

Technical report 24-021

Distributed model predictive control for virtually coupled heterogeneous trains: Comparison and assessment*

X. Liu, A. Dabiri, Y. Wang, J. Xun, and B. De Schutter

If you want to cite this report, please use the following reference instead:

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, pp. 20753–20766, 2024. doi:[10.1109/TITS.2024.3458169](https://doi.org/10.1109/TITS.2024.3458169)

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: <https://www.dcsc.tudelft.nl>

* This report can also be downloaded via https://pub.bartdeschutter.org/abs/24_021

Distributed Model Predictive Control for Virtually Coupled Heterogeneous Trains: Comparison and Assessment

Xiaoyu Liu, Azita Dabiri, Yihui Wang, Jing Xun, and Bart De Schutter, *Fellow, IEEE*

Abstract—Virtual coupling is regarded as an efficient way to improve the line capacity of rail transportation systems by reducing the spacing between consecutive trains. This paper is the first to compare and assess different distributed model predictive control (MPC) approaches, i.e., cooperative distributed MPC, serial distributed MPC, and decentralized MPC, for virtually coupled trains with a nonlinear train dynamic model. To make a balanced trade-off between computational complexity and efficiency, we also propose and assess convex approximations of the above control approaches. Furthermore, we are the first to introduce the relaxed dynamic programming approach to analyze the stability of the MPC-based nonlinear train control problem. By using the relaxed dynamic programming approach, a distributed stopping criterion with a stability guarantee is developed for the cooperative distributed MPC approach. In real life, masses of trains are different and can change at stations due to changes in passenger loads. This change in mass can significantly affect the dynamics and control of the virtually coupled trains when not taken into account in the control design. Therefore, we explicitly consider heterogeneous train masses when designing MPC approaches. We evaluate the different distributed MPC approaches through case studies based on the data of the Beijing Yizhuang Line. Simulation results indicate that the cooperative distributed MPC approach has the best tracking performance, while the serial distributed MPC approach can reduce communication requirements and computation capabilities with sacrifices of tracking performance.

Index Terms—Virtual coupling, train speed control, distributed model predictive control, heterogeneous train masses, relaxed dynamic programming

I. INTRODUCTION

The transport demand for rail transportation systems has increased rapidly, and the need to enhance rail line capacity while ensuring operational safety remains a paramount concern for rail operators. The line capacity is associated with the spacing between consecutive trains, which is determined by the signal systems. Currently, the widely applied signal

This research has received funding from the National Natural Science Foundation of China under Grant (No.62073026, 72071016), the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 101018826 - CLariNet). The work of the first author is also supported by the China Scholarship Council under Grant 202007090003.

Xiaoyu Liu, Azita Dabiri, and Bart De Schutter are with the Delft Center for Systems and Control, Delft University of Technology, 2628CD, Delft, The Netherlands (email: x.liu-20@tudelft.nl, a.dabiri@tudelft.nl, b.deschutter@tudelft.nl).

Yihui Wang and Jing Xun are with the School of Automation and Intelligence, Beijing Jiaotong University, Beijing 100044, China (email: yihui.wang@bjtu.edu.cn, jxun@bjtu.edu.cn).

system in urban rail transit is the moving block system [1], [2], which determines the distance between two consecutive trains based on the absolute braking distance, i.e., the distance a train needs to fully stop from its current speed.

In recent years, an advanced signaling technology, i.e., virtual coupling, has been recognized as an efficient way to further improve the line capacity by reducing the spacing between consecutive trains [3], [2]. In a platoon of virtually coupled trains, the distance between two consecutive trains is determined based on the relative braking distance, which also takes into account the braking characteristics of the predecessor train [4], [5]. Different from platoons of connected and automated vehicles (CAVs) in road traffic [6], a platoon of virtually coupled trains features a long train braking distance, and trains in a platoon should run on the same rail track, leading to larger spacing between trains. Furthermore, the communication between non-adjacent trains is typically not considered due to the longer headway in railway systems as communication over longer distances may become unreliable [7]. Hence, one cannot just adopt control approaches of CAVs to virtually coupled trains.

As a novel signaling technology, virtual coupling significantly relies on vehicle-to-vehicle communication and cooperative train control schemes [8], [9], [10]. Generally, the communication topology and the cooperative control schemes are highly intertwined. Several control approaches have been developed for virtually coupled trains based on different communication topologies. Cao *et al.* [11] applied generalized predictive control (GPC) to virtually coupled trains with the aim to ensure the expected tracking distance and to prevent collisions. Xun *et al.* [12] applied model predictive control (MPC) to realize centralized control and in addition they developed a speed protection mechanism for virtually coupled trains. Su *et al.* [13] developed a centralized MPC approach for virtually coupled trains in the cruising phase, and they applied a generalized minimum-residual-method-based approach to solve the resulting nonlinear optimization problem. The above papers focus on centralized control approaches that rely on a centralized controller, thereby significantly increasing the communication and computation burden [14].

In contrast to those centralized control approaches, decentralized control strategies have gained attention due to their potential to alleviate the communication and computation burden. Felez *et al.* [15] formulated a decentralized MPC approach for virtually coupled trains based on a linear model with nonlinear constraints. Two cases are considered in [15],

i.e., the case that the follower receives predicted states from its predecessor train, and the case that the followers have to predict the states of its predecessor train based on the measured information. Considering uncertainties in the dynamic model and train positioning, Felez *et al.* [16] developed a decentralized robust MPC approach based on the min-max principle. This work is further extended in [17] by including more uncertain factors, such as modeling errors, positioning errors, communication delays, and possible adhesion losses. Di Meo *et al.* [18] developed a decentralized control approach based on local state variables and the information received from other trains, and they analyzed the exponential stability under communication delays by introducing a Lyapunov-Krasovskii function. By using sliding mode control (SMC) and a nonlinear train control model, Park *et al.* [19] developed a robust gap controller based on the measurement of the position and velocity of the predecessor trains. Basile *et al.* [20] developed a deep deterministic policy gradient approach to design a decentralized control law for virtually coupled trains with heterogeneous train dynamics and uncertain disturbances, showing lower computational burden and energy consumption compared to MPC. However, the safety distance in [20] is considered by using a penalty term in the reward function, which does not provide a theoretical guarantee of safety. The above papers primarily emphasize the significance of decentralized control strategies for virtual coupling, highlighting their ability to alleviate the communication burden while ensuring system performance. However, these decentralized approaches often rely on measurement information or limited communication information, and trains make independent decisions without coordinating their actions with those of other trains, highlighting the potential for distributed and/or cooperative control approaches¹ that can leverage communication data even more.

The advanced vehicle-to-vehicle communication technology enables communication-enhanced information exchange between virtually coupled trains [7], [8], prompting the exploration of distributed control methods that can leverage more extensive communication data. Quaglietta *et al.* [23] analyzed the safety margin of virtually coupled trains to handle the safety risk caused by communication delays, control delays, positioning errors, and train braking characteristics. Su *et al.* [24] considered the heterogeneous train braking distance and developed a feedback control law to ensure the string stability of the train platoon. Liu *et al.* [5] linearized the train movement model and developed a distributed MPC approach for a platoon of virtually coupled trains, where trains are assumed to be close to each other, and therefore the slope difference between different trains is ignored; then, they analyzed the local stability of each individual train based on a terminal controller. By ignoring the slope difference between trains, Liu *et al.* [25] developed an optimal control (OC) approach based on Pontryagin's principle, and analyzed the local stability and the head-to-tail string stability. By considering the resistance caused by tracks and winds as bounded disturbances, Luo *et al.* [26] introduced tube-based distributed MPC based on a linear

train model, where the safety constraint can be ensured in any situation in the robust control scheme. In the aforementioned distributed control approaches, each train computes its control input based on the information received from its predecessor train only, and thus the approach is also called the serial distributed control approach. Zhang *et al.* [27] introduced the fixed-time tracking control (FTC) approach and developed a cooperative control approach to achieve virtual coupling within the fixed time. Wang *et al.* [28] introduced a Q-learning-based cooperative control approach for virtually coupled trains where monitoring sensors and wireless communication networks are used to obtain the operational status of trains; however, only two virtually coupled trains are considered in [28], and the extension to more trains still requires further research. In summary, these studies indicate the potential for enhanced control and coordination among virtually coupled trains facilitated by vehicle-to-vehicle communication technologies.

In a set of virtually coupled trains, trains may have different characteristics, resulting in heterogeneity. In particular, heterogeneous trains may have different lengths, masses, and braking characteristics, which should be considered in the controller design to ensure efficient and safe operation [4], [24]. Train mass is a crucial factor influencing train dynamics and varies according to train type and passenger load. Therefore, as an illustration, among all the aspects of heterogeneous trains, we focus on the train mass in the current paper.

Table I summarizes the aforementioned studies, outlining the differences in the model, control scheme, control approach, and train heterogeneity they used. From Table I, we can observe the application of both linear and nonlinear train dynamic models. Notably, the nonlinear model generally yields more accurate results but also comes with a higher computational burden compared to the linear model. According to different communication topologies, different control schemes, i.e., centralized, decentralized, distributed, and cooperative distributed, are studied. We find that MPC stands out as the most widely adopted train control approach in virtual coupling research. For more studies in virtual coupling, we refer to the recent review papers [10], [8], [29]. It is worth noting that only the study presented in [19] explicitly incorporates train masses when designing the controller, and there is still no research on an MPC design for virtually coupled trains explicitly considering masses of trains. Furthermore, a comprehensive comparison and assessment considering different models and different control schemes for virtually coupled trains is still unaddressed in the existing literature.

The paper contributes to the state of the art as follows:

- 1) A comprehensive comparison and assessment of distributed MPC approaches for virtually coupled trains are provided, which would benefit the process of control method design and selection for virtually coupled trains.
- 2) We are the first to incorporate the relaxed dynamic programming (RDP) approach into the train control field and to use it to ensure the stability of the nonlinear train control problem. By using RDP, a stopping criterion under the distributed control scheme with a stability guarantee is developed for the cooperative distributed MPC approach.

¹Each agent in a distributed control scheme only focuses on its own objective, while cooperative distributed control enables agents to take into account the objective of the overall system [21], [22].

