Two-Stage Optimal Trajectory Planning Based on Resilience Adjustment Model for Virtually Coupled Trains

Song, Hongyu; ShangGuan, Wei; Qiu, Weizhi; Sheng, Zhao; Harrod, Steven S.

Published in:
IEEE Transactions on Intelligent Transportation Systems

Link to article, DOI:
10.1109/TITS.2023.3307904

Publication date:
2023

Document Version
Peer reviewed version

Citation (APA):
Song, H., ShangGuan, W., Qiu, W., Sheng, Z., & Harrod, S. S. (Accepted/In press). Two-Stage Optimal Trajectory Planning Based on Resilience Adjustment Model for Virtually Coupled Trains. IEEE Transactions on Intelligent Transportation Systems. https://doi.org/10.1109/TITS.2023.3307904

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
You may not further distribute the material or use it for any profit-making activity or commercial gain
You may freely distribute the URL identifying the publication in the public portal.

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.
Two-Stage Optimal Trajectory Planning Based on Resilience Adjustment Model for Virtually Coupled Trains

Hongyu Song, Wei ShangGuan, Weizhi Qiu, Zhao Sheng, and Steven S. Harrod

Abstract—Virtual coupling is proposed as an innovative solution to meet the growing transport demand and to further improve the service quality of railways. Nevertheless, obtaining the optimal driving strategy that enhances its transport capacity and energy efficiency remains a challenging task. In order to achieve these objectives, this paper presents a novel convoy optimization method that optimizes the recommended trajectories for virtually coupled trains. A resilience adjustment model is firstly proposed to evaluate the coupling process and generate candidate trajectories according to the adjustment rules. In addition, the convoy optimization problem is formulated as a two-stage programming model consisting of a multi-objective programming stage and a least-cost goal programming stage, which determine the optimal trajectories for trains. Taking into account the requirements of practical applications, the solution method is finally designed to solve the proposed model and achieve dynamic updates of the recommended trajectories throughout train operations. Based on the field data from the Wuhan-Guangzhou high-speed railway line, numerical experiments are conducted to validate the effectiveness of the proposed method. The experimental results indicate that the proposed method shows the best performance in terms of infrastructure utilization and energy consumption, and the spacing between virtually coupled trains is well maintained regardless of ideal or disturbing conditions.

Index Terms—High-speed train, trajectory planning, virtual coupling, resilience adjustment, multi-objective optimization.

I. INTRODUCTION

HIGH-SPEED railway have grown dramatically worldwide, and have shown themselves to be competitive with both road transport and short distance flights. Shift2rail (EU) projects that railway transport demand in Europe is expected to increase 50% by the year 2050 [1]. However, the capacity of rail networks in densely populated areas is nearing saturation, primarily measured in terms of the count of train paths on the railway line. An obvious solution is to construct more railway lines, but modern construction is both expensive and difficult in existing developed areas. Construction costs may reach over US$ 10 million per kilometer. And as the HS2 high speed railway project in the United Kingdom has shown, even when a railway project is generally desired by the public, it can meet great opposition at the local level. Therefore, if a way can be found to increase the capacity on existing infrastructure, there is a significant financial incentive to explore this alternative. One alternative is to compress the trains closer together on the railway line, thus providing more train paths per hour on the same infrastructure.

The traditional train control system defines fixed spacings between adjacent trains, called “fixed block”. These spacings are marked as fixed points on the railroad tracks. The philosophy of fixed block system is applied in the European system ETCS level 2 and the Chinese system CTCS level 3. However, these blocks must be designed to accept all trains, and trains in reality differ in size and performance. Consequently, traditional railway systems must make a compromise that focuses on the most shared characteristics of the train traffic.

An improvement on fixed block is “moving block”, which is implemented in the European level 3 and Chinese level 4 signal systems. In these systems, the spacing is not defined by the track infrastructure, but is defined by dynamic calculations onboard the train. It can then be customized for each train, allowing some trains to have a shorter spacing, thus compressing more trains into the same infrastructure. These spacings are most commonly calculated according to the absolute braking distance from the front of one train to the rear of the preceding train. However, the spacing increases dramatically with speed, reaching nearly 6.5 km at 350km/h [2], and thus for high-speed railways still represents a significant consumption of space on the infrastructure.

What if there was absolute knowledge and control of train movements? Modern train-to-train (T2T) communication and...
control systems offer the possibility of coordination between trains. If trains could be coordinated, such that the following train could closely match its movements to the preceding train, then the spacing between trains could be reduced further. This is referred to as the “virtual coupling system” (VCS) [3]. VCS aims to separate trains by the relative braking distance, which is much shorter than the absolute braking distance. In addition, train convoys can be formed through the cooperative control of trains. These convoys can be viewed as one train by the train control system, thus further optimizing the scheduling and coordination of train traffic on the network.

Although the initial idea of virtual coupling was presented in 1999, it was not explored in more depth due to technological limitations [4], [5]. In 2006, Ständer et al. proposed the safety concept of train operations under VCS [6]. Henke et al. investigated the communication protocol and structure of VCS [7]. More recently, under the motivation of research projects [8], related research has further advanced in various aspects, such as the establishment of following strategies, derivation of minimum safe spacing, et.al.

In order to ensure security during convoy operations, Zhang et al. proposed a general modeling method that describes the protection logic for VCS based on topological manifolds [9]. Zhou et al. proposed a safety protection model and investigated the formulation of minimum safe spacing between trains under VCS [10]. Pan et al. analyzed the spacing of virtually coupled trains under eight typical train-following scenarios, including train departures and arrivals [11]. Quaglietta et al. defined the operational states of trains under VCS and developed a multi-state train-following model, which provides driving strategies for rear trains in a convoy [12]. Liu et al. designed an analytic optimal linear feedback that permits the rear train to follow its preceding train with desired distance and consistent speed [13]. Zhang et al. investigated the cooperative control problem for virtually coupling trains with fixed departure and coupling time [14]. Park et al. developed an algorithm that obtains the recommended trajectories for adjacent trains [15], where the proceeding train must reduce its speed to allow the rear train to catch up. Lang et al. developed a Deep Q-network algorithm to generate trajectories for virtually coupled trains, which takes into account four metrics related to safety and punctuality [16].

The concept of virtual coupling is similar to vehicle platooning in road traffic. Hence, some scholars have borrowed mature techniques from the automotive field to study VCS. Meo et al. investigated the portability of achievements obtained in vehicle platooning to the railway domain, including the mathematical models and control algorithms [17]. Referring to the ideas of vehicle platooning, Pan et al. developed the velocity-difference model that enables the synchronous control of virtually coupled trains [18]. Moreover, based on the architecture of model predictive control, some approaches [19], [20] were presented to maintain the minimum safe spacing for trains under VCS.

Most literature primarily focuses on controlling the spacing between adjacent trains to a constant or a value derived from relative braking mode. However, in daily operations, especially for high-speed trains, the actual spacing may fluctuate significantly due to various disturbances, which may result in unsafe situations, or inefficiencies due to wasted capacity. Therefore, a dynamic calculation is desirable to ensure safety and achieve the most efficient use of resources under all conditions.

Currently, scholars have tried to control the spacing within the desired range considering the uncertainty of the operating environment. Ketphat et al. investigated the modified minimum safe spacings for different situations (e.g., sudden deceleration of the preceding train) [21]. ShangGuan et al. introduced the resilience to describe the spacing and proposed an intelligent algorithm to maintain the spacing within a specific range [22]. Wang et al. presented a reinforcement learning-based train control approach and stabilized the spacing within the desired range [23]. Quaglietta et al. established a train following model based on a dynamic safety margin, which can ensure the safety of virtually coupled trains under various scenarios [24].

### B. Motivation and Contributions

Previous studies have devoted significant attention to control methods for maintaining the desired spacing between virtually coupled trains, and have verified the benefits of VCS for railway capacity [25]. However, in addition to infrastructure utilization, the energy consumption should also be considered as a primary objective for convoy optimization.

In this paper, a novel convoy optimization method is proposed to obtain recommended trajectories for virtually coupled trains, aiming to increase railway capacity and decrease energy consumption. Furthermore, the trajectory can be dynamically adjusted throughout operations, including train departure from the station and running in the section (called offline and online scenarios, respectively). The contributions of this paper to the literature are listed as follows.

- The resilience adjustment model is established to evaluate the coupling process of trains under VCS and to generate candidate trajectories for the proposed programming model. Furthermore, a dynamic region is introduced here for the convoy operation, rather than using the minimum safe spacing as the desired coupling spacing. This region can be adjusted based on the actual circumstances.
- An optimal trajectory planning framework is proposed in this paper, which illustrates the optimization mechanism for trains under both offline and online scenarios. In order to satisfy the needs of each scenario, the convoy optimization problem is formulated as the two-stage programming model consisting of a multi-objective programming stage and a least-cost goal programming stage.
- An implementation method based on two parallel loops is provided in this study. One loop searches for the optimal solution of our proposed model by an improved heuristic algorithm, and the other loop is designed to provide real-time recommendations for train operations. Consequently, recommended trajectories can be dynamically optimized while ensuring the operation safety.

