Transportation Research Part C 182 (2026) 105390

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Multivariate event hypergraph diffusion model for train delay prediction

Yi Xu [a], Honghui Li [a,*], Chang Wu [a], Yunjuan Peng [b], Xilu Du [c], Hongwei Wang [d], Sabah Mohammed [e], Alessandro Calvi [f], Dalin Zhang [g,*]

[a] School of Computer Science & Technology, Beijing Jiaotong University, Beijing 100044, China
[b] School of Software Engineering, Beijing Jiaotong University, Beijing 100044, China
[c] China Railway Lanzhou Group Co.,Ltd., Gansu 730000, China
[d] Office of Research and Development Affairs, Beijing Jiaotong University, Beijing 100044, China
[e] Department of Computer Science, Lakehead University, Thunder Bay P7A0A2, Canada
[f] Department of Engineering, Roma Tre University, Tome 00118, Italy
[g] School of Cyberspace Science and Technology, Beijing Jiaotong University, Beijing 100044, China

ARTICLE INFO

Keywords:
Train delay prediction
Denoising diffusion probabilistic models
Regional-level multi-train operation
High-order correlations
Non-Euclidean characteristics
Mixed hypergraph convolution

ABSTRACT

Train delay prediction is a key technology for train scheduling and timetable optimization, and constitutes a critical component of intelligent transportation systems. We present the first study on regional-level multi-train delay prediction problem, and focus on modeling the regional-level delay propagation and evolution process, and capturing coordinated operation status among multiple train clusters in the complex operation network. First, we propose a brand-new Multivariate Event Hypergraph Diffusion (MEHD) model, and introduce a novel data structure, the mixed hypergraph, which accurately models the spatio-temporal high-order correlations between the regional-level multi-train arrival events. Then, we propose a mixed hypergraph convolution method to characterize complex train operation network, which improves the ability to capture the spatio-temporal high-order correlations and non-Euclidean characteristics between events. Finally, we propose an event hypergraph diffusion process, and design a prior operational schedule-conditioned attention denoising module to enhance the ability to learn all train arrival event generation mechanisms. Extensive experiments demonstrate that our MEHD achieves superior performance compared to current state-of-the-art models on actual high-speed rail performance datasets, with an average improvement of 20%-30% on multiple metrics, and performs good robustness and efficiency. Subsequent experiments and analyses demonstrate the unique advantages of MEHD over single-train prediction methods. To the best of our knowledge, this is the first end-to-end model for regional-level multi-train delay prediction. The dataset and source code are available online: https://github.com/bjtuxuyi/MEHD.

1. Introduction

The intelligent high-speed railway operation system is a key link in realizing the Intelligent Transportation System (ITS) (Zhang, Feb 2023), as it is an important part of the modern transportation system and the core carrier of the intelligent upgrading of rail

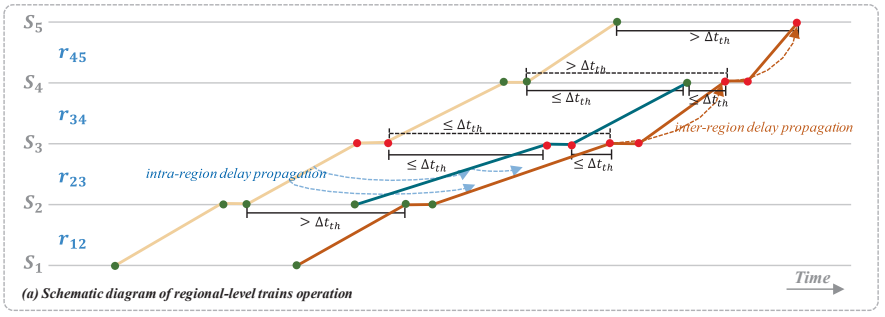
* Corresponding authors.
E-mail addresses: hhli@bjtu.edu.cn (H. Li), dalin@bjtu.edu.cn (D. Zhang).

https://doi.org/10.1016/j.trc.2025.105390
Received 23 April 2025; Received in revised form 7 September 2025; Accepted 12 October 2025

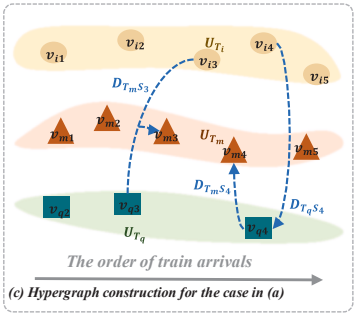
transportation. However, with the continuous expansion of the high-speed railway operation network, trains are often delayed because of equipment, human, and environmental factors. The traditional manual dispatching method has difficulty meeting the complex demands of transport organization nowadays (Wen, 2019). Therefore, carrying out accurate train delay analysis, modeling, prediction, and inference research helps to quickly develop an optimal dispatching schedule. This enables adjustment of the operation strategies within a certain scope, efficiently eliminates the impact of delay propagation caused by disturbances, and rapidly restores the operation order (Zhang et al., Mar 2022).

The train delay propagation and prediction methods have led to significant progress. Traditional model-driven models are mainly based on equation systems (Spanninger et al., 2021)and graph models (Büker and Seybold, 2012; Kecman and Goverde, 2013; Corman and Kecman, 2018)to capture the delay propagation dependency, and they demonstrate high robustness and interpretability. In recent years, deep learning methods have made pioneering progress in prediction accuracy. Wen et al. (2020) and Huang et al. (2020) considered single train delay sequence as homogeneous time series and used RNNs to capture the temporal potential trend of delays. Zhang et al. (2024) considered train delays as event sequences and used temporal point processes to capture the asynchronous characteristics of delay propagation. Ding et al. (2021) and Li et al. (2024) represented train delay data as spatio-temporal graph sequence, which not only captured the long short-term trends of delay propagation, but also aggregated the operational information between adjacent trains and stations within a certain range on the railway network through graph convolution. Since train operations are affected by emergencies, Huang et al. (2022) and Zhang et al. (2022) also considered weather and infrastructure data.

However, high-speed rail trains often operate across multiple regions from their origin to destination stations. Within these regions, dispatch departments are responsible for implementing various strategies to minimize the impact of delays as much as possible and ensure safe train operations (Yuan, 2006; Higgins and Kozan, 1998). Fig. 1(a) takes the cross regional operation of trains T_i, T_q, and T_m from r_{12} to r_{45} as an example, and Fig. 1(b) shows the intra-region and inter-region operational statuses of each train. Within a region, train delays often result from the coupled operation of multiple trains. Each train must arrive at stations in sequence. If the delay time



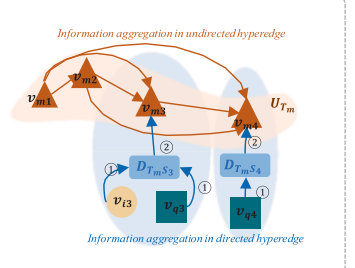
(a) Schematic diagram of regional-level trains operation



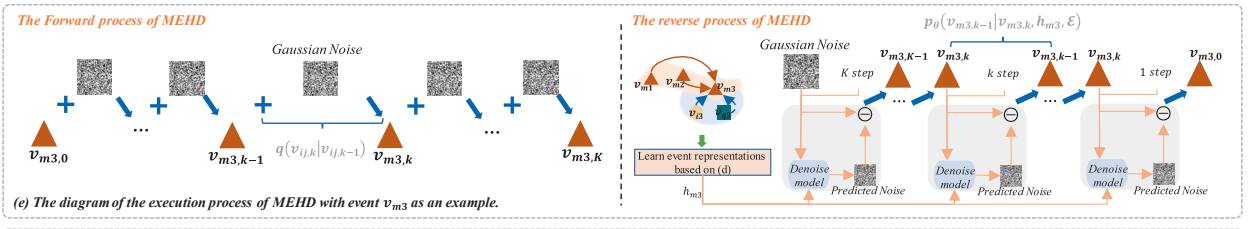
(c) Hypergraph construction for the case in (a)

Train\region	r_{12}	r_{23}	r_{34}	r_{45}
T_m	S_1(departure on time) → S_2(arrival on time)	S_2(departure on time) → S_3(arrival delay)	S_3(departure delay) → S_4(arrival delay)	S_4(departure delay) → S_5(arrival delay)
T_q	--	S_2(departure on time) → S_3(arrival delay)	S_3(departure delay) → S_4(arrival on time)	--
T_i	S_1(departure on time) → S_2(arrival on time)	S_2(departure on time) → S_3(arrival delay)	S_3(departure delay) → S_4(arrival on time)	S_4(departure on time) → S_5(arrival on time)
Delay propagation direction	--	Inter-region: $T_q \rightrightarrows T_m$; $T_i \rightrightarrows T_q$; $T_i \rightsquigarrow T_m$.	Intra-region: $T_q \rightrightarrows T_m$; $T_i \rightrightarrows T_q$ Inter-region: T_m Ongoing delay	Inter-region: T_m Ongoing delay

(b) The regional operational status of trains in (a)



(d) Information aggregation method in hypergraph convolution



(e) The diagram of the execution process of MEHD with event v_{m3} as an example.

S : station. **r :** train operation region. Δt_{th}: headway between non-simultaneous departure-arrival at the station. **⇉ and ⇝ in (b):** direct and indirect delay propagation between trains.

Dashed and solid boundary lines in (a): indirect or direct impact of delays between trains. **Blue and brown dotted lines in (a) and (e):** the directions of intra-region and inter-region delay propagation.

①and ② in (d) : aggregation of tail set node features to hyperedges and updating head nodes. **Golden, green, and brown broken lines in (a):** the operation routes of trains T_i, T_q and T_m respectively.

Blue and brown solid lines in (d): information aggregation directions within directed and undirected hyperedges.

Fig. 1. Diagram of regional-level multi-train operation, event mixed hypergraph construction, information aggregation method in mixed hypergraph convolution and the execution process of MEHD.

of the preceding train exceeds the minimum tracking interval stipulated in the timetable, the arrival of following trains will be postponed, leading to propagated delays (Tiong and Palmqvist, 2023). For instance, in region r_{23}, the delay of T_i propagates to the following train T_q, and the delay of T_q further propagates to T_m, resulting in a delay for T_m at station S_3. That is, T_m is affected by the delays caused by the train clusters consisting of T_i and T_q. As for the inter-region operations, delays in the previous region may either be resolved through operational scheduling or propagate to the next region (Zhang et al., Nov 2024; Ling et al., 2018), and it may affect the normal operation of other trains in the next region. For example, the delay of T_m at S_4 results not only from the impact of train T_q within region r_{34}, but also from the incomplete absorption of the delay of T_m in r_{23}, and further propagates into r_{45}. Therefore, to achieve accurate delay prediction, it is necessary to precisely characterize the inter-region and intra-region delay propagation status. This paper aims to model and capture such multi-level delay propagation status.

Although existing methods have achieved advanced prediction results, they fail to precisely characterize the inter-region and intra-region delay propagation status. Table 1 compares the unique advantages of the regional-level multi-train delay prediction model proposed in this paper with existing models. This paper emphasizes capturing the evolutionary process of regional-level delay propagation (Daniotti et al., 2023) and the collaborative operation status of multi-train clusters (Schälicke and Nachtigall, 2025) in complex operation networks, and realizes efficient inference in a parallel manner. However, the following intractable challenges exist:

Difficulties in capturing regional-level delay propagation and evolution process. The train operation network exhibits high-dimensional spatio-temporal non-Euclidean characteristics, making it highly intractable to capture the propagation and evolution process of delays in the complex operation network. Temporally, arrival events show asynchronous and discrete non-Euclidean features (Zhang et al., Nov 2024), and traditional time-series models such as LSTM have difficulty in capturing the non-uniform interference propagation patterns. Spatially, factors related to the operation of different trains, including headway, departure plan, and operation recovery time, as well as the interaction between adjacent trains, change dynamically with the operation schedule (Ding et al., 2021). Consequently, the delay propagation paths and intensities in the complex operation network are spatio-temporal non-Euclidean and cannot be adequately described through continuous functions in Euclidean space.

Difficulties in capturing coordinated operation status of multiple train clusters. The train operational status is affected by the nonlinear coupling effects of multiple events, and cannot be obtained by simply superimposing the behaviors of individual trains. Additionally, there is a dynamic disturbance coupling effect between adjacent train clusters (Schälicke and Nachtigall, 2025). This effect is manifested in that a train is affected by other trains at its current station, as well as by itself and other trains at multi-order neighboring stations. This indicates that there is spatio-temporal high-order correlation in the coordinated operation of multiple train clusters. However, the existing methods based on simple graphs do not fully represent the delay propagation patterns, they can only describe the chained delay propagation correlations (Antelmi et al., 2023; Gao et al., 2020). These methods essentially constitute a dimensionality reduction projection of real-world multiple train clusters interactions, making them inadequate for characterizing the coordinated operation status among multiple train clusters.

Therefore, to explicitly model the region-level delay propagation status, we consider arrival events as hypergraph nodes and use a mixed hypergraph to model the event interactions. Specifically, we utilize hyperedges with different semantics and structures to model the inter-region and intra-region propagation status of train delays respectively. For the delay propagation within a region, we employ hyperedges to represent the influence of the preceding train clusters on the following trains. To capture the emergent phenomena arising from the coupled operational patterns of train clusters, we utilize directed hyperedges to model these clusters within the region. For example, in Fig. 1(c), we model multiple preceding trains (T_i and T_q) in region r_{23} as the tail node set of the directed hyperedge, and the following train (T_m) as the head node. For the delay propagation across regions, we employ hyperedges to represent the influence of arrival events across regions. We make full use of the event time information and learn the historical representations of events based on sequence learning. This approach is more concise and efficient compared to the method that retains the directed connections between all historical and future events. Hence, these hyperedges are undirected. As shown in Fig. 1(c), we model the upstream and downstream arrival events of each train as undirected hyperedges U_{T_i}, U_{T_q}, and U_{T_m}. Then, we utilize mixed hypergraph convolution to learn the inter-regional and intra-region delay propagation status. Finally, based on the powerful capability of diffusion models in probability distribution modeling (Ho et al., 2020; Song et al., 2010), which not only break free from the parametric distribution assumptions

Table 1
Characteristics and advantages of our regional-level multi-train delay prediction model versus existing models.