TABLE I
SUMMARY OF STUDIES ON CONTROL FOR VIRTUALLY COUPLED TRAINS

Literature	Model	Control scheme	Control approach	Train heterogeneity
Cao <i>et al.</i> (2021) [11]	linear	centralized	generalized predictive control	no
Xun <i>et al.</i> (2020) [12]	linear	centralized	model predictive control	no
Su <i>et al.</i> (2021) [13]	nonlinear	centralized	model predictive control	no
Felez <i>et al.</i> (2019) [15]	nonlinear	decentralized	model predictive control	no
Felez <i>et al.</i> (2022) [16]	nonlinear	decentralized	model predictive control	no
Vaquero-Serrano <i>et al.</i> (2023) [17]	nonlinear	decentralized	model predictive control	no
Di Meo <i>et al.</i> (2019) [18]	linear	decentralized	proportional–integral–derivative	no
Park <i>et al.</i> (2020) [19]	nonlinear	decentralized	sliding mode control	train mass
Basile <i>et al.</i> (2024) [20]	nonlinear	decentralized	deep deterministic policy gradient	train dynamics
Liu <i>et al.</i> (2021) [5]	linear	distributed	model predictive control	no
Liu <i>et al.</i> (2021) [25]	linear	distributed	optimal control	no
Luo <i>et al.</i> (2023) [26]	linear	distributed	model predictive control	no
Su <i>et al.</i> (2023) [24]	nonlinear	distributed	feedback control	braking dynamics
Zhang <i>et al.</i> (2021) [27]	nonlinear	cooperative	fixed-time tracking control	no
Wang <i>et al.</i> (2020) [28]	nonlinear	cooperative	Q-learning	no
This study	linear, nonlinear	cooperative, distributed, decentralized	model predictive control	train mass

3) The mass of trains can significantly affect the dynamics and control of virtually coupled trains if not considered in the control design. We are the first to explicitly account for changes in train masses when designing MPC approaches, and we demonstrate the impact of incorporating train masses in the control design through simulations.

The rest of the paper is structured as follows. In Section II, the problem statement and preliminaries are provided. In Section III, the mathematical model of the system is provided. In Section IV, several distributed MPC approaches are presented. In Section V, we conducted case studies to illustrate the performance of the approaches, and in Section VI, the conclusions and the outlook for future works are provided.

II. PROBLEM STATEMENT AND PRELIMINARIES

A. Problem Statement

In a platoon of virtually coupled trains, trains are coupled virtually through train-to-train communication. We consider heterogeneous trains, and in particular, we focus on heterogeneous masses in this paper. The leader train receives reference signals from the infrastructure and operates following a reference speed profile, and each follower train follows its predecessor train while keeping a safe distance.

Let us define s_i , v_i , and u_i as the position, speed, and control input of train i , respectively, and define Δs_i and Δv_i as the position difference and speed difference between train i and its predecessor train, respectively. As stated in [7], ultra-reliable low-latency communications are typically required when the distance between trains is less than 50 m. Moreover, the latency of 50 ms can be achieved for wireless train control and monitoring system [30], [7]. The field tests and simulations in [31] also indicated that the average transmission delay of train-to-train communications is below 20 ms. Therefore, in the current paper, we only consider the case that a train can communicate with its predecessor train and follower train, and train-to-train communication under a relatively short distance can be ensured. As shown in Fig. 1, three possible communication topologies realized

in practice are considered, i.e., bidirectional communication, unidirectional communication, and measurement, and different communication topologies require different control methods. The bidirectional communication in Fig. 1(a) allows trains to include their neighbors' real-time control inputs, speeds, and positions when generating control inputs. Hence, trains can compute their control inputs in parallel and exchange information with their neighbors [7], [27], which involves adjusting control inputs, to achieve cooperative control; however, the communication burden of bidirectional communication is relatively large.

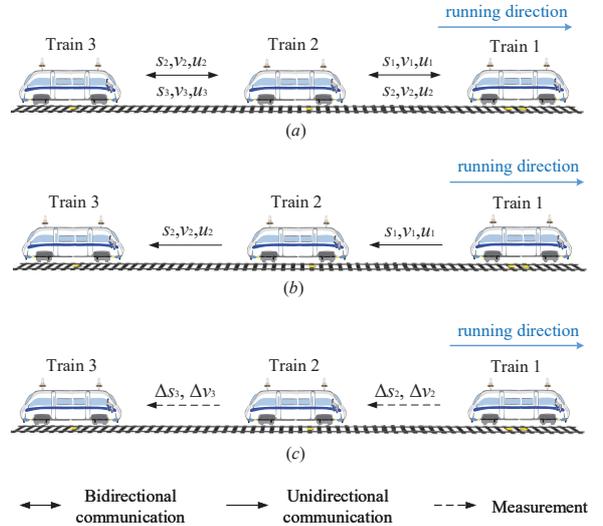


Fig. 1. Illustration of train-to-train communication topologies for virtually coupled trains occurring in practice.

For the unidirectional communication in Fig. 1(b), trains compute control inputs sequentially in the virtual coupled train string: each train computes the control input based on the real-time control input, speed, and position, received from its predecessor train, and then, the computed control input, speed, and position are sent to its successor train. In this context, each train only requires communicating with its neighbors once per

control step.

Fig. 1(c) corresponds to the case when the communication between trains is lost, and thus a train cannot receive the real-time control input, speed, and position of its predecessor train. Then, to ensure safe operation, each train should compute control inputs based on the relative speed and position of its predecessor train measured by onboard sensors, e.g., radars or LiDARs, assuming the predecessor train may brake with the maximum braking force.

In this paper, we consider the three communication topologies depicted in Fig. 1 and introduce different control approaches based on the three communication topologies.

Remark 1. Different from platoons of connected automated vehicles in road traffic, communication between non-adjacent trains is not considered due to the longer headway in railway systems compared to that in road traffic control systems, as communication over longer distances may become unreliable.

The virtual coupling train control problem aims at controlling trains operating with a relatively short headway while ensuring a safe and steady distance between two adjacent trains. The safety distance can be guaranteed by including hard constraints in the control problem. The steady distance between a train and its predecessor train is evaluated by local stability, while the steady distance between any two adjacent trains in the platoon is ensured by the so-called string stability.

B. Preliminaries

To introduce the concept of string stability, let us consider train i in the platoon, and the dynamic of train i is

$$x_{i,k+1} = f_i(x_{1,k}, \dots, x_{i,k}, \dots, x_{I,k}), \quad (1)$$

where $x_{i,k}$ represents the state of train i at time step k , I is the total number of trains in the platoon.

The definitions of local stability and string stability used in this paper are introduced as follows.

Definition 1 (*Lyapunov Local Stability*) [32]: For a given system (1), the equilibrium point x_i^{eq} is said to be Lyapunov local stable if

$$\forall \epsilon > 0, \exists \delta > 0, \|x_{i,0} - x_i^{\text{eq}}\| < \delta \Rightarrow \|x_{i,k} - x_i^{\text{eq}}\| < \epsilon, \forall k \in \mathbb{N}_0. \quad (2)$$

In addition, the equilibrium point x_i^{eq} is said to be asymptotically Lyapunov locally stable if it is Lyapunov locally stable and $x_{i,k} \rightarrow x_i^{\text{eq}}$ as $k \rightarrow \infty$.

Let us further define the dynamic platoon of trains as

$$x_{k+1} = f(x_k), \quad (3)$$

where $x_k = [x_{1,k}, \dots, x_{I,k}]^\top$.

Definition 2 (*Lyapunov String Stability*) [33], [34]: For a platoon of trains described by (3), the equilibrium point x^{eq} is said to be Lyapunov string stable if

$$\forall \epsilon > 0, \exists \delta > 0, \|x_0 - x^{\text{eq}}\| < \delta \Rightarrow \|x_k - x^{\text{eq}}\| < \epsilon, \forall k \in \mathbb{N}_0. \quad (4)$$

In addition, the equilibrium point x^{eq} is said to be asymptotically Lyapunov string stable if it is Lyapunov string stable and $x \rightarrow x^{\text{eq}}$ asymptotically.

Notation: A continuous function $h(\cdot) : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is of class \mathcal{K} , if it is strictly increasing and $h(0) = 0$. A continuous function $h(\cdot) : \mathbb{R}_+ \rightarrow \mathbb{R}_+$ is of class \mathcal{K}_∞ , if it is of class \mathcal{K} and $\lim_{u \rightarrow \infty} h(u) = \infty$. The quadratic norm corresponding to a positive definite symmetric matrix Q is $\|x\|_Q^2 = x^\top Q x$. Given a set $\mathbb{X} \subseteq \mathbb{R}^n$, a scalar $a \in \mathbb{R}$, we define $a\mathbb{X} := \{ax | x \in \mathbb{X}\}$.

III. MATHEMATICAL MODEL FOR VIRTUALLY COUPLED TRAINS

A. Train Dynamic Model

Although the dynamics of a train is continuous, the control input of the automatic train operation (ATO) system is typically implemented in a discrete-time manner due to the implementation of digital computers. Similar to [15], [35], the discrete-time model of longitudinal dynamics of a train can be described as

$$v_{i,k+1} = v_{i,k} + \frac{(u_{i,k} - r_i(v_{i,k}) - w_i(s_{i,k}))T}{M_{i,p}}, \quad (5a)$$

$$s_{i,k+1} = s_{i,k} + v_{i,k}T, \quad (5b)$$

where i is the train index, T represents the sampling time, $v_{i,k}$ and $s_{i,k}$ represent the speed and position of train i at time step k , respectively, $M_{i,p}$ denotes the total mass of train i from station p to its successor station with p being the station index. We assume that $M_{i,p}$ is a piecewise constant function whose value changes at the station in accordance with the variance of the passenger load. Moreover, $u_{i,k}$ is the control input, i.e., the traction/braking force; $r_i(v_{i,k})$ is the basic resistance that is related to the speed of train i ; $w_i(s_{i,k})$ denotes the additional resistance that is determined by the position of train i .

The total mass of train i changes when train i has arrived at a station and can be calculated by

$$M_{i,p} = m_i + n_{i,p}m_{\text{pa}}, \quad (6)$$

where m_i denotes the mass of train i itself; $n_{i,p}$ is the number of passengers on board the train at station p , and the value of $n_{i,p}$ changes when the train has arrived at a station; m_{pa} represents the average mass of a passenger.

