Research Article

Optimization Method of Locomotive Working Diagram Layout

Yajing Zheng and Wenzhou Jin

School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510641, China

Correspondence should be addressed to Yajing Zheng; ctyjzheng@scut.edu.cn

Received 26 September 2018; Revised 5 March 2019; Accepted 25 March 2019; Published 2 May 2019

Academic Editor: Eneko Osaba

Copyright © 2019 Yajing Zheng and Wenzhou Jin. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Rational scheduling of locomotive paths (the locus of the locomotive point in the train working diagram) is an important step in drawing a locomotive working diagram by a computer. But there are some problems in this process, such as the computer usually drawing a locomotive path that overlaps with another locomotive path (in the circumstances, the actual users of the locomotive working diagrams often misread the locomotive planning). At present, there are many studies about assigning sets of locomotives to each train in a preplanned train schedule; in contrast, the studies of visualizing the locomotive planning are relatively rare. Through investigating the locomotive working diagram users, this paper points out that the layout of locomotive paths should put the distance between lines being as large as possible and should put the number of the intersection between lines being as few as possible as the optimization aim which is based to solve the problem of the lines overlap or the problem of the lines beyond the margins for drawing the locomotive paths. This paper also builds the optimization model of locomotive working diagram layout. Based on determining the position of locomotive paths which can be delineated, a genetic algorithm is used to solve the optimizing model of locomotive working diagram layout in this paper. An example of a train working diagram with 36 trains is given at the end of the paper, which indicates that the optimization model of locomotive working diagram layout can better solve the problem of locomotive planning visualization.

1. Introduction

Chinese railways now not only have the computational ability for automatic train scheduling [1–3], but have also made relatively good progress in automatically compiling *locomotive scheduling* ([4, 5]). Locomotive scheduling involves optimally assigning a set of locomotives to trains while simultaneously satisfying a variety of business constraints and minimizing the total scheduling cost [6–8]. It is usually composed of two-step process: ① assigning sets of locomotives to each train in a preplanned train schedule (it is also commonly referred to as the *locomotive planning problem*, [9, 10]); ② visualizing the locomotive planning to the *locomotive working diagram* (a graphic form of locomotive planning).

Previous research in this field has focused mainly on step 1 (for example, Ghoseiri and Ghannadpour [11] consider the locomotive planning problem with homogeneous locomotives; Nourbakhsh and Ouyang [12] and Raviv and Kaspi [13] study the optimal fueling strategies for locomotives; Piu and Speranza [14] give a comprehensive review on the locomotive

planning problem; Xu et al. [15, 16] make train timetabling and locomotive planning decisions simultaneously), whereas studies on step 2, which is just as important, remain fairly scarce.

However, addressing the key theme of this paper visualizes the locomotive planning and depends essentially on optimizing the graphic representation of the locomotive planning. Although the problem resembles the train flow diagram (a train flow diagram is a direct graphic form which shows the category, path and number of all kinds of passenger, and freight train flow in railway network, which includes railway network and train flow line [17–19]), given the very different approach that needs to be taken in this case, available train flow diagram solutions cannot be adapted to the problem of locomotive planning visualization.

The *locomotive path*, which includes the *train paths* and the *connection lines* (a connection line means a locomotive from an inbound train to an outbound train at the same station in practice), is the locus of the locomotive point in the train working diagram. The difficulty in visualizing

FIGURE 1: Line position of connection lines in the cyclic locomotive plan.

locomotive planning is that quantifying the layout of connection lines. Consequently, this paper proposes a method for automatically optimizing locomotive working diagrams based on a detailed investigation into the practices actually employed by those who use the locomotive working diagram. This method can contribute to making better and more systemic automatically generated the locomotive working diagram.

