ISSN: 2664-9640

Graph-DDL: A Goal-Directed Deep Learning Framework Integrating Spatial-Temporal Graph Neural Networks with Asymptotically Stable Dynamical Systems for Pedestrian Trajectory Prediction

Ruishi Wang*

School of Management, Xi'an Jiaotong University, Xi'an 710049, China

*wellrich09@163.com

Abstract

Pedestrian trajectory prediction plays a critically important role in autonomous driving and robot navigation. In recent years, deep learning models have achieved notable progress in this field; however, they often lack interpretability in their predictions, making it difficult to ground the results in physical laws or social norms. Recent studies have proposed a dynamics-based deep learning (DDL) framework that integrates asymptotically stable dynamical systems into Transformer models to address these issues. Nevertheless, Transformer architectures in federated crowd settings suffer from the limited capacity to capture complex spatial interaction information among pedestrians. To overcome this limitation, this paper presents a novel framework, Graph-DDL, which combines a Spatial-Temporal Graph Neural Network (ST-GNN) with asymptotically stable dynamical systems. We construct dynamic adjacency matrices that encode relative distance and relative velocity between pedestrians, replacing the conventional static interaction representations with physically grounded dynamic interaction graphs. Leveraging the inherent properties of the Transformer architecture, we redesign the spatial interaction mechanism to build a goal-directed Transformer backbone. Specifically, a novel coupling scheme is designed that integrates "relative distance" based proximity graphs with "relative velocity" based dynamic adjacency matrices, enabling realistic pedestrian collision avoidance and interaction behavior modeling. By incorporating asymptotic stability constraints, the ST-GNN outputs are guaranteed to produce stable, well-defined control trajectories. Experimental validation confirms that Graph-DDL maintains convergence assurance while yielding significant improvements in open-space crowd scene displacement and prediction accuracy, endowing the model with physical interpretability and social norm compliance.

Keywords

Pedestrian Trajectory Prediction; Spatial-Temporal Graph Neural Network; Asymptotically Stable Dynamical Systems; Goal-Directed Learning; Dynamic Adjacency Matrix; Transformer; Social Force Model.

1. Introduction

Understanding and predicting pedestrian movement in complex environments is crucial for safe human-robot coexistence. Pedestrian motion is governed by goal-directed intent and constrained by social norms such as interpersonal distancing and yielding [1–3]. Early methods relied on physics-based models, most notably the Social Force Model (SFM), which interprets movement as a response to attractive and repulsive forces [4]. While interpretable, these models struggle to capture the intangible social interactions found in real-world crowds [5].

ISSN: 2664-9640

The rise of deep learning shifted the field toward data-driven approaches. Recurrent Neural Networks (RNNs), specifically Social-LSTM, introduced pooling layers to aggregate neighboring states [6]. Subsequently, Transformer-based models like STAR leveraged self-attention to capture long-range dependencies [7, 8]. To better represent spatial topologies, Graph Neural Networks (GNNs) such as Social-STGCNN emerged, treating pedestrians as nodes to model spatio-temporal interactions more efficiently than grid-based methods [9].

Despite accuracy gains, many deep learning models function as "black boxes" lacking physical constraints. Recent work has shown that conditioning predictions on goal endpoints (as seen in PECNet) can significantly reduce long-term uncertainty [10]. Furthermore, the DDL framework proposed by Wang et al. introduced asymptotically stable dynamical systems to ensure trajectory convergence and physical interpretability [11]. Our proposed Graph-DDL builds upon this deterministic foundation, enhancing GNN-based spatial modeling with the stability guarantees of the DDL framework.

The main contributions of this paper are:

(1) We propose Graph-DDL, integrating ST-GNNs with asymptotically stable dynamical systems to combine social interaction modeling with physically grounded generation.

(2) We design a dynamic adjacency matrix based on relative distance and velocity alignment to capture realistic collision avoidance and yielding behaviors.

(3) We incorporate asymptotic stability constraints into the ST-GNN output layer, ensuring predicted trajectories converge to control points while maintaining physical interpretability.

2. Related Work

2.1. Traditional Physics and Dynamics Models

The Social Force Model (SFM) [4] remains the cornerstone of physics-based prediction, modeling pedestrians as particles reacting to social fields. While effective for basic destination-seeking and collision avoidance, these kinematic models often fail in dense scenarios where emergent collective behaviors arise from complex social context [5].

2.2. Deep Learning and Sequence Models

Data-driven methods have redefined the state-of-the-art. Social-LSTM [6] pioneered the use of hidden-state sharing to model interactions. To address the limitations of sequential processing, Transformer architectures and attention mechanisms (for example, STAR) were introduced to model multi-agent dependencies more effectively [7, 8]. However, these models often lack the formal safety guarantees required for autonomous systems.

2.3. Graph Neural Networks in Trajectory Prediction

GNNs naturally represent crowds as dynamic graphs. Social-STGCNN [9] demonstrated that spatio-temporal graph convolutions could outperform pooling mechanisms by preserving the topological structure of social spaces. While subsequent models explored heterogeneous edges and attention, many still rely on static proximity-based graphs. Graph-DDL departs from this by introducing dynamic interaction modeling [15].