The train basic resistance $r_i(v_{i,k})$ can be described by

$$r_i(v_{i,k}) = M_{i,p} (c_0 + c_1 v_{i,k} + c_2 v_{i,k}^2), \quad (7)$$

where c_0 , c_1 , and c_2 are parameters that can be identified based on experiment data [36]. The train basic resistance considers the effects caused by the rotational inertia for wheelsets, the number of axles, the effective frontal cross-section, the air resistance, etc.

The additional resistance $w_i(s_{i,k})$ is related to the total mass of the train and can be approximated as a piecewise constant function of the train position:

$$w_i(s_{i,k}) = M_{i,p} g \sin \theta(s_{i,k}), \quad (8)$$

where $\theta(\cdot)$ is a function of train position representing slope at the corresponding position².

²The additional resistance consists of the resistance caused by slope, curve, and tunnel. Note that the curve resistance and the tunnel resistance can be represented by $w_i(s_{i,k}) = M_{i,p} g \gamma(s_{i,k})$, with $0 \leq \gamma(s_{i,k}) \leq 1$; so they can be transformed into the form of (8).

The decision variable $u_{i,k}$, the train speed $v_{i,k}$, and the train position $s_{i,k}$ should satisfy

$$-B_i^{\text{sb}} \leq u_{i,k} \leq U_i^{\text{max}}, \quad (9)$$

$$0 \leq v_{i,k} \leq v_{\text{lim}}(s_{i,k}), \quad (10)$$

$$s_{i,k} + d_i^{\text{safe}}(v_{i,k}, v_{i-1,k}) \leq s_{i-1,k}, \quad (11)$$

where B_i^{sb} and U_i^{max} are the maximum service braking force and the maximum traction force of train i , respectively, $v_{\text{lim}}(s_{i,k})$ is a piecewise constant function denoting the speed limit for train i at position $s_{i,k}$, $d_i^{\text{safe}}(v_{i,k}, v_{i-1,k})$ is the safety distance between train i and its predecessor train, which can be constrained by

$$d_i^{\text{safe}}(v_{i,k}, v_{i-1,k}) \geq d_i^{\text{sb}}(v_{i,k}) - d_{i-1}^{\text{eb}}(v_{i-1,k}) + L + D_{\text{safe}}, \quad (12a)$$

$$d_i^{\text{safe}}(v_{i,k}, v_{i-1,k}) \geq L + D_{\text{safe}}, \quad (12b)$$

where $d_i^{\text{sb}}(v_{i,k}) = \frac{v_{i,k}^2}{2a_i^{\text{sb}}}$ is the braking distance of train i with the service braking, i.e., when $a_i^{\text{sb}} = \frac{B_i^{\text{sb}}}{M_{i,p}}$, $d_{i-1}^{\text{eb}}(v_{i-1,k}) = \frac{v_{i-1,k}^2}{2a_{i-1}^{\text{eb}}}$ is the braking distance of train $i-1$ with emergency braking, i.e., $a_{i-1}^{\text{eb}} = \frac{B_{i-1}^{\text{eb}}}{M_{i-1,p}}$, where B_{i-1}^{eb} is the emergency braking force of train $i-1$, L denotes the length of a train, and D_{safe} is the safety distance applied to address the safety risk caused by modeling errors, positioning errors, communication delays, etc [23], [5].

B. Dynamic Model for Virtually Coupled Trains

In a platoon of virtually coupled trains, a train is expected to follow its predecessor train at a certain distance. We consider that the relative distance between train i ($i > 1$) and its predecessor train is determined by the speeds of the two trains:

$$e_{i,k} = s_{i-1,k} - s_{i,k} - d_i^{\text{sb}}(v_{i,k}) + d_{i-1}^{\text{eb}}(v_{i-1,k}). \quad (13)$$

The first train ($i = 1$) tracks a desired speed profile with the speed and position represented by $v_{0,k}$ and $s_{0,k}$ respectively, and we define $e_{1,k} = s_{0,k} - s_{1,k}$. The illustration of calculating $e_{i,k}$ in (13) is shown in Fig. 2.

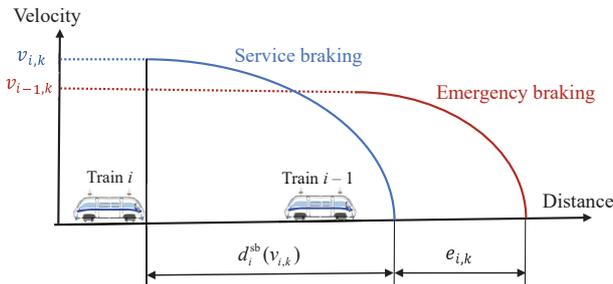


Fig. 2. Illustration of calculating relative distance for train i ($i > 1$).

Let us define the state and input of train i as $x_{i,k} = [v_{i,k}, e_{i,k}]^T$ and $\mu_{i,k} = \frac{1}{M_{i,p}}u_{i,k}$, respectively. Then, the evolution of $x_{i,k}$ can be expressed compactly as

$$x_{i,k+1} = f_i(x_{i,k}, \mu_{i,k}), \quad (14)$$

where $x_{i,k} \in \mathbb{X}_{i,k}$ and $\mu_{i,k} \in \mathbb{W}_{i,k}$, with $\mathbb{X}_{i,k}$ and $\mathbb{W}_{i,k}$ being the feasible sets of $x_{i,k}$ and $\mu_{i,k}$, respectively. If $u_{i,k} \in \mathbb{U}_{i,k}$, then $\mathbb{W}_{i,k} = \frac{1}{M_{i,p}}\mathbb{U}_{i,k}$. Note that, in (14), the states of train i implicitly depend on the position and speed of train $i-1$; hence we have coupled dynamics.

IV. DISTRIBUTED MODEL PREDICTIVE CONTROL FOR VIRTUALLY COUPLED TRAINS

In this section, we apply different distributed MPC approaches for virtually coupled trains based on the nonlinear model in Section III. We first provide the general nonlinear model predictive control problem formulation. Considering the possible communication structures introduced in Fig. 1, the computational complexity, and the model accuracy, we then develop the following six distributed MPC approaches:

- nonconvex cooperative distributed MPC: N-CDMPC;
- convex cooperative distributed MPC: C-CDMPC;
- nonconvex serial distributed MPC: N-SDMPC;
- convex serial distributed MPC: C-SDMPC;
- nonconvex decentralized MPC: N-DMPC;
- convex decentralized MPC: C-DMPC;

N-CDMPC, N-SDMPC, and N-DMPC are related to the bidirectional communication case, the unidirectional communication case, and the measurement case in Fig. 1, respectively. However, as the model (14) and constraints (11), (12), and (13) are nonlinear, the resulting MPC optimization problems of N-CDMPC, N-SDMPC, and N-DMPC are nonlinear and nonconvex, which may increase the computational burden of finding the optimal solution. Hence, we approximate these problems as convex problems to make a balanced trade-off between computational burden and accuracy, and the convex counterparts of the methods are labeled C-CDMPC, C-SDMPC, and C-DMPC, respectively. The details of the above approaches are provided as follows.

A. General Nonlinear MPC Problem Formulation

To ensure that trains run with consistent speed and steady distance, we define the quadratic stage cost for train i at time step k as

$$\ell_i(x_{i,k}, \mu_{i,k}) = \|x_{i,k} - x_{i,k}^{\text{eq}}\|_Q^2 + \|\mu_{i,k}\|_R^2, \quad (15)$$

where $Q \in \mathbb{R}^{2 \times 2}$ is a positive symmetric matrix, and $R \in \mathbb{R}$. The first term in (15) defines the tracking error, while the second term corresponds to the energy consumption of train i . Virtual coupling aims to control trains running with consistent speed and steady relative distance. Thus, the equilibrium state of train i ($i > 1$) is defined as $x_{i,k}^{\text{eq}} = [v_{i,k}^{\text{eq}}, e_{i,k}^{\text{eq}}]^T$ with $v_{i,k}^{\text{eq}} = v_{i-1,k}$ and $e_{i,k}^{\text{eq}} = L + D_{\text{des}}$, where L is the length of a train, D_{des} represents the desired distance between two trains. The equilibrium state of the first train ($i = 1$) is $x_{1,k}^{\text{eq}} = [v_{0,k}, 0]^T$.

The nonlinear MPC optimization problem for train i is

$$\min_{\substack{x_{i,k_0} \\ \mu_{i,k_0}}} J_i(k_0) = \sum_{k=k_0}^{k_0+N-1} \ell_i(x_{i,k}, \mu_{i,k}) \quad (16a)$$

$$\text{s.t. } x_{i,k+1} = f(x_{i,k}, \mu_{i,k}), \quad k = k_0, \dots, k_0+N-1, \quad (16b)$$

$$x_{i,k} \in \mathbb{X}_{i,k}, \quad k = k_0, \dots, k_0+N-1, \quad (16c)$$

$$\mu_{i,k} \in \mathbb{W}_{i,k}, \quad k = k_0, \dots, k_0+N-1, \quad (16d)$$

where $x_{i,k_0} = [x_{i,k_0}^\top, \dots, x_{i,k_0+N}^\top]^\top$ and $\mu_{i,k_0} = [\mu_{i,k_0}, \dots, \mu_{i,k_0+N-1}]^\top$, $\mathbb{X}_{i,k}$ denotes the set defined by constraints (10)-(12), and $\mathbb{W}_{i,k}$ represents the set defined by constraint (9).

The optimization problem (16) is a nonlinear nonconvex optimization problem. Solving (16) at time step k_0 results in the optimized input sequence $\mu_{i,k_0}^* = [\mu_{i,k_0}^*, \dots, \mu_{i,k_0+N-1}^*]$; only the first value μ_{i,k_0}^* is implemented in the system and the procedure is repeated under a moving horizon scheme.

From (16), we formulate a nonlinear model predictive controller, and the stability can be analyzed based on relaxed dynamic programming. The stability condition can be stated as follows.