Modeling	Existing data-driven models	Our model	Ours advantages
Data structure	Time/event sequence (Huang, Sep 2020; Zhang et al., Nov 2024; Li et al., May 2022; Wu, 2021) or physical graph (Ding et al., 2021; Li et al., 2024).	Multivariate spatio-temporal event hypergraph	Can model the spatio-temporal high-order correlations between events.
Data organization	Usually fix the number of adjacent trains in single train delay prediction models (Zhang et al., Nov 2024; Huang et al., 2021; Huang et al., Apr 2020).	adapt to dynamic operation environments	Can utilize the topological information of dynamic events and capture the global operational information.
Data interaction	Fixed-neighborhood train coordinated operation rules learning in single train delay prediction models (Zhang et al., Nov 2024; Huang et al., 2021; Huang et al., Apr 2020).	Multi-order cascade interactions coordinated operation events learning	Can capture regional-level delay propagation and evolution process and coordinated operation status among multiple train clusters.
Inference	Serial inference of single train delays (Huang, Sep 2020; Zhang et al., Nov 2024; Li et al., May 2022; Wu, 2021) or prediction within fixed time intervals (Ding et al., 2021; Li et al., 2024).	Parallel inference of regional-level multi-train delays	Can improve the efficiency of delay prediction application.

of statistical-based explicit variable models but also demonstrate superior modeling ability for high-dimensional complex distributions, we propose an event hypergraph diffusion process. The model projects multi-train arrival events into an efficient latent space, and capture their underlying generative mechanisms. Extensive experiments on the actual high-speed rail performance datasets (Zhang, et al., 2022) show that our model can achieve excellent performance.

In summary, the contributions of this paper are as follows:

(1) We propose a brand-new Multivariate Event Hypergraph Diffusion (MEHD) model to address the regional-level multi-train delay prediction. MEHD conceptualizes train arrivals as events, utilizes an event mixed hypergraph to model delay propagation and evolution process, and learns multiple train clusters coordinated operation status. Notably, we infer delays by predicting arrival time.
(2) In MEHD, we propose a mixed hypergraph convolution module, employing attention-based undirected and directed hyperedge convolution to extract intra-region and inter-region arrival event features respectively. Meanwhile, we design a prior operational schedule-conditioned attention denoising module to enhance the ability to learn the generation mechanisms of all train arrival events.
(3) Extensive experiments demonstrate the outstanding performance of MEHD, reducing the MAE by 35.41 %, the RMSE by 22.91 %, the MAPE by 14.98 %, and the NLL by 22.75 % on average. It also exhibits good robustness and efficiency. Subsequent experiments demonstrate the unique advantages of MEHD over single-train prediction methods. To the best of our knowledge, this is the first end-to-end model for regional-level multi-train delay prediction.

In the following, Section II reviews literature about train delay prediction methods. Apart from that, relevant hypergraph neural networks and diffusion model are introduced in this section. Section III elaborates the necessary definitions and the proposed method. Section IV illustrates the architecture of the proposed model in detail. An overview of datasets and sufficient experiments are given in Section V. Finally, Section VI concludes this paper.

2. Related work

2.1. Train delay prediction works

2.1.1. Model-driven train delay prediction model

Traditional delay prediction models are usually based on activity diagram (Büker and Seybold, 2012), event graph (Kecman and Goverde, 2013), and Markov assumptions (Spanninger et al., 2021) to describe the train operation process, which are well interpretable and enable explicit modeling of train operation, but have poor generalization and difficulties in handling complex data. In the machine learning-based approach, researchers have considered domain knowledge and expert experience such as knowledge related to train (Barbour et al., Aug 2018), network (Wu, Mar 2022), traveling conflicts (Barbour et al., Aug 2018; Huang et al., Feb 2020), and operational disruptions (Pineda-Jaramillo and Viti, Feb 2023), and predict train delays based on clustering (Huang et al., Sep 2022), decision trees (Lulli et al., 2018; Luo et al., 2022), random forests (Nabian et al., May 2019), SVM (Barbour et al., Aug 2018; Huang et al., Feb 2020), and Bayesian networks (Corman and Kecman, 2018).However, these methods heavily rely on feature engineering and require profound professional knowledge of railway operation, in contrast, deep learning-based methods are able to autonomously extract train operation features from data without relying on explicit domain knowledge injection, and show advantages in prediction accuracy and cross-scenario applicability (Wen et al., Apr 2020; Zhang et al., Nov 2024).

2.1.2. Data-driven train delay prediction model

Existing data-driven methods focus on single-train delay prediction, usually based on sequence modeling (Huang, Sep 2020; Zhang et al., Nov 2024). Among them, some works consider train arrival sequences as synchronized time series. For example, Wen et al. (2020) proved the superiority of time-series modeling. Huang (2020) proposed a delay prediction model based on Long Short-Term Memory (LSTM) to capture the delay propagation among trains and stations respectively. On this basis, Li et al. (2022) focused on the train arrival and departure conflicts at the hub station from a microscopic perspective, and used LSTM and Fully Connected Neural Network (FCNN) to learn operation and environment variables respectively. Huang et al. (2021) regarded the trajectory data of multiple continuously running trains at multiple stations as images and used a Convolutional Neural Network (CNN) to learn the operation features, and the features between adjacent pixels form a temporal-dependent correlation. Based on the above research, Huang et al. (2020) used a 3D-CNN to learn spatio-temporal features, using an LSTM to process time-series variables and an FCNN to learn external factors. Meanwhile, Huang et al. (2022) reveal the delay evolution patterns based on K-means, and applied Bayesian models to different patterns. This method considers the delay of all upstream stations to overcome the Markov assumption. Wu et al. (2021) proposed a hybrid deep learning method that combines LSTM and Critical Point Search (CPS), used LSTM to learn the train running time and dwelling time, and employed CPS to identify initial and knock-on delays according to the delay causes, running delays, dwelling delays, and schedule timetables. However, the train operation process consists of asynchronous events that occur discretely in a continuous time domain. Based on this idea, Zhang et al. (2024) first modeled train operation as an asynchronous event sequence and proposed a Train Arrival Neural Temporal Point Process (TANTPP) model. This model introduced a multi-source dynamic spatio-temporal embedding method, which successfully captured the asynchrony of arrival events and achieved SOTA performance on the actual operation dataset.

In addition, there is also some researches that focuses on predicting the arrival delays of a single train within a certain future time

period, and should first define the prediction time period. Among them, Xu et al. (2023) proposed an non-homogeneous Markov chains model, which aims to predict the delay at the station where the train is scheduled to arrive 20 min later (or near). However, dispatchers and passengers often care more about the specific arrival time at a certain station. But those methods lack flexibility in application because some trains may not have arrived at (or have already left) the next station within the given time period. However, delay prediction models based on synchronous time series or asynchronous event sequences need to organize Euclidean data by fixing the historical time period and the number of adjacent trains to adapt to the model structure, this way disrupts the topological structure of train operation and may cause distortion of local features, especially for train groups with a high density of coordinated operation.

Some studies adopted the Spatio-Temporal Graph Convolutional Network (STGCN) to capture the interactions between trains. For example, Ding et al. (2021) constructed a directed graph with trains as nodes and lines as edges, and used an adjacency matrix to represent influence degree between trains. Li et al. (2024) modeled the train operation network as heterogeneous graph, regarding stations and trains as heterogeneous nodes, and constructed heterogeneous edges between stations, between trains, and between trains and stations, and utilized both heterogeneous and homogeneous GNN to capture the interactions between trains and stations. However, these methods establish correlations between entities rather than arrival events, making it difficult to clearly express the causal correlations of events and to learn complex events triggering mechanisms. Besides, these methods only consider the binary interaction correlation in graph, neglects the coordinated operation status among multiple train clusters. Moreover, as mentioned above, these methods can only predict the arrival situation of trains within a given time periods and lacks flexibility.

Absolutely, at the model structure level, it is difficult for existing methods to capture the regional-level delay propagation and evolution process as well as the coordinated operation status among multiple train clusters. At the inference application level, existing methods cannot achieve end-to-end regional-level delay prediction. Therefore, we propose a brand-new prediction model, aiming to address the above challenges and improve the prediction accuracy and inference efficiency of the regional-level delay event.

2.2. Hypergraph convolution network

2.2.1. Development process of hypergraph convolutional networks

With the growing demand for complex system modeling, traditional graph models exhibit significant limitations in representing multi-body interaction correlations. as compressing them into pairwise correlations leads to substantial information loss (Antelmi et al., 2023; Gao et al., 2020). To address this issue, Zhou et al. (2006) introduced hypergraph theory into machine learning, which not only preserves the non-Euclidean properties of nodes but also provides rich higher-order correlation information. Since hypergraphs are a generalized form of graphs, Hypergraph Neural Networks (HGNNs) extend GNNs. The development encompasses undirected hypergraph networks (Zhou et al., 2006), directed hypergraph networks (Luo et al., 2022), dynamic hypergraph networks (Cao et al., 2024) attention-based hypergraph networks (Georgiev et al., 2201), and mixed hypergraph networks (Li et al., 2024). In recent years, hypergraph has been extensively applied in various fields such as action recognition (Hao et al., 2021), visual analysis (Ye et al., Oct 2019), and natural language processing (Bi et al., 2020).

2.2.2. Applications of hypergraphs in ITS

In ITS, hypergraph theory is also used for modeling Higher-order correlations. For example, Borndörfer et al. (2016) used hypergraphs to represent industrial railway requirements and their correlations. Luo et al. (2022) established a directed hypergraph to represent the spatial correlations in the road network and proposed a directed hypergraph neural network for traffic state prediction. Shen et al. (2024) proposed a new HGNN for origin–destination (OD) flow prediction, constructed multiple hypergraphs based on time series patterns and geographic information to extract the peculiar pattern of metro flow. Wang et al. (2024) introduced a multi-channel hypergraph convolutional network, which captured the spatial high-order correlations and semantic correlations among different OD flow channels and addressed the challenge of data sparsity. Cao et al. (2024) integrated spatio-temporal gated hypergraph convolution, combining spatial patterns with dynamic cross-modal traffic data to improve the accuracy of traffic flow prediction. According to the literature review, hypergraph modeling for OD flow or traffic flow aims to capture the high-order correlations among multiple physically adjacent stations, such as multi-directional interactions at intersections, where inter-station physical connections are explicitly defined.

In the regional-level multi-train delay arrival prediction problem, typical hypergraphs and their variants have difficulty effectively addressing the modeling challenges. Among them, the pure directed hypergraph has difficulty capturing the status of delay propagation across regions, while it will lose the coupled operation patterns of multiple trains within the region. In addition, although the nodes in this paper exhibit different semantics within different hyperedges, they only contain one type of arrival event, so it is still not suitable for the heterogeneous hypergraph structure. Finally, the dynamic hypergraph focuses on discovering the hyperedge structure during the learning process, and it is suitable for scenarios where potential connection relationships between nodes are to be discovered. However, the relationships between arrival events are determined based on the train operation schedule. Therefore, we introduce the mixed hypergraph to model the multi-level propagation status of delays.

2.3. Denoising diffusion probabilistic models

Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020; Song et al., 2010) have been widely applied due to their powerful generative capabilities, with applications spanning text generation (Li et al., 2022), graph generation (Niu et al., 2020), image generation (Austin et al., 2021; Yang, 2024), time-series prediction and imputation (Rasul et al., 2021; Yuan et al., 2023), etc. In recent years, researchers have enhanced sample quality and efficiency by refining the denoising process (Watson et al., 2106; Salimans

and Ho, 2202) and improving the likelihood estimation function (Nichol and Dhariwal, 2021; Kingma et al., 2021). Among them, researchers optimize the denoising process according to the characteristics of domain data. For example, Li et al. (2022) proposed a new language model based on continuous diffusion, progressively denoises a sequence of gaussian noises into word vectors, facilitating the generation of hierarchical continuous latent representations. Meanwhile, Niu et al. (2020) designed a DDPM on graphs by incorporating permutation invariance and edge weight constraints. Yuan et al. (2023) proposed a Spatio-temporal Diffusion Point Processes (DSTPP) that utilizes the spatio-temporal correlations of events to guide the denoising process, significantly improving event generation quality. Xu et al. (2023) introduced Markov chains and equivariance mechanism to ensure that the generated molecular conformations satisfy specific data invariance constraints. Yang et al. (2024) proposed a universal contextual diffusion model by integrating cross-modal interaction and alignment mechanisms into both forward and reverse processes, significantly enhancing the performance of text-guided visual generation model.

In this paper, we apply the diffusion generative model to delay prediction for the first time. We fully consider the characteristics of the regional-level multi-train operation data. During the denoising process, we explicitly incorporate the spatio-temporal coupling correlation between arrival events and the prior train operation schedule to guide the arrival time prediction.

3. Methodology

In this section, we present the necessary concepts, the definitions of the research problem, and the execution process of the MEHD. As shown in Table 2, we provide the variables used in this paper and their specific meanings.