The remainder of this paper is organized as follows. Section II provides an overview of the problem. Section III presents the formulations of train motion model, resilience adjustment model, and two-stage programming model. In Section IV, the method for solving and implementing the proposed model is introduced. Section V conducts the case study and discusses the results. Section VI concludes the paper and gives recommendations for future works.
The operational states of the following scenarios. Each scenario is composed of a leader and a follower. Therefore, planning the optimal trajectories for the followers is the key to optimizing the overall performance of VCS. Referring to [24], the operational states of the follower are classified into three categories:

- Coupling state: The follower attempts to catch up with the leader by accelerating and then establishing the coupling spacing.
- Coupled running state: The follower can transition to this state as soon as its speed approaches that of the leader and maintains the desired coupling spacing within tolerances. Hence, the condition for the coupled running state can be formulated as:
  \[
  0 \leq L_{\text{act}} - L_{\text{des}} \leq th_s \\
  |v_{\text{act}} - v_{\text{des}}| \leq th_v
  \]
  where \( i \) is the train number. Train \( i - 1 \) is the predecessor of train \( i \). \( k \) is the time instant. \( v \) is the train speed. \( L_{\text{act}} \) and \( L_{\text{des}} \) are the actual and desired coupling spacing. \( th_s \) and \( th_v \) are the spacing and speed tolerances, respectively.
- Decoupling state: The follower transitions to this state when separation from the convoy is required, such as in scenarios with excessively small spacing that poses a significant safety risk. Thus, the follower must decelerate to prevent the occurrence of hazardous events.

In the literature, the desired coupling spacing \( L_{\text{des}} \) is generally set as the minimum safe spacing \( L_{\text{safe}} \), which is calculated based on the relative braking mode [26]. As presented in Fig. 2, the actual spacing and minimum safe spacing can be expressed as:

\[
L_{\text{act}} = p_{i-1,k} - p_i,k - L_{\text{train}}(2) \\
L_{\text{safe}} = L_{\text{sb}} - L_{i-1,k} + L_{\text{rea}} + L_{\text{sm}}(3)
\]

where \( p \) and \( L_{\text{train}} \) are the position and train length. \( L_{\text{sb}} \) and \( L_{\text{sm}} \) are the service braking distance and emergency braking distance. \( L_{\text{rea}} \) is the traveling distance during the reaction time. \( L_{\text{sm}} \) is the safety margin.

Based on Equation 1 and the separation principle, \( L_{\text{act}} \) must be no less than \( L_{\text{sm}} \) after adjacent trains are stopped. \( L_{\text{sm}} \) is typically chosen as the constant, with a value range from tens of meters to kilometers [25]. A larger value results in more spacing, and more consumption of infrastructure capacity, while a smaller value results in operations closer to the safety limits.

In the real world, various disturbances can occur and affect the expected system state and output. Using the constant \( L_{\text{sm}} \) may result in unsafe situations or wasted capacity, especially in the application of high-speed railways. Therefore, this paper defines a dynamic region for convoy operations, which can be dynamically optimized according to the actual situation.

Furthermore, a recommended trajectory has to be obtained before train departure, which is regarded as the target for train control [27]. When planning trajectories for convoy operations, multiple objectives should be considered to improve the overall performance of the system. Taking capacity consumption and energy consumption as the optimization objectives, a convoy optimization method is proposed to generate and continuously update the recommended trajectories for the followers throughout train operations.

To balance modeling complexity and practical usability, it is assumed that T2T communication is reliable in VCS, enabling each follower to receive real-time updates on the leader’s state and planned trajectory.

III. MODEL FORMULATION

A. Train Motion Model

In this paper, the model used to update the state of virtually coupled trains is based on longitudinal train dynamics, with the train viewed as a point mass with a single degree of freedom. According to Newton’s second law, the train dynamics can be described by [28]:

\[
(1 + \gamma_i) M_i \cdot a_i,k = u_i,k - f_{i,k}^{\text{br}} - f_{i,k}^{\text{ar}}(4)
\]

where \( M \) and \( \gamma \) are the mass and rotary mass coefficient, \( a \) is the acceleration, \( u \) is the output force, \( f^{\text{br}} \) and \( f^{\text{ar}} \) are the basic and additional resistance.

\( f^{\text{br}} \) is related to the characteristics of the high-speed train and its real-time speed, which can be formulated as a quadratic function of speed called the Davis equation [29]:

\[
f_{i,k}^{\text{br}} = M_i \cdot (c_1 + c_2 \cdot v_{i,k} + c_3 \cdot v_{i,k}^2)(5)
\]

where \( c_1, c_2 \) and \( c_3 \) are nonnegative parameters that characterize the basic resistance to train motion, and their values are decided by the train type.

\( f^{\text{ar}} \) represents the additional resistance, which is influenced by the track conditions. In this study, it includes the gradient resistance and curve resistance:

\[
f_{i,k}^{\text{ar}} = M_i \cdot g \left( \theta_{i,k} + \frac{A}{d_{i,k}} \right)(6)
\]

where \( g \) is the gravity acceleration, \( \theta \) and \( d \) are the gradient and curve radius of the track, \( A \) is an empirical constant and equals 600 in this paper [30].
Based on the acceleration \( a_{i,k} \) derived by Equation 4, we can obtain the position and speed of train \( i \) by:

\[
\begin{align*}
    p_{i,k+1} &= p_{i,k} + v_{i,k} \Delta t + 0.5 a_{i,k} \Delta t^2 \\
    v_{i,k+1} &= v_{i,k} + a_{i,k} \Delta t
\end{align*}
\]  

(7)

where \( \Delta t \) is the sampling interval and is set to 1s in this paper.

According to the above train motion model, the train trajectory (formed by \( p \) and \( v \)) can be derived based on the specific value of \( u \). Therefore, we must determine the subsequent output forces in order to plan a train trajectory. Drawing from the research of [31], the optimal train trajectory is composed of four driving modes, i.e., maximum traction, cruising, coasting and maximum service braking, which are applied in this study to obtain the recommended trajectory. The output forces \( u \) in these four modes are represented as:

- \( U_1 \), the maximum tractive force.
- \( U_2 \), the output force during cruising.
- \( U_3 \), the output force during coasting.
- \( U_4 \), the maximum service braking force.

During cruising, the acceleration \( a_{i,k} \) equals zero. Thus, according to Equation 4, \( U_2 = f^{br} + f^{wr} \). During coasting, the traction/braking system will not be engaged, thus \( U_3 = 0 \). \( U_1 \) and \( U_4 \) are subjected to the characteristics of high-speed trains [32], which can be obtained based on the manufacturer’s specifications and actual operational data.

### B. Resilience Adjustment Model

The concept of resilience has been considerably studied and introduced to various domains, such as engineering, ecology, and economics [33]. Recently, resilience has also been utilized to study the performance of railroad systems when they are exposed to risk from disturbances [34], [35], [36]. The resilience can be described by resources required to restore performance level after disturbances [37]. Based on the evaluated resilience, adjustment strategies can be prepared in advance and selected as needed during the recovery process.

In order to maintain or restore the optimal performance level of high-speed trains under VCS, a resilience adjustment model is developed here. This model enables continuous evaluation of the coupling process and then obtains recommendations for trajectory adjustment.

In this study, the optimal coupling spacing \( L_{i,k}^{\text{opt}} \) is defined as the desired performance level for convoy operations:

\[
L_{i,k}^{\text{opt}} = (1+\xi_i)L_{i,k}^{\text{safe}}, \quad \xi_i \in (0, 1)
\]

(8)

where \( \xi \) denotes the safety coefficient.

For train \( i \) at instant \( k \), the resilience of the coupling process is formulated as the ratio of the optimal coupling spacing \( L_{i,k}^{\text{opt}} \) to the actual spacing \( L_{i,k}^{\text{act}} \):

\[
r_{i,k} = \frac{L_{i,k}^{\text{opt}}}{L_{i,k}^{\text{act}}}
\]

(9)

Therefore, the resilience \( r_{i,k} \) represents the extent of deviation from the optimal coupling spacing. Based on the value of resilience, the coupling process are divided into four phases:

\[
r_{i,k} \in \begin{cases} 
    (-\infty, (1+\varepsilon)^{-1}) & \rightarrow \text{Phase } I \\
    [(1+\varepsilon)^{-1}, 1] & \rightarrow \text{Phase } II \\
    (1, 1+\xi_i) & \rightarrow \text{Phase } III \\
    [1+\xi_i, +\infty) & \rightarrow \text{Phase } IV
\end{cases}
\]

(10)

where \( \varepsilon \) is the tolerance coefficient and satisfies \( \varepsilon \in (0, 1) \).

As the actual spacing decreases, the resilience increases and the coupling process sequentially transitions from Phase I to Phase IV. It is noted that, as defined in Equation 3, the minimum safe spacing is primarily influenced by the calculation result of \( L_{i,k}^{\text{safe}} \). If the follower runs much slower than the leader (e.g., the follower just departs from the station), the follower’s service braking distance will be much smaller than the leader’s emergency braking distance. In this case, the calculation results of the minimum safe spacing \( L_{i,k}^{\text{safe}} \), as well as the optimal coupling spacing \( L_{i,k}^{\text{opt}} \) and resilience \( r_{i,k} \), will be less than or equal to 0, which implies that the leader does not impose any restrictions on its follower during this period.

The adjustment model is proposed here for the purpose of maintaining the resilience \( r_{i,k} \) within Phase II or restoring it to Phase I. In addition, when potential risks arise, the defensive adjustment recommendations will be provided to maintain or restore the resilience at a low cost.

Rules of the resilience adjustment model are given in Table I, where \( M_1 \), \( M_2 \), \( M_3 \), and \( M_4 \) represent the driving modes of the leader, namely traction, cruising, coasting, and braking. Specific explanations of the adjustment model are as follows.

- **Phase I**. \( r_{i,k} \in (-\infty, (1+\varepsilon)^{-1}) \)

By default, the follower begins its operation from Phase I, e.g., departing from a station. By substituting Equation 10 into Equation 9, we can derive the relationship between \( L_{i,k}^{\text{act}} \) and \( L_{i,k}^{\text{opt}} \) in Phase I, specifically \( L_{i,k}^{\text{act}} < (1+\varepsilon)L_{i,k}^{\text{opt}} \). Since the actual spacing is much higher than the minimum safe spacing, the follower should drive faster to catch up with the leader.