Theorem 1 (Lyapunov Stability) [37]. Considering system (14) with $x_{i,k} \in \mathbb{X}_{i,k}$, let $\mathbb{X}_{i,k}$ be forward invariant³, and let $\pi_i(\cdot)$ be an admissible control law, i.e., $\pi_i(x_{i,k}) \in \mathbb{W}_{i,k}$, $\forall x_{i,k} \in \mathbb{X}_{i,k}$ such that $f_i(x_{i,k}, \pi_i(x_{i,k})) \in \mathbb{X}_{i,k+1}$. Then, the closed-loop system $x_{i,k+1} = f_i(x_{i,k}, \pi_i(x_{i,k}))$ is asymptotically stable on $\mathbb{X}_{i,k}$ with the equilibrium point $x_{i,k}^{\text{eq}}$ if

$$J_i^N(k) \geq \alpha \ell_i(x_{i,k}, \mu_{i,k}^*) + J_i^N(k+1), \quad (17a)$$

$$\beta_1(\|x_{i,k} - x_{i,k}^{\text{eq}}\|_2) \leq J_i^N(k) \leq \beta_2(\|x_{i,k} - x_{i,k}^{\text{eq}}\|_2), \quad (17b)$$

$$\ell_i(x_{i,k}, \mu_{i,k}) \geq \beta_3(\|x_{i,k} - x_{i,k}^{\text{eq}}\|_2), \quad (17c)$$

where $J_i^N(k)$ represents the optimized value of $J_i(k)$ at time step k with prediction horizon N , for some $\alpha \in (0, 1]$, and $\beta_1(\cdot)$, $\beta_2(\cdot)$, and $\beta_3(\cdot)$ are of class \mathcal{K}_∞ .

Remark 2. Note that $J_i^N(k+1)$ in condition (17a) requires the control law in time step $k+1$, which is not available at time step k . In the MPC scheme, given the optimized control variables at time step k as $\mu_{i,k}^* = [\mu_{i,k}^{*\top}, \dots, \mu_{i,k+N}^{*\top}]^\top$, we can directly build a sequence of feasible control variables for time step $k+1$ as

$$\begin{aligned} \tilde{\mu}_{i,k+1} &= [\tilde{\mu}_{i,k+1}^\top, \dots, \tilde{\mu}_{i,k+N}^\top, \tilde{\mu}_{i,k+N+1}^\top]^\top \\ &= [\mu_{i,k+1}^{*\top}, \dots, \mu_{i,k+N}^{*\top}, \tilde{\mu}_{i,k+N+1}^\top]^\top, \end{aligned} \quad (18)$$

where $\tilde{\mu}_{i,k+1}^\top, \dots, \tilde{\mu}_{i,k+N}^\top$ are the inputs in $\mu_{i,k}^*$, and $\tilde{\mu}_{i,k+N+1}^\top$ can be any admissible control law, e.g., $\mu_{i,k+N} = -\frac{B_{i,p}^{\max}}{M_{i,p}}$. The cost function for $\tilde{\mu}_{i,k+1}$ is represented by $P_i^N(k+1)$. Thus, we can obtain optimized decision variables at time step $k+1$, such that $J_i^N(k+1) \leq P_i^N(k+1)$. Then, the implementable version of (17a) becomes

$$J_i^N(k) \geq \alpha \ell_i(x_{i,k}, \mu_{i,k}^*) + P_i^N(k+1). \quad (19)$$

³A family of sets $\mathbb{X}_{i,k}$ is forward invariant if there exists $\mu_{i,k}$ such that $x_{i,k+1} = f_i(x_{i,k}, \mu_{i,k}) \in \mathbb{X}_{i,k+1}$ holds for all $x_{i,k} \in \mathbb{X}_{i,k}$.

B. Nonconvex Cooperative Distributed MPC

With the bidirectional communication as in Fig. 1(a), trains in a platoon of virtually coupled trains can compute control inputs in parallel and exchange information several times to achieve cooperative control. The alternating direction method of multipliers (ADMM) is an efficient distributed optimization approach for problems with coupled constraints [38]. Therefore, we adopt ADMM to solve the resulting distributed optimization problem in each step of distributed MPC.

For the MPC optimization problem of train i , (16c) and (16d) collect constraints for $x_{i,k}$ and $\mu_{i,k}$, and we can write (16c) and (16d) compactly as:

$$h_i(y_{i-1,k}, y_{i,k}, y_{i+1,k}) \leq E_{1,i,k}, \quad (20)$$

where $y_{i-1,k} = [x_{i-1,k}^\top, \mu_{i-1,k}^\top]^\top$, $y_{i,k} = [x_{i,k}^\top, \mu_{i,k}^\top]^\top$, $y_{i+1,k} = [x_{i+1,k}^\top, \mu_{i+1,k}^\top]^\top$, and $E_{1,i,k}$ is a constant. We can observe from (16) that different subproblems are coupled through constraint (20). The coupled constraints can be relaxed by introducing $\eta_{i,k} \geq 0$ as follows:

$$h_i(y_{i-1,k}, y_{i,k}, y_{i+1,k}) + \eta_{i,k} = E_{1,i,k}. \quad (21)$$

Then, in ADMM, the objective function for train i becomes

$$\begin{aligned} \mathcal{L}_i(k_0) &= J_i(k_0) + \\ &+ \sum_{k=k_0}^{k_0+N-1} \left(\lambda_{i,k}^\top (h_i(y_{i-1,k}, y_{i,k}, y_{i+1,k}) + \eta_{i,k} - E_{1,i,k}) + \right. \\ &\left. + \frac{\rho}{2} \|h_i(y_{i-1,k}, y_{i,k}, y_{i+1,k}) + \eta_{i,k} - E_{1,i,k}\|_2^2 \right), \end{aligned} \quad (22)$$

where $\mathbf{y}_{i,k_0} = [y_{i,k_0}^\top, \dots, y_{i,k_0+N-1}^\top]^\top$, $\rho > 0$ is the augmented Lagrangian parameter, and $\lambda_{i,k}$ represents the Lagrangian multipliers, which are updated by

$$\lambda_{i,k}^{(z+1)} = \lambda_{i,k}^{(z)} + \rho \left(h_i(y_{i-1,k}^{(z)}, y_{i,k}^{(z)}, y_{i+1,k}^{(z)}) + \eta_{i,k} - E_{1,i,k} \right), \quad (23)$$

where z represents the iteration index, and $\lambda_{i,k}^{(z)}$ and $y_{i,k}^{(z)}$ are the values of $\lambda_{i,k}$ and $y_{i,k}$ after iteration z , respectively. For more details about ADMM, we refer the readers to [38], [39].

In each iteration, a nonlinear nonconvex optimization problem should be solved. We can use gradient-based approaches, e.g., sequential quadratic programming, to find a solution. ADMM is a distributed optimization approach, and a stopping criterion that can be applied in a distributed manner is required when implementing ADMM in the distributed control scheme.

Lemma 1. If $\mathcal{L}_i^N(k)$ represents the optimized value of $\mathcal{L}_i(k)$, one sufficient condition for (17a) in the distributed control scheme is

$$\mathcal{L}_i^N(k) \geq \alpha \ell_i(x_{i,k}, \mu_{i,k}^*) + P_i^N(k+1), \quad (24)$$

where $\alpha \in (0, 1]$.

Proof: Based on the weak duality theorem, we have

$$\mathcal{L}_i^N(k) \leq J_i^N(k). \quad (25)$$

Then, according to (19), we have

$$J_i^N(k) \geq \alpha \ell_i(x_{i,k}, \mu_{i,k}^*) + P_i^N(k+1). \quad (26)$$

Hence, we can conclude that (24) implies

$$J_i^N(k) \geq \alpha \ell_i(x_{i,k}, \mu_{i,k}^*) + J_i^N(k+1). \quad (27)$$

The iteration of ADMM for N-CDMPC stops when either the stability condition represented by (24) is satisfied, or the maximum number of iterations Z_{\max} is reached. Based on the aforementioned stopping criteria, ADMM may terminate before reaching its (local) optimal solution. To ensure safe operations, the safety constraints (11)-(12) can be directly incorporated as a constraint when optimizing (22), and the coupled constraint (13) is relaxed by (22).

Lemma 2 (Recursive Feasibility). If a feasible solution that satisfies the stopping criterion (24) is found at time step k , the feasibility for the optimization problem (16) of each agent at time step $k+1$ can be found.

Proof: The proof is based on finding a feasible solution for time step $k+1$. For a solution $\mu_{i,k}^*$ at time step k , a feasible solution at time step $k+1$ can be found as stated in Remark 2. ■

Theorem 2 (Lyapunov String Stability). If a feasible solution that satisfies the stopping criterion (24) can be found, then the platoon of virtually coupled trains is Lyapunov string stable.

Proof: If a feasible solution that satisfies the stopping criterion (24) can be found, according to Theorem 1, we can show that the equilibrium point of each train is Lyapunov stable. Then, the Lyapunov string stability for the platoon of virtually coupled trains can be obtained following the procedure in [33]. ■

Algorithm 1 elaborates the procedure for implementing the cooperative distributed MPC algorithm, where z is the iteration index, and $x_{i,k}^{(z)}$ and $\mu_{i,k}^{(z)}$ represent the values of $x_{i,k}$ and $\mu_{i,k}$ after iteration z , respectively.

Algorithm 1 Cooperative Distributed MPC for Virtually Coupled Trains

Input: x_{i,k_0} , $M_{i,p}$, N , U^{\max} , B^{sb} , B^{eb} , I_{train} , D_{safe} , D_{des} , L , k_{end} , z_{\max} , ρ , $\lambda_{i,k}^{(0)}$, α ; recommend speeds $v_{0,k}$, $s_{0,k}$;

Output: control input $\mu_{i,k}$

```

1:  $k \leftarrow k_0$ 
2: repeat
3:    $q \leftarrow 0$ 
4:   repeat
5:     for  $i = 1, \dots, I_{\text{train}}$  do
6:       minimize objective (22) subject to (5)-(12)
7:       send obtained  $x_{i,k}^{(z+1)}$  and  $\mu_{i,k}^{(z+1)}$  to neighbors
8:       update  $\lambda_{i,k}^{(z+1)}$  subject to (23)
9:     end for
10:     $z \leftarrow z + 1$ 
11:   until  $z = z_{\max}$  or (24) holds for each train  $i$ 
12:   apply control decision  $\mu_{i,k}$  to each train  $i$ 
13:    $k \leftarrow k + 1$ 
14: until  $k = k_{\text{end}}$ 

```

C. Convex Cooperative Distributed MPC

The problem (16) formulated in Section IV-A is a nonlinear nonconvex optimization problem. In the N-CDMPC approach developed in Section IV-B, we cannot ensure the convergence of ADMM and the optimal solution to the optimization problem easily. Moreover, solving nonlinear nonconvex optimization problems typically requires a larger computational burden than its convex counterpart.

