Research Article

Modeling and Algorithms of the Crew Rostering Problem with Given Cycle on High-Speed Railway Lines

Zhiqiang Tian and Huimin Niu

School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China

Correspondence should be addressed to Huimin Niu, hmniu@mail.lzjtu.cn

This paper studies the modeling and algorithms of crew roster problem with given cycle on high-speed railway lines. Two feasible compilation strategies for work out the crew rostering plan are discussed, and then an integrated compilation method is proposed in this paper to obtain a plan with relatively higher regularity in execution and lower crew members arranged. The process of plan making is divided into two subproblems which are decomposition of crew legs and adjustment of nonmaximum crew roster scheme. The decomposition subproblem is transformed to finding a Hamilton chain with the best objective function in network which was solved by an improved ant colony algorithm, whereas the adjustment of nonmaximum crew rostering scheme is finally presented as a set covering problem and solved by a two-stage algorithm. The effectiveness of the proposed models and algorithms are testified by a numerical example.

1. Introduction

Crew planning problem is known as crew scheduling and rostering in the transportation market-airlines, railways, mass transit, and buses. The common features for all these applications are: (i) both temporal and spatial features are involved, that is, each task is characterized by its starting time and location and its finishing time and location and (ii) all tasks to be performed by employees are determined according to a given timetable. Tasks are the smallest elements (or building blocks) and are obtained from the decomposition of flight, train or bus journey, or a trip between two or more consecutive stops in a bus line.

In recent decades, the research and applications of airline crew scheduling and rostering received extensive attention because of its enormous economic benefits and impact, and more articles have been devoted to methodologies and applications in this area than

to any other rostering application areas. The most popular approach for solving airline crew scheduling and rostering is the well-known decomposition technique, in which the overall problem is solved in three major stages: (i) crew pairing generation, (ii) crew pairing optimization, and (iii) crew rostering. Crew pairing generation is a process of constructing all or a large number of feasible pairings/duties from the given timetable. Some of the pairings generated in the first stage will be selected so that all flight legs/trips are covered at a minimum cost. In the last stage, the pairings selected in the second stage are sequenced into rosters that will be assigned to individual crew. For these three stages, the first two are usually called as crew scheduling problem, while the last one is called as crew rostering problem.

Crew rostering problem is combinatorial according to its features. The aim of crew rostering problem is to assign individual crew members to the planned rotations once the set of rotations has been determined (a rotation is understood to be a sequence of crew legs/trips on consecutive days made by a crew that leaves from and returns to the same base). The crew rostering problem is most often treated as a zero-one integer programming problem with a defined objective function and specific set of constraints. Buhr [1] proposed minimizing the discrepancy between the average monthly flight time per crew member and the monthly flight time of individual crew members. When the scale of the problem is large, it is often not possible to find an optimal solution, so the "day-by-day" heuristic algorithm [2, 3] and the "crew-by-crew" heuristic algorithm [4–6] are widely used for generating an initial (or satisfactory) solution. The main drawback of "day-by-day" algorithm is that it usually leads to no available crew members on following days, while the drawback of "crew-by-crew" method very often leads to uneven workloads for the crew members. Furthermore, other techniques including branch-and-bound [7], genetic algorithm [8], and simulated annealing [9] are all used to find the optimal solution.

Ryan [10], Gamache, and Soumis [11] modeled the crew rostering problem as a generalized set partitioning problem. Ryan [10] used linear relaxation and the branch-and-bound method. Gamache and Soumis [11] solved the generalized set partitioning problem by column generation, and the numerical tests on problem from Air France show that the algorithm was capable of solving very large-scale problems.

As discussed above, many articles deal with airline crew planning problem, but the railways applications of crew scheduling and rostering have appeared only recently in the public transport sector literature. Since the development of high-speed railway lines is so fast, the need for efficient methods for generating computerized crew roster schemes is growing because the complexity involved in this issue is too much to be managed manually.

The remainder of this paper is organized as follows: the overall statement of the problem is first presented in Section 2 where the crew planning problem is divided into two subproblems. The formulations of the two subproblems are given in Sections 3 and 4, respectively, while a numerical example with generated data is represented in Section 5.

2. Problem Statement

Crew rostering plan with given cycle is a specific form in which the crew routes (or crew legs) are formed into crew rosters under the premise that the plan cycle T, the work time T_w, and the rest time T_r are all given (where T equals the sum of T_w and T_r). Then the crew routes are assigned to crew members as a continuous work plan. The crew members arranged in crew rostering plan is usually set as the primary objective since it matters greatly in the operation costs.