2. The Position of the Connection Lines

There are three drawing areas in the locomotive working diagram for positioning connection lines (as shown in Figure 1): ① the top (hereinafter referred to as “upper section”); ② the bottom (hereinafter referred to as “lower section”); ③ the middle (hereinafter referred to as “middle section”). In reality, depending on the situation, areas called “middle section” in this paper may not exist at all or there may be several of them at one time. For simplicity and actuality, this paper assumes that there is only one middle section. In general, as shown in Figure 1, each section is equal in height (Δ in this paper indicates the height of each section), and the line position of each connection line is a series of equally spaced horizontal lines in each section. In this paper, each horizontal line, representing the line positions, is numbered 1, 2... as shown in Figure 1. If R represents the distance between adjacent horizontal lines, then $R = \Delta / (K + 1)$ (where K in the formula is the maximum value of the horizontal lines’ number; this value is derived from the train schedule and the locomotive planning and is explained in detail in Section 3.2 of this paper).

Generally, a “complete” connection line consists of three segments (as shown in Figure 2): ① the initial segment of the connection line is the left vertical line (hereinafter referred to as “LL”); ② the middle segment of the connection line is the horizontal line (hereinafter referred to as “ML”); ③ the end segment of the connection line is the right vertical line (hereinafter referred to as “RL”).

The four types shown in Figure 3 show all the possible types of a “complete” connection line. In general, upper sections only display the type shown in Figure 3(a), while lower sections only display the type shown in Figure 3(b). Middle sections may display any of the four types. In order to automatically draw the four types of connection lines according to the locomotive planning on the basic train diagram, the most efficient approach is to confirm a location of the connection lines as drawing the ML of the connection line on the smallest number line position which does not conflict with any ML. In this paper, the above method is referred to as the “smallest distribution” method. However, applying the smallest distribution method to generate locomotive working diagrams by some diagrams editing systems in practice frequently produces cases where the LL or the RL overlap during parallel operation at both ends of a station (as shown in Figure 3(a), two RLs overlap as trains are dispatched simultaneously from this station at each end of the station).

This situation will affect the way that locomotive working diagram users interpret the locomotive planning; therefore, to avoid this becoming a problem, overlapping LLs or RLs can be handled in the way illustrated in Figures 3(b) and 3(c): ① Figure 3(b) swaps the position of the two lines representing the parallel operations and then adapts the other affected connection lines. ② Figure 3(c) adds a new line position,

FIGURE 2: The diagrams of the connection line.

FIGURE 3: Commonly seen problems and solutions in traditional locomotive working diagrams editing systems.

which is used to replace one of the connection lines in the parallel operation. The visual impact of these modifications on the locomotive diagram is clear: in Figure 3(b) the spacing between the MLs is larger, whereas in Figure 3(c), there are fewer crossover points between the connection lines. As such, there are two ways to solve the problem, in visualizing the locomotive planning, caused by parallel operations being conducted at both ends of the station. ① Maximize the spacing between each line position. ② Minimize the number of crossover points between connection lines. According to the data collected for this paper through surveys from the actual users of the locomotive working diagrams, the majority of respondents said that when there were fewer connection lines on the diagram, minimizing crossover points was more important than maximizing the space between lines, whereas the opposite was true when there were a large number of connection lines. Figure 3(d) is an optimized version of the diagram in Figure 3(a), where minimizing crossover points was more important than maximizing the space between lines. In addition, determining the number of the line positions before generating a planning diagram is a precondition to studying the optimization of visualizing the locomotive planning.

Therefore, the issue of visualizing the locomotive planning discussed in this paper can be summarized as follows: assuming that the locomotive planning is known, when

adding the necessary connection lines to the train diagram, it is necessary to avoid overlapping LLs and RLs due to parallel operations at both ends of a station. To this end, it is necessary to ensure maximum spacing between adjacent MLs and minimize the number of crossover points between connection lines. In order to give this problem a general nature, for the purposes of this study a real-life locomotive planning (results of a train schedule) was used, which means that it was not a cycle diagram.