There are two nonconvex components in the N-CDMPC formulation, i.e., the nonlinear model (16b) and constraints (21). By using Taylor expansion at the prior estimate state of the train, we can linearize $d_i^{\text{sb}}(v_{i,k})$ and $d_{i-1}^{\text{eb}}(v_{i-1,k})$ in (12) and (13). The prior estimate state of train i at time step $k+1$ can be calculated according to the current speed $v_{i,k+1}$, the current position $s_{i,k+1}$, and control inputs in (18) [15], [5]. The nonlinear model (16b) can also be linearized at each time step based on the prior estimate state by using Taylor expansion. Other settings are exactly the same as the N-CDMPC approach. Hence, we can simplify the N-CDMPC approach to develop a convex cooperative distributed MPC (C-CDMPC) approach for the platoon of virtually coupled trains.

D. Nonconvex Serial Distributed MPC

For the unidirectional communication in Fig. 1(b), each train only communicates with its neighbors once in one control step. In this context, each train computes control inputs sequentially based on the information received from its predecessor train. Specifically, train i calculates control inputs based on the speed $\bar{v}_{i-1,k}$, position $\bar{s}_{i-1,k}$, and control input $\bar{\mu}_{i-1,k}$ received from train $i-1$, where $\bar{v}_{i-1,k}$, $\bar{s}_{i-1,k}$, and $\bar{\mu}_{i-1,k}$ are the results of the optimization problem in train $i-1$. Thus, the safety constraints in (12) are replaced by

$$d_i^{\text{safe}}(v_{i,k}, \bar{v}_{i-1,k}) \geq d_i^{\text{sb}}(v_{i,k}) - d_{i-1}^{\text{eb}}(\bar{v}_{i-1,k}) + L + D_{\text{safe}}, \quad (28a)$$

$$d_i^{\text{safe}}(v_{i,k}, \bar{v}_{i-1,k}) \geq L + D_{\text{safe}}. \quad (28b)$$

Furthermore, the relative distance with its predecessor train becomes

$$e_{i,k} = \bar{s}_{i-1,k} - s_{i,k} - d_i^{\text{sb}}(v_{i,k}) + d_{i-1}^{\text{eb}}(\bar{v}_{i-1,k}). \quad (29)$$

Then, the cost function becomes

$$\bar{\ell}_i(x_{i,k}, \mu_{i,k}) = \|x_{i,k} - \bar{x}_{i,k}^{\text{eq}}\|_{Q_i}^2 + \|\mu_{i,k}\|_{R_i}^2, \quad (30)$$

where $\bar{x}_{i,k}^{\text{eq}} = [\bar{v}_{i,k}^{\text{eq}}, \bar{e}_{i,k}^{\text{eq}}]^T$ is the equilibrium state of train i , with $\bar{v}_{i,k}^{\text{eq}} = \bar{v}_{i-1,k}$ and $\bar{e}_{i,k}^{\text{eq}} = L + D_{\text{des}}$.

Therefore, in nonconvex serial distributed MPC (N-SDMPC), each train solves the MPC optimization problem as follows

$$\min_{\substack{x_{i,k_0} \\ \mu_{i,k_0}}} J_i(k_0) = \sum_{k=k_0}^{k_0+N-1} \bar{\ell}_i(x_{i,k}, \mu_{i,k}) \quad (31a)$$

$$\text{s.t. } x_{i,k+1} = f(x_{i,k}, \mu_{i,k}), \quad k = k_0, \dots, k_0+N-1, \quad (31b)$$

$$g_i(\bar{y}_{i-1,k}, y_{i,k}) \leq E_{2,i,k}, \quad k = k_0, \dots, k_0+N-1, \quad (31c)$$

where (31c) is the compact form of constraints corresponding to $y_{i,k} = [x_{i,k}^T, \mu_{i,k}^T]^T$, i.e., constraints (9)-(11) and (28).

The MPC optimization problem (31) is a nonlinear non-convex optimization problem, and we can use gradient-based approaches, e.g., sequential quadratic programming, to find a solution. At each MPC step of N-SDMPC, each train calculated its control input $\mu_{i,k}$ for implementation by solving (31) with received $\bar{x}_{i-1,k}$ and $\bar{\mu}_{i-1,k}$, and then send the obtained $x_{i,k}$ and $\mu_{i,k}$ to its succeeding train.

Remark 3. As each train only communicates with its neighbors once per control step in the unidirectional communication case, the global optimal solution to the overall problem cannot be guaranteed. The serial distributed MPC approach follows a first-come first-serve fashion for the coupled constraint (31c), i.e., the predecessor train calculates and sends states and control inputs to its follower train, and the follower train then calculates states and control inputs that satisfy the coupled constraint (31c) based on the received information.

E. Convex Serial Distributed MPC

To reduce the computational burden of solving the nonlinear nonconvex optimization problem (31) for each train, (31) can be approximated to develop convex serial distributed MPC (C-SDMPC) for the platoon of virtually coupled trains based on the prior estimate state [5]. See also Section IV-C for detailed information on the convex approximation using the prior estimate state. Then, we can obtain the convex counterpart of (31) by linearizing $d_i^{\text{sb}}(v_{i,k})$ and $d_{i-1}^{\text{eb}}(\hat{v}_{i-1,k})$ in (28a) and (29). Other settings of the C-SDMPC approach are exactly the same as the N-SDMPC approach in Section IV-D.

F. Nonconvex Decentralized MPC

The virtually coupled train control approaches should be able to ensure safe operation when the communication between trains is lost, i.e., the case in Fig. 1(c). In this context, each train should compute control inputs based on the relative speed and position of its predecessor train measured by onboard sensors, e.g., radars or LiDARs, assuming the predecessor train brakes with the maximum braking force. This leads to a nonconvex decentralized MPC (N-DMPC) approach elaborated in this section.

For train i , the relative speed and position with respect to its predecessor train, i.e., train $i-1$, at time step k are represented by $\Delta v_{i,k}$ and $\Delta s_{i,k}$, respectively, which can be obtained by onboard sensors. At time step k , the estimated speed $\hat{v}_{i-1,k}$ and position $\hat{s}_{i-1,k}$ of train $i-1$ are

$$\hat{v}_{i-1,k} = v_{i,k} + \Delta v_{i,k}, \quad (32a)$$

$$\hat{s}_{i-1,k} = s_{i,k} + \Delta s_{i,k}. \quad (32b)$$

Then, the predicted state of train $i-1$ is estimated by assuming the control value as $\hat{\mu}_{i-1,k} = \frac{-B_{i-1}^{\text{eb}}}{M_{i-1,p}}$. Thus, the safety constraints in (12) are replaced by

$$d_i^{\text{safe}}(v_{i,k}, \hat{v}_{i-1,k}) \geq d_i^{\text{sb}}(v_{i,k}) - d_{i-1}^{\text{eb}}(\hat{v}_{i-1,k}) + L + D_{\text{safe}}, \quad (33a)$$

$$d_i^{\text{safe}}(v_{i,k}, \hat{v}_{i-1,k}) \geq L + D_{\text{safe}}. \quad (33b)$$

Furthermore, the relative distance with its predecessor train becomes

$$e_{i,k} = \hat{s}_{i-1,k} - s_{i,k} - d_i^{\text{sb}}(v_{i,k}) + d_{i-1}^{\text{eb}}(\hat{v}_{i-1,k}). \quad (34)$$

To ensure the safety operation, train i should follow the desired state $\hat{x}_{i,k}^{\text{eq}} = [\hat{v}_{i,k}^{\text{eq}}, \hat{e}_{i,k}^{\text{eq}}]^\top$ with $\hat{v}_{i,k}^{\text{eq}} = \hat{v}_{i-1,k}$ and $\hat{e}_{i,k}^{\text{eq}} = L + D_{\text{des}}$. Then, the cost function for N-DMPC is

$$\hat{\ell}_i(x_{i,k}, \mu_{i,k}) = \|x_{i,k} - \hat{x}_{i,k}^{\text{eq}}\|_{Q_i}^2 + \|\mu_{i,k}\|_{R_i}^2. \quad (35)$$

Hence, the optimization problem of train i for N-DMPC becomes

$$\min_{\substack{x_{i,k_0} \\ \mu_{i,k_0}}} J_i(k_0) = \sum_{k=k_0}^{k_0+N-1} \hat{\ell}_i(x_{i,k}, \mu_{i,k}) \quad (36a)$$

$$\text{s.t. } x_{i,k+1} = f(x_{i,k}, \mu_{i,k}), \quad k = k_0, \dots, k_0+N-1, \quad (36b)$$

$$g_i(\hat{y}_{i-1,k}, y_{i,k}) \leq E_{3,i,k}, \quad k = k_0, \dots, k_0+N-1, \quad (36c)$$

where (36c) collects constraints (9)-(11) and (33). The optimization problem (36) is also a nonlinear nonconvex optimization problem. At each MPC step of N-DMPC, each train calculated its control input $\mu_{i,k}$ for implementation by solving (36) with estimated $\hat{x}_{i-1,k}$.

G. Convex Decentralized MPC

Similarly, we can obtain the convex counterpart of (36), named as convex decentralized MPC (C-DMPC) by linearizing $d_i^{\text{sb}}(v_{i,k})$ and $d_{i-1}^{\text{eb}}(\hat{v}_{i-1,k})$ in (33a) and (34). Then, the nonlinear model (36b) can be linearized at each time step based on the prior estimate state. Other settings of the C-DMPC approach are exactly the same as the N-DMPC approach.

V. CASE STUDY

In this section, we conduct simulations to validate the developed distributed MPC approaches. We first introduce general settings for simulations. Then, we perform simulations for a platoon of trains with uniform masses. Finally, we explore simulations involving trains with varying masses.

A. General Setup

The simulations are conducted based on the real-life train operation data of trains on the Beijing Yizhuang Line from Station YH to Station CQ. The values of the main parameters are provided in Table II. The value of ρ is set as 0.5, and the initial value of $\lambda_{i,k}^{(0)}$ is set as 1. The values of the safety distance and the desired distance are the same as those in papers [15], [16]. The distance from Station YH to Station CQ is 1398.6 m, and the slope and the speed limit information along the line are shown in Fig. 3. Model mismatches exist between the control model and the simulation model. The controller design considers the prediction model with the values of the maximum traction and braking forces U_i^{max} , B_i^{sb} , and B_i^{eb} given in Table II, while the assessment experiments use the simulation model considering the traction and braking characteristics given in Fig. 4 (see also [40]).