The determination of the given cycle of crew rostering plan is mainly dependent on the operating characteristics and the working regulations of crew members of high speed railway lines, which should also guarantee that the working time of crew members is in accordance with the working standard. Because the crew routes (or crew legs) got by the phase of crew scheduling plan generally have different working time, the crew roster schemes consisted by crew routes (or crew legs) usually have variance in working time. In the long run, the unbalance of working time between crew members can be relieved by letting crew members work underdifferent crew roster schemes in several cycles. However, from the viewpoint of plan-making, for all crew roster schemes, minimizing the variance of working time is helpful for keeping the workload balance of all crew members.

In the process of working out the crew scheduling plan, crew legs are further combined into crew routes. However, when it comes to the crew rostering plan, both crew routes and crew legs can be used as the fundamental data. In contrast, using crew legs to make up the crew rostering plan is more flexible, and actually includes the scenarios of making plan with the input of crew routes. Therefore, this paper only focuses on the implementation method of crew rostering plan with the input of crew legs. For the crew rostering plan with given cycles, we have two specific forms.

(i) Dividing all crew legs into several subsets which could form a feasible crew roster scheme subject to all constraints, and then the crew roster scheme is assigned to T crew members (or T crew member groups) to accomplish sequentially.

Crew rostering plan with this pattern has strong regularity, which is easy to be worked out and helpful for increasing the possibility to manage the crew members by groups. However, when applying crew rostering plan with this pattern, the crew roster scheme executed sequentially by T crew members may sometime use more crew members than the lower bound we really need. In order to explain this scenario, a term named as maximum crew roster scheme is defined at the first stage.

Maximum Crew Roster Scheme

For a feasible crew roster scheme, if the total working time exceeds the working time standard T_w while connecting any other crew leg, this scheme will be regarded as a maximum crew roster scheme, and then the crew members needed to accomplish this scheme are equal to the cycle of the crew rostering plan (T).

In crew rostering plan worked out by the first mode, the number of crew members used just equals kT (where k is the number of crew roster schemes) if all crew roster schemes are maximum crew roster scheme. If there is a nonmaximum crew roster scheme, some crews will be unnecessarily used if we still assign this scheme to T crew members.

(ii) Determining crew roster scheme for each crew member directly in crew rostering plan. In this circumstance, the matching problem between crew members and crew legs instead of crew routes is directly considered, which is beneficial to reduce the crew member arranged in crew rostering plan.

A feasible crew rostering plan worked out by this method is shown in Table 1, where the cycle of the plan is 9 days, of 7 days for work and 2 days for rest, 3 crew legs are accomplished by 4 crew members.

Table 1: A crew rostering plan with given cycle of 9 days.

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9
Crew member 1	1	2	3	1	2	3	1	Rest	Rest
Crew member 2	Rest	Rest	1	2	3	1	2	3	1
Crew member 3	3	1	Rest	Rest	1	2	3	1	2
Crew member 4	2	3	2	3	Rest	Rest	Rest	2	3

Table 1 shows that the crew roster schemes for each crew member are not exactly the same. While the scale of the problem is large, the compilation and implementation of the plan will be relatively complicated.

In order to work out the crew rostering plan more easily and more convenient, an implementation strategy by integrating the advantages of the two methods discussed above is put forward. In this strategy, the first method is adapted to work out a crew rostering plan which has maximum crew roster schemes as much as possible. In summary, the implementation method of crew rostering plan with given cycle studied in this paper contains two steps: the first one realizes the decomposition of all crew legs, while the second one gives further adjustment to the nonmaximum crew roster scheme which may be obtained by the first step. For convenience, the two steps above are called as decomposition of crew legs and adjustment of nonmaximum crew roster scheme, and then the optimization model and algorithm for the two subproblems are studied.

3. Model and Algorithm of Decomposition of Crew Legs

3.1. Description

The decomposition problem divides all crew legs into several subsets, and each set could form a feasible crew roster scheme. Because the number of crew members used affects directly the total cost of daily operation, the first target of this problem is usually to minimize the crew members arranged. Meanwhile, for the working time in each crew roster scheme could be different, and consequently, the working load between all crew roster schemes should also be considered when dividing the crew legs. As mentioned above, the crew members needed to accomplish a maximum crew roster scheme is exactly equal to the cycle of crew rostering plan. But for a nonmaximum crew roster scheme with m crew legs, if a crew member could only undertake one crew leg in one day, then the lower bound of crew members needed to accomplish the nonmaximum crew roster scheme is $[(m \times T)/T_w]$.

The decomposition problem is essentially a set partitioning problem, which could be described by the representation according to feasible schemes. While generate feasible crew roster schemes, the following regulations and constraints should be considered.

   (i) One crew member could undertake at most one crew leg in one day.

  (ii) Within the given cycle, the accumulated working time of crew members should not exceed the working time standard T_w.

 (iii) The connection time between two adjacent crew legs should satisfy the corresponding standard.

