one of main capacity-limiting factors instead of section headway. Particularly during the peak hours in workdays, the number of turn-back track has turned into a critical limiting factor of line capacity. The most feasible solution to serve the continuously increasing demand with an appropriate service level is to improve the quality of scheduled timetable. Previous works have paid more attention to line headway, rolling stocks circulation and train route. The research presents a general simulation modeling framework and algorithm for timetable designing on multi-interval rail transit line with turn-back track constraint based on passenger demand in different periods, line capacity, number of rolling stocks, location of depots, train route and turn-back mode.

2 Literature review

In fact, the *RTSchPrb* is a NP-hard problem and the existing researches about the problem have aimed at achieving a rail transit train timetable with arrival and departure time of all trains at the visited stations. In railway system, studies generally began with an infeasible initial timetable with some conflicts, and after these conflicts were solved a feasible train timetable is composed depending on which trains are operated. In most papers about the scheduling problem in railway system the related mathematical models are built such as the studies of Odijk [1], Higgins *et al.* [2], Lee and Chen [3], Brannlund *et al.* [4]. In only a few papers a simulation model was developed for the train scheduling problem. In railway system there are several comprehensive simulation tools to optimize and evaluate timetable, such as OpenTrack and Railsys in studies of Grube *et al.* [5]. In urban rail transit system, YalçInkaya and Mirac Bayhan [6] presented a modeling and solution approach based on discrete-event simulation and response surface methodology to deal with average passenger travel time optimization problem inherent to the metro planning process, Cadarso and Marín [7] proposed an integrated planning model to increase the offered capacity and system frequencies to attend the increased passenger demand and traffic congestion around urban and suburban areas. Jiang *et al.* [8] analyzed the style of shared-path routing based on rail network and the passenger flow characteristics. Also, Jiang *et al.* [9] emphasized the scheduling problem on circle rail transit lines, and a real case study of Shanghai Rail Transit Line 4 illustrated the practical value of the TPM program. Niu and Zhou [10] focused on optimizing the passenger train timetable in a heavily congested urban rail corridor.

As stated above, the *RTSchPrb* has been studied by researchers and so far many efforts have been made to solve the problem, but in the current simulation models, none of them refers to a multi-interval transit line with turn-back track constraint. Although the TPM software has been applied to train timetable programming in urban rail transit system in China, which can generate initial timetable automatically at one time, the old TPM software did not take the number of turn-back track into consideration, leading to a mass of artificial adjustments after initial timetable was generated.

In this paper, the train scheduling algorithm focuses on one-time train scheduling problem on multi-interval rail transit line with turn-back track constraint in order to obtain a feasible train timetable, including train arrival and departure time at all stations visited by trains, different intervals and the number of rolling stocks. Meanwhile, a general simulation modeling framework is proposed and depicted step by step to guide researchers to develop a train scheduling model for rail transit system. To avoid a deadlock, the divide and conquer algorithm is also embedded into the simulation model.

3 A feasible train scheduling framework

Computing a track-constrained timetable on multi-interval line is a complicated task. A lot factors, such as line topology structure, time periods, train running interval, cyclic train route or depot linking route, cyclic time and train turn-back mode, need be considered in the train scheduling framework. The most difficult process is to acquire the train timetable of the whole day at one time.

The divide and conquer (D&C) adopted in the paper is an important algorithm design paradigm based on multi-branched recursion. A divide and conquer algorithm works by recursively breaking down a problem into two or more sub-problems of the same (or related) type, until which become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem. The divide and conquer rule is a powerful tool for solving conceptually difficult problem.

Train scheduling problem with turn-back track constraint on a multi-interval line can be subdivided by three main problems: cyclic train scheduling problem with same interval, train scheduling problem in the transition process between different periods and scheduling problem of depot linking trains. And the scheduling problem in transition process can also be subdivided two main problems: train scheduling in transition process without turn-back track constraint and that with turn-back track constraint. Figure 1 shows the concept of the divide and conquer rule applied to the proposed model in this paper.