If the leader is accelerating or cruising, the follower should accelerate to expeditiously close the spacing. Therefore, if the follower has not reached its maximum speed, it will operate at maximum acceleration with \( u_{i,k} = U_1 \). Otherwise, \( u_{i,k} = U_2 \).

If the leader is decelerating, the resilience may rise rapidly (since the follower is generally accelerating in this phase), and in extreme cases may even reach Phase III or Phase IV. Thus, it is necessary to take defensive measures to avoid these cases. The output force will be \( u_{i,k} = U_2 \).
Optimal solution. Fig. 3 shows the overview of this optimization and the one that fulfills our objectives will be selected as the then be evaluated based on the two-stage programming model, Recommendations will be provided to maintain the resilience at a low energy cost.

If the leader is not under braking in this phase, the follower will cruise with \( u_{i,k} = U_2 \).

Otherwise, referring to the consideration in Phase I, defensive measures will be recommended and the output force will be \( u_{i,k} = U_3 \).

In addition, the coupled running state can be achieved within Phase II if circumstances permit. Based on the resilience, the coupled running condition is converted to:

\[
\begin{align*}
    r_{i,k} &\in [(1 + \varepsilon_i)^{-1}, 1] \\
    |u_{i,k} - u_{i-1,k}| &\leq th_v
\end{align*}
\]  

(11)

• Phase III. \( r_{i,k} \in (1, 1 + \xi_i) \)

Phase III is a buffer zone between Phase II and Phase IV, where \( L_{i,k}^{safe} < L_{i,k}^{act} < L_{i,k}^{opt} \). In this phase, the actual spacing is relatively short but still in safety limits. Recommendations will be provided to restore the resilience to Phase II and prevent the occurrence of Phase IV.

If the leader is not braking, the follower should preventively decelerate in an attempt to reduce the resilience with minimum energy cost, thus adjusting the output force to \( u_{i,k} = U_3 \).

Otherwise, the follower applies brakes and transitions to the decoupling state to avoid risky situations. Thus, \( u_{i,k} = U_4 \) in this case.

• Phase IV. \( r_{i,k} \in [1 + \xi_i, +\infty) \)

Phase IV implies that the follower is exposed to unsafe situations, which must be avoided during train operation. Therefore, the follower must brake and decouple from the convoy regardless of the leader’s driving mode. The output force will be \( u_{i,k} = U_4 \). It should be noted that this phase conflicts with the constraints of the proposed programming model, which will be introduced in the next subsection. Thus, Phase IV will not be triggered using the recommended trajectory derived from the programming model.

In summary, the parameters \( \xi \) and \( \varepsilon \) will decide the regions of different phases, and further affect the results of the planned trajectory. Hence, \( \xi \) and \( \varepsilon \) are chosen as the decision variables in this paper.

C. Two-Stage Programming Model

As mentioned above, candidate trajectories can be generated based on different combinations of \( \xi \) and \( \varepsilon \). All candidates will then be evaluated based on the two-stage programming model, and the one that fulfills our objectives will be selected as the optimal solution. Fig. 3 shows the overview of this optimization process.

The two-stage programming model is designed for convoy optimization in both offline and online scenarios. “Offline” is best interpreted as the time period when the leader is at the beginning of its journey and the follower is still stopping at the station. “Online” is when the follower is departing or driving. In the offline scenario, the multi-objective programming model will be applied, whereas in the online scenario, the least-cost goal programming model will be applied. Since the scheduled departure headway will affect the operational performance, it is selected as a decision variable in the offline scenario. Hence, the decision variables in offline and online scenarios are:

\[
    z_i = \begin{cases} 
    (H_i, \xi_i, \varepsilon_i), & k \in [0, H_i) \\
    (\xi_i, \varepsilon_i), & k \in [H_i, +\infty) 
    \end{cases}
\]

(12)

where \( z \) is the vector of decision variables, \( H \) is the departure headway (the time interval between two trains which depart from the same station consecutively), ranging from 120 to 300 seconds [38].

As indicated in the bottom right part of Fig. 3, the optimal decision variables will be updated when the adjustment criteria are met. The definition of the adjustment criteria will be given and explained in Section IV-B.

Details of the optimization objectives, constraints, and programming models are described in the following.

1) Optimization Objectives and Constraints: In this paper, capacity consumption and energy consumption are taken as the optimization objectives.

Objective 1: Capacity consumption is recognized as one of the most critical metrics for convoy optimization. It refers to how much of the available infrastructure capacity an individual train requires and can be measured in many ways. In this study, the capacity consumption of train \( i \) is defined as:

\[
    \phi_i^{cc} = H_i + T_i
\]

(13)

where \( T \) denotes the driving time over the track segment being measured.
Objective 2: High-speed railways can offer increased mobility but also consume large amounts of energy. Minimizing the energy consumption can both reduce the environmental pollution and lower the transport cost. The energy consumption of train $i$ can be calculated by:

$$\phi_{i}^{ec} = \int_{0}^{T_{i}} u_{i}(t) \cdot v_{i}(t) dt$$  (14)

where $u(t)$ and $v(t)$ are the functions of the output force and speed in the continuous domain.

For simplicity, Equation 14 is discretized as follows:

$$\phi_{i}^{ec} = \sum_{k} (u_{i,k} \cdot v_{i,k} \cdot \Delta t)$$  (15)

Moreover, to normalize the formulation, $F_{n}$ is here defined as the set of optimization objectives:

$$F_{n} = \left\{ \begin{array}{ll}
\phi_{i}^{ec}, & n = 1 \\
\phi_{i}^{ec}, & n = 2
\end{array} \right.$$

(16)

The constraints of convoy operations can be summarized as the speed constraints, which are determined by the separation principle of VCS and the operating environment, respectively.

The maximum speed of train $i$ limited by the minimum safe spacing can be formulated as:

$$v_{i,k}^{safe} : \max \left( v_{i,k} \right) \left| L_{i,k}^{act} - L_{i,k}^{safe} \geq 0 \right.$$  (17)

Moreover, there are speed limits $v_{i,k}^{env}$ along the railway line. Therefore, the operating constraint on the maximum speed is:

$$v_{i,k}^{env} = \min \left\{ v_{i,k}^{safe}, v_{i,k}^{env} \right\}$$  (18)

2) Multi-Objective Programming for Offline Optimization:

The multi-objective programming stage is designed to provide decision-makers with a set of recommended trajectories. In this stage, both objectives are desired to be minimized. The programming model is defined as:

$$\min \mathbb{F} (z_{i}) = \{ F_{1} (z_{i}) , F_{2} (z_{i}) \}$$

s.t.

$$v_{i,max} - v_{i,x} \geq 0, \quad x = k, k + 1, \ldots, F_{1} (z_{i})$$

(19)

where $x$ is the set of sampling instants from the current time instant to the end of the journey, $v_{i,x}$ denotes the recommended speed of train $i$ at instant $x$, $v_{i,max}$ is the corresponding speed limit.

It should be noted that the objectives of capacity consumption and energy consumption can conflict with each other. A Pareto solution set will be obtained to evaluate this trade-off. Finally, the recommended trajectory can be chosen according to the specific transport demand.

3) Least-Cost Goal Programming for Online Optimization:

This stage will be activated after the follower departs the station. It is integrated to find a driving strategy that has the least cost compared with the recommended trajectory obtained from the multi-objective programming stage. The goal programming model [39] can take several goals into account simultaneously, prioritize among them and give specific weights to each goal.

For convoy optimization in the online scenario, deviations of $\phi_{h}$ and $\phi_{e}$ should be considered simultaneously. Thus, instead of defining the priority of each goal, the overall goal weight is here introduced to formulate the least-cost goal programming model. The programming model for this stage is:

$$\mathbb{G} (z_{k}) = \min \sum_{n=1}^{2} w_{n} (\alpha_{n} d_{n}^{+} + \lambda_{n} d_{n}^{-})$$

s.t.

$$v_{i,max} - v_{i,x} \geq 0, \quad x = k, k + 1, \ldots, F_{1} (z_{k})$$

(20)

with

$$d_{n}^{+} = \begin{cases} F_{n} (z_{i}) - \mathbb{F}_{n}, & F_{n} (z_{i}) \geq \mathbb{F}_{n} \\ 0, & F_{n} (z_{i}) < \mathbb{F}_{n} \end{cases}$$

$$d_{n}^{-} = \begin{cases} 0, & F_{n} (z_{i}) > \mathbb{F}_{n} \\ \mathbb{F}_{n} - F_{n} (z_{i}), & F_{n} (z_{i}) \leq \mathbb{F}_{n} \end{cases}$$

(21)

(22)

where $\mathbb{F}_{n}$ is the offline optimization result for $F_{n}$, $w_{n}$ is the overall goal weight, $d_{n}^{+}$ and $d_{n}^{-}$ are the positive and negative deviations between $F_{n} (z_{i})$ and $\mathbb{F}_{n}$, $\alpha_{n}$ and $\lambda_{n}$ are the weights of the corresponding deviations.

$w_{1}$ and $w_{2}$ are the goal weights of capacity consumption and energy consumption, respectively. To make a trade-off between these goals, $w_{1}$ and $w_{2}$ are both set to 0.5 in this paper.