 (iv) Within the given cycle, each crew member used should at least assign a continuous rest T_r.

Through the above analysis, the optimization objectives put forward in this paper for crew legs partitioning problem are as follows.

(i) Minimizing the number of feasible crew roster schemes, which means maximize the number of maximum crew roster schemes at the same time.

(ii) Minimizing the number of crew legs in the nonmaximum crew roster scheme if there is one in the crew rostering plan.

(iii) Realizing the most equalized working time distribution of all maximum crew roster schemes.

3.2. Crew Legs Partitioning Model

$$\min \sum_{i=1}^{m} x_i, \tag{3.1}$$

$$\min \sum_{i=1}^{m} x_i (1 - \sigma_i) k_i, \tag{3.2}$$

$$\min \sum_{i=1}^{m} x_i \sigma_i \left(t_i - \frac{\sum_{i=1}^{m} x_i t_i}{\sum_{i=1}^{m} x_i} \right)^{\alpha}, \tag{3.3}$$

$$\text{subject to} \quad \sum_{i=1}^{m} x_i a_{ij} = 1, \quad j = 1, 2, \ldots, n, \tag{3.4}$$

$$x_i \in \{0, 1\}, \quad i = 1, 2, \ldots, m, \tag{3.5}$$

where m is the total number of feasible crew roster schemes, x_i is a 0-1 decision variable equals 1 if feasible crew roster scheme i is chosen in the final solution, and 0 otherwise, σ_i is a 0-1 variable equals 1 if feasible crew roster scheme i is a maximum crew roster scheme, and 0 otherwise, t_i is the total working time of all crew legs in feasible crew roster scheme i, k_i is the number of crew legs contained in feasible crew roster scheme i, α is the control parameter for calculate the proportionality of crew roster schemes, a_{ij} is a 0-1 variable that equals 1 if feasible crew roster scheme i contains crew leg j, and 0 otherwise, n is the total number of crew legs.

Objective functions (3.1)–(3.3) correspond to the optimization objectives in Section 3.1, respectively. Constraint (3.4) represents that each crew leg belongs to one crew roster scheme; constraint (3.5) is the 0-1 restriction of decision variable.

3.3. Ant Colony Algorithm with Bipheromone and Biheuristic Information

The model founded above is a set partitioning model with multiobjective, for which finding a feasible solution is NP-hard. In order to solve the model easily, the problem is converted into a network optimization problem by treating the crew legs as nodes, and treating the connections between crew legs as arcs, then the new problem needed to be solved is to find a Hamilton chain, which covered all crew legs. The Hamilton chain is composed by some subchains, each of which represents a feasible crew roster scheme. The issue of finding

Hamilton chain in a network could be executed by algorithms based on construction strategy. Ant colony algorithm is a kind of bionic algorithm simulating the mode of finding route in the real world by ants, which has already performed well when applied in problems that can be solved by construction strategy [5, 12]. For this reason, the ant colony algorithm is chosen to solve the converted problem. The critical operations of ant colony algorithm which are different from the common use are as follows.

(i) *The Construction of the Solution*

Ant k starts from the crew leg with the earliest ending time and continueus the construction process by adding new crew legs one by one. If the working time of the sequence of crew legs does not exceed the working standard, then the next crew leg will be chosen by the probability P_{ij}^k, otherwise, it should make clear a new feasible crew route scheme has already been found, after recording the scheme and indexes, setting the value of variable that record the working time of crew leg sequence to 0, and now, the ant chooses a crew leg as the start node for a new crew roster scheme by the probability P_i^k. While all crew legs have been selected, the key issue is to determine whether the last crew roster scheme is a maximum crew roster scheme or not, calculate all indexes, and finish the construction process of ant k.

(ii) *Representation of Phenomena*

The ant colony algorithm in this paper has bi-phenomena representation methods that both record the phenomena of all arcs τ_{ij} and all nodes τ_i in the solution construction graph. τ_{ij} is the degree of the expectation for crew leg i connecting crew leg j, while τ_i is the expectation for selecting crew leg i as the start node for the next crew roster scheme.

(iii) *Selection Strategy*

During the selection process, the probability of choosing next crew leg for ant k located at node i is related with its current state.

When the node i where ant k is located is not an "origin" node, the formula of probability for selecting the next crew leg is as follows:

$$P_{ij}^k = \begin{cases} \dfrac{[\tau_{ij}]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{ij}]^\alpha [\eta_{ij}]^\beta}, & \text{if } j \in N_i^k, \\ 0, & \text{otherwise,} \end{cases} \tag{3.6}$$

where η_{ij} is the heuristic information calculated by the following formula:

$$\eta_{ij} = \frac{1}{t_{ij}} \times \frac{1}{|((T_i + t_j)/m) - t^*| + 1}, \tag{3.7}$$

where T_i is the accumulated working time of all crew legs in current sequence, t_{ij} is the connection time between crew leg i and crew leg j, t_j is the working time of crew leg j, m is the number of all crew legs after connecting crew leg j, and t^* is the average working time of all crew legs.