3. Mathematical Modeling

3.1. Symbol Definition. To facilitate explanations, the count time (start and stop time of the train working diagram) of the locomotive planning in this paper was set to 0:00. The symbol φ represents sections where connection lines can be positioned in the locomotive working diagram. In this paper, the value of φ is 0, 1, and 2 corresponding to the upper section, the lower section, and the middle section, respectively (if there are two or more middle sections, the value can be further increased). Let H^φ be the number set of the connection lines in the range of φ in a certain locomotive working diagram; any “complete” connection line ($h \in H^\varphi$) may consist of two lines shown in Figure 4 (in this paper, this is referred to as the four “basic lines”). For example, the

FIGURE 4: The basic lines of connection lines.

connection line shown in Figure 2(a) is composed of two basic lines from Figures 4(a) and 4(c).

In general, if a connection line does not cross the count time 0:00, it consists of two basic lines in Figure 4, and if a connection line crosses the count time 0:00, it is constituted by one basic line in Figure 4. For convenience, in this paper, all the connection lines across 0:00 will be completed as being formed of two basic lines (generally speaking, there are two ways of completion; each can be chosen arbitrarily), so that all the connection lines will contain the following three attributes: ① t_s^h , time corresponding to LL or start time of connection line h , ② t_e^h , time corresponding to RL or end time of connection line h , and ③ s^h representing the number of the line position of the connection line h . Variables a^h , b^h , c^h , and d^h are defined as 0-1: if the connection line h contains the types of line shown in Figure 4(a), $a^h = 1$. If the connection line h contains the types of line shown in Figure 4(b), $b^h = 1$. If the connection line h contains the types of line shown in Figure 4(c), $c^h = 1$. If the connection line h contains the types of line shown in Figure 4(d), $d^h = 1$. Take the connection line in Figure 2(a) as an example: $a^h = 1$, $b^h = 0$, $c^h = 1$, and $d^h = 0$.

K^φ is the set of numbered line positions within section φ , according to the numbering rules of this paper, which define $k^\varphi \in K^\varphi$ as offset multiplier relative to the horizontal line (the center of intermediate train distancing point).

Variable p_{ij}^φ is defined as 0-1 within section φ ; if the connection line i and the connection line j have an overlap in time, apply 1; otherwise, apply 0.

Variable m_{ij}^φ is defined as 0-1; if the connection line i and the connection line j are all in the range of φ and $t_s^i < t_e^i < t_e^j$, apply 1; otherwise, apply 0.

Variable n_{ij}^φ is defined as 0-1; if the connection line i and the connection line j are all in the range of φ and $t_s^j < t_e^j < t_e^i$, apply 1; otherwise, apply 0.

Define 0-1 variable θ_{ki}^φ ; if the variable is 1, it means that within section φ , connection line i will select line position k^φ . If the variable is 0, it means that connection line i will not select line position k^φ . Then, s^h clearly satisfies the relative formula $s^h = \sum_k k^\varphi \cdot \theta_{kh}^\varphi$.

Define the function $y = f(x)$; when x is true, $y = 1$, and when x is false, $y = 0$. Table 1 summarizes the situation for all eight connection line crossover points.

3.2. Confirming the Line Positions. The premise for optimizing the layout of locomotive working diagrams is to confirm the line positions in each section, that is, to determine the

largest k^φ in the set K^φ . This can be done following the steps set out as follows.

Step 1. Let $\varphi = 0$, $\theta_2 = 0$.

Step 2. Sort all t_s^h and t_e^h within section φ in ascending order by time, forming sequence $\{T_i\}$; when the time is the same, in addition to $t_s^{h_1} = t_e^{h_2}$, $t_s^{h_1}$ should be ranked first, and in the other situations, the order can be determined randomly.

Step 3. Let $\theta_1 = 0$, scanning time sequence $\{T_i\}$ sequentially; if t_s^h is scanned, $\theta_1 = \theta_1 + 1$, $\theta_2 = \max\{\theta_1, \theta_2\}$. If t_e^h is scanned, $\theta_1 = \theta_1 - 1$, $\theta_2 = \max\{\theta_1, \theta_2\}$.

Step 4. $\varphi = \varphi + 1$; if the value of φ is larger than the number of the line positions on the locomotive working diagram, go to step 5. Otherwise, go to step 2.

Step 5. Construct the number set K^φ of the line positions in the range of φ , in which $\max k^\varphi = \theta_2$, $k^\varphi \in K^\varphi$.