Table 1 presents the parameters and variables that are used in this model. The unit of all time-related parameters and variables is 1 second.

3.1 Train scheduling process in the same interval period

In the period with the same interval, the timetable is typically periodic. Cyclic time (T_i^C) is the time of train running a round-trip between the original and terminal stations, including four parts, shown in eqn (1).

$$T_i^C = t_u^r + t_{s_1}^t + t_d^r + t_{s_n}^t \tag{1}$$

The total number of rolling stocks (N_i^S) in period i could be calculated by cyclic time and train interval, shown in eqn (2).

$$N_i^S = \lceil T_i^C / t_i^I \rceil \tag{2}$$

where $\lceil Num \rceil$ is the maximal integer less than *Num*. Thus, the actual interval of trains in period i (t_i^{AI}) can be calculated by eqn (3).

$$t_i^{AI} = T_i^C / N_i^S \tag{3}$$

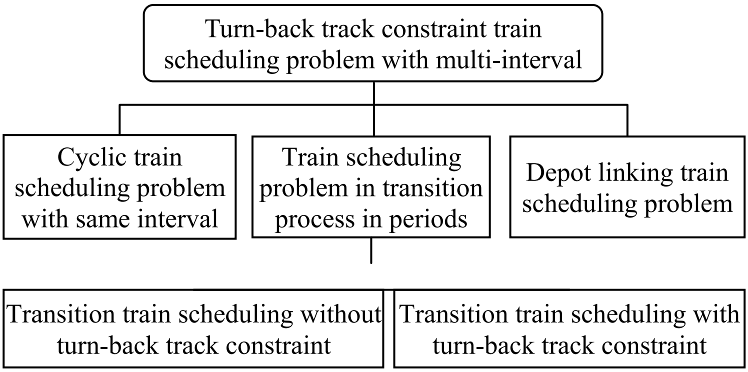

Figure 1: Sub-problems of train scheduling on multi-interval rail transit line.

Table 1: Definition of parameters and variables.

Symbol	Description
$\boldsymbol{i}$	the ID of time period
$\boldsymbol{t_i^I}$	Suggesting train interval in period i
$\boldsymbol{t_i^{AI}}$	actual train interval in period i
$\boldsymbol{T_i^C}$	cyclic time in period i
$\boldsymbol{T_i^{PS}}$	starting time of period i
$\boldsymbol{T_i^{PE}}$	ending time of period i
$\boldsymbol{s_i}$	station i
$\boldsymbol{t_u^r}$	train running time in upward direction
$\boldsymbol{t_d^r}$	train running time in downward direction
$\boldsymbol{t_s^t}$	train turn-back time at station s
$\boldsymbol{t_s^{t,max}}$	maximal turn-back time at station s
$\boldsymbol{t_s^{t,min}}$	minimal turn-back time at station s
$\boldsymbol{LT_i()}$	array of non-linking trains at the beginning (left side) of period i
$\boldsymbol{RT_i()}$	array of non-linking trains at the ending (right side) of period i
$\boldsymbol{n(RT_i/LT_i)}$	the number of trains in RT_i or LT_i
$\boldsymbol{RT_i(n)/LT_i(n)}$	the n^{th} train in RT_i or LT_i
$\boldsymbol{v_k^u}$	train v_k in upward direction
$\boldsymbol{v_k^d}$	train v_k in downward direction
$\boldsymbol{a_{v,s}}$	arrival time of train v at station s
$\boldsymbol{d_{v,s}}$	departure time of train v at station s

According to Figure 2, it is easy to calculate the departure time of the trains with the algorithm shown in Table 2.