When the node i where ant k is located is an "origin" node, the formula of probability for selecting the next crew leg as the start node of a new crew roster scheme is as follows:

$$P_i^k = \begin{cases} \dfrac{[\tau_i]^\alpha [\eta_i]^\beta}{\sum_{\text{Selected}_i \neq 1} [\tau_i]^\alpha [\eta_i]^\beta}, & \text{if Selected}_i \neq 1, \\ 0, & \text{otherwise.} \end{cases} \tag{3.8}$$

The heuristic information η_j is calculated by the following formula, in which t_j^e is the end time of crew leg j:

$$\eta_j = \frac{1}{t_j^e}. \tag{3.9}$$

(iv) *Evaluation Function*

It is inconvenient to evaluate the solutions constructed by the ants since the model contains three optimization objectives. Thus, these objectives are considered in one formula by weight when establishing the evaluation function

$$\min\left\{ \alpha \sum_{i=1}^m x_i + \beta \frac{\sum_{i=1}^m x_i(1-\sigma_i)k_i}{T_w} + \gamma \frac{\sum_{i=1}^m x_i\sigma_i\left(t_i - \left(\sum_{i=1}^m x_i t_i / \sum_{i=1}^m x_i\right)\right)^2}{\left(t_{\max} - \left(\sum_{i=1}^m x_i t_i / \sum_{i=1}^m x_i\right)\right)^2 \sum_{i=1}^m x_i\sigma_i} \right\}, \tag{3.10}$$

where α, β, γ, are the weigh factors of each optimization objective, respectively, $t_{\max}$ is T_w times of the working time of crew leg which has longest working time. It is obvious that the first part of formula (3.10) is an integer and the values of other parts are within the interval [0,1). Since the number of crew member arranged is the objective that most concerned by the management organization, and the importance of other two objectives reduce, gradually, the values of α, β, γ, are set to 1, 1, and 0.1, and the value of each part belongs to [0,m], [0,1), and [0,0.1), respectively.

4. Model and Algorithm of Adjustment of Nonmaximum Crew Roster Scheme

4.1. Description

The essence of the adjustment of nonmaximum crew roster scheme is to work out a new crew rostering plan with the crew legs contained by the nonmaximum crew roster scheme. At this time, the second strategy for the implementation of crew rostering plan is used and its main objectives are still minimizing the crew members arranged and equalizing the working time distribution of all crew roster schemes. While adjusting the nonmaximum crew roster scheme, the constraints in Section 3.1 should be satisfied. Moreover, any crew leg must be assigned to one and only one crew member.

4.2. Multiobjective Model

The optimization model founded for the second subproblem in this paper is as follows

$$\min M, \tag{4.1}$$

$$\min \quad \sum_{i=1}^{M} \left(\sum_{k=1}^{T} \sum_{j=1}^{N} x_{ij}^{k} t_{j} - \frac{T \sum_{j=1}^{N} t_{j}}{M} \right)^{2}, \tag{4.2}$$

$$\text{subject to} \quad \sum_{i=1}^{M} x_{ij}^{k} = 1, \quad j = 1, 2, \ldots, N; \; k = 1, 2, \ldots, T, \tag{4.3}$$

$$\sum_{j=1}^{N} x_{ij}^{k} \le 1, \quad i = 1, 2, \ldots, M; \; k = 1, 2, \ldots, T, \tag{4.4}$$

$$\sum_{k=1}^{T} \sum_{j=1}^{N} x_{ij}^{k} \le T_{w}, \quad i = 1, 2, \ldots, M, \tag{4.5}$$

$$\sum_{j=1}^{N} x_{ij}^{f(k+1,T)} \left(t_{j}^{s} + 1440 \right) - \sum_{j=1}^{N} x_{ij}^{k} t_{j}^{e} - T_{\text{con}} + \theta \left(2 - \sum_{j=1}^{N} x_{ij}^{f(k+1,T)} - \sum_{j=1}^{N} x_{ij}^{k} \right) \ge 0, \tag{4.6}$$

$$i = 1, 2, \ldots, M; \quad k = 1, 2, \ldots, T$$

$$\sum_{k=1}^{T} \prod_{r=0}^{T_{r}-1} \left(1 - \sum_{j=1}^{N} x_{ij}^{f(k+r,T)} \right) \ge 1, \quad i = 1, 2, \ldots, M, \tag{4.7}$$