At this point, the line positions of each drawing area have been determined, and that means the set K^φ is known.

3.3. Model Construction and Solution

3.3.1. Constraint Conditions. ① One connection line can only select one line position:

$$\sum_k \theta_{kh}^\varphi = 1, \quad \forall k^\varphi \in K^\varphi, h^\varphi \in H^\varphi \quad (1)$$

② Connection lines cannot overlap

$$p_{ij} \left(\theta_{ki}^\varphi + \theta_{ky}^\varphi \right) \leq 1, \quad \forall k^\varphi \in K^\varphi, h^\varphi \in H^\varphi \quad (2)$$

③ When $\varphi = 2$, if two locomotives are engaged in parallel operations at both ends of a station, this is reflected in the locomotive working diagram; that is, the LL or RL of a certain connection line and the LL or RL of another connection line are at the same time, and it is obvious that the two vertical lines, which are at the same time, cannot overlap in the diagram.

Parameter u_{ij}^φ is defined as 0-1; if $i \neq j$ and $t_s^i = t_s^j$, apply 1; otherwise, apply 0. Define 0-1 parameter v_{ij}^φ ; if $i \neq j$ and $t_e^i = t_e^j$, apply 1; otherwise, apply 0. Parameter r_{ij}^φ is defined as 0-1; if $i \neq j$ and $t_e^i = t_s^j$, apply 1; otherwise, apply 0. There are four cases shown in Table 2.

TABLE I: Summary of connection lines and analysis of each type of connection lines crossover point.

Example diagram	Constraint condition
(a)	When $\varphi = 0$ or $\varphi = 2$: If $d^i = 1, d^j = 1, p_{ij}^{\varphi} = 1, m_{ij}^{\varphi} = 1$ and $f(s^i > s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{1ij}^{\varphi} = d^i \cdot d^j \cdot p_{ij}^{\varphi} \cdot m_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then the i and j have a crossover point.
(b)	When $\varphi = 2$: If $c^i = 1, c^j = 1, p_{ij}^{\varphi} = 1$ and $n_{ij}^{\varphi} = 1$, then i and j have a crossover point; i.e., if $\omega_{2ij}^{\varphi} = c^i \cdot c^j \cdot p_{ij}^{\varphi} \cdot n_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then i and j have a crossover point;
(c)	When $\varphi = 2$: If $b^i = 1, b^j = 1, p_{ij}^{\varphi} = 1, m_{ij}^{\varphi} = 1$ and $f(s^i < s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{3ij}^{\varphi} = b^i \cdot b^j \cdot p_{ij}^{\varphi} \cdot m_{ij}^{\varphi} \cdot f(s^i < s^j) = 1$, then i and j have a crossover point. When $\varphi = 1$: If $b^i = 1, b^j = 1, p_{ij}^{\varphi} = 1, m_{ij}^{\varphi} = 1$ and $f(s^i > s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{3ij}^{\varphi} = b^i \cdot b^j \cdot p_{ij}^{\varphi} \cdot m_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then i and j have a crossover point.
(d)	When $\varphi = 2$: If $d^i = 1, d^j = 1, p_{ij}^{\varphi} = 1, n_{ij}^{\varphi} = 1$ and $f(s^i < s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{4ij}^{\varphi} = d^i \cdot d^j \cdot p_{ij}^{\varphi} \cdot n_{ij}^{\varphi} \cdot f(s^i < s^j) = 1$, then i and j have a crossover point. When $\varphi = 1$: If $d^i = 1, d^j = 1, p_{ij}^{\varphi} = 1, n_{ij}^{\varphi} = 1$ and $f(s^i > s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{4ij}^{\varphi} = d^i \cdot d^j \cdot p_{ij}^{\varphi} \cdot n_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then i and j have a crossover point.
(e)	When $\varphi = 2$: If $d^i = 1, b^j = 1, p_{ij}^{\varphi} = 1, m_{ij}^{\varphi} = 1$ and $f(s^i > s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{5ij}^{\varphi} = d^i \cdot b^j \cdot p_{ij}^{\varphi} \cdot m_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then i and j have a crossover point.
(f)	When $\varphi = 2$: If $b^i = 1, d^j = 1, p_{ij}^{\varphi} = 1, m_{ij}^{\varphi} = 1$ and $f(s^i < s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{6ij}^{\varphi} = b^i \cdot d^j \cdot p_{ij}^{\varphi} \cdot m_{ij}^{\varphi} \cdot f(s^i < s^j) = 1$, then i and j have a crossover point.
(g)	When $\varphi = 2$: If $c^i = 1, d^j = 1, p_{ij}^{\varphi} = 1, n_{ij}^{\varphi} = 1$ and $f(s^i > s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{7ij}^{\varphi} = c^i \cdot d^j \cdot p_{ij}^{\varphi} \cdot n_{ij}^{\varphi} \cdot f(s^i > s^j) = 1$, then i and j have a crossover point.
(h)	When $\varphi = 2$: If $d^i = 1, c^j = 1, p_{ij}^{\varphi} = 1, n_{ij}^{\varphi} = 1$ and $f(s^i < s^j) = 1$, then i and j have a crossover point; i.e., if $\omega_{8ij}^{\varphi} = d^i \cdot c^j \cdot p_{ij}^{\varphi} \cdot n_{ij}^{\varphi} \cdot f(s^i < s^j) = 1$, then i and j have a crossover point.