$\alpha_{1}$ and $\lambda_{1}$ are the weights of positive and negative costs in capacity consumption, corresponding to $d_{1}^{+}$ and $d_{1}^{-}$. Although it is somewhat more acceptable for a follower to arrive ahead of the scheduled time ($d_{1}^{+} > 0$) than to be delayed ($d_{1}^{-} > 0$), both situations may lead to conflicts with other trains. Hence, our goal is to make $d_{1}^{+}$ and $d_{1}^{-}$ as small as possible, and $\alpha_{1}$ and $\lambda_{1}$ should satisfy:

$$\alpha_{1} > \lambda_{1} > 0$$

(23)

In addition, $\alpha_{2}$ and $\lambda_{2}$ stand for the weights of positive and negative costs in energy consumption. The candidate trajectory with lower energy consumption is more inclined to be selected, which presents a smaller $d_{2}^{+}$ or a larger $d_{2}^{-}$. Therefore, $\alpha_{2}$ and $\lambda_{2}$ are recommended to be:

$$\alpha_{2} > 0, \lambda_{2} < 0$$

(24)

In this paper, $\alpha_{1}$ and $\lambda_{1}$ are set to 0.9 and 0.1, $\alpha_{2}$ and $\lambda_{2}$ are set to 1.0 and -1.0. In practical applications, these parameters can be changed according to the specific transport demand.

IV. SOLUTION METHOD

A. DMPSO Algorithm

As illustrated in our proposed model, the decision variables are associated with the selection of output forces at different time instants, and consequently affect the optimization results. The optimal solution can be obtained by heuristic algorithms, which have been applied and proven to perform well in solving optimization problems in various fields [40]. In this paper, the dynamically modified particle swarm optimization (DMPSO) algorithm is designed to solve the proposed trajectory planning model.

The overview of DMPSO algorithm is shown in Algorithm 1, where $ng$ and $np$ denote the current iteration number and current particle number. $c_{ind}^{np}$ and $c_{inds}^{np}$ represent individual extremum and individual Pareto solution set, which
are obtained by comparing the \(np^{th}\) particle. \(o^{\text{glo}}\) and \(o^{\text{glos}}\) represent global extremum and global Pareto solution set, which are obtained by comparing all particles. In this algorithm, each particle \(o_{np}\) has a position vector \(p_{np}\), which has the same elements as \(z\) vector (Equation 12). Therefore, as presented in the algorithm, its dimension varies at different stages. During iterations, \(p_{np}\) is continuously modified to find the optimal solutions.

\begin{algorithm}
\caption{DMPSO Algorithm}
\begin{algorithmic}[1]
\State \textbf{Input:} maximum iterations \(N_g\) and maximum particle number \(N_p\)
\State Obtain the state of follower and leader
\If {train stops at the station}
\State Particle dimension \(D_p = 3\)
\Else
\State Particle dimension \(D_p = 2\)
\EndIf
\For {\(ng = 1\) to \(N_g\)}
\For {\(np = 1\) to \(N_p\)}
\If {\(ng = 1\)}
\State Initialize particle \(o_{np}^*\)
\Else
\State Obtain candidate particle \(o_{np}^*\)
\EndIf
\EndFor
\EndFor
\EndIf
\State Evaluate candidate trajectory based on Table 1
\State \textbf{end}
\EndFor
\State Obtain individual solution \(o_{np}^{\text{ind}}\)
\State Select global extremum \(o_{np}^{\text{glo}}\)
\For {\(np = 1\) to \(N_p\)}
\If {update this particle}
\State \(o_{np} = o_{np}^*\)
\EndIf
\EndFor
\If {\(D_p = 3\)}
\State Output global Pareto solution set \(o^{\text{glos}}\)
\Else
\State Output global extremum \(o^{\text{glo}}\)
\EndIf
\end{algorithmic}
\end{algorithm}

Key procedures of the adopted solution algorithm, including obtaining candidate particle (Step 12), selecting individual and global extremum (Steps 20 and 21), and updating particle (Step 24), are introduced as follows.

1) Rules of Obtaining Candidate Particle: The update rules for obtaining candidate particle \(o_{np}^*\) at each iteration can be described as:

\begin{equation}
 p_{np}^* = p_{np} + v_{np}^*
\end{equation}

with

\begin{equation}
 v_{np}^* = \varphi \left[ v_{np} + c_1 r_1 (p_{np}^{\text{ind}} - p_{np}) + c_2 r_2 (p_{np}^{\text{glo}} - p_{np}) \right]
\end{equation}

where \(p_{np}^{\text{ind}}, p_{np}^{\text{glo}}\) and \(p_{np}^{\text{glos}}\) are the position vectors of \(o_{np}^{\text{ind}}, o_{np}^{\text{glo}}\) and \(o_{np}^{\text{glos}}\). \(v_{np}\) are the velocity vectors of \(o_{np}\) and \(o_{np}^*\), which denote the position offsets of particles. \(v_{np}\) is initialized as a zero vector in the first iteration. \(c_1\) and \(c_2\) are the coefficients that adjust the search directions towards \(o_{np}^{\text{ind}}\) and \(o_{np}^{\text{glo}}\), respectively. \(r_1\) and \(r_2\) are the random numbers uniformly distributed between 0 and 1. The contraction factor \(\varphi\) is defined according to Clerc [41] and satisfies:

\begin{equation}
 \varphi = 2/(\sqrt{2} - c - \sqrt{c^2 - 4c})
\end{equation}

\begin{equation}
 c = c_1 + c_2 > 4
\end{equation}

2) Rules of Selecting Individual and Global Extremum: According to Equation 26, the individual and global extremum will affect the search direction of the particle. In this paper, the algorithm is designed to obtain Pareto solutions at the multi-objective programming stage and obtain an optimal solution at the least-cost goal programming stage. Therefore, the goals of updating the extremum vary at different stages. The selection mechanism is designed as follows.

For offline optimization at the multi-objective programming stage, more feasible trajectories should be generated. Hence, to maintain the diversity of the particle, solutions in \(o_{np}^{\text{ind}}\) and \(o_{np}^{\text{glos}}\) will be randomly selected as \(o_{np}^{\text{ind}}\) and \(o_{np}^{\text{glo}}\), respectively.

For online optimization at the least-cost goal programming stage, each particle is evaluated by Equation 20. The particle with the minimum cost will then be selected as the extremum.

3) Rules of Updating Particle: This step determines whether the candidate particle \(o_{np}^*\) can be delivered to the next iteration. To mitigate the convergence to local optima, a update mechanism inspired by the simulated annealing method [42] is utilized here. The detailed rules are as follows:

If \(o_{np}^*\) is not worse than \(o_{np}\), the particle must be updated and \(o_{np}\) will be replaced by \(o_{np}^*\). Otherwise, there is a certain probability to select \(o_{np}^*\). The probability of selecting \(o_{np}^*\) is designed to be lower than that of selecting \(o_{np}\) and decreases with iterations.

It should be noted that at the multi-objective programming stage, particles will be evaluated by the quantity of dominated solutions, and the particle with more dominated solutions will be considered as the better one. Hence, the update probability \(P\) is given by:

\begin{equation}
 P\left(o_{np}^*, o_{np}\right) = \min \left\{ 1, \exp \left[ \frac{\Delta E\left(o_{np}^*, o_{np}\right)}{\theta} \right] \right\}
\end{equation}

with

\begin{equation}
 \Delta E\left(o_{np}^*, o_{np}\right) = \begin{cases} 
 \frac{\left| o_{np}^* - o_{np} \right|}{T_{ng}}, & D_p = 3 \\
 \frac{\left[ \mathbb{G}(o_{np}) - \mathbb{G}(o_{np}^*) \right]}{T_{ng}}, & D_p = 2 
\end{cases}
\end{equation}

where \(|o_{np}|\) and \(|o_{np}^*|\) denote the number of solutions dominated by \(o_{np}\) and \(o_{np}^*\) compared to all feasible solutions. \(\mathbb{G}(o_{np})\) and \(\mathbb{G}(o_{np}^*)\) are the costs of \(o_{np}\) and \(o_{np}^*\) obtained by
To ensure that the initial temperature, $T_0$, is large enough, $T_0$ should be set to a large value, e.g., $T_0 = 100$. Furthermore, to ensure that $T_0$ can always be greater than 0, we can set $K = 0.1$. The coefficient $\beta$ is used to adjust the speed of cooling. $\beta$ is defined as a constant between 0 and 1. Taking $\beta = 0.9$ as an example, it can be observed that $K$ decreases with the increasing of $\beta$. Taking $\beta = 0.9$ as an example, $K$ decreases with the increasing of $\beta$. Taking $\beta = 0.9$ as an example, $K$ decreases with the increasing of $\beta$.

Assuming that $N_p = 50$ and $K = 0.1$, $T_0$ is obtained in Fig. 4. It should be noted that $P = 1$ when $|\omega^*_{np}| > |\omega_{np}|$. Therefore, this figure only presents the relationship between $K$ and $P$ when $|\omega^*_{np}| < |\omega_{np}|$. It can be observed that $P$ increases with the increasing of $K$. Taking $K = 0.1$ as an example, $P$ is maintained above 0.37. Although the high probability of selecting the worse solution enhances the search capability, it reduces the efficiency of obtaining the optimal solution. On the contrary, larger values of $K$ will result in lower probabilities, even in early iterations. Therefore, it is recommended to set $K$ to 5 when $N_p = 50$. Similarly, based on this consideration, $T_0$ and $\beta$ are set to 4 and 0.5.

B. Implementation Procedure

In the previous sections, we have introduced the resilience adjustment model, the two-stage programming model, and the DMPSO algorithm. This subsection proposes an effective approach for integrating these models, enabling the dynamic generation of recommended trajectories for practical applications. The designed trajectory planning procedure is presented by the sequence diagram in Fig. 5, which includes two loops running in parallel:

LOOP 1: This loop is designed to update the optimal decision variables for LOOP 2, thus maintaining the performance level of the generated trajectory. In this figure, the blue rectangle denotes the calculation process of the DMPSO algorithm. The heights of the blue rectangles represent its computational costs at multi-objective programming stage and least-cost goal programming stage, respectively denoted by $T_{L1}^{MP}$ and $T_{L1}^{LP}$. The multi-objective programming stage is triggered first when the leader departs the station.