$$x_{ij}^{k} \in \{0, 1\}, \; i = 1, 2, \ldots, M; \quad j = 1, 2, \ldots, N; \; k = 1, 2, \ldots, T, \tag{4.8}$$

where M is the crew members needed to finish all the task; N is the total number of crew legs in nonmaximum crew roster scheme; x_{ij}^{k} is a 0-1 decision variable that equates 1 if crew member i undertakes crew leg j on day k, and 0 otherwise; T is the given cycle of crew rostering plan; T_{w} is the working time while T_{r} is the continuous rest time for crew members in cycle T; T_{con} is the standard for the connection of crew legs; t_{j} is the work time of crew leg j; t_{j}^{s} is the start time while t_{j}^{e} is the end time of crew leg j; θ is an infinite positive number.

The feasible crew roster schemes in crew rostering plan are all loops consisting of crew legs, so the connection time between each two adjacent crew legs should satisfy the standard. Hence, the following function is designed to reflect this characteristic:

$$f(k, T) = \begin{cases} k\%T, & \text{if } k < T, \\ T, & \text{otherwise.} \end{cases} \tag{4.9}$$

Objective function (4.1) is designed to minimize the crew members arranged in the crew rostering plan, while objective function (4.2) means the most equalized working load among all crew roster schemes. Constraints (4.3)–(4.7) correspond to the restrictions in Section 3.1, respectively. The $\theta(2 - \sum_{j=1}^{N} x_{ij}^{f(k+1,T)} - \sum_{j=1}^{N} x_{ij}^{k})$ in constraint (4.6) guarantees that

the formula is tenable while one of the adjacent jobs is not the crew leg. Constraint (4.8) is the restriction of 0-1 decision variable.

4.3. Two-Phase Algorithm

It is hard to solve the model founded above by traditional optimization algorithms because it includes both linear constraints and constraints based on the description. Therefore, this paper designs an algorithm with two phases. The first phase represents the converted problem by set covering model, and then the initial feasible crew rostering plan with minimal crew members arranged is obtained. On the basis of the result of the first phase, an algorithm with the ideology of simulated annealing is used in the second phase to implement further improvement until a better crew rostering plan is obtained.

Phase 1. Initial crew rostering plan with minimal crew members arranged. The adjustment of nonmaximum crew roster scheme is converted into a standard set covering problem by representing the feasible crew roster scheme with 0-1 string, and then all feasible crew roster schemes are found. Finally, the initial crew rostering plan with minimal crew members is obtained by solving the set covering model. The generation of crew roster scheme, the conversion of feasible scheme, and other issues are as follows.

(i) The Generation and Conversion of Feasible Crew Roster Scheme

The crew roster schemes obtained by the decomposition of crew legs contains at least one nonmaximum crew roster scheme, in which the crew legs contained must be less than T_w. Therefore, all the feasible crew roster schemes can be easily found by a deep search method. The generated crew route schemes are represented by the real index of crew legs, thus a 0-1 string with the length of M is used to represent the crew leg for one day in the scheme. The crew leg i is undertaken by crew members when the ith element of the string equals 1, and the crew member is assigned a rest while all elements in the string equal 0. Therefore, a 0-1 string with the length of MT can be used to represent the situation of all the schemes with T days. Thanks to the conversion from feasible crew roster schemes, there does not exist the situation that more than 2 elements in the substring equal 1 in one day.

(ii) The Generation of the Set of All Feasible Crew Roster Schemes

Because the execution process of crew roster scheme is a circulation, the feasible crew roster scheme got by the deep search method can just represent a specific situation in the cycle. In order to generate all feasible crew roster schemes, a duplication process is used to copy the schemes found in step (i). Each crew roster scheme can be duplicated T schemes for the cycle of the scheme equals to T. For a crew rostering plan with the cycle of 4 days, in which 3 days are working and 1 day is resting. Since a feasible crew roster scheme is 2-1-3, the schemes duplicated are shown in Table 2.

All feasible crew roster schemes could be obtained after the duplication to the searched schemes, but some of them are exactly the same ones (e.g., a scheme 1010 will become two 1010 and two 0101 after the duplicate). In order to keep the uniqueness of all schemes, a procedure that checks the repetition and deletes the repeated schemes is then applied. After the above operation, the problem of adjusting nonmaximum crew roster scheme becomes

Table 2: Duplication of feasible crew roster scheme.