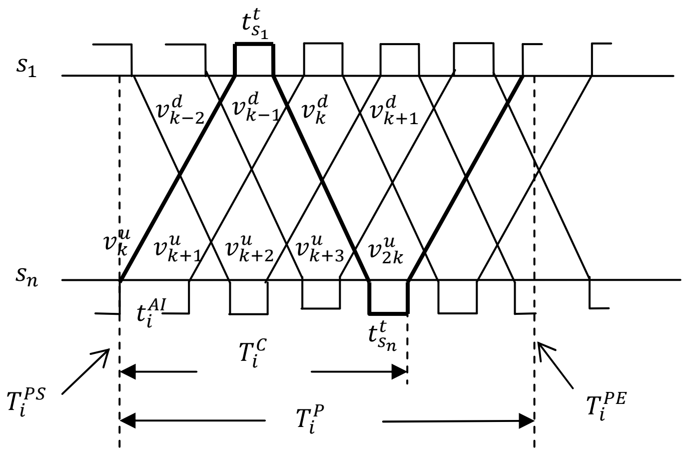

Figure 2: Train scheduling in the same interval period.

Table 2: Algorithm 1 of train scheduling in the same interval period.

Step0: Initialization;

Step1: Calculate the number of rolling stocks in period i, let $k = N_i^S = \lceil T_i^C / t_i^I \rceil$, then $t_i^{AI} = T_i^C / k$;

Step2: Let $d_{v_k^u, s_n} = T_i^{PS}$, and set $p \leftarrow 1,\ q \leftarrow 1$;

Step3: If $q = k$, go to Step6, else go to Step4;

Step4: Calculate $t = d_{v_{k\cdot p+q-1}^u, s_n} + t_u^r + t_{s_1}^t$, if $t \le T_i^{PE}$, let $d_{v_{k\cdot p+q-1}^d, s_1} = t$, and connect train $v_{k\cdot p+q-1}^u$ and train $v_{k\cdot p+q-1}^d$, go to Step5, else $p \leftarrow 1$, $q \leftarrow q+1$, go to Step3;

Step5: Calculate $t = d_{v_{k\cdot p+q-1}^u, s_n} + T_i^C$, if $t \le T_i^{PE}$, let $d_{v_{k\cdot(p+1)+q-1}^u, s_n} = t$, connect train $v_{k\cdot(p+1)+q-1}^u$ and train $v_{k\cdot p+q-1}^d$, then let $p \leftarrow p+1$, go to Step4, else $p \leftarrow 1$, $q \leftarrow q+1$, go to Step3;

Step6: Set $m \leftarrow 1$;

Step7: Calculate $t = d_{v_{(k-m+1)}^d, s_1} - t_i^{AI}$, if $t > T_i^{PS}$, let $d_{v_{(k-m)}^d, s_1} = t$, and connect train $v_{(k-m)}^d$ and train $v_{(2k-m)}^u$, $m \leftarrow m+1$, go to Step7, else go to Step8;

Step8: Record left upward trains $LT_i^u()$, left downward trains $LT_i^d()$, right upward trains $RT_i^u()$ and right downward trains $RT_i^d()$.

3.2 Multi-interval train scheduling without turn-back track constraint

In rail transit system, trains need turn-back at terminal stations. According to station type, the turn-back mode can be characterized by two types: turn-back operation with crossover located in advance of station (TAS) and in back of station (TBS), as illustrated by Figure 3. Compared with TAS, station layout of TBS is relatively more complex.

If there is no turn-back track constraint in station s_1 and s_n, trains at transition period can be linked if the turn-back time is not less than $t_{s_i}^{t,min}$. The scheduling process is presented in Figure 4 and described in detail by Table 3.