LOOP 2: This loop can generate the recommended trajectory based on the real-time states of adjacent trains, thus ensuring the operation safety. Its calculation process is represented by the yellow rectangle. This loop is initially activated after the follower departs the station and periodically runs at intervals of $\Delta T_{L2}$ (default 1 second). The computational cost of LOOP 2 is represented by $T_{L2}$. In each cycle, the computational task involves 2 steps:

- Step 1: calculate and update the recommended trajectory based on the latest decision variables and real-time states of adjacent trains;
- Step 2: update the adjustment criteria, and send them to LOOP 1 if any of the criteria is satisfied and LOOP 1 is not searching for the optimal solution.

The adjustment criteria are defined as:

$$\begin{align*}
\text{Criterion 1:} & \quad T_{\text{up}} \geq \Delta T_{\text{max}} \\
\text{Criterion 2:} & \quad u \neq u'
\end{align*}$$

where $T_{\text{up}}$ represents the accumulated running time since the last update of decision variables. It starts counting at the first cycle of LOOP 2 and is reset to 0 when decision variables are updated by LOOP 1. $\Delta T_{\text{max}}$ is a constant and equals 120s in this paper. $u'$ is the output force according to the recommended trajectory obtained at the previous time instant. $u$ is the latest recommendation.

Criterion 1 decides the maximum update interval for LOOP 1 to recalculate the optimal solution. Criterion 2 represents the inconsistency between the previously recommended trajectory and real-time optimization result obtained by LOOP 2. Once any of the criteria is satisfied, LOOP 1 will be activated to re-calculate the optimal decision variables. Therefore, the update interval of LOOP 1 (as represented by $\Delta T_{L1}$ in Fig. 5) varies according to the actual situation, but will not exceed $\Delta T_{\text{max}}$. 

---

**Fig. 4. Relationship between $K$ and $P$ when $|\omega^*_{np}| < |\omega_{np}|$.**

Equation 20. $T_{ng}$ is the temperature value at iteration $ng$ [42], given by:

$$T_{ng} = \left\{ \begin{array}{ll}
T_0 & \text{for } |\omega^*_{np}| \leq |\omega_{np}| \\
\max \left\{ T_0, \left( N_p/K - |\omega^\text{glos}| \right) \right\} & \text{for } |\omega^*_{np}| > |\omega_{np}| \\
D_p = 2, & \text{for } |\omega^*_{np}| > |\omega_{np}| \\
D_p = 3, & \text{for } |\omega^*_{np}| \leq |\omega_{np}|
\end{array} \right. \quad (30)$$

where $|\omega^\text{glos}|$ denotes the number of solutions in $\omega^\text{glos}$. $T_0$ is the initial temperature. $\beta$ is the speed of annealing. $T_0$ is defined to ensure that $T_{ng}$ can always be greater than 0 and is set to 0.1. $K$ is the coefficient that adjusts the decay rate of $P$.

Assuming that $N_p = 50$ and $K = 0.1$, $T_0$ is obtained in Fig. 4. It should be noted that $P = 1$ when $|\omega^*_{np}| > |\omega_{np}|$. Therefore, this figure only presents the relationship between $K$ and $P$ when $|\omega^*_{np}| < |\omega_{np}|$. It can be observed that $P$ increases with the increasing of $K$. Taking $K = 0.1$ as an example, $P$ is maintained above 0.37. Although the high probability of selecting the worse solution enhances the search capability, it reduces the efficiency of obtaining the optimal solution. On the contrary, larger values of $K$ will result in lower probabilities, even in early iterations. Therefore, it is recommended to set $K$ to 5 when $N_p = 50$. Similarly, based on this consideration, $T_0$ and $\beta$ are set to 4 and 0.5.
Furthermore, it should be noted that the computational cost of LOOP 1 is generally much higher than LOOP 2. While LOOP 1 is still searching for the optimal solution at the least-cost goal programming stage, LOOP 2 can keep adjusting the recommended trajectory according to the latest decision variables (maybe not optimal for the current situation) and the real-time states of adjacent trains. Therefore, when implementing our model in practical applications, the train trajectory can be continuously optimized while ensuring operation safety.

Based on the proposed procedure, candidate trajectories and recommended trajectories will be calculated by LOOP 1 and LOOP 2, respectively. The computational process of planning a trajectory is presented in Fig. 6, where $S_{\text{max}}$ denotes the total mileage of the planned journey. At any instant $k$, the output force $u_{1,k}$ can be firstly obtained by the resilience adjustment model. Based on $u_{1,k}$, the trajectory from instant $k$ ($p_{i,k}$ and $v_{i,k}$) to instant $k + 1$ ($p_{i,k+1}$ and $v_{i,k+1}$) can be derived from Equation 7. These steps are then repeated until the follower can apply brakes to reach its destination ($p_{i,k} + L_{i,k}^{\text{sb}} = S_{\text{max}}$). It is noted that after the leader enters the terminal station, the train control system will assign a different route to the follower. Therefore, the original leader will no longer restrict the operation of the follower. In order to keep the consistency of the computational process, the follower will be assigned a guiding point (can be regarded as a new leader) to complete the remaining journey. The position, speed, and driving mode of this guiding point are specified as $S_{\text{max}} + L_{i,k}^{\text{safe}} = 0$, and M1, respectively. In this way, the trajectory of the follower can be planned to enter the station and reach its destination.

V. CASE STUDY

The Wuhan-Guangzhou high-speed railway line is chosen to perform the simulations here. Trains will be tested to travel 234.1km under the speed limitation of 350km/h. According to the data from Wuhan Railway Bureau, the leader is scheduled to complete its trip in 2912 seconds. From kilometer point 4.97 to 5.98, there is a section with a curve radius of 12000 meters, but otherwise the curvature of the railway is not remarkable. Fig. 7 presents the gradients of the railway.

In addition, three types of trains are utilized in the following experiments, namely CRH380AL, CR400BF, and CRH380BL. In order to provide a unified description, these train types are respectively referred to as HST1, HST2, and HST3. According to the manufacturer’s specifications and actual operational data [43], [44], [45], their maximum tractive force and maximum service braking force are calculated by Equations 32, 33, and 34. The units of speed and force are kilometers per hour (km/h) and kilonewtons (kN). The parameters of the train motion models are given in Table II.

- **HST1**:  
  \[
  U_1 = \begin{cases} 
  520, & 0 \text{km/h} \leq v \leq 50 \text{km/h} \\
  530.7068 - 0.2141 v, & 50 \text{km/h} < v \leq 167 \text{km/h} \\
  82656v^{-1}, & v > 167 \text{km/h} \\
  -704.9085, & 0 \text{km/h} \leq v \leq 70 \text{km/h} \\
  -880.7582 + 2.5113 v, & 70 \text{km/h} < v \leq 118 \text{km/h} \\
  -740.0030 + 1.3186 v, & v > 118 \text{km/h} 
  \end{cases}
  \]  

- **HST2**:  
  \[
  U_1 = \begin{cases} 
  267 - 0.2428 v, & 0 \text{km/h} \leq v \leq 160 \text{km/h} \\
  36504v^{-1}, & v > 160 \text{km/h} \\
  -294.5880, & 0 \text{km/h} \leq v \leq 5 \text{km/h} \\
  -251.7149 - 8.5746 v, & 5 \text{km/h} < v \leq 20 \text{km/h} \\
  -423.4702, & 20 \text{km/h} < v \leq 160 \text{km/h} \\
  -765.44 + 2.1373 v, & 160 \text{km/h} < v \leq 240 \text{km/h} \\
  -436.2007 + 0.7654 v, & v > 240 \text{km/h} 
  \end{cases}
  \]  

- **HST3**:  
  \[
  U_1 = \begin{cases} 
  520 - 0.4167v, & 0 \text{km/h} \leq v \leq 144 \text{km/h} \\
  66240v^{-1}, & v > 144 \text{km/h} \\
  -791.4384, & 0 \text{km/h} \leq v \leq 140 \text{km/h} \\
  -1214.5172 + 3.0229 v, & 140 \text{km/h} < v \leq 240 \text{km/h} \\
  -881.5415 + 1.6378 v, & v > 240 \text{km/h} 
  \end{cases}
  \]  

In numerical experiments, the recommended trajectory for the leader is obtained according to the general optimal driving strategy [31]. The model proposed in this study is then adopted to generate the recommended trajectory for the follower. The constant safety margin $L_{\text{min}}$ and speed tolerance $\theta_{v}$ are set to 40m and 1km/h [12]. The abbreviations used in the subsequent content are listed in Table III. All experiments are conducted on a laptop with a 3.6GHz Intel i7-12700H CPU.