TABLE 2: Schematic representation of overlapping LL and RL in locomotive working diagram.

	Example diagram	Constraint condition
(a)		If $b^i = 1, a^j = 1$ and $u_{ij}^\varphi = 1$, then i and j subject to $s^i - s^j \geq 1$; i.e., $b^i \cdot a^j \cdot u_{ij}^\varphi \cdot (s^i - s^j) \geq b^i \cdot a^j \cdot u_{ij}^\varphi$.
(b)		If $d^i = 1, a^j = 1$ and $r_{ij}^\varphi = 1$, then i and j subject to $s^i - s^j \geq 1$; i.e., $d^i \cdot a^j \cdot r_{ij}^\varphi \cdot (s^i - s^j) \geq d^i \cdot a^j \cdot r_{ij}^\varphi$.
(c)		If $c^i = 1, b^j = 1$ and $r_{ij}^\varphi = 1$, then i and j subject to $s^j - s^i \geq 1$; i.e., $c^i \cdot b^j \cdot r_{ij}^\varphi \cdot (s^j - s^i) \geq c^i \cdot b^j \cdot r_{ij}^\varphi$.
(d)		If $d^i = 1, c^j = 1$ and $v_{ij}^\varphi = 1$, then i and j subject to $s^i - s^j \geq 1$; i.e., $d^i \cdot c^j \cdot v_{ij}^\varphi \cdot (s^i - s^j) \geq d^i \cdot c^j \cdot v_{ij}^\varphi$.

3.3.2. *Objective Function.* ① Minimum number of connection line crossover points:

$$\min Z_1 = \sum_{\varphi} \sum_{\substack{i^\varphi \in H^\varphi \\ j^\varphi \in H^\varphi}} (\omega_{1ij}^\varphi + \omega_{2ij}^\varphi + \omega_{3ij}^\varphi + \omega_{4ij}^\varphi + \omega_{5ij}^\varphi + \omega_{6ij}^\varphi + \omega_{7ij}^\varphi + \omega_{8ij}^\varphi) \quad (3)$$

② To improve the visual aspect of locomotive working diagrams, connection lines should occupy the smaller delineable line positions as far as possible:

$$\min Z_2 = \sum_{\varphi} \sum_{h^\varphi \in H^\varphi} s_h^\varphi \quad (4)$$

3.4. *Model Solution.* The model constructed above in this paper is a nonlinear pure-integer programming model with multiple objects. These two objective functions are incommensurable, which means they have different units and cannot be solved by linear weighting methods effectively. However, these two objective functions can be solved using a delaminating sequence method in order of priority: First, solve the single-object optimized model applying formula (3) as the objective function, and formulae (1) and (2) as constraint conditions. Then, under the conditions of the above calculation results, solve the single-object optimized model, with formula (4) as the objective function, and formulae (1) and (2) as constraint conditions.