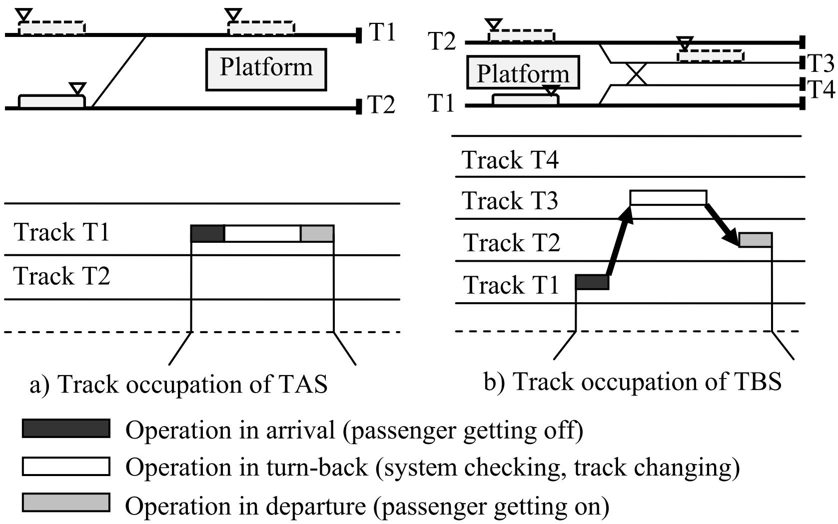

Figure 3: Track occupation of turn-back operation process at terminal station.

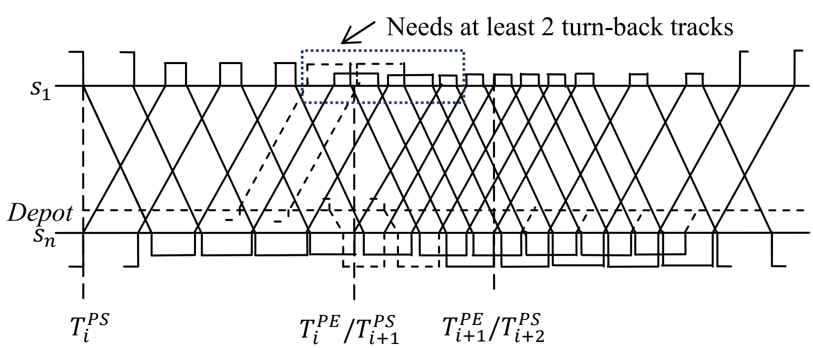

Figure 4: Multi-interval train scheduling without turn-back track constraint.

Table 3: Algorithm 2 of train scheduling without turn-back track constraint.

Step0: Initialization, read the result information of Algorithm 1;
Step1: Let $m \leftarrow 1, n \leftarrow 1$, and set $num1 = n(RT_i^u)$, $num2 = n(LT_{i+1}^d)$;
Step2: If $m \leq num1$, let $v_1 = RT_i^u(m)$, go to Step3, else go to Step6;
Step3: If $n \leq num2$, let $v_2 = LT_{i+1}^d(n)$, go to Step4, else go to Step6;
Step4: Calculate $t_{link} = d_{v_2,s_1} - a_{v_1,s_1}$, if $t_{s_1}^{t,min} \leq t_{link} \leq t_{s_1}^{t,max}$, connect train v_2 and train v_1,remove v_2 from $LT_{i+1}^d()$ and remove v_1 from $RT_i^u()$, let $m \leftarrow m+1$ and $n \leftarrow n+1$, go to Step2, else go to Step5;
Step5: If $num1 < num2$, add a train v_3 from the depot to s_1, let $t_{link} = d_{v_2,s_1} - a_{v_3,s_1}$, and make sure $t_{s_1}^{t,min} \leq t_{link} \leq t_{s_1}^{t,max}$, remove v_2 from $LT_{i+1}^d()$, let $n \leftarrow n+1$, go to Step2, else let train v_1 enter depot, remove v_1 from $RT_i^u()$, $m \leftarrow m+1$, go to Step2;
Step6: If $n(RT_i^u) > 0$, let every train in $RT_i^u()$ enter depot, if $n(LT_{i+1}^d) > 0$, add trains from depot to connect every train in $LT_{i+1}^d()$ respectively;
Step7: Use the same method to deal with the trains in $RT_i^d()$ and $LT_{i+1}^u()$;
Step8: Use the same method to deal with other periods;
Step9: Calculate the timetable of all trains at each station.