A. Parameter Selection and Computational Cost Analysis

As described in Section IV-B, the algorithm is designed to update the optimal solution according to the operational data of adjacent trains at the least-cost goal programming stage, where $N_{g}$ and $N_{p}$ will affect the convergence of the algorithm. Thus,
TABLE II
NUMERICAL PARAMETERS OF HIGH-SPEED TRAINS

<table>
<thead>
<tr>
<th>Train Type</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>HST1</td>
<td>M (ton)</td>
<td>890</td>
</tr>
<tr>
<td></td>
<td>(\gamma)</td>
<td>0.06</td>
</tr>
<tr>
<td></td>
<td>(f_{br}^{L}(kN))</td>
<td>4.6280 + 0.03382v + 0.0009968v^2</td>
</tr>
<tr>
<td></td>
<td>(L_{train}(m))</td>
<td>403</td>
</tr>
<tr>
<td>HST2</td>
<td>M (ton)</td>
<td>501</td>
</tr>
<tr>
<td></td>
<td>(\gamma)</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>(f_{br}^{L}(kN))</td>
<td>1.9990 + 0.00636v + 0.0005471v^2</td>
</tr>
<tr>
<td></td>
<td>(L_{train}(m))</td>
<td>210</td>
</tr>
<tr>
<td>HST3</td>
<td>M (ton)</td>
<td>1037</td>
</tr>
<tr>
<td></td>
<td>(\gamma)</td>
<td>0.06</td>
</tr>
<tr>
<td></td>
<td>(f_{br}^{L}(kN))</td>
<td>4.2683 + 0.06707v + 0.001047v^2</td>
</tr>
<tr>
<td></td>
<td>(L_{train}(m))</td>
<td>400</td>
</tr>
</tbody>
</table>

TABLE III
EXPLANATION OF ABBREVIATIONS IN THE FOLLOWING EXPERIMENTS

<table>
<thead>
<tr>
<th>Abbr.</th>
<th>Explanation</th>
<th>Definition</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAS</td>
<td>Deviation between (L_{act}) and (L_{safe})</td>
<td>(L_{act} - L_{safe})</td>
<td>m</td>
</tr>
<tr>
<td>DAO</td>
<td>Deviation between (L_{act}) and (L_{opt})</td>
<td>(L_{act} - L_{opt})</td>
<td>m</td>
</tr>
<tr>
<td>SD</td>
<td>Speed difference</td>
<td>(v_i - v_{i-1})</td>
<td>km/h</td>
</tr>
<tr>
<td>DCC</td>
<td>Deviation of capacity consumption</td>
<td>(\phi_{cc}^{ref} - \phi_{cc}^{cc})</td>
<td>s</td>
</tr>
<tr>
<td>DEC</td>
<td>Deviation of energy consumption</td>
<td>(\phi_{ec}^{ref} - \phi_{ec}^{ec})</td>
<td>kWh</td>
</tr>
</tbody>
</table>

*tgt indicates the target solution being evaluated.
*ref indicates the reference solution.

The parameters need to be appropriately set. The computational cost of the algorithm is analyzed by:

\[
C_{ng} = \frac{\phi_{cc}^{cc, glo} - \min(\phi_{cc}^{cc})}{\max(\phi^{cc})} - \min(\phi_{cc}^{cc}) + \frac{\phi_{ec}^{cc, glo} - \min(\phi_{ec}^{cc})}{\max(\phi^{cc})} - \min(\phi_{ec}^{cc})
\]

(35)

where \(\min(\cdot)\) and \(\max(\cdot)\) denote the minimum and maximum capacity (energy) consumption among all solutions. \(\phi_{cc}^{cc, glo}\) and \(\phi_{ec}^{cc, glo}\) are the capacity and energy consumption of the global extremum at iteration \(ng\).

Assuming that \(N_g = 100\) and \(N_p = 50\), we can obtain the experimental results in Fig. 8, where the algorithm is tested 100 times. The convergence of each trial is plotted in black, and the average computational cost at each iteration is plotted in red. It should be noted that the average deviation of the solution’s performance (capacity consumption & energy consumption) at iterations 20 and 100 are 0s and 2.1 kWh, respectively. Hence, it is acceptable to use the solution obtained at iteration 20. Furthermore, the algorithm takes about 2.6s with \(N_g = 20\), but about 14.1s with \(N_g = 100\). In order to balance the solution’s performance and computational cost, \(N_g\) and \(N_p\) are set to 20 and 50 in this paper.

Using the parameters selected above, the detailed analysis of the computational cost for implementing our proposed method is provided as follows. It is noted that calculating the trajectory based on train motion model and resilience adjustment model is the most computationally intensive part for each LOOP. In addition, as shown in Fig. 6, the computational tasks increase with the time range covered by the planned trajectory. Therefore, experiments are conducted to evaluate the computational costs (\(T_{L1}^{MP}\), \(T_{L1}^{LP}\), and \(T_{L2}\) in Fig. 5) when following trains that have different total running times. The average computational costs are provided in TABLE IV, where the leaders maintain an average speed of 200 km/h and a total running time ranging from 20 to 70 minutes.

For LOOP 1, at both multi-objective and least-cost goal programming stages, the algorithm obtains candidate trajectories and repeats \(N_g \cdot N_p\) times during iterations. LOOP 2 generates the recommended trajectory in the same way, but only calculates it once per cycle. Therefore, LOOP 1 (\(T_{L1}^{MP}\) and \(T_{L1}^{LP}\)) needs to take much more time than LOOP 2 (\(T_{L2}\)). Moreover, the differences between the multi-objective programming stage and the least-cost goal programming stage mainly lie in the number of decision variables and the objective functions. As shown by the results of \(T_{L1}^{LP}\) and \(T_{L1}^{MP}\), these differences have little impact on the computational cost.

According to the proposed implementation procedure, there are two constraints on the computational costs:

First, at the multi-objective programming stage, the optimal solution must be obtained before the departure of the follower. In this paper, the departure headway is defined within the range of 120s to 300s. Hence, \(T_{L1}^{MP}\) must be smaller than 120s.

Second, the update interval of LOOP 2 is fixed at 1s. Hence, \(T_{L2}\) must be smaller than 1s.

Although there are no constraints on \(T_{L1}^{LP}\), the lower computational cost means that followers can obtain optimal solutions that are closer to the current conditions. Hence, a smaller \(T_{L1}^{LP}\) is preferred.

As presented in above experiments, the computational costs of LOOP 1 (\(T_{L1}^{MP}\), \(T_{L1}^{LP}\)) and LOOP 2 (\(T_{L2}\)) are maintained at the level of seconds and microseconds, respectively. Therefore, the computational costs of the proposed method can meet the implementation requirements.

B. Generation of Recommended Trajectories

The trajectory generation experiment is given in this section, which will illustrate typical outcomes of our proposed model.
The high-speed train HST1 is used to perform the following experiment.

The optimization results at the multi-objective programming stage are shown in Fig. 9. The green line in Fig. 9a presents the trajectory of the leader. Its capacity consumption and energy consumption are 2912s and 9494.22kWh.

Based on the proposed algorithm, candidate trajectories can be generated by applying the resilience adjustment model with different decision variables. Then, all candidates are evaluated by Equation 16 and their evaluation results are represented by black dots in Fig. 9b. For the trajectory optimization problem at the multi-objective programming stage, the Pareto solutions will be selected, as shown by blue dotted lines in Fig. 9a. The corresponding Pareto frontier is plotted in blue in Fig. 9b. It can be observed that the minimum energy consumption of the follower is higher than the leader by about 584.74kWh. This is because the follower needs to accelerate to a higher speed in order to catch up with the leader, leading to the higher energy consumption.

Using the solution with the minimum capacity consumption as the recommended trajectory (red line in Fig. 9a, cc = 2960s and ec = 10100.13kWh), we will analyze its operating results in details, as shown in Fig. 10.

The variations of the resilience and output force are shown in Fig. 10a. This trajectory, the solution for the resilience adjustment model, is x = 0.186 and e = 0.284. In addition, uncertainties are not considered here, and the follower can accurately follow the recommended trajectory during operation. Thus, there is no deviation between the planned trajectory and actual operating result. The updated solution can always be the same, or sometimes with slight variations.

Fig. 10b shows the variations in spacings during operation. It is noted that the calculation results of the minimum safe spacing Lsafe and the optimal coupling spacing Lopt are less than zero until 8.27km. In order to show more details of the coupled running state, this figure only displays the portion of the y-axis that is greater than zero.

Based on Equation 11, we can further identify whether the follower is coupled to the leader. The coupled running state is maintained from 120.91km to 217.38km (from 1526s to 2657s in temporal) with an average spacing of 1.01km. During this period, the follower maintains its actual spacing at Lsafe within tolerance while ensuring that the actual spacing is greater than Lsafe. Moreover, at 230.09km, the follower applies the brake and transitions to the decoupling state. Hence, there is a buffer zone between the coupled running state and the decoupling state, from 217.38km to 230.09km. In this region, the follower runs in the coasting mode (Fig. 10a), holds the same speed as the leader (Fig. 10c), and maintains the actual spacing unchanged (Fig. 10b). However, Lsafe gradually changes during this period. This is attributed to the different decay rates between the follower’s service braking distance Lsb,i,k and the leader’s emergency braking distance Leb,i−1,k. As defined in Equation 3, Lsafe is mainly affected by the calculation result of Lsb,i,k − Leb,i−1,k. Even if the follower and its leader run at the same speed, the result varies depending on the specific speed value. The effect of the braking characteristics on the convoy operation will be further studied in the next subsection and therefore will not be explained in detail here. In summary, during this period, the follower uses Phase III as the buffer zone and continues coasting instead of applying brakes, thereby saving energy and simultaneously guaranteeing efficiency.