Sequential quadratic programming (SQP) is an efficient gradient-based algorithm for solving nonlinear programming

TABLE II
PARAMETERS FOR THE CONTROLLER DESIGN

Parameter	Symbol	Numerical value
Prediction horizon	N	5
Sampling time	T	0.2 s
Number of trains	I_{train}	4
Average train mass	$M_{i,p}$	60 t
Resistance parameter	c_0	0.0078
Resistance parameter	c_1	0.00085
Resistance parameter	c_2	0.000076
Maximum traction force	U_i^{max}	60000 N
Maximum service braking force	B_i^{sb}	48000 N
Emergency braking force	B_i^{eb}	60000 N
Safety distance	D_{safe}	5 m
Desired distance	D_{des}	10 m
Train Length	L	10 m
Weight of tracking error	Q_1	100
Weight of relative speed error	Q_2	1
Weight of control variable	R	1
RDP parameter	α	0.5
Maximum iterations	q_{max}	5

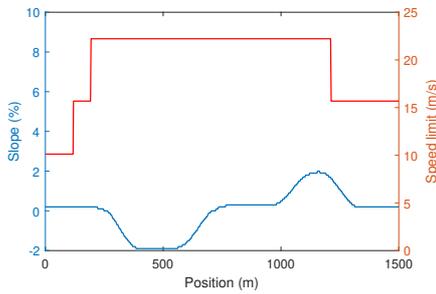


Fig. 3. Line information from Station YH to Station CQ.

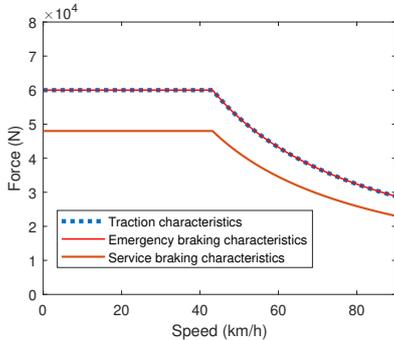


Fig. 4. Traction and braking characteristics of the simulation model.

problems [41] and has also been applied to solve the optimization problem of virtually coupled trains [5]. Similar to [5], in each MPC step, the resulting optimization problem is solved by SQP through the `fmincon` function in the MATLAB optimization toolbox. All simulations are implemented in MATLAB (R2019b) on a computer with an Intel Xeon W-2223 CPU and 8GB RAM.

B. Control Performance with Uniform Train Masses

This case study is conducted to evaluate the performance of the distributed MPC approaches in the case that trains in the platoon have the same mass. The parameters are provided

in Section V-A. We consider a platoon of 4 trains in the simulation, and all trains weigh 60 t. The simulation results are presented in Table III, wherein the relative distance is calculated as defined in (13), and the speed difference represents the velocity error between a train and its predecessor train.

Note that the relative distance represents the distance between two trains, assuming that the predecessor train performs emergency braking and the succeeding train performs service braking. As the train length is set as 10 m and the safety distance D_{safe} is 5 m, the relative distance should be larger than 15 m to ensure safe operation. Furthermore, since the desired distance is 10 m, the ideal relative distance should be 20 m considering the length of the train (i.e., 10 m).

It can be observed from Table III that convex cooperative distributed MPC (C-CDMPC), convex serial distributed MPC (C-SDMPC), and convex decentralized MPC (C-DMPC) exhibit a performance that is comparable to that of their nonconvex counterparts in terms of the relative distance and the speed difference. The average CPU time is reduced when the underlying problem is convex, with a reduction of 64.25%, 17.86%, and 17.86% for C-CDMPC, C-SDMPC, and C-DMPC, respectively, compared with their corresponding original approaches, indicating that a computational burden reduction is achieved by transforming these problems to their corresponding convex problems. As the performance, in terms of the relative distance and the speed difference, of the original approaches is comparable with their corresponding convex counterparts, we will focus on C-CDMPC, C-SDMPC, and C-DMPC to compare the performance of different distributed control schemes in the following for brevity.

Table III shows that all approaches can ensure safe operation when trains have the same mass with a minimum relative distance larger than 15 m. The average relative distance of C-CDMPC and C-SDMPC is close to the ideal relative distance (20 m), while C-DMPC has the largest average relative distance. Furthermore, C-CDMPC exhibits the smallest fluctuation, with the relative distance fluctuating within the range [18.47 m, 22.62 m] and the speed difference fluctuating within [-1.1059 m/s, 1.3927 m/s].

For further demonstration, the speed profiles obtained by C-CDMPC, C-SDMPC, and C-DMPC are provided in Fig. 5, Fig. 6, and Fig. 7, respectively, where we include the speed difference between a train and its predecessor train. For the first train, the speed difference denotes the difference with the reference speed. It can be observed from Fig. 6 that due to the speed limit, train 4 cannot accelerate, causing a rapid change in speed difference. Thanks to the bidirectional communication as represented in Fig. 1(a), the rapid change is avoided in Fig. 5, i.e., by using C-CDMPC, a train can include the information of its follower train when calculating its control input, thereby achieving a more homogeneous speed profile via cooperative control. Table III and Fig. 7 show that C-DMPC exhibits the largest fluctuation in both relative distance and the speed difference. As a train cannot receive information from its predecessor train in Fig. 1(c), a train should always assume its predecessor train will perform emergency braking. The decentralized control scheme tends to be conservative; thus, the relative distance and the speed difference of C-DMPC

TABLE III
SIMULATION RESULTS FOR DIFFERENT APPROACHES WITH UNIFORM TRAIN MASSES

Approach	RDP	Total cost	Relative distance (m)			Speed difference (m/s)			CPU time (s)	
			max	average	min	max	average	min	max	average
N-CDMPC	yes	$2.1506 \cdot 10^4$	22.63	19.89	18.50	1.3927	0.0099	-1.1059	4.74	4.00
C-CDMPC	yes	$1.8412 \cdot 10^4$	22.62	19.91	18.47	1.3886	0.0083	-1.1071	1.62	1.43
N-SDMPC	no	$2.8826 \cdot 10^4$	27.64	20.02	19.35	1.6091	0.0025	-1.2273	0.40	0.28
C-SDMPC	no	$2.8825 \cdot 10^4$	27.64	20.02	19.35	1.6091	0.0025	-1.2273	0.30	0.23
N-DMPC	no	$9.1687 \cdot 10^5$	31.50	21.91	19.08	1.7582	0.0035	-1.3627	0.37	0.28
C-DMPC	no	$9.2349 \cdot 10^5$	31.64	21.92	19.08	1.7488	0.0035	-1.3778	0.30	0.23

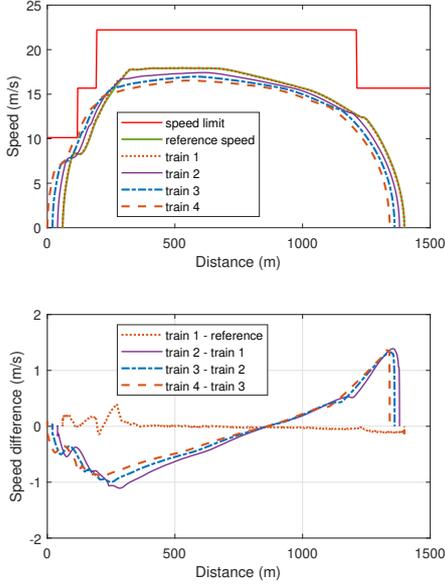


Fig. 5. Speed profiles and speed difference of C-CDMPC (with the same mass).

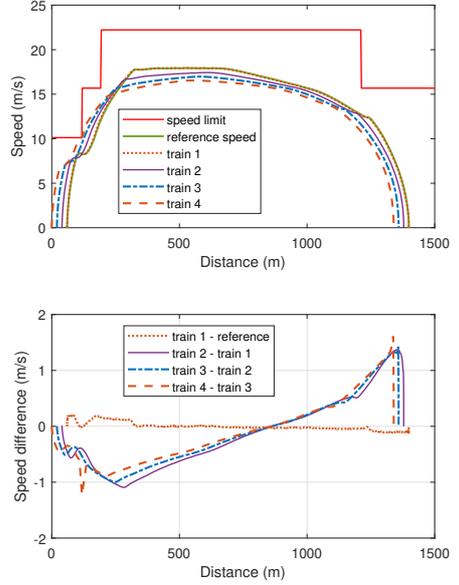


Fig. 6. Speed profiles and speed difference of C-SDMPC (with the same mass).

are larger than those of C-CDMPC and C-SDMPC.

C. Control Performance with Heterogeneous Train Mass

In general, the masses of trains within a platoon are different due to variations in the total passenger loads on each train. The mass of a train influences the acceleration and deceleration (see (5a)), and determines the upper bound and the lower bound of the control input. Therefore, the mass inconsistency will influence the control performance of the platoon. In this case study, we consider a platoon of 4 trains, where the weights of the trains, from the leader train to the follower trains, are 60 t, 66 t, 57 t, and 66 t, respectively; so the heaviest train is more than 15% heavier than the lightest train.

In order to show the importance of incorporating the information on weights into the control design, we first conduct simulations with all trains assumed to have the same mass in the control design. Then, we compare the results with the true masses of the trains used in the control design. The simulation results are provided in Table IV.

From Table IV, we can find that all approaches have comparable performance in terms of the relative distance and speed difference with their convex counterpart under both cases.

Therefore, in the following, we only use convex approaches for the comparison between the two cases in which the mass information is disregarded or included in the design.