3.3 Multi-interval train scheduling with turn-back track constraint

Due to the turn-back track constraint at station s_1, in the process of off-peak period converting to peak period, the transition of different periods should start from the other station s_n, that is to say, trains out of depot need turn-back at station s_n at first, which is different from that trains out of depot turn-back at station s_1 in Figure 4, the same with the process of trains entering depot. At this situation, although less turn-back track is needed, few more additional trains need be added, as the dotted lines shown in Figure 5. Table 4 presents the train scheduling algorithm in multi-interval period with turn-back track constraint.

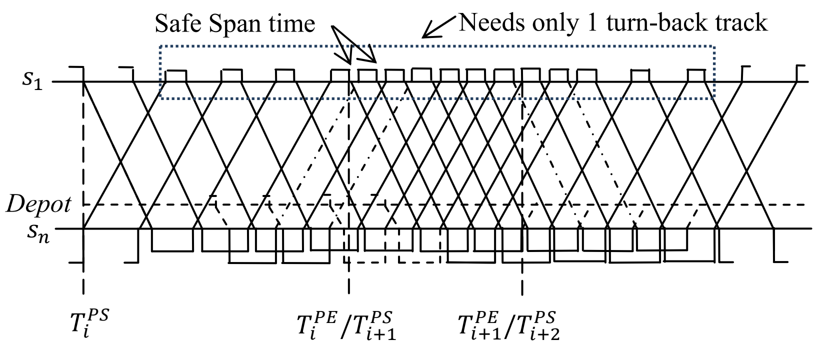

Figure 5: Multi-interval train scheduling with turn-back track constraint.

Table 4: Algorithm 3 of train scheduling with turn-back track constraint.

Step0: Initialization;

Step1: Calculate the number of rolling stocks in period i,
let $k = N_i^S = \lceil T_i^C / t_i^I \rceil$, then $t_i^{AI} = T_i^C / k$;

Step2:Set $d_{v_k^u, s_n} = T_i^{PS}$, $p \leftarrow 1$, $q \leftarrow 1$;

Step3: If $q = k$, go to Step6, else go to Step4;

Step4: Calculate $t = d_{v^u_{k \cdot p+q-1}, s_n} + t_u^r + t_{s_1}^t$, if $t \le T_i^{PE}$, let
$d_{v^d_{k \cdot p+q-1}, s_1} = t$, and connect train $v^u_{k \cdot p+q-1}$ and train $v^d_{k \cdot p+q-1}$, go to Step5, else $p \leftarrow 1$, $q \leftarrow q + 1$, go to Step3;

Step5: Calculate $t = d_{v^d_{k \cdot p+q-1}, s_1} + T_i^C$, if $t \le T_i^{PE}$, let $d_{v^u_{k \cdot (p+1)+q-1}, s_n} = t$, connect train $v^u_{k \cdot (p+1)+q-1}$ and train $v^d_{k \cdot p+q-1}$, then let $p \leftarrow p + 1$, go to Step4, else $p \leftarrow 1$, $q \leftarrow q + 1$, go to Step3;

Step6: Set $m \leftarrow 1$;

Step7: Calculate $t = d_{v^d_{(k-m+1)}, s_1} - t_i^{AI}$, if $t > T_i^{PS}$, let $d_{v^d_{(k-m)}, s_1} = t$, connect train $v^d_{(k-m)}$ and train $v^u_{(2k-m)}$, go to Step8, else go to Step10;

Step8: Calculate $d_{v^u_{(k-m)}, s_n} = d_{v^d_{(k-m)}, s_1} - t_u^r - t_{s_1}^t$, connect train $v^u_{(k-m)}$ and train $v^d_{(k-m)}$;

Step9: Let $m \leftarrow m + 1$, go to Step7;

Step10: Record left upward trains $LT_i^u()$ and right downward trains $RT_i^d()$ in period i;

Step11: Deal with other periods with the same method;

Step12: Deal with non-linking trains $RT_i^d()$ and $LT_{i+1}^u()$ with Algorithm 2;

Step13: Calculate the timetable of all trains at each station.

