Cooperative Control of Virtually Coupled Train Sets through the Fusion of Multiple Car-Following Models

Mingjun Qian, Zhiwen Huang, Chao Yin

Abstract—**To address the challenges of stability, safety, and computational efficiency in the cooperative control of virtually coupled train sets (VCTS) under complex operational scenarios, this study proposes an integrated OPO-MCF-DMPC method that combines Online Parameter Optimization (OPO), Multiple Car-Following models (MCF), and Distributed Model Predictive Control (DMPC). First, an MCF-DMPC architecture is developed for virtual coupling by enhancing and integrating the Intelligent Driver Model (IDM), Cooperative Adaptive Cruise Control (CACC), and Full Velocity Difference Model (FVD), thereby improving precision in velocity-displacement tracking and achieving the control objective. Next, to overcome the model selection challenge, a Multi-Strategy Moth-Flame Optimization (MSMFO) algorithm is designed, incorporating chaotic mapping, adaptive flame adjustment, and stochastic mutation. Compared to conventional optimization algorithms, MSMFO improves solution fitness by 2.2%–10.2% and reduces computation time by 4.4%–16.7%. Finally, an OPO mechanism is introduced for dynamic online correction of car-following model parameters, further enhancing adaptive control capabilities. Comprehensive simulations demonstrate that the proposed OPO-MCF-DMPC method achieves a 35.8% improvement in computational efficiency over standard DMPC under long prediction horizons while satisfying the stringent stability and real-time performance requirements of VCTS operations. This advancement provides a practical and efficient solution for realizing virtual coupling in railway systems.**

Index Terms—**Virtual coupling, virtually coupled train sets, car-following model, distributed model predictive control, moth-flame optimization algorithm**

I. Introduction

DUE to the spatiotemporal distribution imbalance of passenger flow along railway transit lines, flexible train formation or dynamic reconfiguration is often required to align transport capacity with passenger demand and to achieve green operational objectives. However, the conventional fixed formation mode lacks real-time adaptability to dynamic transportation demands, leading to issues such as low operational efficiency, suboptimal resource utilization, and poor environmental performance. In contrast, Virtual Coupling (VC) technology obviates the necessity for physical couplings between conventional train units, relying instead on vehicle-to-vehicle communication and distributed control to establish virtual connections among train units [1, 2]. This technology enables real-time dynamic reconfiguration of operational relationships and coupled cooperative control mechanisms among train units, facilitating stable operation under a Relative Distance Braking Mode (RDBM) with minimized relative braking distances [3, 4]. VC effectively overcomes the spatial and temporal constraints of traditional fixed formation modes [5]. This advancement enhances rail network throughput efficiency and service quality while reducing energy consumption and operational safety risks for train sets [6]. Consequently, ensuring safe, stable, and precise following operations of train units remains a critical challenge for VC implementation [7].

In recent years, the rapid advancement of autonomous train driving and wireless communication technology has garnered significant scholarly attention toward the development of safe and efficient tracking control strategies for Virtually Coupled Train Sets (VCTS) operating under RDBM. Among these, key contributions include a variety of control methodologies [8], such as Car-Following (CF) models [9], Model Predictive Control (MPC) [10], artificial potential field control [7], optimization and optimal control, sliding mode control [11], and feedback control approaches. Among these methodologies, CF models and MPC have emerged as particularly prominent due to their computational efficiency and precise control performance, leading to their widespread adoption in relevant applications.

CF models, which describe the dynamic response characteristics of trailing vehicles to preceding ones, are also applicable to VC systems. In existing studies, Quaglietta et al. [9] developed a train CF model that incorporated multi-state transition mechanisms to facilitate smooth transitions across different operational modes. Pan et al. [12] proposed a headway calculation method for VC trains and conducted comparative analyses across eight typical CF scenarios. To enhance the efficiency of implementing VC in trains, Shuai et al. [13] designed a desired-spacing-based train CF model to achieve the coupling of multiple high-speed trains. However, existing research predominantly employs single-model architectures with offline-calibrated parameters, which

Manuscript received June 6, 2025; revised August 20, 2025.

This work was supported in part by Double First Class Major Scientific Research Project of Gansu Provincial Department of Education (GSSYLXM-04); Gansu Provincial Department of Education Higher Education Innovation Fund Project (2020A-038); and Youth Fund Project of Lanzhou Jiaotong University (2014029).

Mingjun Qian is an Associate professor at the School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China (corresponding author, e-mail: qianmingjun@mail.lzjtu.cn).

Zhiwen Huang is a Postgraduate student at the School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China (e-mail: 12241118@stu.lzjtu.edu.cn).

Chao Yin is a Postgraduate student at the School of Traffic and Transportation, Lanzhou Jiaotong University, Lanzhou 730070, China (e-mail: 2055375426@qq.com).

results in limitations regarding environmental adaptability, objective function-driven orientation, and coupling precision in complex traffic scenarios.

MPC achieves optimal system control by formulating and solving constrained optimal control problems over a finite time horizon, resulting in an optimal sequence of control inputs for future operations. In the context of Centralized MPC (CMPC) applications for VCTS, Xun et al. [14] proposed a multi-train cooperative CMPC strategy that incorporates train dynamics to regulate inter-train spacing. Meanwhile, Su et al. [15] developed an adaptive CMPC framework for nonlinear safety-constrained VC scenarios, establishing cruise-phase tracking objectives based on braking distance differentials and speed disparities. However, CMPC implementations encounter challenges related to high computational complexity and significant communication dependencies. Distributed MPC (DMPC) addresses these limitations through localized optimization and information exchange mechanisms [16, 17]. For instance, Liu et al. [18] innovatively introduced a terminal constraint-based DMPC architecture to ensure safe and stable control of high-speed VCTS under complex operating conditions. J. Li et al. [19] proposed a distributed robust MPC framework combined with a discrete Kalman filter to address structural and external uncertainties in VCTS. Furthermore, Z. Li et al. [20] formulated a bidirectional topological DMPC method for leader-follower VCTS configurations, overcoming the limitations of unidirectional architectures and enabling more robust distributed coordination.

Although DMPC has shown considerable advantages in VCTS cooperative control, its dependence on conventional solvers leads to bottlenecks in both computational efficiency and convergence performance when addressing complex model structures, long prediction horizons, and high-frequency sampling requirements. Matrone et al. [21] emphasized that achieving high-performance MPC requires long prediction horizons, which substantially exacerbate computational complexity, with the complexity growing superlinearly as the prediction length increases. Lin et al. [22] experimentally demonstrated that while control accuracy improves with longer prediction horizons, the computational time increases accordingly.

In summary, classical CF models suffer from insufficient environmental adaptability due to their single-model architectures and reliance on offline-calibrated parameters. Concurrently, existing MPC approaches face computational challenges when addressing complex system models. These observations reveal that single-control approaches for VCTS exhibit critical technical bottlenecks, including limited generalization capabilities and suboptimal cooperative control performance, when applied to dynamic and complex rail transit operational scenarios. Therefore, to address these limitations, there is an urgent need for multi-model fusion frameworks to enhance the system's environmental adaptability and operational robustness.

In response to these challenges, this paper proposes a collaborative control method for VCTS that integrates an Online Parameter Optimization (OPO) mechanism with a Multiple Car-Following models and Distributed Model Predictive Control (MCF-DMPC) method. The methodology primarily achieves dynamic control and collaborative optimization of velocity-displacement tracking for VCTS through the fusion of an enhanced intelligent optimization algorithm and the OPO method. Experimental results demonstrate that the proposed method exhibits significant advantages in computational efficiency, fitness, and control stability, thereby providing a theoretical foundation and technical support for the engineering implementation of VC technology.

II. Preliminaries

A. VCTS Model

1) Dynamic modeling of VCTS: In sectional operational scenarios, the dynamic analysis of VCTS primarily focuses on the dominant mechanism of longitudinal acceleration. Guided by Newtonian mechanics and engineering simplification principles, this study rationally neglects secondary factors such as grade resistance and curvature resistance. Consequently, a single-mass point train model is formulated for flat track conditions:

$$\begin{cases} \dot{s}_i(t) = v_i(t) \\ \dot{v}_i(t) = u_i(t) - \omega_i(t) \\ \omega_i(t) = c_0 + c_1 v_i(t) + c_2 v_i(t)^2 \end{cases} \quad i = 1, \cdots, N \tag{1}$$

where i denotes the i-th train unit within the VCTS. N represents the total number of train units in the set. $s_i(t)$, $v_i(t)$, $u_i(t)$, and $\omega_i(t)$ correspond to the displacement, velocity, control acceleration, and fundamental resistance of the i-th train unit at time t, respectively. c_0, c_1 and c_2 are the Davis resistance coefficients.

By selecting state variables $x_i = [s_i, v_i]^T$ and control variable u_i, the corresponding state transition equations were discretized using the Euler method as follows:

$$x_i(t+1) = Ax_i(t) + Bu_i(t) - B\omega_i(t) \tag{2}$$

where $A = \begin{bmatrix} 1 & \tau \\ 0 & 1 \end{bmatrix}$. $B = \begin{bmatrix} \frac{\tau^2}{2} & \tau \end{bmatrix}^T$. τ is the sampling time.

Within the prediction horizon $\left[t_k, t_{k+Np}\right]$, the system predicted state sequence $X_i^{(P)}(t_k)$ and the predicted control input sequence $U_i^{(P)}(t_k)$ are defined as:

$$\begin{cases} X_i^{(P)}(t_k) = \left[x_i(t_k \mid t_{k+1}), \cdots, x_i\left(t_k \mid t_{k+Np}\right)\right]^T \\ U_i^{(P)}(t_k) = \left[u_i(t_k \mid t_{k+1}), \cdots, u_i\left(t_k \mid t_{k+Np-1}\right)\right]^T \end{cases} \tag{3}$$

where $x_i(t_k \mid t_{k+1})$ and $u_i(t_k \mid t_{k+1})$ represent the system state and control input at time t_{k+1} predicted at time t_k, respectively. Np is the prediction horizon.

2) Expected spacing: Under VC conditions, train unit i must satisfy synchronization targets with its immediately preceding train unit $i-1$ in both velocity and displacement:

$$\begin{cases} v_i(t) \to v_{i-1}(t) \\ \Delta S_{i,i-1}(t) \to D_{i,i-1}^{(e)}(t) \end{cases} \tag{4}$$

where $\Delta S_{i,i-1}(t)$ is the actual real-time spacing between the leading and trailing trains. $D_{i,i-1}^{(e)}(t)$ represents the expected spacing between train unit i and its preceding train unit $i-1$.