3.1. Problem statement

Definition 1. ***Train operation network and region***:Railway operations have strict scheduling characteristics, and the timetable provides a basis for the train operation network $G = (V, E, A, R, T)$, where $V = \{v_1, v_2, \cdots, v_n\}$ is the set of stations, $E = \{\cdots, e_{pq}, \cdots\} \subseteq V \times V$ is the set of lines, $A \in \mathbb{R}^{n \times n}$ is adjacency matrix, represents station connectivity in the network, $A_{pq} = 0$ indicates no direct connection between v_p and v_q, otherwise there is a connection. $R = \{r_{pq} | A_{pq} = 1\}$ is the set of train operation regions, where each region $r_{pq} = (S_p, e_{pq}, S_q) \in R$ corresponds to a directed line $e_{pq} \in E$ from upstream station S_p to downstream station S_q. And the number of regions $|R|$ equals the number of lines $|E|$. $T = \{T_1, \cdots, T_i, \cdots\}$ represents the set of trains on the train operation network, the stop sequence of train T_i is $\{S_{i1}, \cdots S_{im_i}\}$, and these stops can be adjacent-region or cross-region, m_i is the total number of stops of train T_i. In section r_{pq}, multiple trains $\left\{T^1_{pq}, \cdots, T^g_{pq}\right\}$ operating within it must run in sequence and adhere to the timetable-specified rules for arrival, departure, dwell for overtaking, where $|g(t, pq)|$ represents the number of trains operating in section r_{pq} at time t, and the coupled operation of these trains occurs within the region.

Definition 2. **Train arrival and departure event sequence:** Train scheduled and actual arrival event are formulated as $E^a_{ij,p} = \left\{S_{ij}, t^a_{ij,p}\right\}$ and $E^a_{ij,a} = \left\{S_{ij}, t^a_{ij,a}\right\}$ respectively. Train scheduled and actual departure event are formulated as $E^d_{ij,p} = \left\{S_{ij}, t^d_{ij,p}\right\}$ and $E^d_{ij,a} = \left\{S_{ij}, t^d_{ij,a}\right\}$ respectively, where $T_i \in T, S_j \in V$, $t^a_{ij,p}, t^a_{ij,a}, t^d_{ij,p}, t^d_{ij,a}$ represent the corresponding event occurrence time respectively. The multi-train scheduled and actual arrival event sequence are formulated as $Plan^a = \left\{\mathbb{P}^a_1, \mathbb{P}^a_2, \cdots, \mathbb{P}^a_n\right\}$ and $Act^a_t = \left\{\mathbb{H}^{t,a}_1, \cdots, \mathbb{H}^{t,a}_n\right\}$, $\mathbb{P}^a_i = \left\{E^a_{i1,p}, \cdots, E^a_{im_i,p}\right\}$ and $\mathbb{H}^{t,a}_i = \left\{E^a_{i1,a}, \cdots, E^a_{ia_i,a}\right\}$ denote the scheduled and arrival sequence of T_i respectively, where $t > t^a_{ia_i,p}$ is the query time, $a_i \leq m_i$ is the number of actual stops, n is the total number of trains. Correspondingly, we define the multi-train scheduled and actual departure event sequence as $Plan^d = \left\{\mathbb{P}^d_1, \mathbb{P}^d_2, \cdots, \mathbb{P}^d_n\right\}$ and $Act^d_t = \left\{\mathbb{H}^{t,d}_1, \cdots, \mathbb{H}^{t,d}_n\right\}$, $\mathbb{P}^d_i = \left\{E^d_{i1,p}, \cdots, E^d_{i(m_i-1),p}\right\}$ and $\mathbb{H}^{t,d}_i = \left\{E^d_{i1,a}, \cdots, E^d_{i(a_i-1),a}\right\}$ denote

Table 2
Variables involved in this paper and their specific meanings.

Variables	Variable meanings	Variables	Variable meanings
G	train operation network	$\{S_{i1}, \cdots S_{im_i}\}$	the stop sequence of train T_i
r_{pq}	train operation region	m_i	the total number of stops of train T_i
$\left\{T^1_{pq}, \cdots, T^g_{pq}\right\}$	multiple trains in region r_{pq}	$E^a_{ij,p}$	scheduled arrival event of train T_i at station S_j
$E^a_{ij,a}$	actual arrival event of train T_i at station S_j	$E^d_{ij,p}$	scheduled departure event of train T_i at station S_j
$E^d_{ij,a}$	actual departure event of train T_i at station S_j	$t^a_{ij,p}, t^a_{ij,a}$	scheduled and actual arrival event occurrence time
$t^d_{ij,p}, t^d_{ij,a}$	scheduled and actual departure event occurrence time	$Plan^a, Act^a_t$	multi-train scheduled and actual arrival event sequence
$Plan^d, Act^d_t$	multi-train scheduled and actual departure event sequence	Fut^a_t, Fut^d_t	the future scheduled arrival and departure sequence
a_i	the number of actual stops of train T_i	n	the total number of trains
$\mathscr{H}$	train arrival event mixed hypergraph	U, D	the set of undirected and directed hyperedges
$\mathbb{C}_{ij}$	the set of influencing events of v_{ij}	Δt_{th}	headway between non-simultaneous departure-arrival at the station
H	the incidence matrix of U	H^{head}, H^{tail}	the head and tail incidence matrix of D
$\widehat{F}_t$	the predicted arrival time of trains at subsequent stations	$g_\theta(\bullet)$	the MEHD operator

the scheduled and arrival sequence of T_i respectively. The future scheduled arrival and departure sequence are formulated as $Fut_t^a = \left\{\mathbb{F}_1^{t,a}, \mathbb{F}_2^{t,a}, \cdots, \mathbb{F}_n^{t,a}\right\}$ and $Fut_t^d = \left\{\mathbb{F}_1^{t,d}, \mathbb{F}_2^{t,d}, \cdots, \mathbb{F}_n^{t,d}\right\}$, where $\mathbb{F}_i^{t,a} = \left\{E^p_{i(a_i+1),a}, \cdots, E^p_{im_i,a}\right\}$, $\mathbb{F}_i^{t,d} = \left\{E^d_{i(a_i+1),a}, \cdots, E^d_{i(m_i-1),a}\right\}$ and $t^p_{i(a_i+1)} > t$.

Definition 3. ***Train arrival event mixed hypergraph***:Let $\mathscr{H} = \left(\mathscr{V}, U, D, H, H^{head}, H^{tail}\right)$ denotes the train arrival event mixed hypergraph, where $\mathscr{V}$ denotes the set of scheduled arrival event nodes, and $v_{ij} = E^a_{ij,p}$ denotes that T_i arrives at S_j at time t_{ij}, and simply denoted as $\left(t_{ij}, S_{ij}\right)$ hereinafter. $U = \{U_{T_1}, \cdots, U_{T_M}\}$ and $D = \left\{D_{T_1S_1}, \cdots, D_{T_iS_j}, \cdots\right\}$ denotes the set of undirected and directed hyperedges respectively. $U_{T_i} = \{v_{i1}, \cdots, v_{im_i}\} \in U$ contains all the arrival events of train T_i, $D_{T_iS_j} = \left\{v_{ij}, v_k \middle| v_k \in \mathbb{C}_{ij}\right\} \in D$ contains the event v_{ij} and all the influencing events of v_{ij}, we denote the set of influencing events of v_{ij} as $\mathbb{C}_{ij}$. If the scheduled departure time $t^p_{ij,d}$ of train T_i at station S_j is earlier than the scheduled arrival time $t^p_{kj,a}$ of train T_k at station S_j, and $t^p_{kj,a} - t^p_{ij,d} \le \Delta t_{th}$, then we consider that v_{kj} may have an impact on v_{ij}, that is, $v_{kj} \in \mathbb{C}_{ij}$. Δt_{th} is the headway between non-simultaneous departure-arrival at the station, which represents the minimum interval from the moment a train leaves a station to the moment another train arrives at the same station. In essence, this is a constraint on the station route. The path from one track to another track or platform can only be occupied by one train within a certain period. For the preceding trains that have reached the terminal station, we use the actual arrival time to replace the departure time. $H \in \mathbb{R}^{N\times M}$ represents the incidence matrix of U, $H^{head} \in \mathbb{R}^{N\times M}$ and $H^{tail} \in \mathbb{R}^{N\times M}$ represent the head and tail incidence matrix of D respectively.

Problem Formulation: On the daily operation network, for the query time t, given the regional-level multi-train scheduled arrival event sequence $Plan = \left\{Plan^a, Plan^d\right\}$, the historical actual arrival and departure event sequence $Act_t = \left\{Act_t^a, Act_t^d\right\}$, the future arrival and departure sequence $Fut_t = \left\{Fut_t^a, Fut_t^d\right\}$, and fully considering the network G constraints affecting the train operation. The aim is to predict the arrival time of each train at subsequent stations $\widehat{F}_t = \{\widehat{t}_{i(a_i+1)}, \cdots, \widehat{t}_{im_i}\}$, that is:

$$\widehat{F}_t = g_\theta(Plan, Act_t, Fut_t, G) \tag{1}$$

Where, g_θ is the MEHD operator. Meanwhile, the real-time scheduling of train operation strategies and the dynamic constraints have a significant impact on the train operation status, so this paper focuses more on short-term prediction (Caimi et al., Nov 2012).

3.2. MEHD

In this section, we model the regional-level multi-train delay prediction problem as a multivariate event hypergraph diffusion process, and establish an efficient and unified learning strategy for Conditional Probability Density (CPD) of events to capture the triggering mechanisms of multi-train arrival events. The prediction diagram of MEHD is illustrated in Fig. 2.

MEHD consists of a forward noising process and a multivariate hypergraph conditional denoising process, for the train arrival event set $\mathscr{V}$, we train the parameters by maximizing the likelihood of the CPD:

$$log p_\theta(\mathscr{V}_0 | H_{\mathscr{V}}, \mathscr{E}) \tag{2}$$

where $\mathscr{V}_0$ denotes the result of the final generation step, $H_{\mathscr{V}}$ denotes the historical influencing events, that is the historical arrival events within one train and the events between adjacent trains. and $\mathscr{E}$ stands for the hypergraph structure.

In the forward noising process, we iteratively add noise into the event distribution independently (Ho et al., 2020), thereby transforming the original event distribution $\mathscr{V}_0$ into a standard normal distribution $\mathscr{V}_K$. The forward diffusion process is defined by a Markov chain, that is, the noise addition for each arrival event is performed independently.

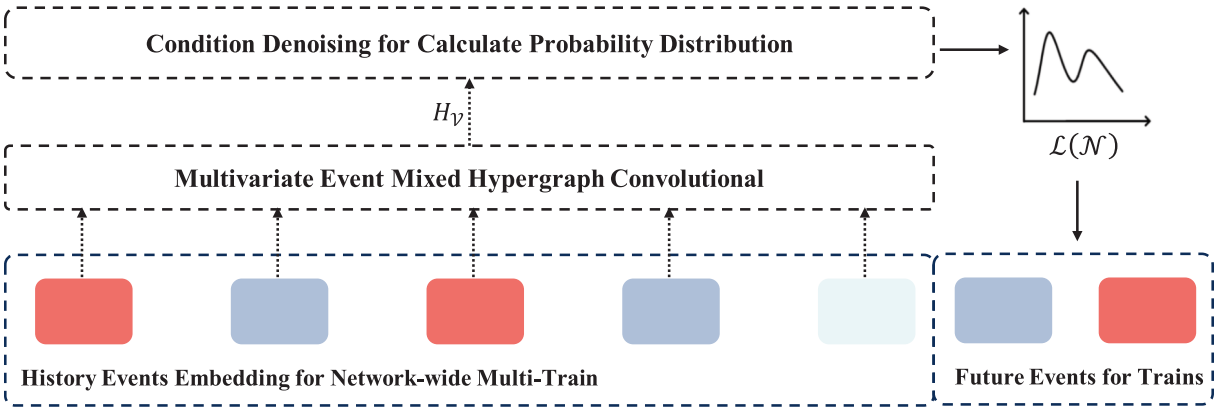


Fig. 2. The prediction diagram of MEHD.

$$q(\mathscr{V}_k|\mathscr{V}_{k-1}) = \prod_{v_{ij}\in\mathscr{V}} q(v_{ij,k}|v_{ij,k-1}) \tag{3}$$

where $\mathscr{V}_k$ denotes the event distribution at the *k-th* diffusion step. The conditional diffusion distribution $q(v_{ij,k}|v_{ij,k-1})$ for event v_{ij} is defined as a gaussian distribution. As shown in Fig. 1(e), we iteratively add Gaussian noise to node v_{m3} independently. At the *k-th* step, that is:

$$q(v_{ij,k}|v_{ij,k-1}) = N\left(v_{ij,k}; \mu(v_{ij,k-1}); \sum(v_{ij,k-1})\right) \tag{4}$$

In the reverse process, we start from the noise distribution $\mathscr{V}_K$ and conduct conditional denoising, progressively generating the event distribution $v_{ij,0}$ at historical diffusion steps. The conditions include the historical arrival events within one train, the events between adjacent trains, future train operation plans, external factors, etc. Here, we use the hypergraph structure and event representation to represent them.

$$p_\theta(\mathscr{V}_0|H_{\mathscr{V}}, \mathscr{E}) = \int p(\mathscr{V}_K)\prod_{k=1}^{K} p_\theta(\mathscr{V}_{k-1}|\mathscr{V}_k, H_{\mathscr{V}}, \mathscr{E})d(\mathscr{V}_{1:K}) \tag{5}$$

$$p_\theta(\mathscr{V}_{k-1}|\mathscr{V}_k, H_{\mathscr{V}}, \mathscr{E}) = \prod_{v_{ij}\in\mathscr{V}} p_\theta(v_{ij,k-1}|v_{ij,k}, h_{ij}, \mathscr{E}) \tag{6}$$

where, θ denotes the model parameters, $\mathscr{E}$ is the hypergraph structure, guides the learning of the event representation. The conditional diffusion distribution $p_\theta(v_{ij,k-1}|v_{ij,k}, h_{ij}, \mathscr{E})$ for event v_{ij} is defined as a gaussian distribution. As shown in Fig. 1(e), we iteratively input the current diffusion step and the noise into the noise prediction model starting from pure Gaussian noise, and obtain a clean prediction result at the last step. At the *k-th* step, that is:

$$p_\theta(v_{ij,k}|v_{ij,k-1}, h_{ij}, \mathscr{E}) = N\left(v_{ij,k-1}; \mu_\theta(v_{ij,k}, h_{ij}, \mathscr{E}); \sum\nolimits_\theta(v_{ij,k}, h_{ij}, \mathscr{E})\right) \tag{7}$$

where, h_{ij} denotes the event representation of v_{ij}, and h_{ij} is obtained by the mixed hypergraph convolutional module designed in Section 4.3. For example, following the information aggregation diagram depicted in Fig. 1(d), we utilize v_{m1} and v_{m2} within the hyperedge U_{T_m}, as well as v_{i3} and v_{q3} within the hyperedge $D_{T_mS_3}$, to obtain the feature representation of v_{m3}.