In Table IV, if we assume all trains have the same mass in the control design, the minimum relative distance across all approaches is less than 0 m, implying the potential collision between trains, i.e., a train cannot ensure safety operation by using service braking when the predecessor train performs emergency braking. The relative distance between trains during the operation process of each approach when assuming all trains have the same mass in the control design is shown in Fig. 8. Fig. 8 shows that the relative distance between Train 1 and Train 2, Train 3 and Train 4 are lower than the given threshold. As the follower train has a larger inertia than its predecessor train, if the predecessor train starts to perform emergency braking, the follower train cannot perform braking with the same deceleration. Therefore, when a train is heavier than its predecessor train, the required safe tracking distance becomes difficult to ensure. Train 3 is lighter than Train 2; thus, the braking distance of Train 3 is shorter than expected and the safety distance between trains can be ensured. However, the relative distance between Train 2 and Train 3 is

TABLE IV
SIMULATION RESULTS FOR DIFFERENT APPROACHES WITH DIFFERENT TRAIN MASSES

	Approach	RDP	Total cost	Relative distance (m)			Speed difference (m/s)			CPU time (s)	
				max	average	min	max	average	min	max	average
With all trains assumed to have the same mass	N-CDMPC	yes	$3.9359 \cdot 10^7$	44.26	16.34	-4.39	1.3943	0.0102	-1.1032	4.71	4.02
	C-CDMPC	yes	$3.9348 \cdot 10^7$	44.26	16.35	-4.39	1.3902	0.0087	-1.1049	1.78	1.41
	N-SDMPC	no	$3.9198 \cdot 10^7$	44.26	16.44	-4.40	1.5643	0.0030	-1.2313	0.55	0.26
	C-SDMPC	no	$3.9198 \cdot 10^7$	44.26	16.44	-4.40	1.5643	0.0030	-1.2313	0.30	0.22
	N-DMPC	no	$3.7318 \cdot 10^7$	46.31	18.35	-2.66	1.7237	0.0041	-1.3598	0.47	0.26
	C-DMPC	no	$3.7330 \cdot 10^7$	46.31	18.35	-2.67	1.7254	0.0041	-1.3747	0.27	0.22
With true masses of trains in the control design	N-CDMPC	yes	$2.5846 \cdot 10^4$	21.96	19.86	17.86	2.2300	0.0126	-1.5705	5.04	4.35
	C-CDMPC	yes	$1.9643 \cdot 10^4$	21.92	19.88	18.16	2.2387	0.0106	-1.5715	1.82	1.39
	N-SDMPC	no	$3.0343 \cdot 10^4$	27.42	20.01	18.69	2.5732	0.0042	-1.6251	0.34	0.26
	C-SDMPC	no	$3.0342 \cdot 10^4$	27.42	20.01	18.69	2.5732	0.0042	-1.6251	0.31	0.23
	N-DMPC	no	$9.2421 \cdot 10^4$	31.33	21.87	18.18	2.7396	0.0057	-1.7608	0.35	0.27
	C-DMPC	no	$9.3093 \cdot 10^4$	31.47	21.88	18.17	2.7339	0.0057	-1.7748	0.30	0.23

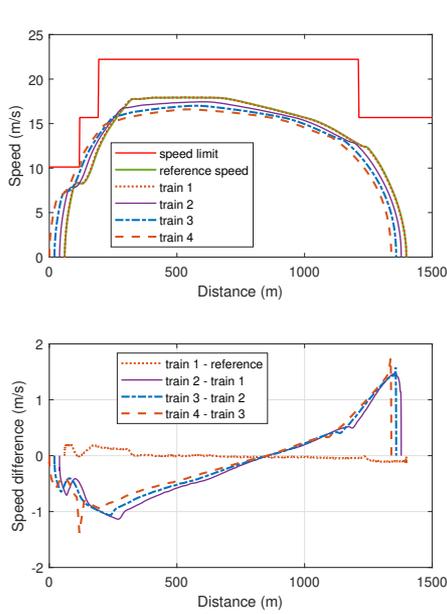


Fig. 7. Speed profiles and speed difference of C-DMPC (with the same mass).

larger than the desired distance, with the maximum value being more than twice the desired distance, which is unnecessary and negatively influences the tracking performance.

The relative distance between trains of each approach when considering the true masses of the trains is shown in Fig. 9. From Table IV and Fig. 9, we can find that by including train masses explicitly, the safety distance between trains can be ensured, and the relative distance between trains is comparable to the case of uniform masses in Table III.

The speed profiles obtained by C-CDMPC, C-SDMPC, and C-DMPC considering the true masses of trains are provided in Fig. 10, Fig. 11, and Fig. 12, respectively. The C-CDMPC approach has the smallest fluctuation, with the relative distance fluctuating between [18.16 m, 21.92 m] and the speed difference fluctuates between [-1.5715 m/s, 2.2387 m/s]. In the cooperative control scheme, a subsystem can include the status of its neighbors and try to reach consistency with its neighbors regarding the relative distance and speed difference. Fig. 10,

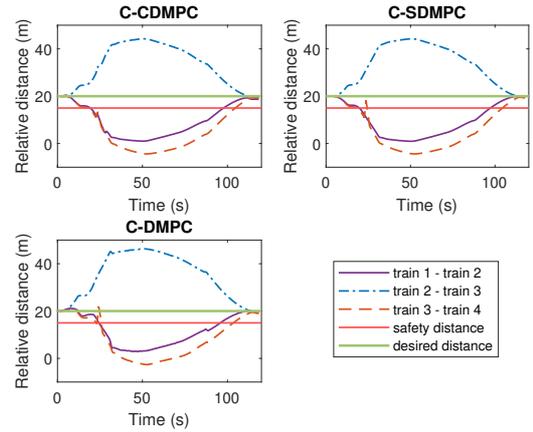


Fig. 8. Relative distance of different approaches (with all trains assumed to have the same mass in the control design).

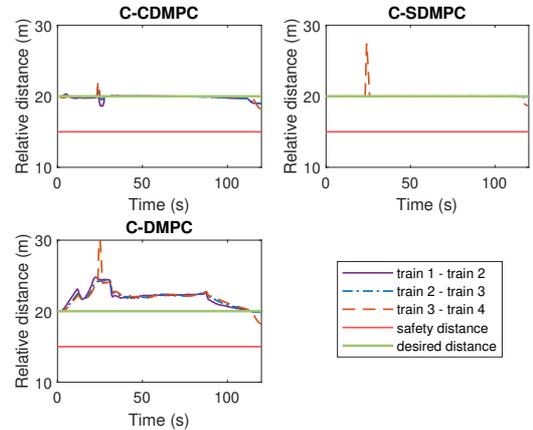


Fig. 9. Relative distance of different approaches (with the true masses of the trains used in the control design).

Fig. 11, and Fig. 12 show that for all three control methods, the speed difference between Train 2 and Train 3 is lower than the speed difference between Train 1 and Train 2, Train 3 and Train 4, implying that if the follower train is lighter than the predecessor train, the tracking performance would be better.

From the above simulations, we can conclude that the

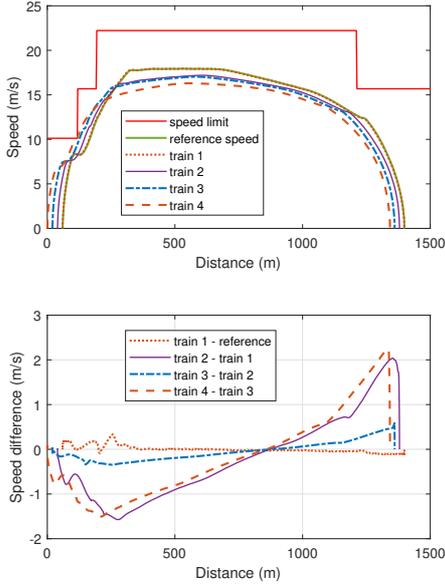


Fig. 10. Speed profiles and speed difference of C-CDMPC (with the true masses of the trains used in the control design).

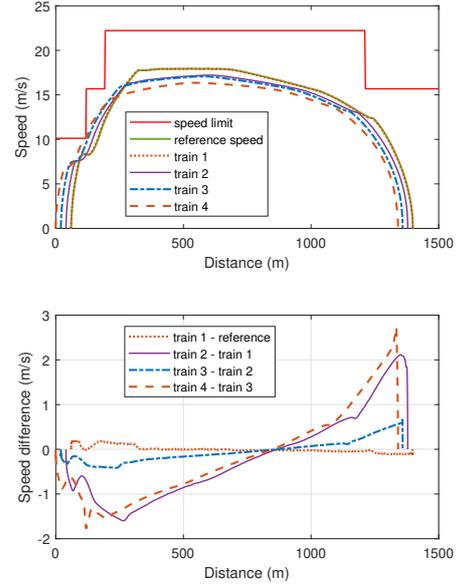


Fig. 12. Speed profiles and speed difference of C-DMPC (with the true masses of the trains used in the control design).

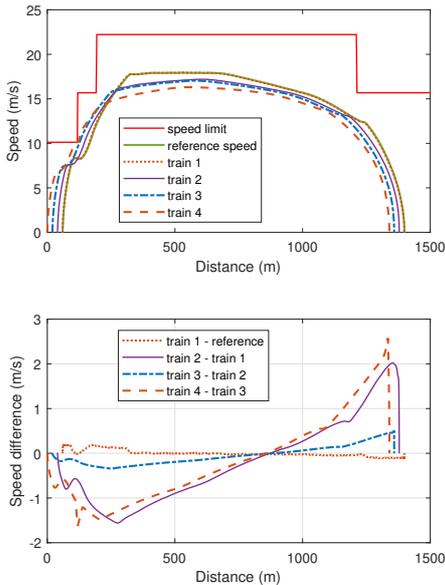


Fig. 11. Speed profiles and speed difference of C-SDMPC (with the true masses of the trains used in the control design).

cooperative control approach has the best tracking performance while requiring ample communication and computation capabilities. Hence, C-CDMPC can be selected when sufficient communication bandwidth and computation power are available. The C-SDMPC approach can be selected in case of limited communication bandwidth and limited computation power. Moreover, in the worst case when two neighbor trains cannot communicate with each other, C-DMPC can be selected

to control trains in a decentralized manner. Moreover, the simulation results also indicate that arranging heavier trains at the front of the platoon can help to improve the control performance of the virtually coupled trains.

D. Highlights of Results

1) *Convex approximation*: In Sections V-B and V-C, we have conducted simulations for cooperative distributed MPC, serial distributed MPC, and decentralized MPC under both the cases of uniform masses and heterogeneous masses. For all approaches and cases, we have tested nonlinear MPC approaches and their convex approximations. The simulation results indicate that MPC with convex approximation can achieve a speed tracking accuracy that is comparable to that of the original nonconvex counterpart, while significantly reducing the computation time. Therefore, using convex approximation is an effective way to improve the computational efficiency of MPC in virtually coupled trains.