On the premise of ensuring operational safety based on relative braking distance, $D_{i,i-1}^{(e)}(t)$ is calculated as:

$$D_{i,i-1}^{(e)}(t)=L_i^{(s)}(t)+L_{i-1}^{(t)}+\left|\frac{v_i^2(t)}{2b_i^{(c)}}\right|-\left|\frac{v_{i-1}^2(t)}{2b_{i-1}^{(m)}}\right| \tag{5}$$

where $L_i^{(s)}(t)$ denotes the safety margin maintained between the rear of unit $i-1$ and the front of unit i after braking, incorporating equipment response latency, human reaction time, and communication delays. $L_i^{(t)}$ represents the physical length of train unit i. $b_i^{(c)}$ and $b_i^{(m)}$ are the normal service braking deceleration and maximum emergency deceleration for unit i, respectively.

Notably, in this study, to facilitate efficient and safe VC, $D_{i,i-1}^{(e)}(t)$ does not represent the minimum safe spacing $D_{i,i-1}^{(s)}(t)$ (it is permissible that: $\Delta S_{i,i-1}(t)\le D_{i,i-1}^{(e)}(t)$, while enhancing safety redundancy). The minimum safe spacing $D_{i,i-1}^{(s)}(t)$ is formally defined as:

$$D_{i,i-1}^{(s)}(t)=L_i^{(s)}(t)+L_{i-1}^{(t)}+\left|\frac{v_i^2(t)}{2b_i^{(m)}}\right|-\left|\frac{v_{i-1}^2(t)}{2b_{i-1}^{(m)}}\right| \tag{6}$$

B. CF Models Selection and Enhancement

CF, as a fundamental microscopic traffic phenomenon, characterizes the dynamic interaction mechanisms between consecutive vehicles in single-lane scenarios with overtaking is prohibited [23]. These interaction mechanisms exhibit a strong similarity to the cooperative control characteristics of VCTS—both require dynamic speed-spacing adjustments of following vehicles based on feedback of preceding vehicle states.

Consequently, CF models demonstrate significant compatibility advantages in addressing VCTS cooperative control challenges. Meanwhile, physics-based CF models exhibit broad applicability in traffic control applications due to their high computational efficiency and reproducibility. Therefore, from the dual perspectives of functional adaptability and parametric complexity, this study selects three physics-based models for integration: the Intelligent Driver Model (IDM), Cooperative Adaptive Cruise Control (CACC), and Full Velocity Difference Model (FVD).

1) IDM: As one of the most widely adopted CF models [24], the IDM features physically interpretable parameters, calibration simplicity, and broad applicability. The original IDM formulation is expressed as:

$$a_i(t)=a_i^{(m)}\left\{1-\left[\frac{v_i(t)}{v_i^{(d)}(t)}\right]^{\delta}-\left[\frac{D_{i,i-1}^{(e)}(t)}{\Delta S_{i,i-1}(t)}\right]^2\right\} \tag{7}$$

where $a_i^{(m)}$ denotes the maximum traction acceleration of train unit i. $v_i^{(d)}(t)$ represents the desired velocity of train unit i. δ is the sensitivity coefficient.

However, when $v_i(t)\to v_i^{(d)}(t)$ and $\Delta S_{i,i-1}(t)\to D_{i,i-1}^{(e)}(t)$, the model results in $a_i(t)$ approaching $-a_i^{(m)}$ instead of 0. This result violates the VC acceleration requirements, indicating that the original IDM cannot achieve VC.

To address this limitation, this study proposes an enhanced IDM by redefining the desired velocity as $v_{i-1}(t)$. The optimized IDM is formulated as:

$$a_i(t)=a_i^{(m)}\left\{2-\left[\frac{v_i(t)}{v_{i-1}(t)}\right]^{\theta_1}-\left[\frac{D_{i,i-1}^{(e)}(t)}{\Delta S_{i,i-1}(t)}\right]^{\theta_2}\right\} \tag{8}$$

where θ_1 and θ_2 are the sensitivity coefficients for velocity and displacement, respectively. To substantially enhance the model's sensitivity to relative safe spacing, this study fixed parameter in the original formulation is replaced with the tunable parameter θ_2. This modification prevents the term $\left[D_{i,i-1}^{(e)}(t)/\Delta S_{i,i-1}(t)\right]^{\theta_2}$ from reaching unity, particularly in scenarios where $\left|\Delta S_{i,i-1}(t)-D_{i,i-1}^{(e)}(t)\right|\gg 0$, which arises due to the large orders of magnitude of $\Delta S_{i,i-1}(t)$ and $D_{i,i-1}^{(e)}(t)$.

2) CACC: The CACC model, proposed by the PATH Laboratory at the University of California, Berkeley, during systematic research on traffic flow models with constant inter-vehicle time gaps [25], is formulated as:

$$\begin{aligned}a_i(t)=\theta_3 a_{i-1}(t)+\theta_4\left[\Delta S_{i,i-1}(t)-t_g v_i(t)-L_{i-1}^{(t)}\right.\\ \left.-L_i^{(s)}(t)\right]+\theta_5\left[v_{i-1}(t)-v_i(t)\right]\end{aligned} \tag{9}$$

where θ_3, θ_4, and θ_5 denote acceleration, spacing, and velocity sensitivity coefficients, respectively. t_g represents desired time headway.

To adapt to the dual-dimensional consistency control requirements of displacement and velocity in VC scenarios, this study structurally refines the original model by introducing the desired spacing $D_{i,i-1}^{(e)}(t)$ as a control target. The improved CACC model is expressed as:

$$\begin{aligned}a_i(t)=\theta_3 a_{i-1}(t)+\theta_4\left[\Delta S_{i,i-1}(t)-D_{i,i-1}^{(e)}(t)\right]\\ +\theta_5\left[v_{i-1}(t)-v_i(t)\right]\end{aligned} \tag{10}$$

This enhanced model accounts for acceleration variations of the immediately preceding train unit, thereby maintaining robust tracking performance even under non-uniform velocity conditions.

3) FVD: The FVD model proposed by Jiang et al. [26], incorporates the coupling effects of both velocity difference and displacement difference between adjacent vehicles. The governing equations of the model are expressed as:

$$a_i(t)=\theta_6\left[v_i^*(t)-v_i(t)\right]+\theta\left[v_{i-1}(t)-v_i(t)\right] \tag{11}$$

$$\theta=\begin{cases}\theta_7, & \Delta S_{i,i-1}(t)\le D_{i,i-1}^{(e)}(t)\\ 0, & \Delta S_{i,i-1}(t)>D_{i,i-1}^{(e)}(t)\end{cases} \tag{12}$$

where $v_i^*(t)$ denotes the spacing-dependent velocity function, and θ_6, θ, and θ_7 represent sensitivity coefficients.

To adapt to consistency objectives, this study proposes a novel velocity function that integrates the desired spacing:

$$v_i^*(t)=\frac{v_{i-1}(t)\left[v_{i-1}(t)+\Delta S_{i,i-1}(t)-D_{i,i-1}^{(e)}(t)\right]}{v_i(t)} \tag{13}$$

By introducing $v_i(t)$ into the denominator, this function achieves the unified control of displacement convergence precision and speed regulation magnitude, thereby improving spacing regulation precision.

However, despite refinements to IDM, CACC, and FVD models, conventional CF frameworks still exhibit inherent limitations in goal-oriented adaptability, particularly when confronted with multi-objective optimization requirements.

C. Optimization Objective Design for VCTS Cooperative Control

With the VCTS framework and CF models established, research emphasis shifts toward formulating cooperative control objectives. To ensure operational safety and cooperative control within VCTS while satisfying velocity-displacement consistency requirements, this study formulates optimization objectives incorporating energy consumption minimization, control stability enhancement, and terminal state constraints in the MPC framework. The objective function is mathematically defined as:

1) Velocity Objective:

$$J_i^{(v)} = \xi_v \sum_{j=k+1}^{k+Np} \left[v_i\left(t_k \mid t_j\right) - v_{i-1}\left(t_k \mid t_j\right) \right]^2 \tag{14}$$

2) Displacement Objective:

$$J_i^{(s)} = \xi_s \sum_{j=k+1}^{k+Np} \left[\Delta S_{i,i-1}\left(t_k \mid t_j\right) - D_{i,i-1}^{(e)}\left(t_k \mid t_j\right) \right]^2 \tag{15}$$

3) Energy Consumption Objective:

$$J_i^{(u)} = \xi_u \sum_{j=k}^{k+Np-1} \left[u_i\left(t_k \mid t_j\right) \right]^2 \tag{16}$$

4) Control Increment Objective:

$$\begin{aligned} J_i^{(\Delta u)} = \xi_{\Delta u} \sum_{j=k+1}^{k+Np-1} \Big\{ & \left[u_i\left(t_k \mid t_j\right) - u_i\left(t_k \mid t_{j-1}\right) \right]^2 \\ & + \left[u_i\left(t_{k-1}\right) - u_i\left(t_k \mid t_k\right) \right]^2 \Big\} \end{aligned} \tag{17}$$

5) MPC Terminal Constraint Objective:

$$J_i^{(term)} = \xi_{term} \left[J_i^{(v)}\left(t_k \mid t_{k+Np}\right) + J_i^{(s)}\left(t_k \mid t_{k+Np}\right) \right] \tag{18}$$

where ξ_v, ξ_s, ξ_u, $\xi_{\Delta u}$, and ξ_{term} are the weight coefficients for each term in the objective function, respectively.

Meanwhile, the train operation must also adhere to several constraints: (1) minimum safety spacing, (2) block maximum speed limit, and (3) performance criteria for each train.

Therefore, within the prediction horizon Np, the DMPC optimization objective function $J_i^{(DMPC)}$ and constraints for train unit i are formulated as:

$$\min_{U_i(t_k)} J_i^{(DMPC)} = J_i^{(v)} + J_i^{(s)} + J_i^{(u)} + J_i^{(\Delta u)} + J_i^{(term)} \tag{19}$$

Subject to:

$$\begin{cases} \Delta S_{i,i-1}\left(t_k \mid t_{j+1}\right) \geq D_{i,i-1}^{(s)}\left(t_k \mid t_{j+1}\right) \\ 0 \leq v_i\left(t_k \mid t_{j+1}\right) \leq v_m \\ b_i^{(c)} \leq u_i\left(t_k \mid t_j\right) \leq a_i^{(m)}, \\ \forall j = k, k+1, \cdots, k+Np-1 \end{cases} \tag{20}$$

where v_m denotes the block maximum speed limit.