It is very difficult to obtain these solutions using this method with of-the-shelf commercially available software (such as Lingo), and even if this kind of software could produce an optimal solution, the computation time required

to reach the solution would be too long to be practically viable (take the 4th section of this paper for example; the layout of the connection lines in the middle section was generated by Lingo, with Intel Core i3 CPU and 4G memory, and it still took 4 hours to reach the optimal solution). To this end, the genetic algorithm is used to solve the model in this paper. Although it is impossible to guarantee that the output will be the optimal solution, it is effective enough for practical purposes. The algorithm is described as follows.

3.4.1. *Chromosome Encoding and Initialization.* Chromosome encoding uses digital string coding. In the drawing section, each chromosome represents one combination of all the selected line positions selected by connection lines. The number of chromosome genes is equal to the number of connection lines in the drawing area. The genes of each chromosome are sorted according to the given sequence of the connection line numbers, and each gene corresponds to the drawing area of the corresponding connecting line. For example, the chromosome “2, 1, 2” indicates that there are 3 connection lines, and the corresponding solution is $\theta_{2,1}^\varphi = 1$, $\theta_{1,2}^\varphi = 1$, $\theta_{2,3}^\varphi = 1$. It means that the connection line numbered 1 corresponds to the line position 2, the connection line numbered 2 corresponds to the line position 1, and the connection line numbered 3 corresponds to the line position 2.

The integer *pop_size* is defined as the number of chromosomes in the population; at the beginning of the genetic algorithm, the chromosomes need to be initialized.

Step 1. Randomly generate an unmarked integer h_x^φ between 1 and $h_{\max}^\varphi$.

Step 2. According to the smallest distribution method, draw the connection line h_x^p to the smallest line position label that does not conflict with other connection lines. And the labeling value of the line position was filled in the corresponding position of the chromosome.

Step 3. Mark the integer h_x^p .

Step 4. Verify if all integers in 1-100 are marked or not. If so, turn to Step 5; otherwise, turn to Step 1.

Step 5. The obtained chromosomes are tested and placed into the initial population if the constraints of Section 2 model construction are satisfied. Otherwise, restart the first step to construct a new chromosome. When the initial population reaches the size of a chromosome, the initialization of the population is completed.

When the initial population reaches *pop_size* chromosomes, the population initialization stage is complete.

3.4.2. Calculation of Chromosome Fitness. First, calculate the value Z defined by the following formula for each chromosome, which reflects the pros and cons of the two objective functions of a certain chromosome:

$$Z = Z_1 + \frac{Z_2}{M} \quad (5)$$

In the formula, M is a sufficiently large real number, set at 10,000 in this paper.

Then the chromosomes are sorted according to their value (ascending) to obtain the chromosomal sequence $V_1, V_2, \dots, V_{pop_size}$. Then the order-based evaluation function is defined:

$$eval(V_i) = 0.5^i \quad i = 1, 2, \dots, pop_size \quad (6)$$

In the formula, $i = 1$ means the chromosome is best, and $i = pop_size$ means worst.

3.4.3. Chromosome Selection Process. The chromosome selection process is based on its fitness; the greater the fitness of a chromosome, the greater the probability that it will be selected. The selection process steps are as follows.

Step 1. For each chromosome V_i , calculate the cumulative probability q_i :

$$q_0 = 0$$

$$q_i = \sum_{j=1}^i eval(V_j), \quad i = 1, 2, \dots, pop_size \quad (7)$$

Step 2. Generate a random number r from the interval $(0, q_{pop_size}]$.

Step 3. If $q_{i-1} < r \leq q_i$, select chromosome V_i as No. i .

Step 4. Repeat step 2 and step 3 for a total of *pop_size* times to obtain *pop_size* replicated chromosomes.

FIGURE 5: Chromosome crossover operation.

FIGURE 6: Chromosome mutation.