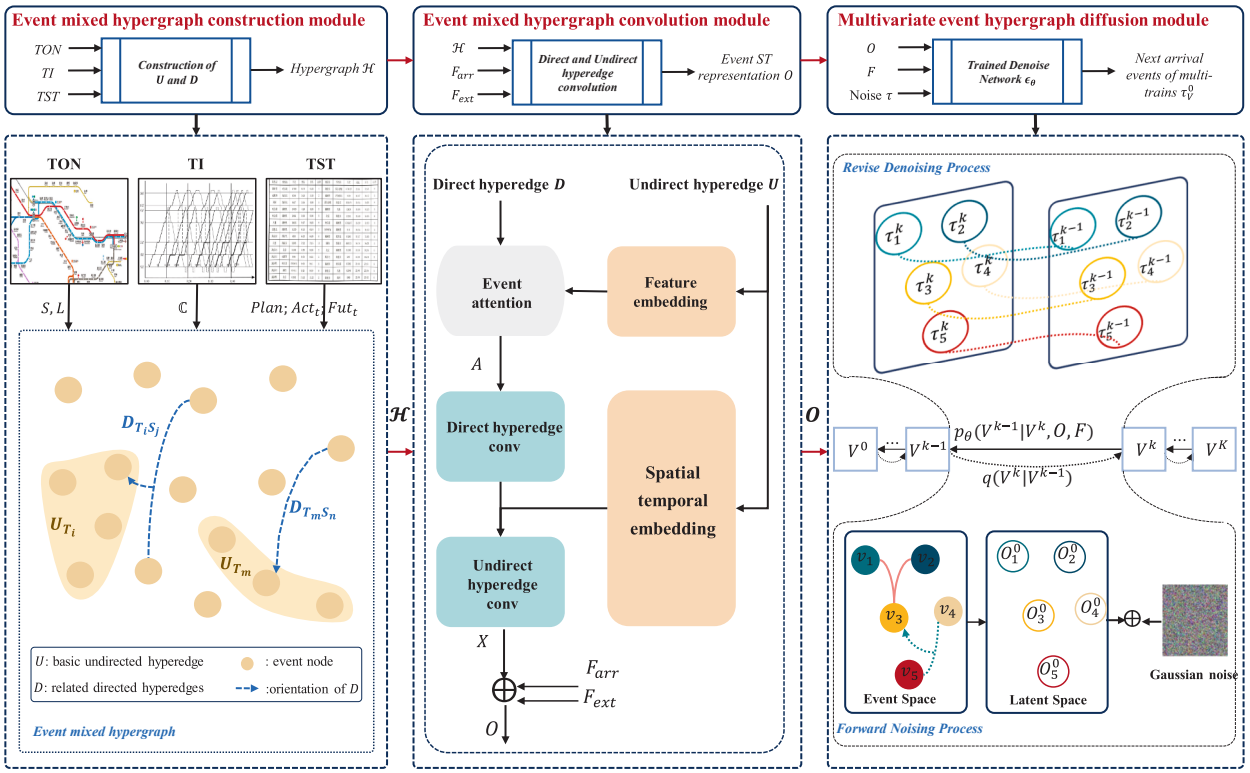


Fig. 3. Overall Architecture of MEHD. TON: Train Operation Network, TI: Train Interlocking, TST: Train Schedule Timetable.

4. MEHD model

4.1. Overall architecture

Fig. 3 shows the overall architecture of MEHD, which includes the following modules: the event mixed hypergraph construction module, the event mixed hypergraph convolution module, and the multivariate event hypergraph diffusion module. Among them, the first module constructs the hypergraph $\mathscr{H} = (\mathscr{V}, U, D)$ based on the train operation network topology and the real-time coordinated operation correlation, and for details of this construction, refer to section 4.2. The second module extracts the event representation O_t through mixed hypergraph convolution, to provide condition for the diffusion process, and the details is given in Section 4.3. The last module conducts diffusion denoise process on the hypergraph to learn the triggering mechanism of multi-train arrival events in Sections 4.4 and 4.5.

4.2. Event mixed hypergraph construction module

As the high-order correlation between events is different, hyperedges have various semantics. For each train arrival sequence $\mathbb{P}_i$, we construct an undirected hyperedge U_{T_i} that contains all the arrival events of train T_i, representing the dependencies among this sequence. Since the stops generally remain unchanged during one single timetable planning cycle, so the structure of U_{T_i} is static. For each event v_{ij}, we construct a directed hyperedge $D_{T_iS_j}$, where the set of head nodes consists of the events in $\mathbb{C}_{ij}$, and the event v_{ij} serves as the tail node, utilizes the information flow of directed edges to capture the influence of adjacent train. Affected by operational schedule and dynamic coordination correlations, this structure is dynamic. The complete set of hyperedges forms a multivariate event mixed hypergraph, which models higher-order correlations among events.

Construction of basic Undirected hyperedges. To model dependencies within one train arrival sequence, the hyperedge U_{T_i} incorporates all arrival event nodes $\{v_{i1}, v_{i2}, \cdots, v_{im_i}\}$ of the train T_i, and $t_{i1} < v_{i2} < \cdots < t_{im_i}$. We construct an incidence matrix to describe the correlation between events and undirected hyperedges:

$$H(v, u_i) = \begin{cases} 1, if v \in u_i \\ 0, if v \notin u_i \end{cases} \tag{8}$$

where, the representation of U_{T_i} is $\mathbb{x}_i^b = \{\mathscr{X}_{i,1}, \mathscr{X}_{i,2}, \cdots, \mathscr{X}_{i,m_i}\} \in \mathbb{R}^{m_i \times F}$, $\mathscr{X}_{i,j} \in \mathbb{R}^F$ represents the feature of node v_{ij}. As shown in Fig. 1 (b), according to Definition 3, we construct three undirected hyperedges U_{T_i}, U_{T_M}, and U_{T_Q}, respectively. Each hyperedge contains all the arrival events of the corresponding train, where, $U_{T_i} = \{v_{i1}, v_{i2}, v_{i3}, v_{i4}, v_{i5}\}$, $U_{T_M} = \{v_{m1}, v_{m2}, v_{m3}, v_{m4}, v_{m5}\}$, and $U_{T_Q} = \{v_{q2}, v_{q3}, v_{q4}\}$.

Construction of related Directed hyperedges. To model the coordinated operation between adjacent trains, the hyperedge $D_{T_iS_j}$ incorporates the event v_{ij} and its adjacent event nodes $\{v_{ij}\} \cup \{v_k | v_k \in \mathbb{C}_{ij}\}$. We construct a head and tail incidence matrix to describe the correlation between events and directed hyperedges, respectively:

$$H^{head}(v, d_i) = \begin{cases} 1, if v \in d_i^{head} \\ 0, if v \notin d_i^{head} \end{cases} \tag{9}$$

$$H^{tail}(v, d_i) = \begin{cases} 1, if v \in d_i^{tail} \\ 0, if v \notin d_i^{tail} \end{cases} \tag{10}$$

where, the representation of $D_{T_iS_j}$ is $\mathbb{x}_{i,j}^r = \{\mathscr{X}_{i,j,1}, \mathscr{X}_{i,j,2}, \cdots, \mathscr{X}_{i,j,K_{ij}}\} \in \mathbb{R}^{K_{ij} \times F}$, the element $\mathscr{X}_{i,j,k} \in \mathbb{R}^F$ represents the feature of node v_{ij}^k. K_{ij} is the length of $\mathbb{x}_{i,j}^r$. As shown in Fig. 1(b), according to Definition 3, in r_{23}, if trains satisfy the conditions $t_{m3,a}^p - t_{i3,d}^p \le \Delta t_{th}$ and $t_{m3,a}^p - t_{q3,d}^p \le \Delta t_{th}$, then let $\mathbb{C}_{m3} = \{v_{i3}, v_{q3}\}$, then construct the directed hyperedge $D_{T_mS_3} := \{v_{i3}, v_{q3} \to v_{m3}\}$. Similarly, we construct $D_{T_qS_4} := \{v_{i4} \to v_{q4}\}$ and $D_{T_mS_4} := \{v_{q4} \to v_{m4}\}$.

4.3. Event mixed hypergraph convolution module

In the multivariate event mixed hypergraph convolutional module, we first encode the spatiotemporal locations of arrival events, then update the feature representations of event nodes by attention-based mixed hypergraph convolution, thereby capturing both historical event dependencies within one train and coordinated influences from adjacent trains.

Event embedding layer To obtain unique representations for each arrival event $v = (t,s)$, we encode the event time by the positional encoding (Vaswani, 2017), and employ a linear layer $x_s = W_x s$ to encode stations $s \in \mathbb{R}^n$. Here we use a simplified event representation.

$$[x_t]_j = \begin{cases} cos\left(t \Big/ 10000^{\frac{j-1}{M}}\right) if j is odd \\ sin\left(t \Big/ 10000^{\frac{j-1}{M}}\right) if j is even \end{cases} \tag{11}$$

where, x_t denotes the time embedding, M is the embedding dimension, and $W_x \in \mathbb{R}^{M\times n}$ is learnable parameters, n depends on the station location representation, we use latitude and longitude to represent location in this paper, hence $n = 2$. And we obtain event encoding through summation, for a hyperedge $e = \{(t_i, s_i)\}_{i=1}^{L}$, the encoding is $e_{st} = \{e_{st,1}, e_{st,2}, \cdots, e_{st,L}\} \in \mathbb{R}^{L\times M}$, where $e_{st,i} = e_{t,i} + e_{s,i}$. Additionally, to address issues where spatiotemporal coupling between some trains may be weak, we retain time and location encodings separately, that is, $e_t = \{e_{t,1}, e_{t,2}, \cdots, e_{t,L}\}$ and $e_s = \{e_{s,1}, e_{s,2}, \cdots, e_{s,L}\}$. Then, we concatenate the e_{st}, e_t, e_s, x_o to obtain the features of the event nodes X for hypergraph convolution, where, x_o represents the train operation features, such as the scheduled arrival time, the scheduled departure time, and the train speed limit.

Direct hyperedge convolution layer The multivariate event mixed hypergraph convolution follows a two-stage message-passing paradigm, iteratively aggregating node representations and hyperedge representations (Zhou et al., 2006; Li et al., 2024). As shown in Fig. 1(d), which is a schematic diagram of information aggregation flow in the convolution, we aggregate the node features of the tail nodes v_{i3} and v_{q3} based on Formula (12) to update the representation of the hyperedge $D_{T_m S_3}$, that is ①. Then, we use the hyperedge representation to update the head node v_{m3} based on Formula (13), that is ②.

It aggregates node features within directed hyperedges to capture the coordinated influence of adjacent trains. At the *l-th* layer, the convolution is defined as:

$$D^{(l+1)} = D_h^{tail^{-1}} H^{tail^T} A X^{(l)} W_1 \tag{12}$$

where $X^{(l)} \in \mathbb{R}^{N\times F_1}$ is the node feature at layer l, $H^{tail} \in \mathbb{R}^{N\times M_2}$, and $W_1 \in \mathbb{R}^{F_1\times F_2}$ are the learnable parameters. F_1 and F_2 are the feature dimensions at layer l and $l+1$, respectively, and M_2 is the number of directed hyperedges.

Then, we apply nonlinear transformation to directed hyperedge to update the node representations:

$$X_D^{(l+1)} = D_v^{head^{-1}} H^{head} W_3 D^{(l+1)} W_4 \tag{13}$$

where, $H^{head} \in \mathbb{R}^{N\times M_2}$, $W_3 \in \mathbb{R}^{M_2\times M_2}$, and $W_4 \in \mathbb{R}^{F_1\times F_2}$ are learnable parameters. $D_h^{tail} \in \mathbb{R}^{M_2\times M_2}$ is the tail degree diagonal matrix of hyperedges, and $D_v^{head} \in \mathbb{R}^{N\times N}$ is the head degree diagonal matrix of nodes, computed as follows:

$$D_h^{tail}(j,j) = \sum_{i=1}^{N} H^{tail}(i,j) D_v^{head}(i,i) = \sum_{j=1}^{M} W(j,j).H^{head}(i,j) \tag{14}$$

Undirect hyperedge convolution layer To capture the temporal relationships of events within one train, we employ Transformer (Vaswani, 2017) to aggregate node features within undirected hyperedges. It is entirely feasible and has significant advantages to use Transformer to model the temporal relationships of events, its self-attention mechanism can dynamically learn the complex temporal dependencies between events, effectively capture long-range temporal correlations. Moreover, since the lengths of the arrival sequences of different trains may vary, and Transformer is naturally suitable for variable-length sequences, it provides more flexible temporal modeling capabilities for the hypergraph structure than fixed aggregation functions.

$$X_U^{(l+1)} = Transformer\left(A; X^{(l)}\right) \tag{15}$$

$$A(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d}}\right) \tag{16}$$

where, $A \in \mathbb{R}^{N\times N}$ denotes attention weight matrix, $Q = X^{(l)}W^Q$, $K = X^{(l)}W^K$, $V = X^{(l)}W^V$, W^Q, W^K, W^V are learnable parameters. As shown in Fig. 1(d), within the undirected hyperedge U_{T_m}, we aggregate all historical arrival events v_{m1} and v_{m2} to obtain the

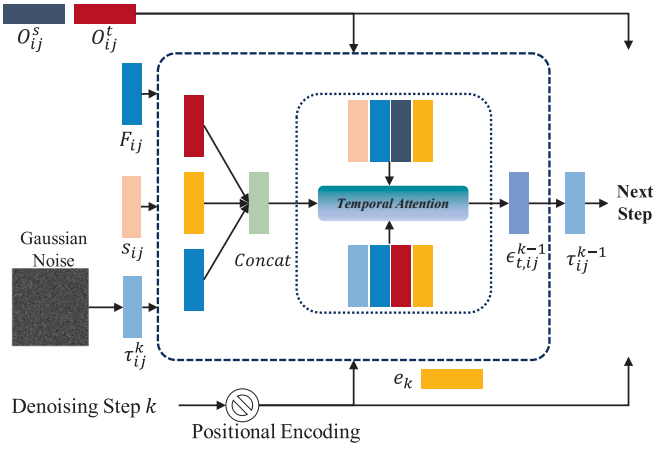


Fig. 4. Prior operational schedule-conditioned attention denoising.

representation of the current arrival event v_{m3}.