3.4 Computing process of train scheduling

The computing process of train scheduling with turn-back track constraint is by the sequence of the following: (1) line topology structure defining and time parameter inputting; (2) time period and interval inputting; (3) cyclic train scheduling with same interval; (4) train scheduling in transition process with turn-back track constraint or without turn-back track constraint; (5) depot linking train scheduling; (6) timetable adjusting and optimizing by user interface; (7) indexes calculating; (8) graphics and tables outputting.

The train scheduling process is shown in Figure 6, and this framework is computed by the new version of TPM software (Version 5.1), of which the interface is shown in Figure 7.

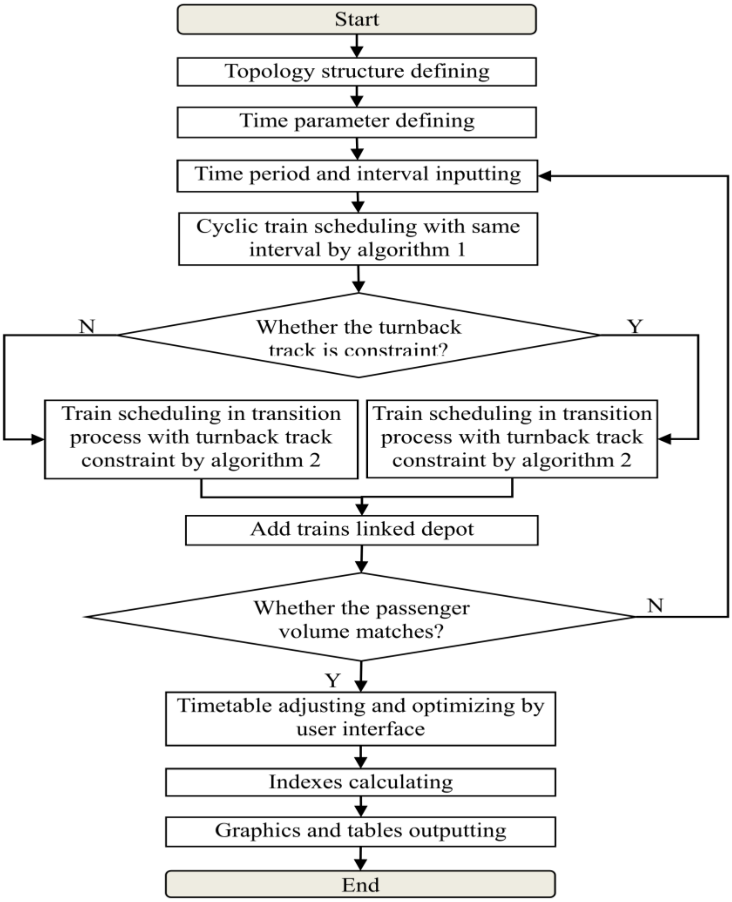

Figure 6: Train scheduling process with turn-back track constraint.

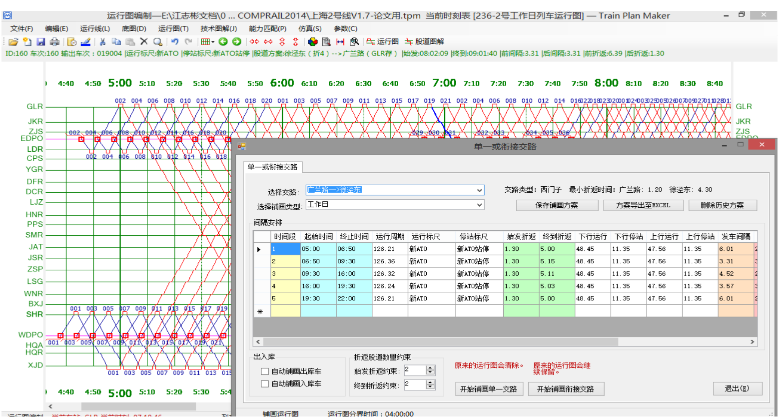

Figure 7: Interface of TPM software (Version 5.1).