Schemes	Before conversion	After conversion
Initial scheme	2-1-3-0	0100-1000-0010-0000
Duplicated scheme 1	1-3-0-2	1000-0010-0000-0100
Duplicated scheme 2	3-0-2-1	0010-0000-0100-1000
Duplicated scheme 3	0-2-1-3	0000-0100-1000-0010

a standard set covering problem with the objective of covering all crew legs in the plan cycle with minimal number of crew roster schemes

$$\min \sum_{u=1}^{U} y_u$$

$$\text{subject to} \quad \sum_{u=1}^{U} y_u a_{uv} \geq 1, \quad v = 1, 2, \ldots, MT,$$

(4.10)

where y_u is a 0-1 decision variable that equals 1 if crew roster scheme u is chosen in final solution and 0 otherwise, a_{uv} is a 0-1 variable that represents whether crew roster scheme u covers crew leg v.

(iii) *The Generation of Initial Crew Tostering Plan*

The set covering problem above can be solved by various mature algorithms, such as genetic algorithm, ant colony algorithm, and algorithms based on the application of column generation principle, and interested readers are referred to the related references [13–15].

By solving the set covering model, an initial crew rostering plan with minimal crew members arranged can be obtained accompanied by the scenario where some of the crew legs may be covered more than once. We assume that crew leg i is covered k_i times in the solution of set covering problem, then the initial feasible crew rostering plan of this problem will be obtained by deleting k_i-1 extra covers of crew leg i randomly. Using the opposite process of the duplication while all crew legs appear only one time during the solution, then, the crew rostering plan based on the original description is obtained with minimal crew members.

Phase 2. Adjustment for equalized workload based on the ideology of simulated annealing.

When designing the algorithm for most equalized workload of crew roster schemes, two concepts are defined firstly.

Feasible Swap

In the kth day of crew rostering plan, while the crew roster schemes obtained by swapping the jobs (work or rest) of crew member i and j still satisfy all the restrictions, the operation adopted is called as a feasible swap.

Positive Swap

A feasible swap is called a positive swap while the equilibrium of workload has improved after the adoption of this swap (the variation of equilibrium Δ_i is a positive number). On the contrary, a feasible swap, which reduces the equilibrium, is a negative swap.

According to the definitions, the essence of the adjustment of nonmaximum crew roster scheme is to improve the equilibrium of workload by finding and adopting positive swaps, until a crew rostering plan with high equilibrium is found. For providing a relative lager searching space, ideology of simulated annealing is merged in the designed algorithm, which means that the selection of negative swaps is allowed in the operation.

The details of the algorithm are as follows.

Step 1. Let the initial temperature t equal t_0, the start day k for searching feasible swap equals 1, calculate the working time T_i of each crew roster scheme in initial crew rostering plan, the average working time T^* of all crew roster schemes, and the equilibrium of initial plan.

Step 2. Search all feasible swaps from the first crew roster scheme at day k.

Step 3. Let $k = k + 1$, if $k > T$, which means that all feasible swaps has already been found and turn to the next step, otherwise, turn to Step 2.

Step 4. Calculate the variation Δ_i of each feasible swap, and then calculate the probability P_i for each swap to be selected.

Step 5. Select a feasible swap by probability P_i, the probability for accepting the swap selected equals 1 when the variation Δ_i is a positive number, and equals P_{accept} when the variation Δ_i is a negative number. Record the swap used and update the equilibrium of crew rostering plan, and then turn to next step. This step is repeated once the selected swap is refused because the probability P_i is lower than P_{accept}.

Step 6. Implement the update of temperature by the update function and determine whether the updated temperature reaches the lowest point $t_{\min}$. Turn to next step if the temperature equals $t_{\min}$, otherwise, let k equals 1 and return to Step 2.

Step 7. Once the adjusting process of the equilibrium of all crew schemes is finished, the best crew rostering plan is outputted.

The parameters in the simulated annealing algorithm above are similar to the common usage except the selection probability of feasible swaps which is described here only. Assume that there are n feasible swaps at one status in Step 5 of the algorithm above, then the probability P_i for choosing feasible swap i is represented as follows:

$$P_i = \frac{\Delta_i - \Delta_{\min} + \varepsilon}{\sum_{i=1}^{n}(\Delta_i - \Delta_{\min} + \varepsilon)}, \tag{4.11}$$

where $\Delta_{\min}$ is the minimal variation of workload equilibrium of all feasible swaps, which is usually a negative number. Δ_i is the variation of workload equilibrium of feasible swap i. ε is a small positive number used to ensure that the probability of the feasible swap with minimal variation does not equal zero.

Table 3: Details of generated crew legs.

Crew legs	Start time	End time	Working time
1	868	1240	372
2	708	1109	401
3	652	1116	464
4	829	1195	366
5	984	1360	376
6	775	1146	371
7	701	1179	478
8	999	1432	433
9	812	1239	427
10	450	826	376
11	665	1064	399
12	569	1020	451
13	414	817	403
14	931	1393	462
15	543	978	435
16	784	1153	369
17	604	1035	431
18	491	897	406
19	852	1275	423
20	392	859	467
21	730	1152	422
22	971	1354	383
23	512	949	437
24	574	996	422

In the algorithm designed above, the probability for the acceptance of negative swap is relatively high when at a high temperature and reduced gradually with the decreasing of temperature. When the temperature reaches the lowest level, this probability tends to zero and the algorithm converges finally.