For the event spatiotemporal encodings e_t, e_s, and e_{st}, we use the aforementioned convolution operations to derive the temporal representation X_U^t, spatial representation X_U^s, and spatiotemporal representation X_U^{st}. To obtain a comprehensive spatiotemporal representation, we further integrate event features X_D from directed hyperedges:

$$X^* = W_D X_D + W_U X_U^* \tag{17}$$

where $W_D \in \mathbb{R}^{N\times N}$, and $W_U \in \mathbb{R}^{N\times N}$ are learnable parameters, M is the embedding dimension. Finally, we incorporate the train and section information F_{arr}, and external factors F_{ext} to derive final representations for each event (Zhang, et al., 2022):

$$O = MLP(F_{arr}||F_{ext})||X \tag{18}$$

where $X = [X^s||X^t||X^{st}] \in \mathbb{R}^{N\times 3^*M}$, train and section information includes features such as station grades and regional mileage, while the external factors include meteorological and holiday information.

4.4. Multivariate event hypergraph diffusion module

Based on Section 3.2, this section focuses on the forward process $q(v_{ij,k}|v_{ij,k-1})$ and the revise process $p_\theta(v_{ij,k-1}|v_{ij,k}, h_{ij}, \mathscr{E})$ for individual event. Specifically, for each event $v_{ij} = (\tau_{ij}, s_{ij})$, where τ_{ij} denotes the inter-event interval, we gradually add gaussian noise to v_{ij}^0 until it is corrupted into pure gaussian noise v_{ij}^K, the noise is independently applied to pixel-level features, and the forward process is modeled as a Markov process in both temporal and spatial domains.

$$q_{st}\left(v_{ij}^k|v_{ij}^{k-1}\right) := \left(q\left(\tau_{ij}^k|\tau_{ij}^{k-1}\right)|q\left(s_{ij}^k|s_{ij}^{k-1}\right)\right) \tag{19}$$

$$q\left(x^k|x^{k-1}\right) := N\left(x^k; \sqrt{1-\beta_k}\,x^k, \beta_k I\right) \tag{20}$$

where, $\alpha_k = 1-\beta_k$, $\overline{\alpha}_k = \prod_{i=1}^{k}\alpha_k$.

Specifically, unlike earthquake or virus outbreak event, the future schedule is predetermined, is closely linked to the actual arrival, so the event generation process must fully consider the prior schedule. As shown in Fig. 4, we designed a prior operational schedule-conditioned attention denoising method, which conditions on both the future schedule F_{ij} and the historical event spatio-temporal representation O_{ij}, while capturing the coupling correlation between temporal and spatial domains. Therefore, we iteratively denoise from v_{ij}^K to v_{ij}^0 to learn the reconstruction process of the event v_{ij}:

$$p_\theta\left(v_{ij}^{k-1}|v_{ij}^k, O_{ij}, F_{ij}\right) = p_\theta\left(\tau_{ij}^{k-1}|\tau_{ij}^k, O_{ij}, F_{ij}\right) \tag{21}$$

Specifically, at each diffusion step, we perform conditional spatio-temporal attention denoising, the parameters are calculated as follows:

$$e_k = SinusoidalPosEmb(k) \tag{22}$$

$$\alpha = Softmax\left(W_{ta}Concat\left(F_{ij}, O_{ij}^t, e_k\right) + b_{ta}\right) \tag{23}$$

where $F_{ij} = MLP\left(F_{s_i}||F_{\tau_i}\right)$. F_{s_i} and F_{τ_i} denote the future stop information and timetable-based arrival schedule, respectively. W_{ta} and b_{ta} are learnable parameters. It is worth noting that this attention mechanism is applied to different diffusion steps of certain event, aiming to precisely guide the direction of denoise at each step. In contrast, the attention mechanism in Section 4.3 works between different events, aiming to extract accurate representations of events with complex spatio-temporal dependencies from the historical events.

Then, we integrate the spatio-temporal conditions $O_{ij} = \left\{O_{ij}^t, O_{ij}^s\right\}$ and F_{ij} into the predicted values for the $k+1$ step as follows:

$$x_{ij,t}^k = \sigma\left(W_t\tau_{ij}^{k+1} + W_f F_{ij} + W_{ot}O_{ij}^t + b_t + e_k\right) \tag{24}$$

$$x_{ij,s} = \sigma\left(W_s s + W_f F_{ij} + W_{os}O_{ij}^s + b_s + e_t\right) \tag{25}$$

where W_f, W_t, W_{ot}, W_{os}, b_t, and b_s are learnable parameters, and σ denotes the activation function. The output noise at step k is given by:

$$\epsilon_{t,ij}^k = \sum \alpha . x_{ij}^k \tag{26}$$

where $x_{ij}^k = \left[x_{ij,t}^k, x_{ij,s}\right]$. $\epsilon_{t,ij}^k$ represents the predicted noise for event v_{ij} at step k. According to Eq. (29), we can derive the predicted value τ_{ij}^k at step k. In the next step, the predicted value is fed back into the denoising network, and iteratively approximates the temporal

ground-truth.

4.5. Training and inference

Training we optimize the parameters θ by maximizing the log-likelihood to fit the spatio-temporal distribution of arrival events:

$$\sum_{i=1}^{L}\sum_{j=1}^{m_i} log p_\theta\left(v_{ij}^0 | O_{ij}, F_{ij}\right) \tag{27}$$

According to Reference (Ho et al., 2020), we train the model by the following loss function. For event v_{ij} and diffusion step k, the function is defined as:

$$L = \mathbb{E}_{v_{ij}^0, \epsilon, k}\left[\left|\left|\epsilon - \epsilon_\theta\left(\sqrt{\overline{\alpha}_k}\, v_{ij}^0 + \sqrt{1-\overline{\alpha}_k}\,\epsilon, O_{ij}, F_{ij}, k\right)\right|\right|^2\right] \tag{28}$$

where $\epsilon \sim \mathcal{N}(0, I)$, $\beta_1, \cdots, \beta_K \in (0,1)$ is a given increasing variance schedule, representing a particular noise level, $\alpha_k = 1 - \beta_k$, $\overline{\alpha}_k = \prod_{i=1}^{k}\alpha_k$, F_{ij} is the future schedule representation and O_{ij} is the historical event spatio-temporal representation, v_{ij}^0 is the clean observation. we train the model composed of mixed hypergraph convolution and diffusion models in an end-to-end manner. The pseudocode of the training procedure is shown in *Algorithm 1*.

Inference Using the trained MEHD model to predict future arrival events, we first obtain the hidden representations O of all historical arrival events, and extract events F from the future schedule. Conditioned on O and F we predict the next event starting from gaussian noise $\tau_{ij}^K \sim \mathcal{N}(0, I)$:

$$\tau_{ij}^{k-1} = \frac{1}{\sqrt{\alpha_k}}\left(\tau_{ij}^k - \frac{\beta_k}{\sqrt{1-\overline{\alpha}_k}}\epsilon_\theta\left(v_{ij}^k, O_{ij}, F_{ij}, k\right)\right) + \sqrt{\beta_k} z_t \tag{29}$$

where z_t is an auxiliary variable sampled from the standard gaussian distribution, and ϵ_θ denotes the trained denoising network, and the meaning of β_k, α_k and $\overline{\alpha}_k$ is the same as that described in the previous section. The pseudocode for inference procedure is shown in *Algorithm 2*.

Algorithm 1 Training Procedure for MEHD

Target: *Train the hypergraph convolution module and the denoising network in an end-to-end manner.*
Input: $\mathscr{H}$, F_{arr}, F_{ext}, *future operation schedule* F
1. $O \leftarrow formula(11)-(18)$ // *Get the representation of the event hypergraph network.*
2. for U_{T_i} in $\mathscr{H}$ **do** // *Loop through each train.*
3. Repeat:
4. $v_{ij}^0\ q\left(v_{ij}^0 \in U_{T_i}\right)$ // *Sample clean train arrival events.*
5. $k\ Uniform(1, 2, \cdots, K)$ // *Randomly sample diffusion steps.*
6. $\epsilon \sim \mathcal{N}(0, I)$ // *Sample standard Gaussian noise.*
7. $v_{ij}^k = \sqrt{\overline{\alpha}_k} v_{ij}^0 + \sqrt{1-\overline{\alpha}_k}\epsilon$ // *Construct noisy samples.*
8. Take gradient descent step on:
9. $\nabla_{\phi,\theta}\left[\left|\left|\epsilon - \epsilon_\theta\left(v_{ij}^k, O_{ij}, F_{ij}, k\right)\right|\right|^2\right]$ // *Train the network through MAE.*
10. Until: Converged
Output: *The hypergraph convolution network and the denoising network* ϵ_θ

Algorithm 2 Inference Procedure for MEHD

Target: *Generate data samples from noise by the trained network.*
Input: $\mathscr{H}$, F_{arr}, F_{ext}, *future operation schedule* F, *well-trained* ϵ_θ, K
1. $O \leftarrow formula(11)-(18)$ // *Get the representation of the event hypergraph network.*
2. for U_{T_i} in $\mathscr{H}$ **do** // *Loop through each train.*
3. **for** v_{ij} in U_{T_i} **do** // *Loop through each arrival event to be predicted.*
4. $\tau_{ij}^K \sim \mathcal{N}(0, I)$ // *Sample standard Gaussian noise.*
3. for $k = K$ to 1 **do** // *Loop through denoising steps.*
4. $z_t \sim \mathcal{N}(0, I) If k > 1 else z_t = 0$ // *Sample auxiliary noise.*
5. $\epsilon_{t,ij}^k = \epsilon_\theta\left(v_{ij}^k, O_{ij}, F_{ij}, k\right)$ // *Predict the noise at the current step.*
6. $\tau_{ij}^{k-1} = \frac{1}{\sqrt{\alpha_k}}\left(\tau_{ij}^k - \frac{\beta_k}{\sqrt{1-\overline{\alpha}_k}}\epsilon_{t,ij}^k\right) + \sqrt{\beta_k} z_t$ // *Perform denoising update.*
7. end for
7. end for
8. end for
Output: *The set of arrival times* $\left\{\tau_{ij}^0\right\}$ *of all events to be predicted.*

5. Experiments and evaluations

In this section, we perform experiments to answer the following research questions:
RQ1: How does the proposed model perform compared with existing baseline approaches?
RQ2: Is the key component of the proposed model effective?
RQ3: Does the mixed hypergraph convolution successfully capture the high-order correlations in train operation network?
RQ4: Does the proposed model have unique advantages in regional-level multi-train prediction scenarios?

5.1. Experimental setup

5.1.1. Datasets

We utilize an actual high-speed rail operation dataset (Zhang, et al., 2022) from China's railway ticketing system, covering the period from October 8 to December 31, 2019. The dataset includes train operation records, station information, and external factors such as weather, temperature, wind direction, and holiday schedules. We select two representative rail lines, including J-G Line (China's most critical north–south rail artery) and the J-H Line (a primary route connecting Beijing and Shanghai), both ranking among the nation's busiest and most important corridors. Table 3 provides a summary of these two railway lines, the arrival delay is abbreviated as AD.

5.1.2. Data preprocess

The data is categorized into: (1) Actual operations data, including train, station, scheduled/actual arrival/departure times, and directions; (2) Station-route data, including station types, mileage, and sections; (3) External factors, including precipitation, temperature, and holidays. Data preprocessing involved: (1) Removing outlier delays caused by equipment failures which depends more on repairs time and interim decisions; (2) Mapping non-numeric data to low-dimensional spaces, which is more efficient than one-hot encoding; (3) Standardizing all inputs to reduce feature-scale bias.

5.1.3. Baselines

We compared and evaluated the MEHD model with a large number of advanced baselines, including deterministic baseline, probabilistic temporal baseline, and probabilistic spatio-temporal baseline:

Deterministic baseline:We include the best deep learning and machine learning models widely used in train delay prediction, including random forest (RF), bayesian network (BN), long short term memory (LSTM), gated recurrent unit (GRU), CLF-NET (Huang et al., Apr 2020), and FCLL-NET (Huang, Sep 2020).

Probabilistic temporal baseline:We include state-of-the-art temporal baselines, including transformer Hawkes process (THP) (Zuo et al., 2020), the recurrent marked temporal point process (RMTPP) (Du et al., 2016) as baseline methods.