III. Design of OPO-MCF-DMPC

Building on the enhanced CF models and optimization objectives introduced in Section II, this section details the integration of OPO, MCF, and DMPC into a unified framework. In developing this framework, it is essential to address the limitations of conventional single-model CF architectures: these rely on offline parameter calibration to passively adapt to predefined objective functions, resulting in inadequate disturbance rejection capabilities. Moreover, their adaptability to multi-objective dynamic optimization requirements in complex traffic scenarios is insufficient. In contrast, DMPC leverages its intrinsic fault-tolerant mechanisms and distributed computational advantages to effectively distribute computational loads across individual train units, thereby reducing system-level controller complexity while enhancing both control precision and real-time responsiveness.

Building upon this comparative analysis, this study integrated Online Parameter Optimization with Multiple Car-Following models (OPO-MCF) into the DMPC framework. The hybrid architecture preserved traditional CF models' computational efficiency for local dynamics while adding distributed cooperative mechanisms that enhance environmental adaptability and goal-oriented performance.

As shown in Fig. 1, under the OPO-MCF-DMPC, each train unit is equipped with an independent DMPC controller for one-to-one operational control. In this system, $x_i^*\left(t_k\right)$ represents the initial state of the train at time t_k. $U_i^{(C)}\left(t_k\right)$ denotes the control acceleration sequence for train unit i obtained through DMPC within the control horizon Nc.

Within each DMPC, $\Theta\left(t_k\right) = \left[\theta_1\left(t_k\right), \theta_2\left(t_k\right), \cdots, \theta_7\left(t_k\right)\right]^T$ is the sequence of undetermined parameters of the CF model solved by OPO. $M_i^*\left(t_k\right) = \left[m\left(t_k\right), m\left(t_{k+1}\right), \cdots, m\left(t_{k+Nc-1}\right)\right]^T$ represents the optimal real-number-encoded sequence for selecting CF models within the prediction horizon, where $m\left(t_k\right) \in \{1, 2, 3\}$ uniquely maps to the predefined models.

A. CF Models Selection Based on MSMFO

Given the critical role of CF models in controlling VCTS, selecting an optimal model structure is essential. However, the CF model selection problem presents a combinatorial optimization challenge characterized by significant non-convexity, complicating the use of conventional solvers, particularly over long prediction horizons or with high-density sampling. Consequently, this study employed heuristic optimization algorithms to mitigate computational bottlenecks and efficiently approximate optimal solutions.

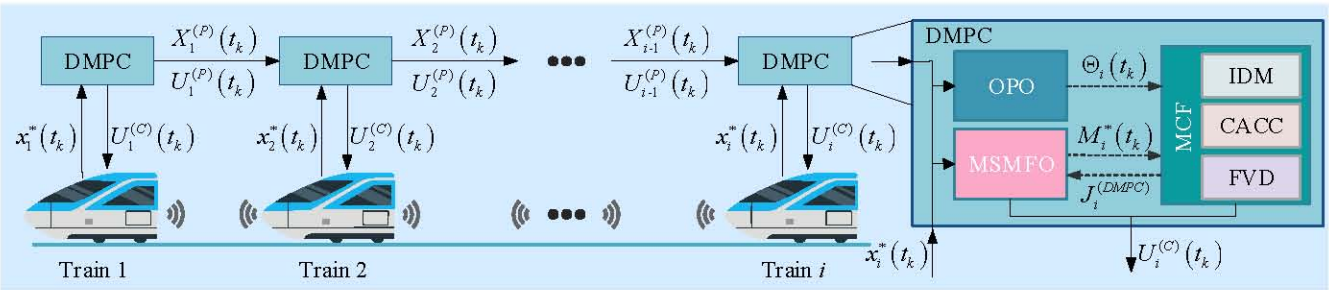

Fig. 1. Operating mechanism of OPO-MCF-DMPC.

The Moth-Flame Optimization (MFO) algorithm is a heuristic swarm intelligence method. Its design draws inspiration from moths' spiral flight patterns of moths around light sources [27]. This algorithm demonstrates strong global optimization capabilities and high solution accuracy when tackling complex constrained optimization problems with undefined search spaces, making it particularly suitable for the combinatorial optimization problem examined in this study. To further enhance the global search capability and computational accuracy of the MFO, this research introduces a Multi-Strategy Moth-Flame Optimization (MSMFO) algorithm.

1) Cyclic Boundary Rounding: Under the constraint of integer solutions, the positions of moths and flames must be discretized. However, the spiral motion of moths often results in out-of-bounds values. Direct truncation rounding can lead to solution aggregation at the boundaries, thereby diminishing the global search capability. To address this issue, this study proposes a cyclic boundary rounding method that employs modular arithmetic for the periodic mapping of out-of-bounds values. This method preserves integer solutions while avoiding boundary aggregation and maintaining population diversity. The cyclic boundary rounding formula is expressed as follows:

$$Round(m)=\left[\left(\lfloor m \rfloor - lb\right) \bmod ub\right]+lb \tag{21}$$

where $Round(m)$ denotes the rounding operation on m; lb and ub represent the lower and upper bounds of the solution, respectively.

2) Logistic-Tent Chaotic Map: The MFO demonstrates a significant dependency on the initial population distribution during its search process. To mitigate premature convergence resulting from uneven population distribution, this study utilizes the Logistic-Tent chaotic map for population initialization. This approach integrates the rapid iteration of Tent maps with the complex dynamic characteristics of the Logistic map [28]. The mapping formula is defined as:

$$C_{p+1}=\begin{cases} Round\left[r_1 C_p\left(1-C_p\right)+0.5\left(4-r_1\right)C_p\right] \\ \text{, if } C_p<0.5 \\ Round\left[r_1 C_p\left(1-C_p\right)+0.5\left(4-r_1\right)\left(1-C_p\right)\right] \\ \text{, otherwise} \end{cases} \tag{22}$$

where p is iteration index ($p=1,2,\cdots,P-1$). P represents the population size. C_p represents the mapping function value at the p-th iteration, and C_0 is a random number within [0,1]. r_1 is a random number within [0,4], representing the control parameter of logistic mapping.

3) Elite Opposition-Based Learning Strategy: To enhance search capabilities and increase population diversity, the population undergoing elite opposition-based learning is sorted by fitness value, with the top 40% selected as elite individuals $m_p^{(e)}$. These elite individuals are then processed through the using formula to generate new individuals:

$$\tilde{m}_p^{(e)}=Round\left[r_2\left(lb+ub\right)-m_p^{(e)}\right] \tag{23}$$

where $\tilde{m}_p^{(e)}$ denotes the opposition solution of $m_p^{(e)}$. r_2 is a p-dimensional random vector with each component in $[0,1]$.

Finally, the population generated through elite learning is combined with the original population and ranked according to their fitness values. The top p individuals are then selected to form the new population.

4) Adaptive Adjustment of Flame Count: In the classical MFO, the linear reduction of flame count hinders the balance of convergence performance between the initial and final stages. To address this limitation, an adaptive flame reduction strategy is proposed:

$$N_k^{(f)}=Floor\left\{N_0^{(f)}-0.5\left[\left(N_0^{(f)}-1\right)\cdot\tanh\left(0.1\left(k/K-0.5\right)N_0^{(f)}\right)+1\right]\right\} \tag{24}$$

where $Floor\{\cdot\}$ denotes the floor function for downward rounding. k represents the current iteration number. K denotes the maximum allowable iterations. $N_k^{(f)}$ indicates the flame quantity at the k-th iteration. $N_0^{(f)}$ is the initial flame quantity.

5) Adaptive Stochastic Flame Mutation: Moths exhibit a limited ability to search around flames, which makes them susceptible to local optima. Stochastic flame mutation mitigates this limitation by introducing an adaptive perturbation probability to determine both the timing and step size of mutations. The perturbation probability is defined as:

$$r_k^{(g)}=1-\sin\left(\frac{k\pi}{2K}\right) \tag{25}$$

where $r_k^{(g)}$ denotes the adaptive mutation probability at the k-th iteration. If a uniformly distributed random number $\tilde{r}_k^{(g)}\in[0,1]$ satisfies $\tilde{r}_k^{(g)}\le r_k^{(g)}$, stochastic mutation is activated.

As iterations progress, $r_k^{(g)}$ decreases, reducing mutation probability to shift focus from global exploration to local exploitation, thereby accelerating convergence. The mutation formula is:

$$\tilde{F}_k^{(n)}=Round\left[r_k^{(g)}\left(ub-lb\right)\left(1-\frac{k}{K}\right)F_k^{(n)}\right] \tag{26}$$

where $\tilde{F}_k^{(n)}$ represents the mutated position of the n-th flame ($n=1,2,\cdots,N_k^{(f)}$) at the k-th iteration.

The fitness values of mutated flames are evaluated against original values. Mutated positions replace original ones if they demonstrate superior fitness.

6) Archimedean Spiral-Based Moth Flight: The position update equation for moths based on the Archimedean spiral is formulated as:

$$S\left(M_p,F_n\right)=D_p e^{\alpha r_s}\sin\left[2\beta\pi\left(r_s-r\right)\right]+F_n \tag{27}$$

where M_p and F_n denote the positions of the p-th moth and n-th flame, respectively. $D_p=\left|F_n-M_p\right|$ represents the straight-line distance between M_p and F_n. r_s is a random number within the interval $[r,1]$, $r=-1-k/K$. α and β are spiral shape parameters, set to $\alpha=0.5$, $\beta=4$.

7) Greedy Algorithm for Refining the Solution: During iterative processes, the local search capability of the MFO algorithm gradually diminishes, while MPC control problems demand higher real-time solution efficiency. To address this, we propose a hierarchical optimization framework: MFO is first employed for global exploration, followed by a greedy algorithm for intensive refinement of optimal solutions. The

greedy algorithm updates each dimensional solution as follows:

$$\min_{m_k^{(t)} \in \{lb,\ldots,ub\}} f\left(\left[m_1^{(t-1)},\cdots,m_{k-1}^{(t-1)},m_k^{(t)},m_{k+1}^{(t-1)},\cdots,m_{Np}^{(t-1)}\right]^T\right)$$
$$,t=\{1,2,\cdots,\infty\},k\in\phi \tag{28}$$

where $f(\cdot)$ denotes the fitness function. $m_k^{(t)}$ represents the k-dimensional solution in the t-th iteration of the greedy algorithm. $\phi \sim Perm(Np)$ indicates a random permutation of the set $\{1,2,\cdots,Np\}$. This stochastic dimensional update strategy enhances the local search capability of the greedy algorithm.