4 Case study: Shanghai Rail Transit Line 2

In this section, based upon the above simulation framework, a case study is described to illustrate the application of the proposed simulation model to Shanghai Rail Transit Line 2 by the software tool named TPM (Version 5.1). Before the timetable is programmed, much initialization information is needed such as the track map of rail transit line, time period and interval, section running time, stopping time at stations, turn-back time at station and so on.

Figure 8 shows the track map of Line 2, including all the stations, depots, tracks as well as their layouts, and the time period division and detailed information are shown in Table 5.

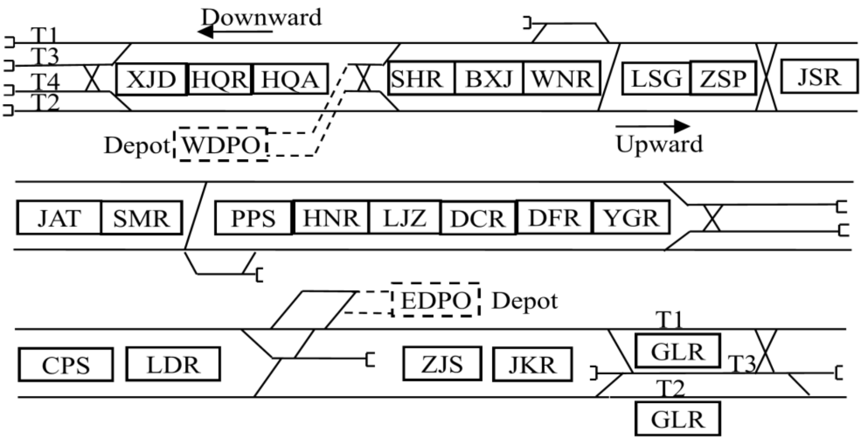

Figure 8: Track map of Line 2.

Table 5: Time period division and detailed information of Line 2.

Period ID	Start time	End time	Cyclic time	Interval	Number of rolling stocks
1	05:00	07:00	126min 21s	6min 01s	21
2	07:00	09:30	126min 36s	3min 31s	36
3	09:30	16:00	126min 32s	4min 52s	26
4	16:00	19:30	126min 24s	3min 57s	32
5	19:30	22:00	126min 21s	6min 01s	21

According to the above information, a feasible timetable with five-interval period, 535 trains and one turn-back track constraint at GLR station is generated in 15 seconds by TPM (as shown in Figure 9). Figure 10 presents the track occupation in the turn-back process at station GLR where the track can’t be occupied by more than one trains simultaneously, and that is in accordance with Figure 9.

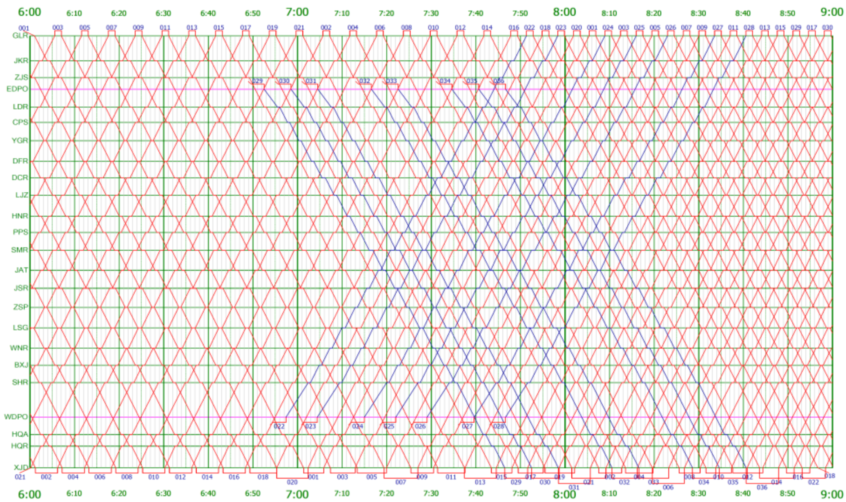

Figure 9: Train diagram of Line 2 by TPM with the proposed algorithm.