5. Numerical Example

According to the operational characteristics of China high-speed railway currently, this paper generates several crew legs as the basic data to work out the crew rostering plan with given cycle. Assume that the earliest starting time and the latest ending time of all crew legs are 6:00 and 24:00, respectively, which will be 360 and 1440 when represented by minute. The working time of each crew leg equals the time span between start time and end time. The connection time standard between two crew legs is 960 min, and all crew legs start and end at the same crew base. The details of the 24 generated crew legs are shown in Table 3, in which the start time, end time, and working time of all crew legs are represented by minute.

First, with the data generated above, this paper works out a crew rostering plan with the cycle of 8 days, in which 6 days are for working and 2 days for rest, the results are shown in Table 4.

Table 4: Crew rostering plan with the cycle of 8 days.

Roster schemes	Days						Crew members arranged	Total crew member arranged	Lower bound of crew members	Operation time(s)
	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6				
Crew roster scheme 1	13	20	10	18	23	15	8			
Crew roster scheme 2	24	12	17	3	11	7	8			
Crew roster scheme 3	2	21	6	16	9	4	8	32	32	3.5172
Crew roster scheme 4	1	19	14	22	5	8	8			

Table 5: Crew rostering plan with the cycle of 7 days.

Roster schemes	Days					Crew members arranged	Total crew member arranged	Lower bound of crew members	Operation time(s)
	Day 1	Day 2	Day 3	Day 4	Day 5				
Crew roster scheme 1	13	20	10	18	23	7			
Crew roster scheme 2	15	12	24	17	3	7			
Crew roster scheme 3	11	7	2	21	6	7	34	34	2.4618
Crew roster scheme 4	16	9	4	19	1	7			
Crew roster scheme 5	22	14	5	8		6			

It is easily obtained from Table 4 that the 4 crew roster schemes in the crew rostering plan are all maximum crew roster schemes which means each scheme contains 6 crew legs that equals the working time criterion. Under this circumstance, the adjustment procedure is not needed and the number of crew members arranged in the crew rostering plan equals the product of the plan cycle and the total number of crew schemes, which has already been minimal and the plan can be executed with high regularity.

Moreover, the crew legs above are used to work out a crew rostering plan with the cycle of 7 days in which 5 days for work and 2 days for rest, the outcome obtained by the decomposition of crew legs is shown in Table 5.

It can be concluded from Table 5 that the previous 4 crew roster schemes are all maximum schemes, and the crew members used in each scheme are equal exactly to the cycle of the crew rostering plan. The 5th crew roster scheme (see the last row in Table 5) is a nonmaximum crew roster scheme. If still using 7 crew members to undertake this scheme at this time, one or more crew members will be not required actually. Actually, the lower bound of crew members needed for the last crew roster scheme is $[(4 \times 7)/5] = 6$. Therefore, the second phase is needed to adjust the 5th crew roster scheme. The initial crew rostering plan is worked out with the crew legs in the 5th crew roster scheme by the algorithm designed in Section 4.3. The initial crew rostering plan of adjustment of nonmaximum crew roster scheme is shown in Table 6, in which the number 0 represents no crew leg is assigned to this crew member on this day.

Table 6: Initial crew rostering plan of adjustment of non-maximum crew roster scheme.

Crew	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Accumulated working time (min)	Proportionality (min^2)
1	22	14	8	22	8	0	0	2094	
2	0	0	22	14	14	8	22	2123	
3	5	22	0	0	22	14	8	2037	
4	14	5	14	5	0	0	5	2052	529799.33
5	0	8	5	8	5	22	0	2001	
6	8	0	0	0	0	5	14	1271	

Table 7: Optimal crew rostering plan of adjustment of nonmaximum crew roster scheme.

Crew	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Accumulated working time (min)	Proportionality (min^2)
1	14	14	14	14	0	0	0	1848	
2	0	0	22	5	22	8	22	1958	
3	5	5	0	0	8	5	8	1994	
4	22	22	8	22	0	0	5	1958	29749.33
5	0	8	5	8	5	22	0	2001	
6	8	0	0	0	14	14	14	1819	

It is obvious that the crew member arranged to the 5th crew roster scheme is 6 which is equal to the lower bound of the crew member needed. However, the accumulated working time varies from 1271 min to 2123 min, and the proportionality is not optimal. In order to improve the proportionality, the algorithm based on the ideology of simulated annealing is used and the improved crew rostering plan is shown in Table 7.