The greedy algorithm terminates when consecutive iterations yield identical solutions. The overall workflow of the MSMFO is illustrated in Fig. 2.

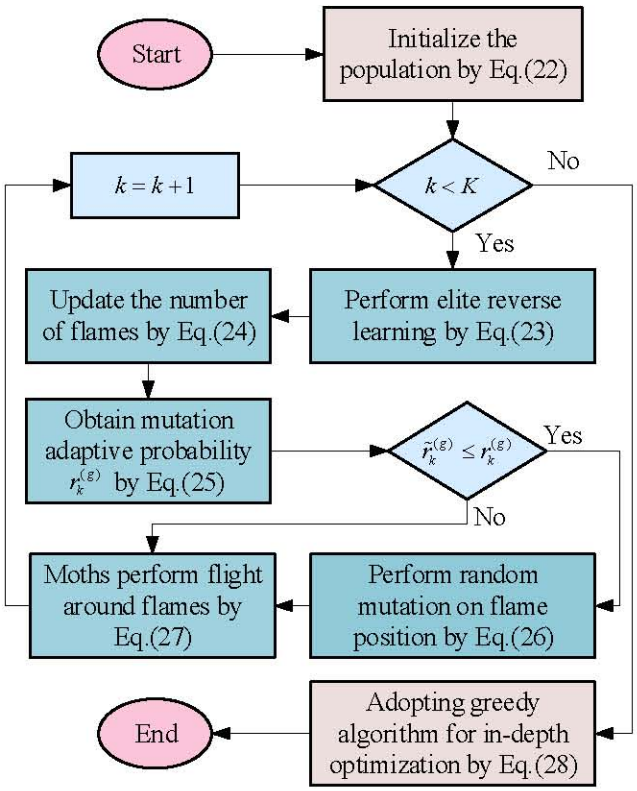

Fig. 2. Flow chart of MSMFO.

B. Online Parameter Optimization Method for CF Models

Conventional CF models predominantly rely on offline parameter calibration strategies, which introduce inherent structural limitations such as response latency and operational rigidity in dynamic environments. These deficiencies significantly degrade formation coupling robustness and situational adaptability. To mitigate these challenges, we propose a predictive control framework-based Online Parameter Optimization (OPO) methodology. The core principle involves constructing and solving a prediction horizon parameter optimization model using real-time system states, enabling dynamic self-adjustment of the mission-critical CF model's parameters. The comprehensive implementation architecture comprises the following procedures:

Step 1. At time instant t_k, obtain the current system states, including: initial state $x_i(t_k)$ of train i, predicted state sequence $X_{i-1}^{(P)}(t_k)$, and control sequence $U_{i-1}^{(P)}(t_k)$ of the preceding train $i-1$.

Step 2. Based on the acquired system states and pending parameters, solve for the predicted system state sequence $\tilde{X}_i^{(P)}(t_k)$ and control sequence $\tilde{U}_i^{(P)}(t_k)$ using the MCF:

$$\left[\tilde{X}_i^{(P)}(t_k),\tilde{U}_i^{(P)}(t_k)\right]=MCF\left[x_i(t_k);\Theta_i(t_k)\right] \tag{29}$$

where $\Theta_i(t_k)=\left[\theta_1(t_k),\theta_2(t_k),\cdots\theta_7(t_k)\right]^T$ is the sequence of pending parameters for the CF models at time t_k. $MCF[\cdot]$ represents the MCF used to solve the system state sequence and control sequence within the prediction horizon.

Based on the objective function in (19), the optimization problem can be reformulated to determine the pending parameter sequence:

$$\min_{\Theta_i(t_k)} J_i^{(DMPC)}\left\{MCF\left[x_i(t_k);\Theta_i(t_k)\right]\right\} \tag{30}$$

The constraints are maintained consistent with (20). To improve computational efficiency, the optimization problem can be decomposed into model-specific subproblems, enabling distributed execution of OPO for each model.

The OPO approach proposed within the MCF-DMPC incorporates low-dimensionality and decomposability principles, ensuring reasonable acceleration output ranges while substantially reducing computational complexity and optimization time.

IV. Simulation and Analysis

This study validates the effectiveness of the proposed collaborative control method through systematic simulation experiments. Firstly, the dynamic coupling capabilities of the three enhanced CF models are assessed. Next, the optimization effect of the MCF-DMPC integrated model on coupling performance is verified while benchmarking the MSMFO against classical intelligent optimization algorithms. Furthermore, the positive impact of the OPO mechanism on MCF-DMPC is evaluated through performance comparisons with conventional solvers, further demonstrating the effectiveness and superiority of the OPO-MCF-DMPC system. Finally, the performance of the following train was compared under different control methods when the leading train exhibited speed variations.

Assuming the simulated environment is a straight track, additional resistances are neglected. All train units in these simulations possess identical performance characteristics, with their parameters and configurations specified in Table I.

TABLE I
Performance Parameters of Train Units

Parameter	Value	Parameter	Value
τ / s	1.0	c_2 / (m^{-1})	1.0×10^{-5}
v_m / (m·s^{-1})	350/3.6	$b_i^{(m)}$ / (m·s^{-2})	-1.0
$L_i^{(s)}$ / m	$1.5v_m$	$b_i^{(c)}$ / (m·s^{-2})	-0.8
$L_i^{(t)}$ / m	203	$a_i^{(m)}$ / (m·s^{-2})	1.0
c_0 / (m·s^{-2})	1.18×10^{-2}	Np	60
c_1 / (s^{-1})	7.76×10^{-4}	Nc	5

A. CF-DMPC Simulation Results

The simulation was run on MATLAB R2024b with an Intel(R) Core (TM) i7-11800H CPU @ 2.30GHz and 32GB RAM. Three train units were initialized at positions 2000 m, 1500 m, and 0 m with initial velocities of 220 km/h, 200 km/h, and 180 km/h, respectively. Train 1 (leader) maintained constant velocity while following trains coupled with their immediate predecessors. To replicate real-world operational disturbances, control acceleration perturbations were modeled as normally distributed noise (mean = 0, SD = 0.01). The pending parameters for the CF models were defined as:

$$\begin{bmatrix} \Theta_2^T \\ \Theta_3^T \end{bmatrix} = \begin{bmatrix} 1.2 & 12.1 & 0 & 0.011 & 0.013 & 0.012 & 4.2\times10^{-4} \\ 1.4 & 16.7 & 0.35 & 0.012 & 0.015 & 0.015 & 4.3\times10^{-4} \end{bmatrix}^T$$

Using these parameters, the tracking performance of the three CF models is illustrated in Fig. 3.

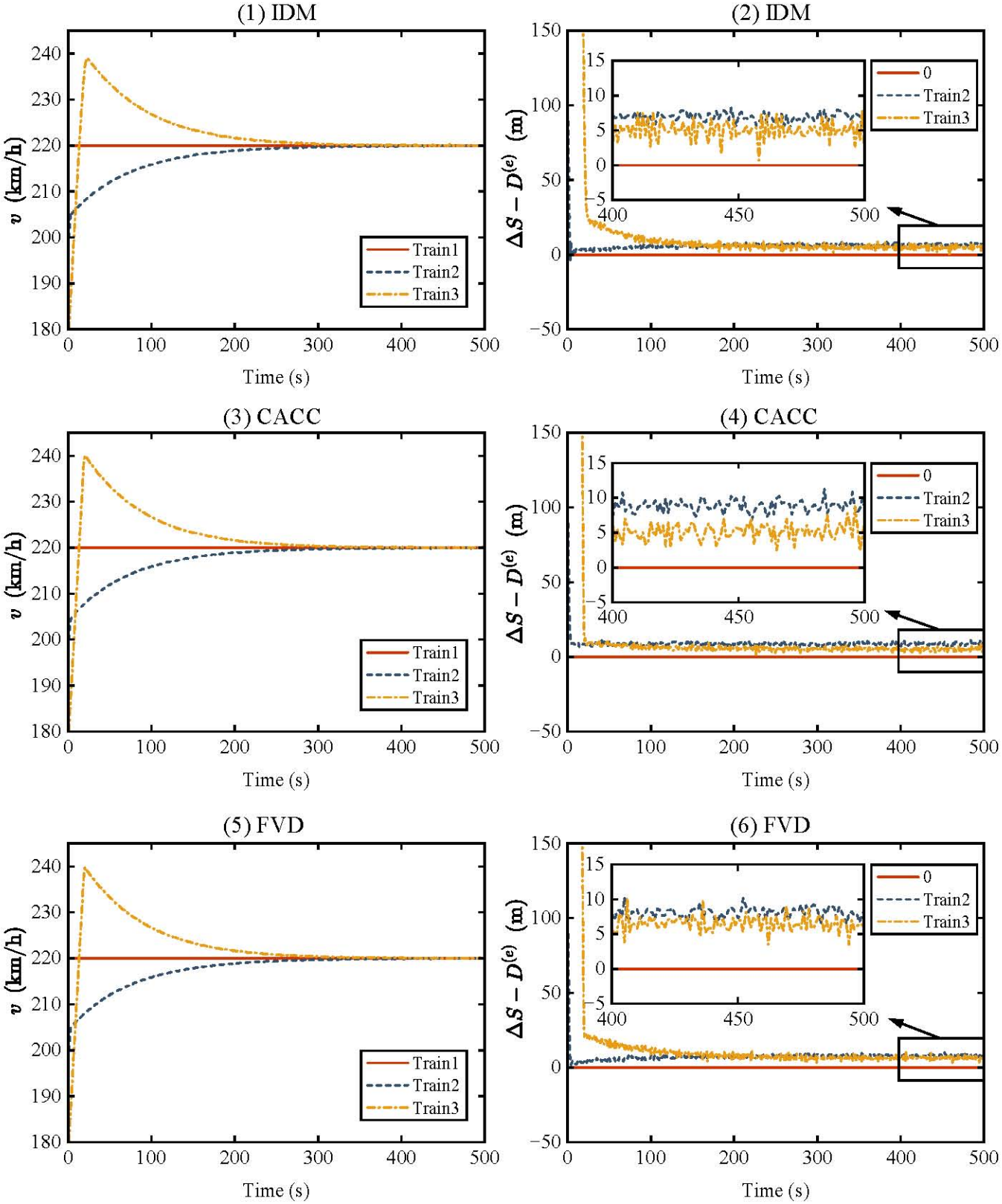

Fig. 3. Performance comparison of IDM, CACC, and FVD in terms of speed v and desired spacing deviation $\Delta S - D^{(e)}$.