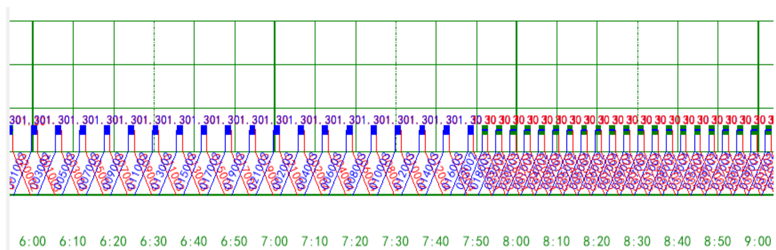

Figure 10: Track occupation in the turn-back process at station GLR.

5 Conclusion

Nowadays, one-time diagram programming problem has become a key and difficult point in urban rail transit system. Meanwhile, with the growth of passenger demand, there will be more and more capacity-limiting constraints on rail transit lines in China. The algorithm proposed in the paper with divide and conquer rule can solve the rail transit scheduling problem with turn-back track constraint in a relatively short time. In practice, the algorithm and model in the paper have been adopted by the new version of TPM software which has been successfully applied in most rail transit lines in China, such as Shanghai, Beijing, Guangzhou, Shenzhen, Kunming and Suzhou. Although the algorithm is very effective on rail transit lines with only one depot, for those with two or more depots, user interface adjustments are still necessary. Further research will focus on the algorithm optimization of rail transit lines with multi-depots.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 51008229), the Fundamental Research Funds for the Central Universities (Grant No. 20123228). The acquisition of the analysis data in this paper is supported by the Shanghai Shentong Metro Operation Management Center. The authors appreciate this support.

References

[1] Odijk, M.A., A constraint generation algorithm for the construction of periodic railway timetables, *Transportation Research Part B-Methodological*, 30(6), pp. 455-464, 1996.
[2] Higgins, A., Kozan, E., & Ferreira, L., Optimal scheduling of trains on a single line track, *Transportation Research Part B-Methodological*, 30(2), pp. 147-161, 1996.
[3] Lee, Y., & Chen, C.Y., A heuristic for the train pathing and timetabling problem, *Transportation Research Part B: Methodological*, 43(8-9), pp. 837-851, 2009.
[4] Brannlund, U., Lindberg, P.O., Nou, A., & Nilsson, J.E., Railway timetabling using Lagrangian relaxation. *Transportation Science*, 32(4), pp. 358-369, 1998.
[5] Grube, P., Núñez, F., & Cipriano, A., An event-driven simulator for multi-line metro systems and its application to Santiago de Chile metropolitan rail network, *Simulation Modelling Practice and Theory*, 19(1), pp. 393-405, 2011.
[6] YalçInkaya, Ö., & Mirac Bayhan, G., Modelling and optimization of average travel time for a metro line by simulation and response surface methodology, *European Journal of Operational Research*, 196(1), pp. 225-233, 2009.
[7] Cadarso, L., & Marín, Á., Integration of timetable planning and rolling stock in rapid transit networks, *Annals of Operations Research*, 199(1), pp. 113-135, 2012.
[8] Jiang, Z., Xu, R., Wu, Q., & Lv, J., Shared-path routing timetable computer designing in rail transit system, *Tongji Daxue Xuebao/Journal of Tongji University*, 38(5), pp. 692-696, 2010.
[9] Jiang, Z., Gao, J., & Xu, R., Circle rail transit line timetable scheduling using Rail TPM , *Proc. of the 12th Int. Conf. On Design and Operation In Railway Engineering*, eds. B. Ning & C.A. Brebbia, WITPress: Southampton, pp. 945-952, 2010.
[10] Niu, H., Zhou, X., Optimizing urban rail timetable under time-dependent demand and oversaturated conditions, *Transportation Research Part C: Emerging Technologies*, 36(11), pp. 212-230, 2013.