3.4.4. Chromosome Crossover Operation. Define P_c as the probability of a crossover operation, indicating an average of $P_c \cdot pop_size$ chromosome crossover operations in each population. The crossover operation uses a partial matching crossover method, so the chromosomes after crossover must be viable. Since chromosome genes are sorted according to the number of connection lines, crossover operations are performed as follows.

Step 1. Generate a random number ε_1 between $[0, 1]$ from $i = 1$ to *pop_size* repeatedly; if $\varepsilon_1 < P_c$, choose V_i as a parent.

Step 2. Pair the parents $V'_1, V'_2, V'_3 \dots$ generated in Step 1 into groups randomly.

Step 3. Generate a random integer ε_2 between $[1, h_{max}^p]$ randomly, then select a group of parent chromosomes that have not been selected in Step 2, and swap all the genes of these two chromosomes from ε_2 to the end, as shown in the Figure 5.

If the generation of descendants is viable, they can replace the parent. Otherwise, keep the viable subject and regenerate a new random number and repeat the crossover operation until two viable descendants or a given number of cycles are used up. Only the viable descendant can replace the parent in the chromosome crossover operation.

3.4.5. Chromosome Mutation Operation. Define P_m as the probability of mutation operation, which indicates that there is an average of $P_m \cdot pop_size$ chromosomes in each population mutating. Selecting the parent of a chromosome mutation is similar to parent selection in the crossover operation, and mutation is performed as follows.

Step 1. Generate two unequal random integers ε_3 and ε_4 between $[1, h_{max}^p]$.

Step 2. Exchange the gene at the position ε_3 with the gene at the position ε_4 in the chromosome, as shown in Figure 6.

If the chromosome obtained after mutation is viable, the original chromosome will be replaced. Otherwise, repeat Step

FIGURE 7: Locomotive working diagram generated using the smallest distribution method.

FIGURE 8: The locomotive operations diagram drawn by the method of this paper.

1 and Step 2 until a viable chromosome mutation is achieved or the given number of cycles is used up.

4. Examples

Figure 7 shows a locomotive working diagram corresponding to a certain train operations diagram, which uses the smallest distribution method to draw the connection lines, and there is an overlap of two RLs caused by parallel operations in an intermediate station. As a comparison, the author used the method in this paper to optimize the layout of the locomotive working diagram shown in Figure 7.

When the size of the population is 30, the probability of crossover (P_c) is 0.3, the probability of mutation (P_m) is 0.2,

and the number of random models is 1000, and the system calculates three traceable areas separately on a computer which has Intel Core i3 CPU and 4G memory. The middle area takes the longest time, about 1 minute, to get the best chromosome in this area (6, 5, 4, 1, 3, 2, 2, 6, 3, 5, 2, 1, 3, 1, 3, 4, 1, 5), and the target value of this chromosome is 24.0057, which means that the number of crossover points between connection lines in the optimal plan in the middle section is 24. According to the three chromosomes obtained, the locomotive turnover diagram as shown in Figure 8 can be obtained.

In the locomotive working diagram generated using the method in this paper, the number of crossover points between the connection lines is 42, 5 less than the number of

crossover points in Figure 7 which was drawn manually. The displacement per connection line is 9mm, 2mm more than in Figure 7 (the height of the middle section in the original diagram was 63mm). With regard to the visualization of these examples, the locomotive working diagram generated using the method presented in this paper is clearer and easier to understand and is more relevant and more legible for the purposes of front line staff.

5. Conclusions

Up until now, there had been little research into the visual representation of locomotive working diagrams. This paper proposes an improved and more systematic planning method to solve some of the problems encountered with computer-generated plans, such as lack of capacity or overlapping of connection lines in locomotive working diagrams, and it also dramatically improves the legibility of locomotive working diagrams. In addition, by adjusting the model slightly, the proposed method could also be applied for other purposes, such as deadhead or light travel, indicating that this method can be very versatile.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61603140).