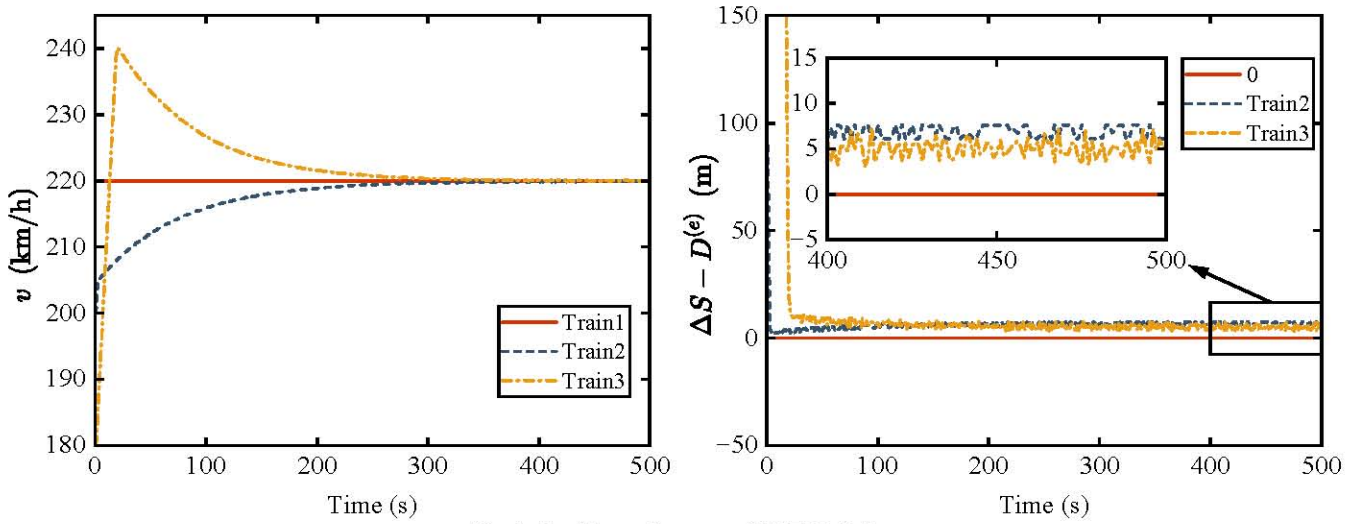

Fig. 4. Tracking performance of MCF-DMPC.

As illustrated in Fig. 3, slight performance differences existed among the three models; yet all models achieved VC functionality. Furthermore, the combination of CF model and DMPC enables the CF-DMPC method to possess a certain anti-interference ability. However, parameters of CF models require independent calibration for different train units.

B. MCF-DMPC Simulation Results

The simulation scenario remains consistent with that of the previous section, and the parameter values for the objective function and the MSMFO are specified in Table II.

TABLE II
PARAMETERS OF THE OBJECTIVE FUNCTION AND THE ALGORITHM

Parameter	Value	Parameter	Value
ξ_v	1.0	ξ_{term}	10
ξ_s	0.05	P	50
ξ_u	200	K	100
$\xi_{\Delta u}$	400	$N_0^{(f)}$	50

As illustrated in Fig. 4, the MCF-DMPC effectively enabled the train to achieve VC. Compared to single CF model approaches, the MCF demonstrated significant superiority in minimizing desired spacing deviations.

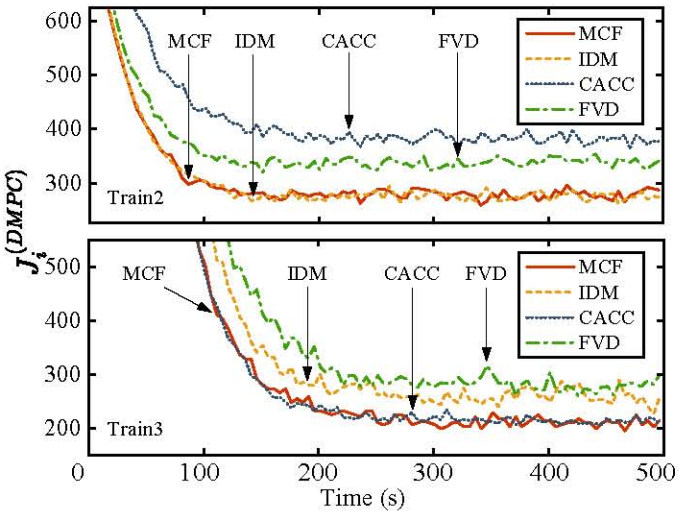

Fig. 5. Comparison of $J_i^{(DMPC)}$ among CF models and MCF.

Results in Fig. 5 show that MCF's objective function curve closely approximated the optimal model's trajectory across diverse operational scenarios. This close agreement demonstrates MCF's enhanced alignment capability with target objectives. Furthermore, it empirically validates the optimization performance of MSMFO.

To validate the superiority of MSMFO in model selection, a benchmark scenario was designed using Train 2 as the reference. The first prediction horizon of Train 2 was selected for fitness convergence analysis, with consistent parameter settings (population size $P = 50$, iteration count $K = 100$) across all comparative algorithms. To validate the performance of MSMFO, it was compared with several algorithms: the Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Simulated Annealing (SA) from the MATLAB Optimization Toolbox, and the MFO. For fair comparison, SA was configured with 5,000 iterations. The minimum fitness was recorded every 50 iterations and normalized to a scale of 100. These comparative results are shown in Fig. 6-7.

As demonstrated in Fig. 6-7, the proposed MSMFO exhibited significant advantages in computational efficiency and solution quality for CF model selection. Specifically, Fig. 6 revealed markedly accelerated convergence of the best fitness function under the identical iteration counts.

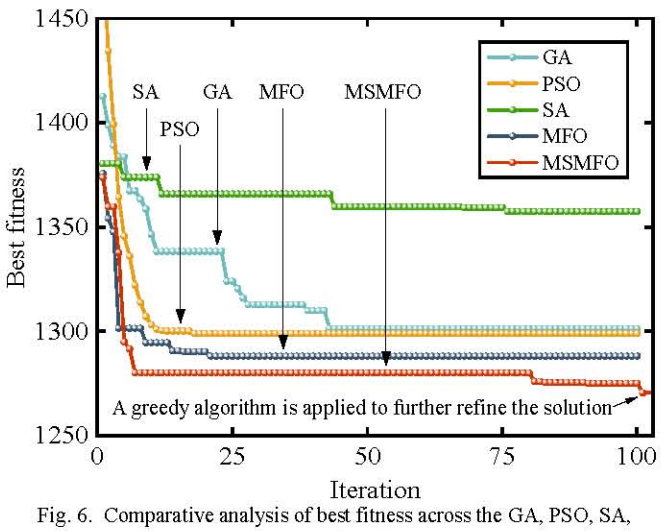

Fig. 6. Comparative analysis of best fitness across the GA, PSO, SA, MFO, and MSMFO.

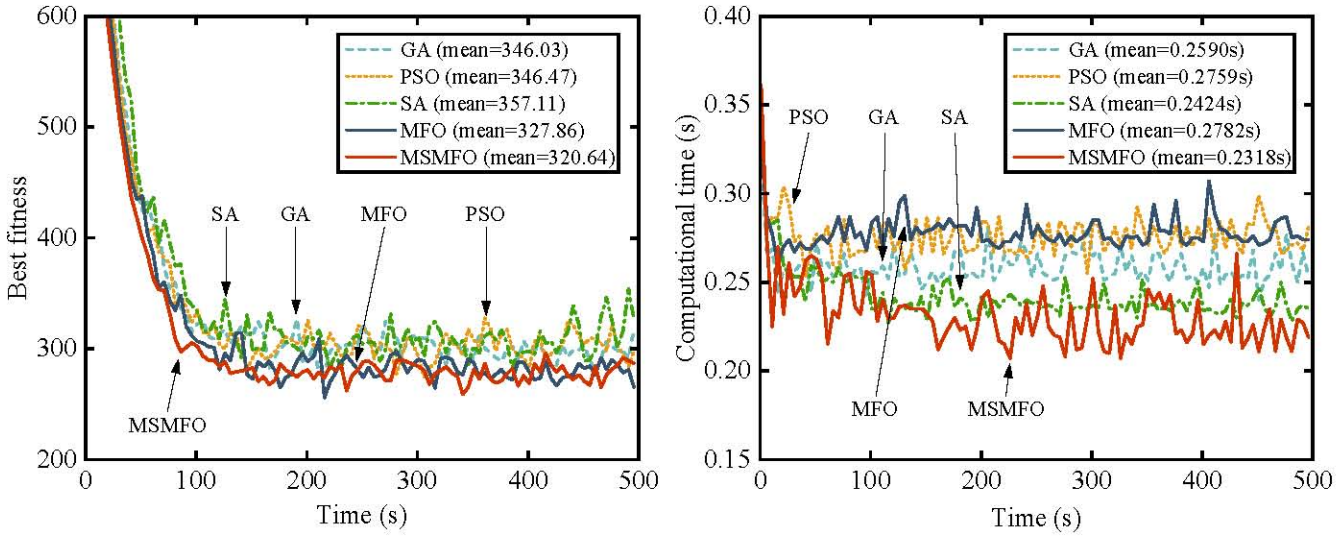

Fig. 7. Comparison of algorithms' the best fitness and computational time for solving.