C. Convoy Optimization for Heterogeneous Trains

As illustrated in the section of resilience adjustment model, the maximum tractive force and the maximum service braking force are applied to calculate the recommended trajectories for...
TABLE V
PERFORMANCE OF OUR PROPOSED MODEL WHEN USING DIFFERENT TYPES OF TRAINS AS LEADERS

<table>
<thead>
<tr>
<th>ID</th>
<th>Position</th>
<th>Type</th>
<th>$\phi^c$ (s)</th>
<th>$\phi^{ec}$ (s)</th>
<th>DCC (kWh)</th>
<th>DEC (kWh)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1 Leader</td>
<td>HST1</td>
<td>2912</td>
<td>9494.22</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Follower 1</td>
<td>HST1</td>
<td>2960</td>
<td>10100.13</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Follower 2</td>
<td>HST2</td>
<td>2960</td>
<td>5370.38</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Follower 3</td>
<td>HST3</td>
<td>2965</td>
<td>11651.15</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>S2 Leader</td>
<td>HST2</td>
<td>2912</td>
<td>5059.45</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Follower 1</td>
<td>HST1</td>
<td>2952</td>
<td>10131.23</td>
<td>-8</td>
<td>+31.09</td>
<td>+32.54</td>
</tr>
<tr>
<td>Follower 2</td>
<td>HST2</td>
<td>2952</td>
<td>5398.47</td>
<td>-8</td>
<td>+28.09</td>
<td>+30.35</td>
</tr>
<tr>
<td>Follower 3</td>
<td>HST3</td>
<td>2958</td>
<td>11741.13</td>
<td>-7</td>
<td>+85.98</td>
<td>+92.21</td>
</tr>
<tr>
<td>S3 Leader</td>
<td>HST3</td>
<td>2912</td>
<td>11433.04</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Follower 1</td>
<td>HST1</td>
<td>2960</td>
<td>10527.14</td>
<td>0</td>
<td>+427.01</td>
<td>+439.42</td>
</tr>
<tr>
<td>Follower 2</td>
<td>HST2</td>
<td>2958</td>
<td>5538.22</td>
<td>-2</td>
<td>+167.84</td>
<td>+180.04</td>
</tr>
<tr>
<td>Follower 3</td>
<td>HST3</td>
<td>2964</td>
<td>11979.40</td>
<td>-1</td>
<td>+328.25</td>
<td>+340.46</td>
</tr>
</tbody>
</table>

*To calculate DCC and DEC in S2 and S3, $\phi^c$ and $\phi^{ec}$ in the same type of the follower in S1 are set as $\phi^{ec}_{off}$ and $\phi^{ec}_{off}$.

This higher force requirement translates into a higher energy demand, resulting in increased energy consumption. Therefore, in each experimental group, HST3 (1037ton) > HST1 (890ton) > HST2 (501ton) in terms of the energy consumption.

- Effect of train length on capacity consumption:
  Comparing the results shown in Table V and Fig.11a, it can be concluded that achieving smaller coupling spacing does not necessarily guarantee lower capacity consumption. The length of the leader is one of the key factors influencing efficiency. A longer leader imposes more limitations on the follower (Equation 2), i.e., the follower must travel a longer distance to reach its destination after the leader enters the terminal station. Therefore, for the same type of followers, S1 ($L_{\text{train}}^{i-1} = 403m$) ≥ S3 ($L_{\text{train}}^{i-1} = 400m$) > S2 ($L_{\text{train}}^{i-1} = 210m$) in terms of the capacity consumption (Note that HST1 and HST3 are very close in train length).

- Effect of traction characteristic on coupling point:
  The coupling point is the time/position at which the follower transitions to the coupled running state. In Fig.11a, squares and dots are used to represent where the followers transition to and transition out of the coupled running state. The detailed ranges for maintaining the coupled running state are provided in Table VI. For the followers with the same leader (in the same group), they transition out of the coupled running state at almost the same time point, which are mainly influenced by the trajectory of the leader. However, the result shows significant variations in the coupling point:
  1. In each group, the follower HST1 consistently arrives at the coupling point first, followed by HST2 and HST3.
  2. For the same type of followers, the follower in S3 (HST3 is the leader) can reach the coupling point earlier, followed by S2 and S1.

These phenomena can be attributed to the differences in traction characteristics among trains. Fig.12a presents the traction distances achieved by different trains, where vertical axis indicates the distance required to accelerate from zero to a certain speed. Conspicuously, HST1 has the best traction performance, being able to reach the same speed with the shortest traction distance, while HST3 shows the poorest traction performance. The follower with better traction performance (i.e., HST1) can catch up with the leader faster and thus reach the coupling point earlier. Furthermore, the traction characteristics of the leader have the opposite effect on the coupling process, accordingly exhibiting a contrasting trend.
Fig. 12. Comparison of traction and braking distances among heterogeneous trains. (a) Traction distance. (b) Service braking distance. (c) Emergency braking distance.

<table>
<thead>
<tr>
<th>ID</th>
<th>Follower</th>
<th>MIN (km)</th>
<th>AVG (km)</th>
<th>RMSE (km)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>HST1</td>
<td>1.00</td>
<td>1.01</td>
<td>0.0038</td>
</tr>
<tr>
<td></td>
<td>HST2</td>
<td>2.23</td>
<td>2.25</td>
<td>0.0053</td>
</tr>
<tr>
<td></td>
<td>HST3</td>
<td>2.72</td>
<td>2.74</td>
<td>0.0090</td>
</tr>
<tr>
<td>S2</td>
<td>HST1</td>
<td>0.89</td>
<td>0.95</td>
<td>0.0048</td>
</tr>
<tr>
<td></td>
<td>HST2</td>
<td>2.09</td>
<td>2.15</td>
<td>0.0061</td>
</tr>
<tr>
<td></td>
<td>HST3</td>
<td>2.61</td>
<td>2.67</td>
<td>0.0088</td>
</tr>
<tr>
<td>S3</td>
<td>HST1</td>
<td>0.49</td>
<td>0.51</td>
<td>0.0040</td>
</tr>
<tr>
<td></td>
<td>HST2</td>
<td>1.81</td>
<td>1.83</td>
<td>0.0053</td>
</tr>
<tr>
<td></td>
<td>HST3</td>
<td>2.32</td>
<td>2.35</td>
<td>0.0070</td>
</tr>
</tbody>
</table>

- Effect of braking characteristic on operating spacing:

As demonstrated in Section V-B, the follower in the coupled running state will keep the actual spacing at the optimal coupling spacing within tolerance. Therefore, the optimal coupling spacing is not presented in Fig.11. The statistical results of the actual spacing in coupled running state are presented in Table VII, which includes the minimum (MIN), average (AVG), and root mean square error (RMSE). From Fig.11b and Table VII, it can be observed that:

1. In each group, the follower HST1 presents smaller values for both $L_{act}$ and $L_{safe}$, whereas the follower HST3 has larger $L_{act}$ and $L_{safe}$.

2. For the same type of followers, $L_{act}$ and $L_{safe}$ are always smaller when following the leader HST3 (S3), and larger when the leader is HST1 (S1).

These two phenomena are caused by the characteristics of service braking and emergency braking, respectively. Fig.12b and Fig.12c show the service braking distance and emergency braking distance for each type of train. It can be observed that $L_{sb}^{HST3} > L_{sb}^{HST2} > L_{sb}^{HST1}$ and $L_{eb}^{HST3} > L_{eb}^{HST2} > L_{eb}^{HST1}$ at the same speed. In accordance with the definition provided in Equation 3, $L_{safe}$ is directly proportional to the service braking distance of the follower, and inversely proportional to the emergency braking distance of the leader. Moreover, $L_{act}$ is close to $L_{opt}^{act}$ during coupled running, and is directly proportional to $L_{safe}$. Therefore, spacings in coupled running state is smaller when the train with better braking performance (HST1) is used as the follower, but become larger when that train is the leader. Conversely, HST3 has the opposite characteristics and thereby exhibits the reversed effect compared to HST1.

Based on the above analysis, it can be concluded that our method can generate recommended trajectories for heterogeneous trains under VCS. Different types of trains have different operational outcomes, which can be attributed to the effect of train characteristics on the fundamental principles of train operation, namely the laws of motion and basic safety constraints. Thus, these differences cannot be eliminated by optimizing the trajectory planning model. Furthermore, from Table VI and Table VII, it can be observed that trains with better traction and braking performance (HST1) are more suitable for convoy operations, considering both the coupling ranges and operating spacings. Therefore, the following experiments will be conducted based on the high-speed train HST1.

D. Performance Comparison

Experiments in above sections have illustrated the capabilities of our proposed model, which can generate recommended trajectories for convoy operations involving identical or heterogeneous trains. In this section, two other models are employed to demonstrate the superiority of our proposed model. A brief description of these two models is provided below.

Model 1 [21] establishes the modified safe spacing for train operations in coupling, coupled running, and decoupling state. On this basis, it generates recommended trajectories for virtually coupled trains under the constraints of safety boundaries. Model 2 [22] is developed to evaluate the operating spacing of high-speed trains, and then to obtain recommended trajectories based on a resilience optimization mechanism.

The experimental results of different models under ideal and disturbing conditions are presented as follows.

1) Convoy Optimization Under Ideal Conditions:

The experiment on trajectory generation under ideal conditions, that has already been presented by our model in Section V-B, is again conducted to illustrate the optimization process of different models. By following the same leader described in Section V-B, we obtain the optimization results using different models, as shown in Fig.13, Table VIII, and Table IX. It can be observed that different models show diverse characteristics on various indicators, and the detailed analysis is provided as follows.

Based on Model 1, the follower is coupled to the leader from 143.97km to 225.26km. According to the modified safe spacing derived by Model 1, the follower must adopt a conservative driving mode when the leader decelerates and approaches its destination. As depicted in Fig.13a, the follower starts braking to decouple from the convoy at 225.26km, and keeps cruising at a lower speed between 227.53km and 230.51km. However, the utilization of braking and low-speed cruising will result in increased energy consumption and capacity consumption. As shown in Table IX, compared with Model 1, our model saves the capacity consumption and energy consumption by 2.1% and 13.9%, respectively. Furthermore, our model demonstrates an average reduction of 61.5% in coupling spacing.
By Model 2, the follower transitions to the coupled running state earlier, at the distance of 108.26 km (temporally at 1427 s). Furthermore, it transitions out of this state in a time similar to that of our model, but smaller in position. These phenomena imply that, compared to our model, less time is consumed for catching up with the leader and the average speed is maintained at a lower level from 0 km to 213.05 km, as shown by the green curve in Fig. 13a. Therefore, during the period of coupled running state, the spacing between virtually coupled trains is relatively large. Compared with Model 2, our model can reduce the coupling spacing by an average of 80.8%.