Probabilistic Spatiotemporal baseline:We include state-of-the-art spatiotemporal baselines, including Neural Spatio-temporal Point Process (NSTPP) (Chen et al., 2020), DeepSTPP (Zhou et al., 2022), and Train Arrival Neural Temporal Point Process (TANTPP) (Zhang et al., Nov 2024).

5.1.4. Evaluation metrics

We compute the Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) to measure temporal prediction accuracy, and add the Negative Log-Likelihood (NLL) to evaluate probabilistic baselines. Although we cannot obtain the exact likelihood value, we derive the Variational Lower Bound (VLB) based on reference (Ho et al., 2020) and employ it as a proxy for the NLL. So, the true likelihood will outperform the reported values. These metrics are defined as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{30}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{31}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} \times 100\% \tag{32}$$

$$NLL \leq \mathbb{E}_{q(x^{0:K})}\left[-logp(x^K) - \sum_{k=1}^{K} log\frac{p_\theta(x^{k-1}|x^k)}{q(x^k|x^{k-1})}\right] \tag{33}$$

Table 3
Message summary of J-G Line and J-H Line.

Dataset	Mileage(km)	Station number	Train number	Operation records	Arrival punctuality	AD min	AD max	AD mean	AD std
J-G line	2291	37	286	133,165	96.93 %	−44	86	−0.4853	3.6488
J-H line	1318	24	109	53,445	96.58 %	−46	57	−0.0016	3.6088

where y_i denotes the ground-truth, $\widehat{y}_i$ denotes the predicted value, n is the number of valid samples.

5.1.5. Experimental settings

We use PyTorch to develop our MEHD model and baselines, and is trained on NVIDIA GeForce RTX 4090Ti GPU, subsequent inference experiments are conducted on AMD Ryzen 7 7840HS processor with 8 cores and 16 threads, and GPU acceleration is disabled to accurately evaluate algorithmic efficiency. In the experiments, the dataset is divided into training, validation, and test sets, with 70 % for training, 15 % for validation, and the remaining 15 % for testing. For the MEHD model, the batch size is set to 1, to process daily train operation data per batch, and the batch size is set to 128 for other models, that means we train 128 single-train arrival sequences per batch, and we set $\Delta t_{th} = 30min$. The learning rate is set to 0.001 with a warmup strategy, the encoding dimension is set to $M = 256$, and diffusion step is set to $K = 100$.

5.2. Overall performance

Table 4 and Table 5 present the overall performance of deterministic and probabilistic prediction models on the J-G and J-H lines, respectively. From these results, we draw the following conclusions:

The MEHD model achieved the best performance. Experimental results show that RF and BN models attained the score combinations of 1.430, 2.687, and 4.219 on both lines, and these models lack temporal dependency modeling capabilities, with RF particularly struggling to process variable-length arrival sequences, resulting in anomalous RMSE metrics and subpar outcomes. Besides, temporal baselines like LSTM and GRU achieved the score combinations of 1.141, 1.856, and 4.009, because these sequence models capture inter-event temporal dependencies, but they underperform compared to RMTPP, as RMTPP further models the asynchrony of arrival events. Models such as CLF-NET and FCLL-NET achieve superior performance by combining temporal modules to capture coupling relationships between adjacent trains. Compared to deterministic baselines, THP, DeepSTPP, NSTPP, and TANTPP demonstrate suboptimal yet competitive results across all four metrics, indicating that neural point processes can effectively learn the triggering patterns of arrival events. The proposed MEHD model achieves state-of-the-art results across all metrics, outperforming the suboptimal baselines by approximately 20 %-30 % in all four metrics. This advancement in MEHD is due to addressing the spatio-temporal non-Euclidean characteristics of complex railway networks and capturing high-order dependencies among arrival events, thereby precisely modeling both the regional-level delay propagation and evolution process and the coordinated operation status among multiple train clusters. Furthermore, while other models can only predict future arrival times for individual trains, MEHD enables end-to-end prediction of arrival sequences for all trains across the entire network.

Improper modeling of train arrival events degrades performance. Experimental results show that models based on asynchronous event sequences universally outperform synchronous time-series approaches. The reason is that train arrival events are discrete and asynchronous sequences in temporal dimension. The occurrence time of these events can be any point within the continuous time domain, and there is a strong correlation between adjacent events. Consequently, synchronous models, e.g., LSTM, GRU, CLF-NET, FCLL-NET, are structurally constrained from capturing event asynchrony to enhance prediction accuracy, thus failing to achieve optimal results. Moreover, improper assumptions about event triggering mechanism significantly degrade performance. For instance, Hawkes processes assume that historical events increase the probability of future occurrences, but their self-exciting hypothesis may fail in scenarios where prior events inhibit subsequent ones. In our research, delays at upstream stations do not necessarily propagate to downstream stations, as buffer times and headway margins are preemptively introduced in the operation timetable to mitigate delay propagation. Consequently, classical models relying on specific assumptions fail to cover dynamic and heterogeneous cases, further demonstrating the superior capability of MEHD in capturing the coordinated operation status among multiple train clusters.

5.3. Ablation experiments

To evaluate the effectiveness of the key components in the MEHD model, we conducted ablation experiments from two perspectives to verify (1) the necessity of directed hyperedge convolution in the mixed hypergraph convolution; (2) the effectiveness of the spatio-temporal attention denoising method in MEH Diffusion Module. We visualize the results of the ablation experiments in Fig. 5, and draw the following conclusions:

It is necessary to capture the coordinated operation status among multiple train clusters. To examine the necessity of capturing the influence of adjacent events $\mathbb{C}_{ij}$ on the target event v_{ij}, we degrade our MEHD into a base model, MEHD − B, which does not consider the

Table 4
Performance evaluation of deterministic models in predicting the time of the next arrival event. Bold denotes the best result and underline denotes the second-best results.

Dataset	Metric	RF	BN	LSTM	GRU	CLF-NET	FCLL-NET	MEH-DPPM
J-G	MAE	1.513	1.595	1.145	1.311	0.714	<u>0.645</u>	**0.273**
	RMSE	4.853	2.727	1.856	2.145	1.565	<u>1.433</u>	**0.347**
	MAPE(%)	–-	4.479	4.029	4.501	2.491	<u>2.241</u>	**1.699**
J-H	MAE	1.430	1.565	1.141	1.294	0.701	<u>0.595</u>	**0.253**
	RMSE	4.696	2.687	1.876	2.128	1.577	<u>1.407</u>	**0.328**
	MAPE(%)	–-	4.219	4.009	4.463	2.421	<u>2.391</u>	**1.521**

Table 5
Performance evaluation of probabilistic models in predicting the time of the next arrival event. Bold denotes the best results and underline denotes the second-best results.

Dataset	Metric	RMTPP (LSTM)	RMTPP (GRU)	DeepSTPP	NSTPP	THP	TANTPP	MEH-DPPM	#Improvement
J-G	MAE	1.081	1.122	0.797	0.739	0.414	0.462	**0.273**	**34.06 %**
	RMSE	1.573	1.908	0.899	0.973	0.514	0.455	**0.347**	**23.74 %**
	MAPE	2.048	2.643	2.796	2.901	2.657	2.403	**1.699**	**17.04 %**
	NLL	0.145	0.241	0.644	−1.65	−1.21	−1.36	**−2.179**	**24.28 %**
J-H	MAE	0.839	1.108	0.765	0.709	0.402	0.400	**0.253**	**36.75 %**
	RMSE	1.352	1.706	0.733	0.985	0.508	0.421	**0.328**	**22.09 %**
	MAPE	1.747	2.033	2.661	2.607	1.951	2.057	**1.521**	**12.93 %**
	NLL	0.079	0.093	0.317	−1.73	−1.17	−1.27	**−2.196**	**21.22 %**

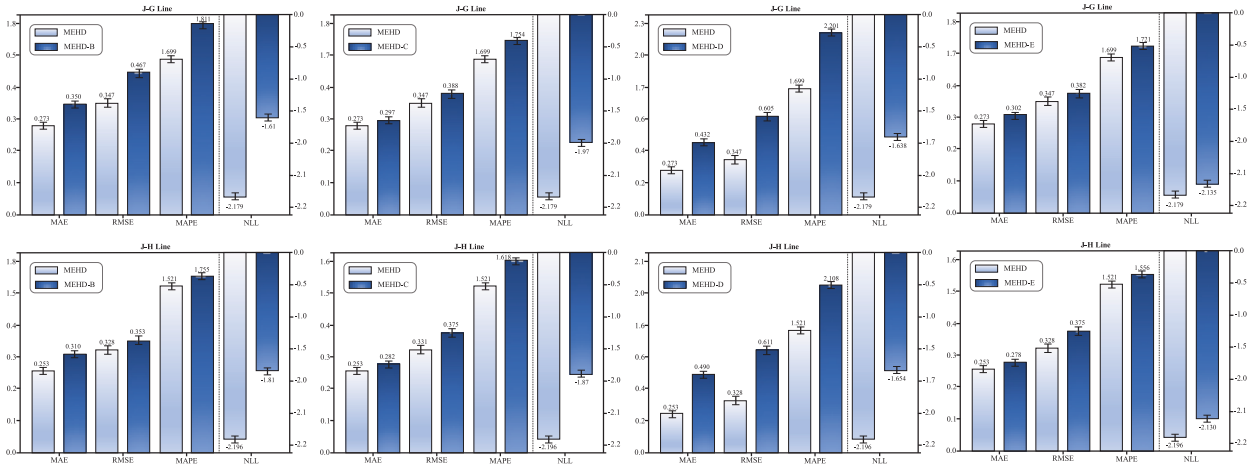


Fig. 5. Performance comparison of MEHD, MEHD-B, MEHD-C, MEHD-D and MEHD-E.

influence of the adjacent events. Specifically, we modify Eq. (17) to $X^* = X^*_U$, and the update of X is updated as $X = concat(X^s_U, X^t_U, X^{st}_U)$. As shown in Fig. 5, it can be observed that MEHD-B achieved the score combinations of 0.35, 0.467, 1.811, and −1.61 on the J-G line, and 0.31, 0.353, 1.755, and −1.81 on the J-H line. And MEHD outperformed MEHD-B in all metrics. This indicates that it is necessary to capture the high-order spatio-temporal correlations between events. and also proves the effectiveness of the directed hyperedge convolution module.

The prior operational schedule-conditioned attention denoising method is effective. To examine the effectiveness of the spatio-temporal attention denoising method guided by the prior schedule, we degrade our MEHD into a base model, MEHD-C, which does not consider the influence of the train operation schedule on event generation process. Specifically, we modify the denoising reconstruction process $p_\theta\left(\tau^{k-1}_{ij}|\tau^k_{ij}, O_{ij}, F_{ij}\right)$ in Eq. (21) to $p_\theta\left(\tau^{k-1}_{ij}|\tau^k_{ij}, s_{ij}, O_{ij}\right)$. As shown in Fig. 5, it can be observed that MEHD-C achieved the score combinations of 0.297, 0.388, 1.754, and −1.97 on the J-G line, and 0.282, 0.375, 1.618, and −1.87 on the J-H line. And MEHD outperformed MEHD-C in all metrics. This indicates that it is necessary to capture the influence of the train operation schedule on learning multi-train arrival event triggering mechanism, and also demonstrates the effectiveness of our denoising method.

It is reasonable to use the diffusion model for train delay prediction. To verify the rationality of using the diffusion model for train delay prediction, we degrade our MEHD into a base model, MEHD-D, which uses the log-normal mixture method to model the probability density function of train operation events instead of the diffusion model. This is an explicit statistical model that solves the conditional probability density of events by fusing multiple log-normal distributions. This is also the solution method applied in the existing SOTA model TANTPP (Zhang et al., Nov 2024). Specifically, we use the event hypergraph representation O obtained by Formula (18) to calculate the probability density function of the arrival time interval, which is defined as follows:

$$p(\tau_{ij}|\omega,\mu,\sigma) = \sum_{k=1}^{K} \omega_k \frac{1}{\tau_{ij}\sigma_k\sqrt{2\pi}} \exp\left(-\frac{(\log\tau_{ij}-\mu_k)^2}{2\sigma_k^2}\right) \tag{34}$$

Where, K represents the log-normal distribution components, ω is the mixing weight, μ is the mixing mean, and σ is the standard deviation. The linear transformation of the event hypergraph representation O will serve as the parameters of the distribution $p(\tau_{ij}|\omega,\mu,\sigma)$:

$$\omega = Softmax\left(f_\omega\left(O_{ij}\right)\right); \mu = f_\mu\left(O_{ij}\right); \sigma = exp\left(f_\sigma\left(O_{ij}\right)\right) \tag{35}$$

Where, $f_\omega(.)$, $f_\mu(.)$ and $f_\sigma(.)$ are three different linear transformations. We train the model by maximizing the log-likelihood to fit the spatio-temporal distribution of arrival events, this is consistent with Formula (27).

$$\sum_{i=1}^{L}\sum_{j=1}^{m_i} logp\left(\tau_{ij}|\omega,\mu,\sigma\right) \tag{36}$$

As shown in Fig. 5, it can be observed that MEHD-D achieved the score combinations of 0.432, 0.605, 1.699, and −1.638 on the J-G line, and 0.49, 0.611, 2.108, and −1.654 on the J-H line. And MEHD outperformed MEHD-D in all metrics. DDPM can help the model breaks away from the parametric distribution assumptions of traditional methods and has a stronger ability to model high-dimensional complex distributions.