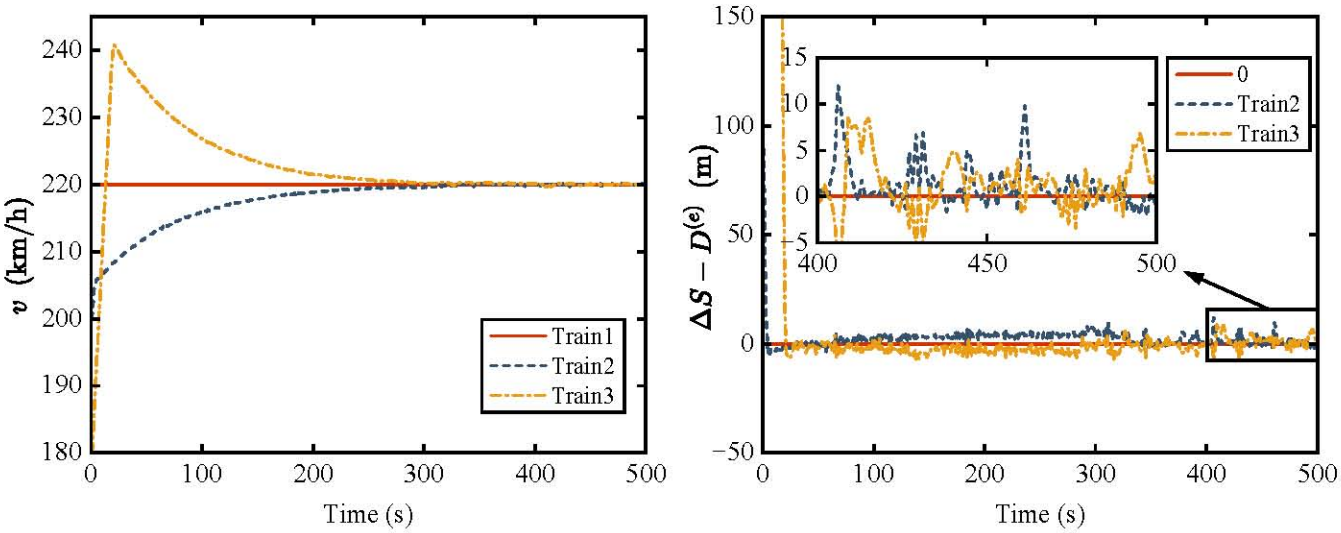

Fig. 8. Tracking performance of OPO-MCF-DMPC.

Fig. 7 demonstrates that with the equivalent parameter, MSMFO achieved a 2.2% to 10.2% improvement in the mean best fitness values and a 4.4% to 16.7% reduction in computation time compared to alternative intelligent optimization algorithms. Furthermore, Fig. 6 confirms that post-iteration refinement via the greedy algorithm facilitates thorough exploration of the solution space, further enhancing solution precision and algorithmic stability.

C. OPO-MCF-DMPC Simulation Results

System parameters remain consistent with those in the previous section, and the tracking performance of the OPO-MCF-DMPC method is illustrated in Fig. 8.

As evidenced in Fig. 8, the OPO-MCF-DMPC method was validated to possess technical feasibility for achieving VC. Empirical analysis confirmed that while exhibiting marginally higher volatility in desired spacing deviation dynamics compared to MCF-DMPC, its deviation magnitude consistently remains within safe thresholds (approximating a near-zero mean distribution) while maintaining a smaller spacing than MCF-DMPC. Furthermore, the comparative distribution of model selection frequencies between MCF-DMPC and OPO-MCF-DMPC is illustrated in Fig. 9.

Fig. 9 shows that OPO-MCF-DMPC exhibited a more balanced distribution pattern in CF model selection compared to MCF-DMPC. This observation indicates that the OPO adjusts CF model parameters to enhance adaptability.

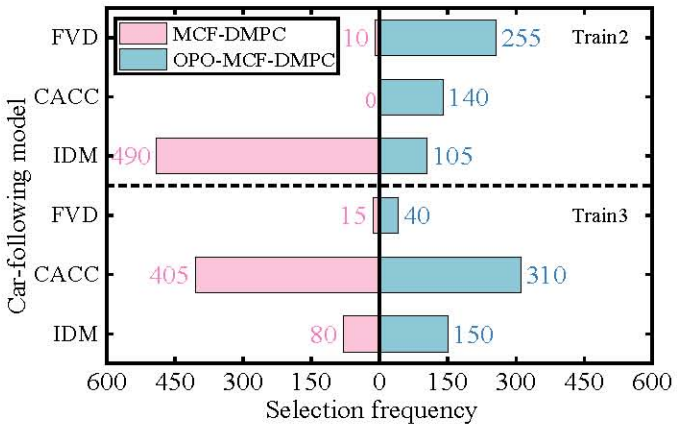

Fig. 9. Selection frequency comparison of CF models.

D. Simulation Comprehensive Comparison

Having validated the efficacy of the OPO mechanism in Sections IV-B–IV-C, this study conducted a comprehensive

performance comparison of three methods: MCF-DMPC, OPO-MCF-DMPC, and DMPC (using MATLAB's built-in fmincon solver). The comparison evaluated three metrics: (1) velocity tracking error of following trains, (2) spacing deviation from desired spacing, and (3) objective function value. The results for Trains 2 and 3 are shown in Fig. 10.

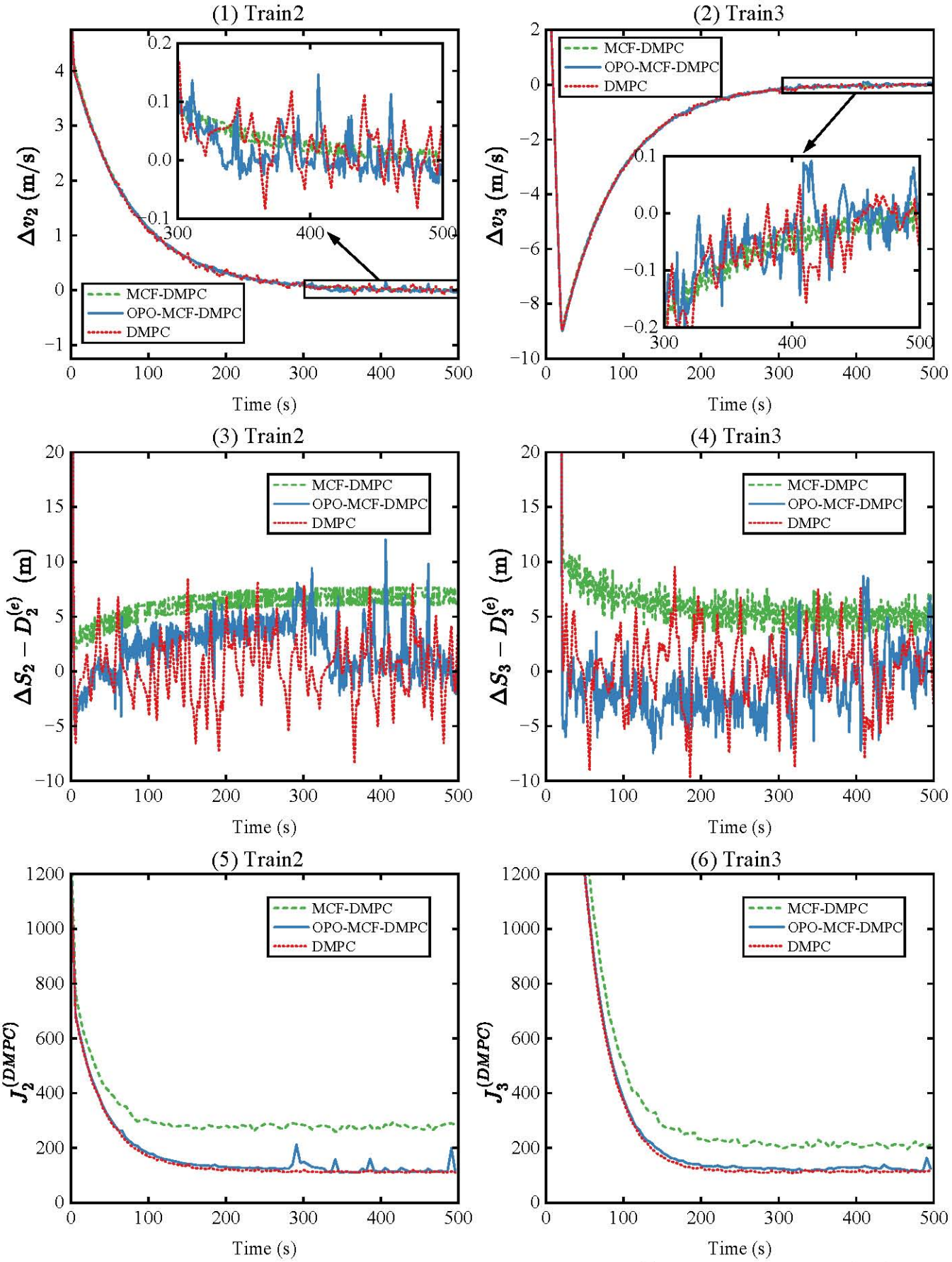

Fig. 10. Simulation comparison of velocity tracking error $\Delta v_i = v_{i-1} - v_i$, spacing deviation $\Delta S_i - D_i^{(e)}$, and objective function value $J_i^{(DMPC)}$.

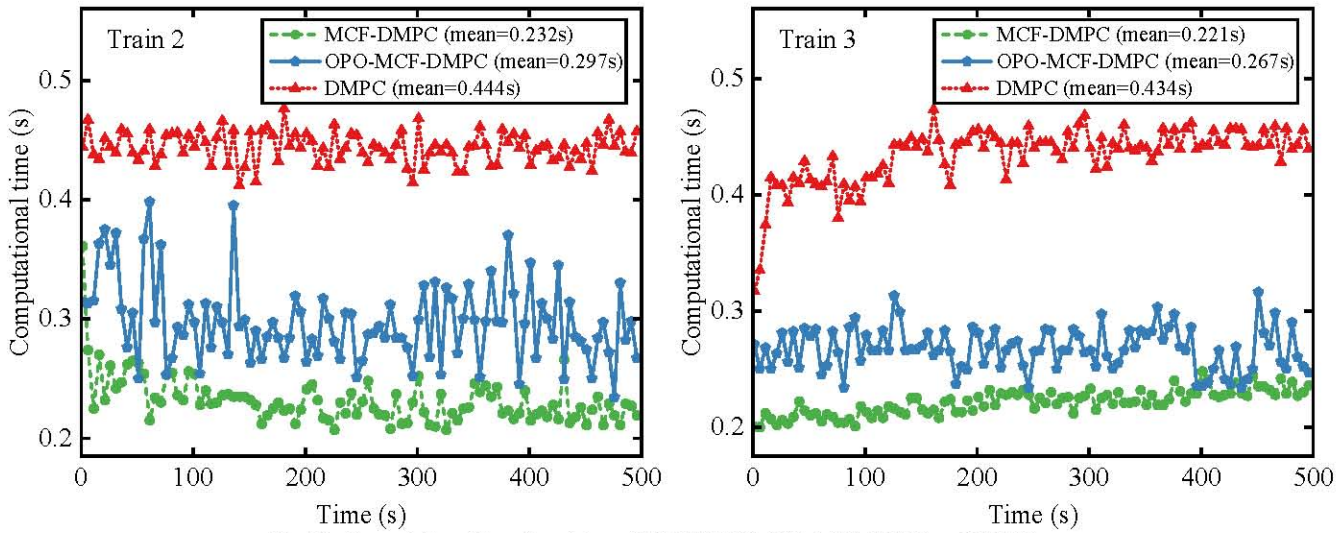

Fig. 11. Computational time for solving of MCF-DMPC, OPO-MCF-DMPC, and DMPC.