Additionally, with regards to the capacity consumption and energy consumption, our model shows a reduction of 0.8% and 3.8%, respectively.

In addition to the typical experimental scenario illustrated above, the experiment for convoy optimization under different speed conditions are conducted here. In following experiment, the leader is allocated trajectories with varying average speeds, and then different models are employed to generate trajectories for the follower. The statistical results are shown in Fig. 14.

As presented in Fig. 14a and Fig. 14b, the capacity consumption and energy consumption exhibit a conflicting relationship. As the speed increases, it is observed that the capacity consumption decreases, while the energy consumption increases. Comparing the results between different models, it is evident that our model always presents lower consumption levels under all speed conditions. Compared with Model 2, our model saves the capacity and energy consumption by an average of 0.5% (DCC = 16.82 s) and 9.2% (DEC = 755.95 kWh). Particularly, as demonstrated in Fig. 14d, in terms of the energy consumption, our model performs better when following the leader with a lower average speed. Compared with Model 2, our proposed model saves the energy consumption by an average of 14.0% (DEC = 1100.50 kWh) when the average speed of the leader is below 260 km/h. Otherwise, our model presents a reduction of 3.5% (DEC = 342.50 kWh) on average.

2) Convoy Optimization Under Disturbances:

In the real world, various disturbing factors can occur and affect the operation of the high-speed train, including the error in data acquisition, the impact of extreme weather, and the execution deviation of controller. These circumstances may lead to large deviations between the actual and planned trajectories, which pose a huge challenges to convoy optimization methods. Hence, to verify the effectiveness of the proposed model under disturbances, the state of the leader is simulated based on field data with positioning and speed errors. Then, different models are adopted to optimize the follower’s trajectory.

Moreover, to validate the significance of dynamically adjusting parameters, a modified version of our model is included here, called “our model (constant)”. Using this model, the decision variables obtained at the multi-objective programming stage will not be changed during operation, i.e., the least-cost goal programming stage is blocked in LOOP 1. As shown by the black line in Fig. 15a, due to disturbances, the trajectory of the leader (received by the follower via T2T communication) deviates from the original plan during operation. By using our proposed model, the coupling
process will be continuously evaluated, and then we can decide whether the configuration of the adjustment model and the recommended trajectory should be adjusted. The adjustment process is shown in Fig.16a. Since resilience is defined as $L_{\text{opt}}/L_{\text{act}}$ in Equation 9, the disturbances will directly affect the value of resilience. Furthermore, compared with the constant version of our model in Fig.16b, the solution of LOOP 1 ($\xi$ and $\epsilon$) also varies over a wider range, as shown by the boundaries of different phases in Fig.16. Due to changes in resilience parameters, the optimal coupling spacing will also change, which will eventually lead to larger fluctuations in the actual spacing.

Table X presents the statistical results of the actual spacing during the planned coupled running period under disturbances. We can observe that, in the presence of disturbances, it becomes challenging to keep the spacing at a constant value. In order to ensure safety, larger spacings are maintained compared with the results under ideal conditions. In this experiment, compared to Model 1 and Model 2, our model reduces the actual spacing by an average of 26.0% and 56.1%, respectively.

Although a smaller spacing can be obtained by our model (constant), it also results in more frequent adjustments of the output forces, as presented by the purple dotted line in Fig.16. When using constant decision variables, the follower needs to frequently accelerate to catch up or decelerate to avoid conflict with the leader, and therefore will consume more energy. Moreover, as shown in Tables X and XI, smaller spacing does not guarantee lower capacity consumption. Overall, compared with other models, our model proposed in this study presents the optimal performance in terms of capacity consumption and energy consumption.

Apart from the operational performance, the computational cost is the crucial factor determining the feasibility of implementing a model in the real world. Based on the experiment illustrated above, the computational costs of different models (excluding the constant version of our model) are analyzed as follows.

Differing from other models, this paper provides two loops to obtain the optimal solution and the recommended trajectory. The detailed design has been illustrated in Section IV-B. The triggering positions and computational costs of LOOP 1 are demonstrated in Fig.17a, which indicates that this loop is triggered 78 times during operation. The multi-objective programming stage is firstly calculated in LOOP 1 after the leader departs the station, and its computational cost $T_{\text{MP}}^{L1}$ equals 3.11s in this experiment. Then based on the adjustment criteria defined in Equation 31, the least-cost goal programming stage is triggered with the update interval $T_{\text{LP}}^{L1}$ ranging from 1s to 120s, with an average of 35.79s. The computational cost $T_{\text{LP}}^{L1}$ varies from 2.98s to 0.08s.

Furthermore, the update interval of LOOP 2, Model 1 and Model 2 is 1s, which is smaller than LOOP 1. Thus, Fig.17b, Fig.17c, and Fig.17d have more sampling points than Fig.17a. LOOP 2 of our model is designed to update recommended trajectories based on decision variables derived from LOOP 1. As shown in Fig.17b, its computational cost is observed to be below 0.004s in this experiment.

Model 1 generates the recommended trajectory according to a series of predefined rules, but does not adjust its parameters during operation. As shown in Fig.17c, its computational cost is in the sub-second range, consistently below 0.09s.

As for Model 2, its parameters are determined by traversing feasible solutions, which may require several hours. Therefore,

![Fig. 15. Optimization results of different models under disturbances. (a) Recommended trajectory. (b) Actual spacing.](image)

![Fig. 16. The resilience and output force of the recommended trajectory under disturbances. (a) Our Model. (b) Our Model (constant).](image)
This paper presents an optimal trajectory planning method that aims to generate time-efficient and energy-saving driving strategies for high-speed trains under the virtual coupling system. The resilience adjustment model is proposed to evaluate the coupling process and to define a dynamic optimal region for convoy operations. By varying the parameters in this model, candidate trajectories with different characteristics (capacity and energy consumption) can be generated. Considering the optimization goals in both offline and online scenarios, a two-stage programming model is then formulated to quantify the optimal solutions. Finally, the solution method is provided to integrate and implement our proposed models.

Numerical experiments based on field data from the Wuhan-Guangzhou high-speed railway line are conducted to verify the effectiveness of the proposed method. Compared with two recent representative models from the literature, our method presents the optimal performance in terms of both capacity consumption and energy consumption. Furthermore, the optimal coupling spacing can be dynamically adjusted according to the actual circumstances.

Further research will be devoted to enhance the stability of convoy operations under disturbances, taking into account the communication delay. The resilience adjustment model will be improved by integrating more factors, such as the predicted state of the leader. Moreover, planning trajectories for multiple convoys will also be an important research focus in our future work.

VI. CONCLUSION

This paper presents an optimal trajectory planning method that aims to generate time-efficient and energy-saving driving strategies for high-speed trains under the virtual coupling system. The resilience adjustment model is proposed to evaluate the coupling process and to define a dynamic optimal region for convoy operations. By varying the parameters in this model, candidate trajectories with different characteristics (capacity and energy consumption) can be generated. Considering the optimization goals in both offline and online scenarios, a two-stage programming model is then formulated to quantify the optimal solutions. Finally, the solution method is provided to integrate and implement our proposed models.

Numerical experiments based on field data from the Wuhan-Guangzhou high-speed railway line are conducted to verify the effectiveness of the proposed method. Compared with two recent representative models from the literature, our method presents the optimal performance in terms of both capacity consumption and energy consumption. Furthermore, the optimal coupling spacing can be dynamically adjusted according to the actual circumstances.

Further research will be devoted to enhance the stability of convoy operations under disturbances, taking into account the communication delay. The resilience adjustment model will be improved by integrating more factors, such as the predicted state of the leader. Moreover, planning trajectories for multiple convoys will also be an important research focus in our future work.

REFERENCES

SONG et al.: TWO-STAGE OPTIMAL TRAJECTORY PLANNING BASED ON RESILIENCE ADJUSTMENT MODEL


Hongyu Song received the B.S. degree from Southwest Jiaotong University, Sichuan, China, in 2017. She is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering, Beijing Jiaotong University.

Her research interests include train operation optimization, intelligent transportation systems, resilience theory, and its application in transportation systems.

Wei Shang Guan was born in Xianyang, Shanxi, China, in 1979. He received the B.S., M.S., and Ph.D. degrees from Harbin Engineering University, Harbin, China, in 2002, 2005, and 2008, respectively.

From 2008 to 2011, he was a Lecturer with Beijing Jiaotong University. From 2013 to 2014, he was an Academic Visitor with University College London. From 2012 to 2016, he was an Associate Professor. Since 2017, he has been a Professor with Beijing Jiaotong University. He is the author of three books, more than 100 articles, and more than 30 inventions. His research interests include train control systems (CTCS/ETCS/ERTMS), system modeling, simulation and testing, GNSS/GIS, integrated navigation, intelligent transportation systems, and cooperative vehicle infrastructure system of China (CVIS-C).

Weizhi Qiu received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2017, where he is currently pursuing the Ph.D. degree with the School of Electronic and Information Engineering.

His research interests include modeling and simulation of transportation systems, virtual–real interactive testing, and stochastic programming.

Zhao Sheng received the B.S. degree in automation and the Ph.D. degree in traffic information engineering and control from Beijing Jiaotong University, in 2015 and 2022, respectively.

He is currently a Research Assistant at the China Academy of Railway Sciences. His research interests include energy-efficient train control and train scheduling.

Steven S. Harrod received the M.Sc. degree from MIT and the Ph.D. degree from the University of Cincinnati.

He is an Associate Professor at the Technical University of Denmark. He is a specialist in transportation management, transportation analysis, and supply chain networks. He has authored or coauthored numerous research articles in journals, such as Transportation Science, Transportation Research Part E, International Journal of Production Economics, and Simulation. He also has extensive experience in industry, including many years in road transport and information systems.