References

- [1] L. Nie, Y. Zhang, and X. Wu, "Key technologies for computer generation of cyclic train timetable. China Railway Science," in *Wu X. Key technologies for computer generation of cyclic train timetable. China Railway Science*, vol. 35, pp. 114–121, 114–121, 35(1), 2014.
- [2] Q. Y. Peng, M. L. Yang, and S. Q. Ni, "A system of making train working graph on single-track lines with computer," *Journal of Southwest Jiaotong University*, vol. 30, no. 5, pp. 537–542, 1995.
- [3] L. S. Zhou, S. J. Hu, J. Ma, and Y. X. Yue, "Network hierarchy parallel algorithm of automatic train scheduling," *Journal of The China Railway Society*, vol. 20, no. 5, pp. 15–21, 1998.
- [4] F. D. He, X. Y. Liang, and D. Y. He, "Self-adaptive genetic algorithm for locomotive diagram," *Journal of Southwest Jiaotong University*, vol. 41, no. 3, pp. 273–278, 2006.
- [5] H. F. Yan and Y. Cui, "Optimization model of locomotive working diagram," *China Railway Science*, vol. 27, no. 4, pp. 123–128, 2006.
- [6] R. K. Ahuja, J. Liu, J. B. Orlin, D. Sharma, and L. A. Shughart, "Solving real-life locomotive-scheduling problems," *Transportation Science*, vol. 39, no. 4, pp. 503–517, 2005.
- [7] B. Vaidyanathan, R. K. Ahuja, and J. B. Orlin, "The locomotive routing problem," *Transportation Science*, vol. 42, no. 4, pp. 492–507, 2008.
- [8] B. Vaidyanathan, R. K. Ahuja, J. Liu, and L. A. Shughart, "Real-life locomotive planning: new formulations and computational results," *Transportation Research Part B: Methodological*, vol. 42, no. 2, pp. 147–168, 2008.
- [9] J. M. P. Boaler, "The solution of a railway locomotive scheduling problem," *Journal of the Operational Research Society*, vol. 31, no. 10, pp. 943–948, 1980.
- [10] M. A. Forbes, J. N. Holt, and A. M. Watts, "Exact solution of locomotive scheduling problems," *Journal of the Operational Research Society*, vol. 42, no. 10, pp. 825–831, 1991.
- [11] K. Ghoseiri and S. F. Ghannadpour, "A hybrid genetic algorithm for multi-depot homogenous locomotive assignment with time windows," *Applied Soft Computing*, vol. 10, no. 1, pp. 53–65, 2010.
- [12] S. M. Nourbakhsh and Y. Ouyang, "Optimal fueling strategies for locomotive fleets in railroad networks," *Transportation Research Part B: Methodological*, vol. 44, no. 8–9, pp. 1104–1114, 2010.
- [13] T. Raviv and M. Kaspi, "The locomotive fleet fueling problem," *Operations Research Letters*, vol. 40, no. 1, pp. 39–45, 2012.
- [14] F. Piu and M. G. Speranza, "The locomotive assignment problem: a survey on optimization models," *International Transactions in Operational Research*, vol. 21, no. 3, pp. 327–352, 2014.
- [15] X. Xu, C. Li, and Z. Xu, "Integrated train timetabling and locomotive assignment," *Transportation Research Part B: Methodological*, vol. 117, pp. 573–593, 2018.
- [16] X. Xu, K. Li, and X. Lu, "Simultaneous locomotive assignment and train scheduling on a single-track railway line: a simulation-based optimization approach," *Computers & Industrial Engineering*, vol. 127, no. 1, pp. 1336–1351, 2019.
- [17] X. Q. Cheng and Y. Pu, "Automation for offset plot of train flow lines," *Journal of Southwest Jiaotong University*, vol. 43, no. 6, pp. 692–696, 2008.
- [18] X. Q. Cheng, Y. Pu, Y. X. Lu, and C. Z. Yin, "Automatic plot algorithm of railway train flow diagram," *China Railway Science*, vol. 28, no. 1, pp. 113–117, 2007.
- [19] F. Shi, P. Ren, J. Qin, Y. Chen, and W. L. Zhou, "Optimization methods for arranging train flow in rail network," *China Railway Science*, vol. 27, no. 2, pp. 120–125, 2006.