As illustrated in Fig. 10, the comparative analysis of three core performance metrics reveals distinct characteristics among the evaluated approaches. The MCF-DMPC demonstrated relatively stable yet comparatively weaker performance across all indicators. While the DMPC achieved superior fitness values, it exhibited significant fluctuations in controlling velocity deviation and spacing deviation metrics.

In contrast to the MCF-DMPC method, the proposed OPO-MCF-DMPC approach, despite demonstrating marginally lower stability than MCF-DMPC during specific operational phases, exhibited unique convergence characteristics attributed to OPO's parameter correction mechanism. Throughout mid-to-late simulation stages, both velocity deviation and spacing deviation metrics converged closely to zero, indicating superior convergence efficacy. In terms of objective function performance, the proposed OPO-MCF-DMPC demonstrates tight trajectory alignment with DMPC during early-to-mid stages. While there is a transient overshoot relative to DMPC's levels in the later phases, the values remain closely aligned overall. Notably, OPO-MCF-DMPC achieves consistent superiority over MCF-DMPC throughout the entire simulation cycle.

Although Figure 10 illustrates notable differences in objective function among the three methods, the overlapping curves impede detailed evaluation of velocity and spacing deviations, necessitating clear quantitative analysis. Table III provides consolidated results of per-train mean absolute deviations for the both full simulation and the late phase.

TABLE III
COMPARISON OF VELOCITY DEVIATION AND SPACING DEVIATION BETWEEN ENTIRE SIMULATION AND LATE PHASE

Train	Method	Velocity deviation (m/s)		Spacing deviation (m)	
		Entire Run	Late Phase	Entire Run	Late Phase
Train 2	MCF-DMPC	0.650	0.031	6.365	6.805
	OPO-MCF-DMPC	**0.644**	**0.028**	**2.967**	**2.042**
	DMPC	0.642	0.040	2.409	2.106
Train 3	MCF-DMPC	1.467	0.061	30.950	5.073
	OPO-MCF-DMPC	**1.481**	**0.056**	**27.840**	**2.218**
	DMPC	1.480	0.065	27.815	2.382

A quantitative analysis of convergence behavior, focusing on speed and spacing deviations as summarized in Table III, reveals distinct performance characteristics. Throughout the entire simulation duration, the proposed OPO-MCF-DMPC consistently demonstrates lower mean absolute deviations compared to the MCF-DMPC, with this advantage being particularly pronounced for spacing deviations. However, the DMPC approach maintains a slight, albeit marginal, performance edge over OPO-MCF-DMPC during this overall phase for both deviation metrics. In the late simulation phase, corresponding to the period when stable VC is typically established, OPO-MCF-DMPC exhibits marked superiority, achieving significantly lower mean absolute deviations in both velocity and spacing than both the baseline MCF-DMPC and the DMPC. This pronounced enhancement in convergence precision during the late phase strongly indicates that OPO-MCF-DMPC delivers exceptional tracking performance specifically under the operational conditions of established VC.

In practical engineering applications, the real-time requirements for the system are extremely stringent, and the time cost for establishing VC in a VCTS is also one of the important indicators for measuring performance. When $\left|\Delta S_{i,i-1}(t) - D^{(e)}_{i,i-1}(t)\right| \le 10$ (m) and $\left|v_i(t) - v_{i-1}(t)\right| \le 0.01$ ($\mathrm{m}\cdot s^{-1}$), it can be considered that the train i has successfully achieved VC. Thus, this article employs two core metrics—computational time and VC stabilization time—to conduct a detailed comparative analysis of the three methods, as illustrated in Fig. 11.

As shown in Fig. 11 under long-prediction-horizon conditions, MCF-DMPC achieved the shortest solution time. Although the incorporation of the OPO optimization mechanism resulted in OPO-MCF-DMPC having a slightly higher average computational time (approximately 0.055 seconds) than MCF-DMPC, its average solution time of 0.282 seconds remained markedly superior to conventional DMPC's 0.439 seconds—representing a 35.8% improvement. This performance level satisfies practical engineering requirements.

The comparison of VC establishment time is presented in Fig. 12:

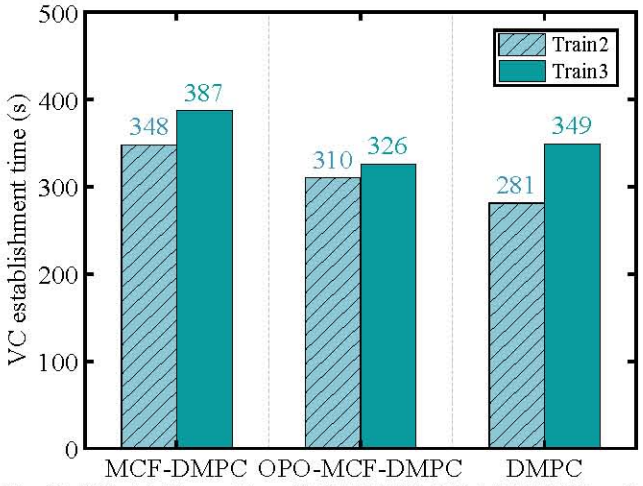

Fig. 12. VC establishment time of MCF-DMPC, OPO-MCF-DMPC, and DMPC.

As shown in Fig. 12, in terms of VC establishment time for the VCTS, both DMPC and OPO-MCF-DMPC outperformed MCF across all train units. However, the comparative performance between the DMPC and OPO-MCF-DMPC in VC establishment time exhibited train-specific dominance, with neither method universally superior across all trains.

In summary, experimental results demonstrated that the proposed OPO-MCF-DMPC collaborative control approach, achieved by integrating MCF-DMPC and OPO, significantly enhanced the collaborative control performance of VCTS. Compared with classical single CF models and standard DMPC, the proposed method demonstrated significant advantages in computational resource usage, real-time response capability, and global convergence, showcasing promising prospects for engineering applications.

E. Variable Speed Simulation

Based on the theory of VC, a strong coupling exists between the desired spacing and train operating speed. This coupling makes it challenging for VCTS to simultaneously meet the dual requirements of both velocity and displacement accuracy during continuous variable-speed operations.

Therefore, considering the frequent occurrence of such variable-speed scenarios in real-world rail operations, the following simulation case was designed to systematically validate the dynamic tracking performance of the proposed control methods under the scenarios where the leader changes speed:

The VCTS consists of a two-train formation. At the simulation start time, Train 2 (the following train) had completed VC with the leading train. Both trains were initialized to an operating speed of 220 km/h, while all other system parameters retained their initial configurations.

The speed control strategy for the leading train was designed as follows: First, it decelerated from the initial speed of 220 km/h to a target speed of 180 km/h with a braking deceleration of 0.2 m/s². Subsequently, it maintained this target speed for 100 seconds. Finally, it accelerated back to the initial speed of 220 km/h with a tractive acceleration of 0.2 m/s². The dynamic response processes of each model under this variable-speed scenario are shown in Fig. 13.

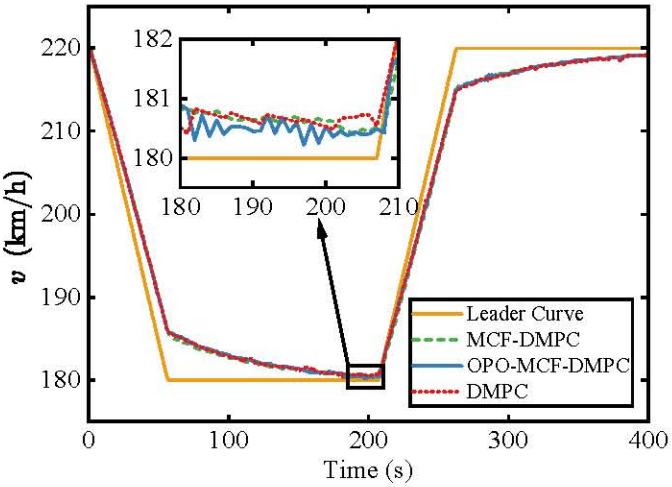

Fig. 13. Comparison of speeds of various methods under a variable-speed scenario.

Fig. 13 reveals that the velocity curves of all models exhibit high consistency. Notably, the OPO-MCF-DMPC maintained excellent tracking precision even during speed transition phases.

Further analyzing performance differences across methods, Fig. 14 presents a comparative analysis of velocity tracking errors and relative spacing deviations under variable-speed operational scenarios.