5.4. Higher-order correlation capture experiments

To demonstrate the unique advantages and necessity of the hypergraph modeling for capturing the high-order correlations in the complex train operation network, we degrade our MEHD into a base model, MEHD-E, which uses directed event graph modeling and convolution to replace the hyperedge modeling and convolution module. Specifically, we construct an event graph based on Definition 3 and modify Equations (12) − (14) as follows (Tong et al., 2004):

$$X^{(l+1)} = \sum_{k=1}^{K}\beta_k\left(\widehat{A}_{in}^{k}\odot\alpha_{in}^{(k)}X^{(l)}W_{in}^{(k)} + \widehat{A}_{out}^{k}\odot\alpha_{out}^{(k)}X^{(l)}W_{out}^{(k)}\right) \tag{37}$$

where k is the neighborhood order, β_k is the fusion weight, $\widehat{A}_{in}^{k}$ and $\widehat{A}_{out}^{k}$ are the normalized in-edge and out-edge adjacency matrices respectively, and W is the learnable parameter.

We visualized the prediction results of MEHD and MEHD-E on the J-H and J-G lines in Fig. 5. As we can observed, MEHD outperforms MEHD-E in all metrics. The reason is that directed graph modeling and learning can only capture pairwise correlation between nodes and it is difficult to model the delay propagation across stations and trains, while hypergraphs have advantages in capturing high-order correlations between events.

5.5. Experiments in multi-train delay prediction scenarios

To prove the unique advantages of the MEHD model in multi-train delay prediction scenarios, we conducted experiments from three perspectives to verify that: (1) MEHD can effectively learn the generation process of the arrival time distribution of each train; (2) MEHD has good robustness in dynamic and stochastic train operation network; (3) The MEHD model has higher efficiency in parallel inference of multiple trains compared to the serial inference of a single train. We draw the following conclusions:

MEHD can effectively learn the generation process of the multi-train arrival time distribution in the network. In Fig. 6, we visualize the arrival time distribution of G101, G109 and G115 during the training process based on the gaussian kernel density estimation. The density estimation method is shown in Eq. (35), where $\widehat{f}(x)$ is the density estimate at point x, n is the number of samples, h is the

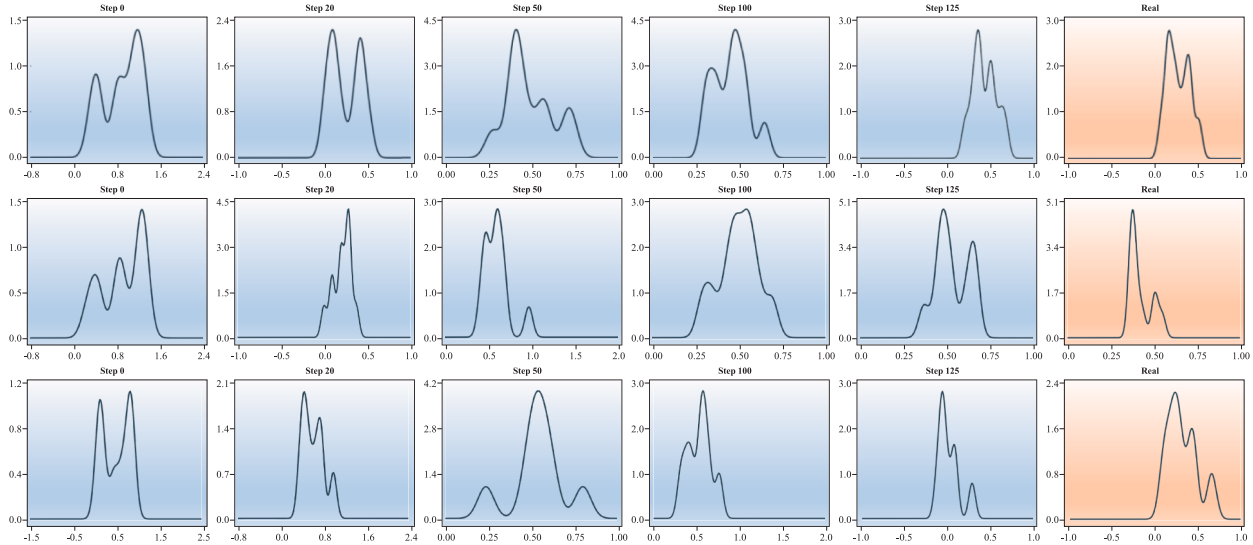


Fig. 6. Visualization of the arrival time distribution of G101, G109 and G115 during the training process.

bandwidth, and K is the kernel function. As we can observed, at the beginning of training, the time distribution exhibits in a chaotic and random state. As the training progresses, the data distribution gradually deforms and becomes concentrated. Finally, the predicted time distribution almost completely coincides with the ground-truth distribution, which indicates that MEHD can successfully learn the generation process of the arrival time distribution of each train.

$$\widehat{f}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right) \tag{38}$$

$$K(u) = \frac{1}{\sqrt{2\pi}} e^{-\frac{u^2}{2}} \tag{39}$$

MEHD has good robustness in dynamic and stochastic train operation network. Influenced by factors such as extreme weather and periodic passenger flow pressure, capacity adjustment strategies such as train service cuts, detour operations, or temporary additional train services are derived, making the railway network dynamic and stochastic, which places higher robustness requirements on the model. Therefore, we assess the effects of missing values and noise on the model's performance separately.

We randomly inserted labels of suspended trains into the training set, with a train cancellation ratio ranging from 5 % to 55 %, while keeping the validation set unchanged. Fig. 7 shows the prediction results under different train cancellation ratios. As we can observed, as the train cancellation ratio increases, the MAE increases from 0.253 to 0.443, the RMSE increases from 0.328 to 0.522, the MAPE increases from 1.521 to 1.965, and the NLL fluctuates within a range. The result indicates that as the training data decreases, the learning effect of the model deteriorates. However, MEHD trained on datasets with 55 % train cancellation ratio still achieves performance close to state-of-the-art baseline, which shows that MEHD can meet the prediction requirements in a dynamic and stochastic train operation network.

We inject Gaussian noise $\eta\ \mathcal{N}(0, \sigma^2)$ into each arrival event, with $\sigma = \varepsilon.\overline{\sigma}$, where $\overline{\sigma}$ is the average standard deviation of arrival times. We evaluate robustness over $\varepsilon \in \{0.05, 0.3, 0.5\}$ and report the prediction results under different intensities of noise in Fig. 7. As we can observed, as the intensity of the noise increases, the MAE increases from 0.253 to 0.968, the RMSE increases from 0.328 to 1.224, the MAPE increases from 1.521 to 3.011, and the NLL increases from -2.196 to -0.437. The result indicates that as the noise disturbance increases, The model performance gradually declines. But, when the noise intensity is 0.5, the performance of our model is still better than that of the RMTPP model. This indicates that our model has a certain degree of noise robustness.

The MEHD model has higher efficiency in parallel inference of multi-train. As shown in Fig. 8, we counted the inference time of MEHD and TANTPP for predicting the arrival event of 100 trains, and counted the number of model parameters (the hidden layer was uniformly set to 256). In TANTPP, we accumulate the inference time of each train in a serial inference manner to get the total time, while MEHD uses a parallel inference manner. As we can observed, the total inference time of MEHD is 0.9130 s, with the initial noise sampling time being 0.7632 s and the forward inference time being 0.1498 s. Compared with the forward inference time of 0.3658 s of TANTPP, the inference efficiency of MEHD is improved by about 1.44 times, and the number of parameters of MEHD is only 48.36 % of that of TANTPP. The results show that MEHD not only has higher accuracy than TANTPP, but also demonstrates obvious advantages in computing efficiency and model complexity.

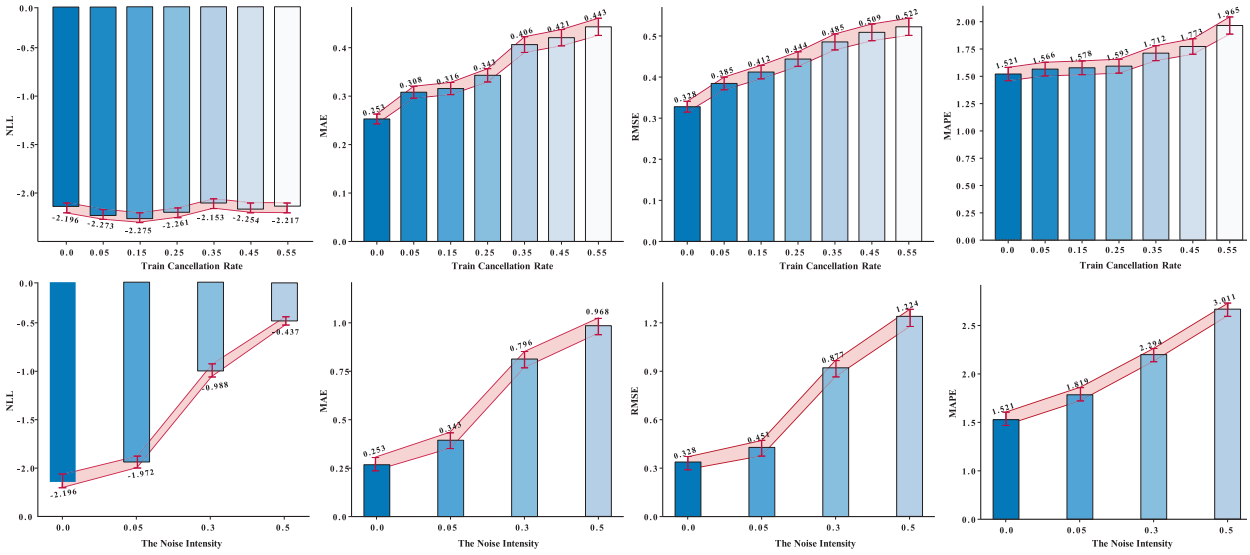


Fig. 7. Prediction performance of MEHD under varying suspension data ratios and noise intensity.

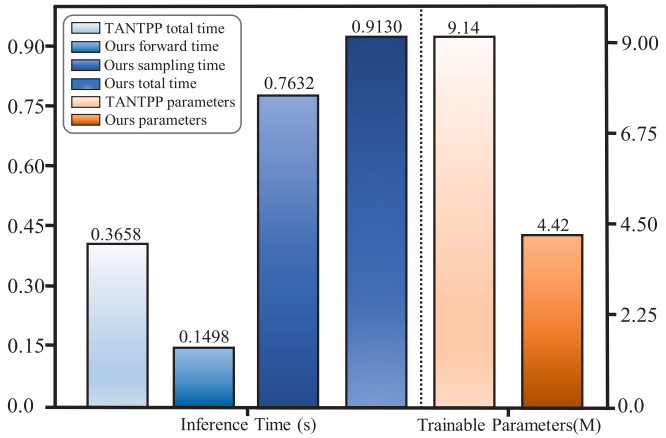


Fig. 8. Comparison of model complexity and inference efficiency of MEHD and TANTPP.

6. Conclusion

This paper focuses on regional-level multi-train delay prediction, which is more challenging compared to single-train arrival prediction or delay prediction within fixed time intervals. We propose a brand-new Multivariate Event Hypergraph Diffusion Model, transforming the regional-level multi-train delay prediction problem into the probability distribution solving problem for train arrival events. Extensive experiments demonstrate that our MEHD achieves superior performance compared to current state-of-the-art models on actual high-speed rail performance datasets, and it also exhibits good robustness and efficiency. Then we verify the effectiveness of the key components through ablation experiments. Subsequent experiments and analyses demonstrate the unique advantages of MEHD over single-train prediction methods.

In the following research, we will collect more data on train operation disturbances and fully consider the time-varying characteristics of disturbance propagation, including the differences between peak and off-peak operation periods. And we will introduce a spatio-temporal adaptive hypergraph generation module to learn the intensity between events based on features such as real-time position and speed. Intuitively, this method explicitly models the impact of train operation disturbances on train running and improves the prediction accuracy.

CRediT authorship contribution statement

Yi Xu: Writing – original draft, Validation, Methodology, Data curation, Conceptualization. **Honghui Li:** Writing – review & editing, Validation, Project administration, Investigation. **Chang Wu:** Validation, Data curation. **Yunjuan Peng:** Writing – original draft. **Xilu Du:** Visualization, Validation. **Hongwei Wang:** Formal analysis. **Sabah Mohammed:** Investigation, Data curation. **Alessandro Calvi:** Supervision, Data curation. **Dalin Zhang:** Supervision, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal correlations that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 52572351) and China Railway Corporation Foundation (No. N2023X027).

Data availability

Data will be made available on request.

References

Zhang, D.L., et al., Feb 2023. “An Interpretable Station Delay Prediction Model based on Graph Community Neural Network and Time-Series Fuzzy Decision tree,” (in English). Ieee T Fuzzy Syst 31 (2), 421–433. https://doi.org/10.1109/Tfuzz.2022.3181453.

Wen, C., et al., 2019. “Train Dispatching Management with Data-Driven Approaches: a Comprehensive Review and Appraisal,” (in English). IEEE Access 7, 114547–114571. https://doi.org/10.1109/Access.2019.2935106.