After the adjustment of the initial crew rostering plan, the variations of accumulated working time among all crew roster schemes are relatively small, and the proportionality of the plan is much better than before.

6. Conclusion

This paper studies the compilation method of crew rostering plan with given cycle of high-speed railway lines. By merging two different compilation principles, the problem is divided into two subproblems which are the crew legs partitioning problem and the adjustment of nonmaximum crew roster scheme. The division of the problem gives consideration to both of the complexity for compilation and the convenience for execution, and the algorithm designed is effective for finding an optimal solution. By studying this problem, we found that under the assumption that each crew member undertakes at most one crew leg in a day, the lower bounds of crew members needed are $\lfloor m/T_w \rfloor \times T$ and $\lceil (m \times T)/T_w \rceil$ when work out the crew rostering plan with the two forms discussed in the beginning separately, respectively. While compiling the crew rostering plan by merging the two strategies, which is used in this paper, the lower bound is $\lfloor m/T_w \rfloor \times T + \lceil (m - \lfloor m/T_w \rfloor) \times T/T_w \rceil$. When m/T_w is an integer, the value of all lower bound are equal no matter what compiling pattern we use, and the minimal crew members arranged may be still equal. The compiling method of crew rostering plan put forward in this paper has its own advantages, but unexpectedly, sacrifices both objectives to

some extent. The algorithms applied in this paper can also be used separately when working out the crew rostering plan with single requirement. When compile crew rostering plan by set covering model, it should be aware that the scale of the problem will affect both the difficulty of the algorithm design and the execution efficiency.

Acknowledgment

References

[1] J. Buhr, "Four methods for monthly crew assignment-a comparison of efficiency," in *Proceedings of the AGIFORS Symposium*, vol. 18, pp. 403–430, 1978.

[2] B. Nicoletti, "Automatic crew rostering," *Transportation Science*, vol. 9, no. 1, pp. 33–42, 1975.

[3] D. Teodorović and P. Lučić, "A fuzzy set theory approach to the aircrew rostering problem," *Fuzzy Sets and Systems*, vol. 95, no. 3, pp. 261–271, 1998.

[4] R. Moore, J. Evans, and H. Noo, "Computerized tailored blocking," in *Proceedings of the AGIFORS Symposium*, vol. 18, pp. 343–361, 1978.

[5] J. Byrne, "A preferential bidding system for technical aircrew," *Proceedings of the AGIFORS Symposium*, vol. 28, pp. 87–99, 1988.

[6] P. Lučić and D. Teodorović, "Simulated annealing for the multi-objective aircrew rostering problem," *Transportation Research A*, vol. 33, no. 1, pp. 19–45, 1999.

[7] H. Dawid, J. König, and C. Strauss, "An enhanced rostering model for airline crews," *Computers and Operations Research*, vol. 28, no. 7, pp. 671–688, 2001.

[8] A. Monfroglio, "Hybrid genetic algorithms for a rostering problem," *Software: Practice and Experience*, vol. 26, no. 7, pp. 851–862, 1996.

[9] A. Ernst, M. Krishnamoorthy, and D. Dowling, "Train crew rostering using simulated annealing," in *Proceedings of the International Conference on Optimization: Techniques and Applications (ICOTA '98)*, Perth, Australia, 1998.

[10] D. M. Ryan, "The solution of massive generalized set partitioning problems in air crew rostering," *Journal of the Operational Research Society*, vol. 43, pp. 459–467, 1992.

[11] M. Gamache and F. Soumis, "A method for optimally solving the rostering problem," Cahier Du GERAD, G-90-40 H3T 1V6, Ecole des Hautes Etudes Commerciales, Montreal, Canada, 1993.

[12] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *BioSystems*, vol. 43, no. 2, pp. 73–81, 1997.

[13] F. D. Croce, R. Tadei, and G. Volta, "A genetic algorithm for the job shop problem," *Computers and Operations Research*, vol. 22, no. 1, pp. 15–24, 1995.

[14] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 26, no. 1, pp. 29–41, 1996.

[15] M. Desrochers and F. Soumis, "A column generation approach to the urban transit crew scheduling problem," *Transportation Science*, vol. 23, no. 1, pp. 1–14, 1989.

Submit your manuscripts at
http://www.hindawi.com

Advances in
Operations Research

Advances in
Decision Sciences

Mathematical Problems
in Engineering

Algebra

Journal of
Probability and Statistics

The Scientific
World Journal

International Journal of
Combinatorics

International Journal of
Differential Equations

Advances in
Mathematical Physics

Journal of
Complex Analysis

Journal of
Mathematics

International Journal of
Stochastic Analysis

Abstract and
Applied Analysis

Discrete Dynamics in
Nature and Society

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Applied Mathematics

Journal of
Function Spaces

Journal of
Optimization