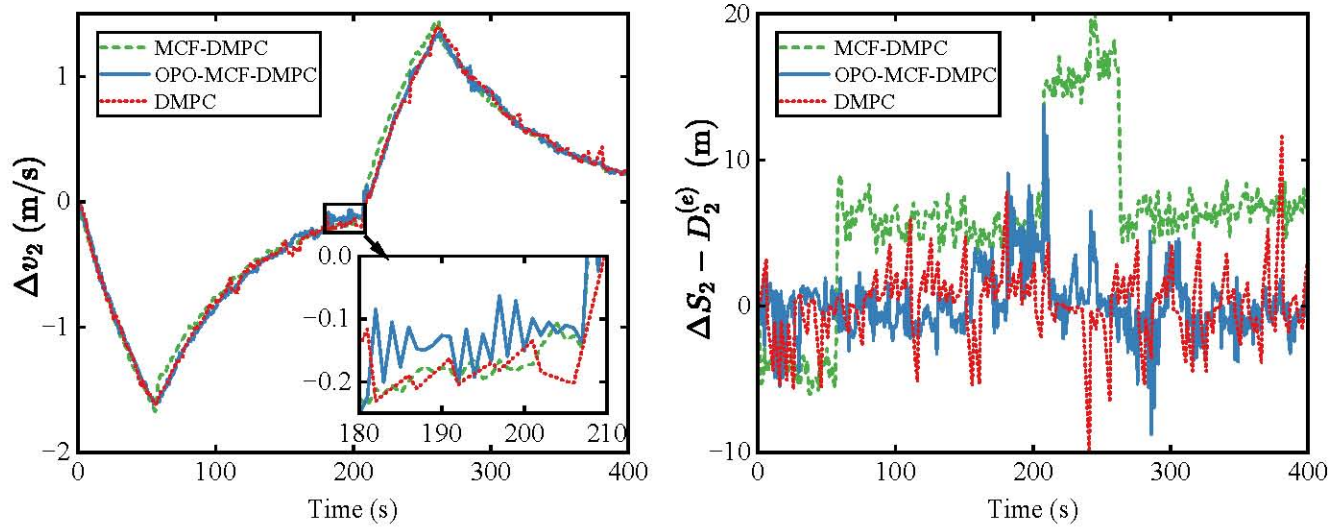

Fig. 14. Comparison of velocity tracking error $\Delta v_2 = v_1 - v_2$ and spacing deviation $\Delta S_2 - D_2^{(e)}$ among various methods.

As illustrated in Fig. 14, the velocity deviation curves of all methods exhibited similar dynamic characteristics; however, they demonstrated differentiated performance in spacing control. Specifically, the MCF-DMPC exhibited marginally higher spacing deviations compared to other methods, while both OPO-MCF-DMPC and DMPC demonstrated superior spacing maintenance capabilities, with all deviations remaining consistently within the 5 m threshold.

To comprehensively quantify dynamic performance disparities across control strategies, Table IV presents the mean absolute velocity and spacing deviations recorded over the entire test cycle for all evaluated methods.

TABLE IV
COMPARISON OF VELOCITY DEVIATION AND SPACING DEVIATION UNDER THE VARIABLE-SPEED OPERATIONAL SCENARIOS

Method	Velocity deviation (m/s)	Spacing deviation (m)
MCF-DMPC	0.677	7.153
OPO-MCF-DMPC	**0.674**	**1.628**
DMPC	0.680	1.747

The quantitative evidence in Table IV clearly demonstrates the consistent superiority of OPO-MCF-DMPC over both MCF-DMPC and conventional DMPC in velocity deviation and spacing deviation. These findings conclusively establish the enhanced efficacy of the OPO-MCF-DMPC architecture in preserving operational stability and tracking precision during complex variable-speed operations.

V. CONCLUSION

To address the complex dynamic requirements of VCTS in distributed cooperative control systems, an integrated OPO-MCF-DMPC method was established in this study. Key contributions are summarized as follows:

(1) Three CF models (IDM, CACC, and FVD) have been refined to better satisfy speed and displacement consistency objectives in VC scenarios. The enhanced models, through synergistic integration with the DMPC framework, strengthen the system's adaptability to complex dynamic environments and deliver higher-precision foundational model support for VCTS control.

(2) A MSMFO algorithm incorporating chaotic mapping initialization, adaptive flame quantity adjustment, and stochastic mutation strategies has been developed. This significantly enhances the global search capability and computational precision of conventional MFO. Simulation results demonstrate that MSMFO improves fitness values by 2.2%–10.2% and reduces computational time by 4.4%–16.7% compared to traditional algorithms (GA, PSO, SA, and MFO) when addressing CF model selection problems.

(3) Integrated within the MCF-DMPC framework, the OPO mechanism significantly enhances VCTS coordination control. By dynamically calibrating CF model parameters, it overcomes static parameter limitations inherent to conventional approaches. Simulation-verified results demonstrate closer alignment between train states and desired states, with significant improvements in coordination precision and environmental adaptability across constant- and variable-speed operations.

(4) Comparative analysis confirms OPO-MCF-DMPC achieves 35.8% faster computational efficiency than DMPC while demonstrating superior control accuracy in both velocity and spacing deviations relative to MCF-DMPC, with objectives comparable to DMPC. This validates the method's responsiveness to real-time engineering requirements.

This research advances theoretical VCTS control methodologies and introduces novel implementation frameworks. The proposed strategy delivers substantial gains in computational efficiency and environmental resilience, thereby enhancing operational safety and performance. Future work will develop neural network-enhanced OPO mechanisms to enable systematic advancement and scaled deployment of VC technology.

REFERENCES

[1] Y. Dun, W. ShangGuan, H. Song, and B. Cai, "Time-space-based virtual coupling high-speed train separation model and trajectory planning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 18573–18590, 2024.

[2] Z. Hou, R. He, C. Liu, S. Wan, K. Yang, and C. Dai, "Optimization of passenger and freight collaborative transportation for urban rail transit under virtual coupling condition," *Engineering Letters*, vol. 32, no. 1, pp. 179–192, 2024.

[3] S. Su, W. Liu, Q. Zhu, R. Li, T. Tao, and J. Lv, "A cooperative collision-avoidance control methodology for virtual coupling trains," *Accident Analysis & Prevention*, vol. 173, pp. 106703–106714, 2022.

[4] Q. Zhang *et al.*, "A Novel Vibration-Based Self-Adapting Method to Acquire Real-Time Following Distance for Virtually Coupled Trains," *IEEE/CAA Journal of Automatica Sinica*, vol. 12, no. 1, pp. 27–39, 2025.

[5] C. Di Meo, M. Di Vaio, F. Flammini, R. Nardone, S. Santini, and V. Vittorini, "ERTMS/ETCS virtual coupling: proof of concept and numerical analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2545–2556, 2020.

[6] Q. Wu, X. Ge, Q.-L. Han, and Y. Liu, "Railway virtual coupling: a survey of emerging control techniques," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 5, pp. 3239–3255, 2023.

[7] Y. Ji, E. Quaglietta, R. M. P. Goverde, and D. Ou, "An artificial potential field approach for virtual coupling train control with complete braking curve supervision," *Transportation Research Part C: Emerging Technologies*, vol. 173, pp. 105050–105078, 2025.

[8] J. Felez and M. A. Vaquero-Serrano, "Virtual coupling in railways: a comprehensive review," *Machines*, vol. 11, no. 5, pp. 521–558, 2023.

[9] E. Quaglietta, M. Wang, and R. M. P. Goverde, "A multi-state train-following model for the analysis of virtual coupling railway operations," *Journal of Rail Transport Planning & Management*, vol. 15, pp. 100195–100212, 2020.

[10] X. Wang, K. Xing, J. Wang, and W. Zheng, "Predictive control algorithm for speed and displacement tracking of urban rail trains," *IAENG International Journal of Applied Mathematics*, vol. 54, no. 7, pp. 1362–1370, 2024.

[11] D. Guo, Z. Zhang, H. Zhou, and X.-B. Chen, "Finite-time sliding mode control of vehicle formations considering performance constraints," *Engineering Letters*, vol. 33, no. 7, pp. 2610–2619, 2025.

[12] J. Pan, Q. Peng, S. Zhan, and J. Bai, "Multiscenario-based train headway analysis under virtual coupling system," *Journal of Advanced Transportation*, vol. 2021, pp. 1–20, 2021.

[13] B. Shuai, J. Luo, X. Feng, and W. Huang, "A train group control method based on car following model under virtual coupling," *Journal of Transportation Systems Engineering and Information Technology*, vol. 24, no. 3, pp. 1–11, 2024.

[14] J. Xun, J. Yin, R. Liu, F. Liu, Y. Zhou, and T. Tang, "Cooperative control of high-speed trains for headway regulation: A self-triggered model predictive control based approach," *Transportation Research Part C: Emerging Technologies*, vol. 102, pp. 106–120, 2019.

[15] S. Su, J. She, K. Li, X. Wang, and Y. Zhou, "A nonlinear safety equilibrium spacing-based model predictive control for virtually coupled train set over gradient terrains," *IEEE Transactions on Transportation Electrification*, vol. 8, no. 2, pp. 2810–2824, 2022.

[16] Z. He, Z. Hou, N. Xu, D. Liu, and M. Zhou, "A distributed model predictive control approach for virtually coupled train set with adaptive mechanism and particle swarm optimization," *Mathematics*, vol. 13, no. 10, pp. 1641–1660, 2025.
[17] X. Liu, A. Dabiri, Y. Wang, J. Xun, and B. De Schutter, "Distributed model predictive control for virtually coupled heterogeneous trains: comparison and assessment," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 12, pp. 20753–20766, 2024.
[18] Y. Liu, R. Liu, C. Wei, J. Xun, and T. Tang, "Distributed model predictive control strategy for constrained high-speed virtually coupled train set," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 1, pp. 171–183, 2022.
[19] J. Li *et al.*, "Distributed robust model predictive control for virtual coupling under structural and external uncertainty," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 8751–8769, 2024.
[20] Z. Li, L. Zhong, and H. Yang, "Distributed cooperative predictive control of virtual coupled trains," *Journal of Traffic and Transportation Engineering*, vol. 24, no. 5, pp. 362–378, 2024.
[21] S. Matrone, A. Pozzi, E. Ogliari, and S. Leva, "Deep learning-based predictive control for optimal battery management in microgrids," *IEEE Access*, vol. 12, pp. 141580–141593, 2024.
[22] J. Lin and M. Ni, "Adaptive model predictive control of virtual coupled based on artificial potential field," *Journal of Beijing University of Aeronautics and Astronautics*, pp. 1–23, 2023.
[23] M. Brackstone and M. McDonald, "Car-following: a historical review," *Transportation Research Part F: Traffic Psychology and Behaviour*, vol. 2, no. 4, pp. 181–196, 1999.
[24] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, 2000.
[25] J. VanderWerf, S. Shladover, N. Kourjanskaia, M. Miller, and H. Krishnan, "Modeling effects of driver control assistance systems on traffic," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1748, no. 1, pp. 167–174, 2001.
[26] R. Jiang *et al.*, "Experimental and empirical investigations of traffic flow instability," *Transportation Research Procedia*, vol. 23, pp. 157–173, 2017.
[27] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-Based Systems*, vol. 89, pp. 228–249, 2015.
[28] S. Rajendran and M. Doraipandian, "A nonlinear two dimensional logistic-tent map for secure image communication," *International Journal of Information and Computer Security*, vol. 10, no. 2/3, pp. 201–215, 2018.