Zhang, D.L., Peng, Y.J., Zhang, Y.M., Wu, D.H., Wang, H.W., Zhang, H.L., Mar 2022. "Train Time Delay Prediction for High-speed Train Dispatching based on Spatio-Temporal Graph Convolutional Network," (in English). Ieee T Intell Transp 23 (3), 2434–2444. https://doi.org/10.1109/Tits.2021.3097064.
Spanninger, T., Büchel, B., Corman, F., 2021. In: IEEE, pp. 1–6.
Büker, T., Seybold, B., 2012. Stochastic modelling of delay propagation in large networks. Journal of Rail Transport Planning & Management 2 (1–2), 34–50.
P. Kecman and R. M. Goverde, "An online railway traffic prediction model," in *RailCopenhagen2013: 5th International Conference on Railway Operations Modelling and Analysis, Copenhagen, Denmark*, 2013: International Association of Railway Operations Research (IAROR), pp. 13-15.
Corman, F., Kecman, P., 2018. Stochastic prediction of train delays in real-time using Bayesian networks. Transp. Res. Part C Emerging Technol. 95, 599–615.
Wen, C., Mou, W.W., Huang, P., Li, Z.C., Apr 2020. "A predictive model of train delays on a railway line," (in English). J Forecasting 39 (3), 470–488. https://doi.org/10.1002/for.2639.
Huang, P., et al., Sep 2020. "Modeling train operation as sequences: a study of delay prediction with operation and weather data," (in English). Transport Res E-Log 141. https://doi.org/10.1016/j.tre.2020.102022.
Zhang, D.L., Du, C.Y., Peng, Y.J., Liu, J.Q., Mohammed, S., Calvi, A., Nov 2024. "A Multi-Source Dynamic Temporal Point Process Model for Train Delay Prediction," (in English). Ieee T Intell Transp 25 (11), 17865–17877. https://doi.org/10.1109/Tits.2024.3430031.
Ding, X., Xu, X., Li, J., Shi, R., 2021. In: IEEE, pp. 2387–2392.
Li, Z., Huang, P., Wen, C., Dong, W., Ji, Y., Rodrigues, F., 2024. Railway network delay evolution: a heterogeneous graph neural network approach. Appl. Soft Comput. 159, 111640.
Huang, P., Spanninger, T., Corman, F., Sep 2022. "Enhancing the Understanding of Train Delays with Delay Evolution Pattern Discovery: a Clustering and Bayesian Network Approach," (in English). Ieee T Intell Transp 23 (9), 15367–15381. https://doi.org/10.1109/Tits.2022.3140386.
D. L. Zhang *et al.*, "Prediction of Train Station Delay Based on Multiattention Graph Convolution Network," (in English), *J Adv Transport,* vol. 2022, Feb 21 2022, doi: Artn 7580267. 10.1155/2022/7580267.
Yuan, J., 2006. *Stochastic modelling of train delays and delay propagation in stations*. Eburon Uitgeverij BV.
Higgins, A., Kozan, E., 1998. Modeling train delays in urban networks. Transp. Sci. 32 (4), 346–357.
Tiong, K.Y., Palmqvist, C.-W., 2023. Quantitative methods for train delay propagation research. Transp. Res. Procedia 72, 80–86.
Ling, X., Peng, Y., Sun, S., Li, P., Wang, P., 2018. Uncovering correlation between train delay and train exposure to bad weather. Phys. A 512, 1152–1159.
S. Daniotti, V. P. Servedio, J. Kager, A. Robben-Baldauf, and S. Thurner, "Systemic risk approach to mitigate delay cascading in railway networks," (in en-US), 2023-10 2023.
M. Schälicke and K. Nachtigall, "Solving the Real-Time Train Dispatching Problem by Column Generation," (in English), *Transportation Science,* Feb 25 2025, doi: 10.1287/trsc.2023.0215.
Antelmi, A., Cordasco, G., Polato, M., Scarano, V., Spagnuolo, C., Yang, D., 2023. A survey on hypergraph representation learning. ACM Comput. Surv. 56 (1), 1–38.
Gao, Y., Zhang, Z., Lin, H., Zhao, X., Du, S., Zou, C., 2020. Hypergraph learning: Methods and practices. IEEE Trans. Pattern Anal. Mach. Intell. 44 (5), 2548–2566.
Ho, J., Jain, A., Abbeel, P., 2020. Denoising diffusion probabilistic models. Adv. Neural Inf. Proces. Syst. 33, 6840–6851.
J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," *arXiv preprint arXiv:2010.02502,* 2020.
D. L. Zhang *et al.*, "A high-speed railway network dataset from train operation records and weather data," (in English), *Sci Data,* vol. 9, no. 1, May 27 2022, doi: ARTN 244. 10.1038/s41597-022-01349-8.
Li, Z.C., Huang, P., Wen, C., Jiang, X., Rodrigues, F., May 2022. "Prediction of train arrival delays considering route conflicts at multi-line stations," (in English). Transport Res C-Emer 138. https://doi.org/10.1016/j.trc.2022.103606.
Wu, J., et al., 2021. A hybrid LSTM-CPS approach for long-term prediction of train delays in multivariate time series. Future Transportation 1 (3), 765–776.
P. Huang, Z. C. Li, C. Wen, J. Lessan, F. Corman, and L. P. Fu, "Modeling train timetables as images: A cost-sensitive deep learning framework for delay propagation pattern recognition," (in English), *Expert Syst Appl,* vol. 177, Sep 1 2021, doi: ARTN 114996. 10.1016/j.eswa.2021.114996.
Huang, P., Wen, C., Fu, L.P., Peng, Q.Y., Tang, Y.X., Apr 2020. "A deep learning approach for multi-attribute data: a study of train delay prediction in railway systems," (in English). Inform Sciences 516, 234–253. https://doi.org/10.1016/j.ins.2019.12.053.
Barbour, W., Mori, J.C.M., Kuppa, S., Work, D.B., Aug 2018. "Prediction of arrival times of freight traffic on US railroads using support vector regression," (in English). Transport Res C-Emer 93, 211–227. https://doi.org/10.1016/j.trc.2018.05.019.
Wu, J.Q., et al., Mar 2022. "The Bounds of Improvements Toward Real-Time Forecast of Multi-Scenario Train Delays," (in English). Ieee T Intell Transp 23 (3), 2445–2456. https://doi.org/10.1109/Tits.2021.3099031.
Huang, P., Wen, C., Fu, L.P., Peng, Q.Y., Li, Z.C., Feb 2020. "A hybrid model to improve the train running time prediction ability during high-speed railway disruptions," (in English). Safety Sci 122. https://doi.org/10.1016/j.ssci.2019.104510.
Pineda-Jaramillo, J., Viti, F., Feb 2023. "Identifying the rail operating features associated to intermodal freight rail operation delays," (in English). Transport Res C-Emer 147. https://doi.org/10.1016/j.trc.2022.103993.
A. Lulli, L. Oneto, R. Canepa, S. Petralli, and D. Anguita, "Large-Scale Railway Networks Train Movements: a Dynamic, Interpretable, and Robust Hybrid Data Analytics System," (in English), *Pr Int Conf Data Sc,* pp. 371-380, 2018, doi: 10.1109/Dsaa.2018.00048.
J. Luo, Q. Y. Peng, C. Wen, W. Wen, and P. Huang, "Data-driven decision support for rail traffic control: A predictive approach," (in English), *Expert Syst Appl,* vol. 207, Nov 30 2022, doi: ARTN 118050. 10.1016/j.eswa.2022.118050.
Nabian, M.A., Alemazkoor, N., Meidani, H., May 2019. "Predicting Near-Term Train Schedule Performance and Delay using Bi-Level Random Forests," (in English). Transport Res Rec 2673 (5), 564–573. https://doi.org/10.1177/0361198119840339.
Xu, J., Wang, W.Q., Gao, Z.M., Luo, H.C., Wu, Q., Jul 2023. "A novel Markov model for near-term railway delay prediction," (in English). Comput. Ind. Eng. 181. https://doi.org/10.1016/j.cie.2023.109302.
Zhou, D., Huang, J., Schölkopf, B., 2006. Learning with hypergraphs: Clustering, classification, and embedding. Adv. Neural Inf. Proces. Syst. 19.
Luo, X., Peng, J., Liang, J., 2022. Directed hypergraph attention network for traffic forecasting. IET Intel. Transport Syst. 16 (1), 85–98.
Cao, S., Wu, L., Zhang, R., Chen, Y., Li, J., Liu, Q., 2024. A spatial-temporal gated hypergraph convolution network for traffic prediction. IEEE Trans. Veh. Technol. 73 (7), 9546–9559.
D. Georgiev, M. Brockschmidt, and M. Allamanis, "Heat: Hyperedge attention networks," *arXiv preprint arXiv:2201.12113,* 2022.
Li, J., Zhang, D., Gao, S., Xu, W., 2024. A mixed Hypergraph Convolutional Network for Session-based Recommendation. In: *International Conference on Intelligent Computing*. Springer, pp. 306–317.
Hao, X., Li, J., Guo, Y., Jiang, T., Yu, M., 2021. Hypergraph neural network for skeleton-based action recognition. IEEE Trans. Image Process. 30, 2263–2275.
Ye, Z.L., Zhao, H.X., Zhang, K., Zhu, Y., Xiao, Y.Z., Oct 2019. "Tri-party deep network representation learning using inductive matrix completion," (in English). J. Cent. South Univ. 26 (10), 2746–2758. https://doi.org/10.1007/s11771-019-4210-8.
Bi, Z., Zhang, T., Zhou, P., Li, Y., 2020. Knowledge transfer for out-of-knowledge-base entities: improving graph-neural-network-based embedding using convolutional layers. IEEE Access 8, 159039–159049.
Borndörfer, R., Reuther, M., Schlechte, T., Waas, K., Weider, S., 2016. Integrated optimization of rolling stock rotations for intercity railways. Transp. Sci. 50 (3), 863–877.
Shen, L., Li, J., Chen, Y., Li, C., Chen, X., Lee, D.-H., 2024. Short-term metro origin-destination passenger flow prediction via spatio-temporal dynamic attentive multi-hypergraph network. Ieee T Intell Transp 25 (8), 9945–9957.
Wang, M., Zhang, Y., Zhao, X., Hu, Y., Yin, B., 2024. Traffic origin-destination demand prediction via Multichannel Hypergraph Convolutional Networks. IEEE Trans. Comput. Social Syst.
X. L. Li, J. Thickstun, I. Gulrajani, P. Liang, and T. B. Hashimoto, "Diffusion-LM Improves Controllable Text Generation," (in English), *Advances in Neural Information Processing Systems 35 (Neurips 2022),* 2022. [Online]. Available: <Go to ISI>://WOS:001213927502043.
C. H. Niu, Y. Song, J. M. Song, S. J. Zhao, A. Grover, and S. Ermon, "Permutation Invariant Graph Generation via Score-Based Generative Modeling," (in English), *International Conference on Artificial Intelligence and Statistics, Vol 108,* vol. 108, pp. 4474-4483, 2020. [Online]. Available: <Go to ISI>://WOS:000559931302058.

J. Austin, D. D. Johnson, J. Ho, D. Tarlow, and R. van den Berg, "Structured Denoising Diffusion Models in Discrete State-Spaces," (in English), *Advances in Neural Information Processing Systems 34 (Neurips 2021),* 2021. [Online]. Available: <Go to ISI>://WOS:000925183301020.

Yang, L., et al., 2024. Cross-modal contextualized diffusion models for text-guided visual generation and editing. In *the Twelfth International Conference on Learning Representations*.

K. Rasul, C. Seward, I. Schuster, and R. Vollgraf, "Autoregressive Denoising Diffusion Models for Multivariate Probabilistic Time Series Forecasting," (in English), *International Conference on Machine Learning, Vol 139,* vol. 139, 2021. [Online]. Available: <Go to ISI>://WOS:000768182705002.

Yuan, Y., Ding, J.T., Shao, C.Y., Jin, D.P., Li, Y., 2023. Spatio-temporal Diffusion Point Processes. In: (in English), *Proceedings of the 29th Acm Sigkdd Conference on Knowledge Discovery and Data Mining, Kdd 2023*, pp. 3173–3184. https://doi.org/10.1145/3580305.3599511.

D. Watson, J. Ho, M. Norouzi, and W. Chan, "Learning to efficiently sample from diffusion probabilistic models," *arXiv preprint arXiv:2106.03802,* 2021.

T. Salimans and J. Ho, "Progressive distillation for fast sampling of diffusion models," *arXiv preprint arXiv:2202.00512,* 2022.

A. Nichol and P. Dhariwal, "Improved Denoising Diffusion Probabilistic Models," (in English), *International Conference on Machine Learning, Vol 139,* vol. 139, 2021. [Online]. Available: <Go to ISI>://WOS:000768182704030.

D. P. Kingma, T. Salimans, B. Poole, and J. Ho, "Variational Diffusion Models," (in English), *Advances in Neural Information Processing Systems 34 (Neurips 2021),* 2021. [Online]. Available: <Go to ISI>://WOS:000925183304035.

M. Xu, L. Yu, Y. Song, C. Shi, S. Ermon, and J. Tang, "Geodiff: A geometric diffusion model for molecular conformation generation," *arXiv preprint arXiv:2203.02923,* 2022.

Caimi, G., Fuchsberger, M., Laumanns, M., Lüthi, M., Nov 2012. "A model predictive control approach for discrete-time rescheduling in complex central railway station areas," (in English). Comput. Oper. Res. 39 (11), 2578–2593. https://doi.org/10.1016/j.cor.2012.01.003.

Vaswani, A., et al., 2017. Attention is all you need. Adv. Neural Inf. Proces. Syst. 30.

S. Zuo, H. Jiang, Z. Li, T. Zhao, and H. Zha, "Transformer hawkes process," in *International conference on machine learning*, 2020: PMLR, pp. 11692-11702.

Du, N., Dai, H.J., Trivedi, R., Upadhyay, U., Gomez-Rodriguez, M., Song, L., 2016. In: "recurrent Marked Temporal Point Processes: Embedding Event History to Vector," (in English), *Kdd'16*, pp. 1555–1564. https://doi.org/10.1145/2939672.2939875.

R. Q. Chen, B. Amos, and M. Nickel, "Neural Spatio-Temporal Point Processes," (in en-US), *arXiv: Learning,arXiv: Learning,* 2020-11 2020.

Z. H. Zhou, X. Y. Yang, R. A. Rossi, H. D. Zhao, and R. Yu, "Neural Point Process for Learning Spatiotemporal Event Dynamics," (in English), *Pr Mach Learn Res,* vol. 168, 2022. [Online]. Available: <Go to ISI>://WOS:001227737300060.

Z. Tong, Y. Liang, C. Sun, D. S. Rosenblum, and A. Lim, "Directed graph convolutional network," *arXiv preprint arXiv:2004.13970,* 2020.