2) *Relaxed dynamic programming (RDP)*: Sections V-B and V-C provide case studies for uniform and heterogeneous masses, respectively. The simulations indicate that cooperative distributed MPC, when accompanied by RDP, can achieve better performance with lower speed and distance tracking differences. By using RDP, we can develop a stopping criterion for the string stability of the platoon, which, in general, cannot be achieved with serial distributed MPC and decentralized MPC. Overall, RDP is an effective approach to analyze the stability of MPC approaches. Moreover, sufficient computational capacity should be ensured to support the efficient implementation of the RDP-based stopping criterion developed.

3) *Heterogeneous masses*: Train masses influence the dynamics of trains and should be considered explicitly in the

controller design to improve control performance. In Section V-C, we have conducted simulations for cases with and without true masses of trains. The simulation results indicate that incorporating the true masses of trains in the controller design ensures safety and achieves the desired tracking performance while significantly reducing the total costs for all the mentioned MPC approaches. In this context, we conclude that, in general, detailed train information should be included to improve control performance when designing control approaches for virtually coupled heterogeneous trains.

TABLE V
CHARACTERISTICS OF INCLUDING DIFFERENT ELEMENTS

Elements	Characteristics
Convex approximation	Reduce computational burden while maintaining tracking accuracy
RDP	Incorporate a stopping criterion into cooperative distributed MPC for string stability
Heterogeneous masses	Improve tracking accuracy and ensure safety

To summarize, the advantages of considering convex approximation, relaxed dynamic programming, and heterogeneous masses are listed in Table V. In this paper, we consider communication between two consecutive trains (as stated in Fig. 1). In this context, each train only needs to consider the status of its preceding and succeeding trains when calculating its control decision. Thus, the approaches can be extended to larger train platoons without increasing the computational burden for each individual train.

VI. CONCLUSIONS

In this paper, cooperative distributed MPC, serial distributed MPC, and decentralized MPC have been compared and assessed for controlling virtually coupled trains, considering the nonlinear train model and changes in the masses of trains. We introduced the relaxed dynamic programming approach into the train control field, and a distributed stopping criterion with a stability guarantee has been developed for the cooperative distributed MPC approach. We have also proposed and assessed convex approximations of the above control approaches to make a balanced trade-off between computational burden and accuracy. The three control approaches and their convex counterparts have been evaluated considering their distance tracking error, speed tracking error, and CPU time. Simulation results indicate that: 1) the convex approaches can achieve a performance that is comparable to that of their original nonconvex version, while the computational burden is reduced; 2) the cooperative control approach has the best tracking performance while requiring ample communication and computation capabilities; 3) by considering heterogeneous train masses explicitly, the safety distance between trains and the desired tracking performance can be ensured while the total objective function value is significantly reduced.

Future research could explore uncertainties related to resistances and train dynamics to enhance the performance of the control methods. Additionally, distributed control under conditions of intermittent communication is also promising, which can be achieved by designing appropriate self-triggered or event-triggered control strategies to address communication

latency. Furthermore, future work could involve extending the research into other types of rail transportation modes, such as freight and heavy haul trains.

REFERENCES

- [1] B. Ning, T. Tang, K. Qiu, and C. Gao, "CBTC (communication based train control): System and development," *WIT Transactions on State-of-the-art in Science and Engineering*, vol. 46, 2010.
- [2] J. Aoun, E. Quaglietta, and R. M. Goverde, "Roadmap development for the deployment of virtual coupling in railway signalling," *Technological Forecasting and Social Change*, vol. 189, p. 122263, 2023.
- [3] R. M. Borges and E. Quaglietta, "Assessing hyperloop transport capacity under moving-block and virtual coupling operations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12 612–12 621, 2021.
- [4] E. Quaglietta, M. Wang, and R. M. Goverde, "A multi-state train-following model for the analysis of virtual coupling railway operations," *Journal of Rail Transport Planning & Management*, vol. 15, p. 100195, 2020.
- [5] 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, 2021.
- [6] Q. Li, Z. Chen, and X. Li, "A review of connected and automated vehicle platoon merging and splitting operations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 22 790–22 806, 2022.
- [7] M. Soliman, J. Siemons, J. Kochems, W. Alshrafi, J. Shamshoom, D. Heberling, and A. Dekorsy, "Automatic train coupling: Challenges and key enablers," *IEEE Communications Magazine*, vol. 57, no. 9, pp. 32–38, 2019.
- [8] 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.
- [9] J. Aoun, E. Quaglietta, and R. M. Goverde, "Investigating market potentials and operational scenarios of virtual coupling railway signaling," *Transportation Research Record*, vol. 2674, no. 8, pp. 799–812, 2020.
- [10] J. Xun, Y. Li, R. Liu, Y. Li, and Y. Liu, "A survey on control methods for virtual coupling in railway operation," *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 838–855, 2022.
- [11] Y. Cao, J. Wen, and L. Ma, "Tracking and collision avoidance of virtual coupling train control system," *Future Generation Computer Systems*, vol. 120, pp. 76–90, 2021.
- [12] J. Xun, M. Chen, Y. Liu, and F. Liu, "An overspeed protection mechanism for virtual coupling in railway," *IEEE Access*, vol. 8, pp. 187 400–187 410, 2020.
- [13] 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, 2021.
- [14] J. M. Maestre, R. R. Negenborn *et al.*, *Distributed model predictive control made easy*. Springer, 2014, vol. 69.
- [15] J. Felez, Y. Kim, and F. Borrelli, "A model predictive control approach for virtual coupling in railways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 7, pp. 2728–2739, 2019.
- [16] J. Felez, M. A. Vaquero-Serrano, and J. de Dios Sanz, "A robust model predictive control for virtual coupling in train sets," *Actuators*, vol. 11, no. 12, p. 372, 2022.
- [17] M. A. Vaquero-Serrano and J. Felez, "A decentralized robust control approach for virtually coupled train sets," *Computer-Aided Civil and Infrastructure Engineering*, vol. 38, no. 14, pp. 1896–1915, 2023.
- [18] 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, 2019.
- [19] J. Park, B.-H. Lee, and Y. Eun, "Virtual coupling of railway vehicles: Gap reference for merge and separation, robust control, and position measurement," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1085–1096, 2020.
- [20] G. Basile, D. G. Lui, A. Petrillo, and S. Santini, "Deep deterministic policy gradient virtual coupling control for the coordination and manoeuvring of heterogeneous uncertain nonlinear high-speed trains," *Engineering Applications of Artificial Intelligence*, vol. 133, p. 108120, 2024.

- [21] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pan-nocchia, "Cooperative distributed model predictive control," *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.
- [22] B. T. Stewart, S. J. Wright, and J. B. Rawlings, "Cooperative distributed model predictive control for nonlinear systems," *Journal of Process Control*, vol. 21, no. 5, pp. 698–704, 2011.
- [23] E. Quaglietta, P. Spartalis, M. Wang, R. M. Goverde, and P. van Koningsbruggen, "Modelling and analysis of virtual coupling with dynamic safety margin considering risk factors in railway operations," *Journal of Rail Transport Planning & Management*, vol. 22, p. 100313, 2022.
- [24] S. Su, J. She, D. Wang, S. Gong, and Y. Zhou, "A stabilized virtual coupling scheme for a train set with heterogeneous braking dynamics capability," *Transportation Research Part C: Emerging Technologies*, vol. 146, p. 103947, 2023.
- [25] Y. Liu, Y. Zhou, S. Su, J. Xun, and T. Tang, "An analytical optimal control approach for virtually coupled high-speed trains with local and string stability," *Transportation Research Part C: Emerging Technologies*, vol. 125, p. 102886, 2021.
- [26] X. Luo, T. Tang, J. Yin, and H. Liu, "A robust MPC approach with controller tuning for close following operation of virtually coupled train set," *Transportation Research Part C: Emerging Technologies*, vol. 151, p. 104116, 2023.
- [27] Z. Zhang, H. Song, H. Wang, X. Wang, and H. Dong, "Cooperative multi-scenario departure control for virtual coupling trains: A fixed-time approach," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 9, pp. 8545–8555, 2021.
- [28] H. Wang, Q. Zhao, S. Lin, D. Cui, C. Luo, L. Zhu, X. Wang, and T. Tang, "A reinforcement learning empowered cooperative control approach for IIoT-based virtually coupled train sets," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 7, pp. 4935–4945, 2020.
- [29] J. Felez and M. A. Vaquero-Serrano, "Virtual coupling in railways: A comprehensive review," *Machines*, vol. 11, no. 5, p. 521, 2023.
- [30] R. Sattiraju, J. Siemons, M. Soliman, W. Alshrafi, F. Rein, and H. D. Schotten, "Design of a highly reliable wireless module for ultra-low-latency short range applications," in *2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS)*. IEEE, 2017, pp. 1–4.
- [31] X. Wang, L. Liu, T. Tang, and W. Sun, "Enhancing communication-based train control systems through train-to-train communications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1544–1561, 2019.
- [32] H. K. Khalil, *Nonlinear Systems (Third Edition)*. Patience Hall, 2002.
- [33] D. Swaroop and J. K. Hedrick, "String stability of interconnected systems," *IEEE Transactions on Automatic Control*, vol. 41, no. 3, pp. 349–357, 1996.
- [34] S. Feng, Y. Zhang, S. E. Li, Z. Cao, H. X. Liu, and L. Li, "String stability for vehicular platoon control: Definitions and analysis methods," *Annual Reviews in Control*, vol. 47, pp. 81–97, 2019.
- [35] S. Li, L. Yang, and Z. Gao, "Distributed optimal control for multiple high-speed train movement: An alternating direction method of multipliers," *Automatica*, vol. 112, p. 108646, 2020.
- [36] W. J. Davis, *The tractive resistance of electric locomotives and cars*. General Electric, 1926.
- [37] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*. Springer, 2017.
- [38] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [39] D. Han and X. Yuan, "A note on the alternating direction method of multipliers," *Journal of Optimization Theory and Applications*, vol. 155, pp. 227–238, 2012.
- [40] X. Sun, Z. Yao, C. Dong, and D. Clarke, "Optimal control strategies for metro trains to use the regenerative braking energy: A speed profile adjustment approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 6, pp. 5883–5894, 2023.
- [41] P. T. Boggs and J. W. Tolle, "Sequential quadratic programming," *Acta Numerica*, vol. 4, pp. 1–51, 1995.