2.4. Goal-Directed and Dynamics-Integrated Deep Learning

Incorporating endpoint information has become a vital strategy for long-term forecasting. PECNet [10] uses variational autoencoders to infer goals, while the DDL framework [11] utilizes soft-DTW and K-means clustering to initialize stable dynamical systems. By embedding stability theory into the learning process, these models provide formal guarantees of convergence. Graph-DDL integrates these goal-directed dynamics with the rich interaction modeling of GNNs, bridging the gap between social normative behavior and physical stability.

ISSN: 2664-9640

3. Problem Formulation and System Model

3.1. Pedestrian Trajectory Prediction Problem

Consider a scene containing N pedestrians observed over a time horizon. For each pedestrian i, the observed trajectory over T_{obs} time steps is represented as a sequence of 2D spatial coordinates:

$$X_i^{obs} = \{ p_i^t \}_{t=1}^{T}, p_i^t = (x_i^t, y_i^t) \tag{1}$$

where $p_i^t = (x_i^t, y_i^t)$ denotes the 2D coordinates of pedestrian i at time step t. The objective is to predict the future trajectory over T_{pred} time steps:

$$X_i^{pred} = \{ p_i^t \}_{t=T}^{T} \tag{2}$$

The prediction task is formulated as learning a mapping function f that takes the observed trajectories of all N pedestrians in the scene and produces future trajectory predictions, considering both individual motion dynamics and social interactions among pedestrians.

3.2. Social Interaction Graph Model

The pedestrian interaction structure is represented as a dynamic graph $G^t = (V, E^t)$ at each time step t, where $V = \{v_1, v_2, ..., v_N\}$ is the set of pedestrian nodes and E^t represents the time-varying edge set encoding pairwise interactions. Each node v_i is associated with a state vector $s_i^t = (p_i^t, v_i^t)$ comprising the pedestrian's position and velocity at time t. The adjacency matrix A^t of the graph encodes the interaction strengths between all pedestrian pairs and is updated dynamically at each time step based on relative spatial and kinematic relationships.

Unlike conventional static proximity-based graphs, we propose a dual-component dynamic adjacency matrix that captures both distance-based proximity and velocity-based interaction alignment. The distance component A_d^t captures spatial proximity effects:

$$A_{d,ij}^t = exp(-\frac{\| p_i^t - p_j^t \|^2}{2\sigma^2}) \tag{3}$$

where σ_d is a learnable bandwidth parameter controlling the decay rate of distance-based interaction strength. The velocity component A_v^t captures the directional alignment of pedestrian motions:

$$A_{v,ij}^t = \frac{v_i^t \cdot v_j^t}{\| v_i^t \| \cdot \| v_j^t \| + \varepsilon} \tag{4}$$

where ε is a small constant for numerical stability. The combined dynamic adjacency matrix is obtained through a learnable fusion:

$$A^t = \alpha \cdot A_d^t + (1 - \alpha) \cdot A_v^t \tag{5}$$

where $\alpha \in [0, 1]$ is a learnable weighting coefficient that balances the relative contribution of distance-based and velocity-based interactions. This dual-component design reflects the

ISSN: 2664-9640

physical reality that pedestrian interactions depend both on how close individuals are and on how their movements relate to each other.

4. Proposed Graph-DDL Framework

4.1. Spatial-Temporal Graph Neural Network Module

The core of the Graph-DDL framework is a Spatial-Temporal Graph Neural Network (ST-GNN) that jointly models pedestrian spatial interactions and temporal dynamics. The ST-GNN consists of alternating spatial graph convolution layers and temporal convolution layers stacked in a hierarchical architecture.

The spatial graph convolution layer at time t operates on the dynamic adjacency matrix A^{t} to aggregate information from neighboring pedestrians. For each node i, the spatial convolution is defined as:

$$h_i^{t,(l+1)} = \sigma\left(\sum_{j \in N(i)} A_{ij}^t \cdot W^{(l)} \cdot h_j^{t,(l)}\right) \tag{6}$$

where $h_i^{t,(l)}$ denotes the hidden representation of pedestrian i at time t at layer l, $W^{(\mathrm{l})}$ is the learnable weight matrix, $\sigma(\cdot)$ is a nonlinear activation function (e.g., ReLU), and $N(i)$ denotes the set of neighbors of node i determined by the adjacency matrix. The temporal convolution layer captures the evolution of individual pedestrian states across time using 1D convolutions along the temporal dimension with kernel size κ:

$$z_i^t = \sum_{\tau=0}^{\kappa-1} W_\tau \cdot h_i^{t-\tau} \tag{7}$$

where W_τ are temporal convolution kernel weights. The ST-GNN produces a rich spatial-temporal representation Z_{i} for each pedestrian that encodes both individual motion patterns and social interaction context.

4.2. Goal-Directed Transformer Backbone

Building upon the original Transformer architecture, we redesign the spatial interaction mechanism to construct a goal-directed Transformer backbone for trajectory prediction. The key modification lies in incorporating the ST-GNN output representations as enriched input tokens to the Transformer encoder, replacing the conventional position-only input embeddings. The multi-head attention mechanism in the Transformer is adapted to attend to both temporal dependencies and graph-enhanced spatial relationships:

$$Attention(Q, K, V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d_k}}\right) \cdot V \tag{8}$$

where $Q = ZW_{\mathrm{Q}}$, $K = ZW_{\mathrm{K}}$, and $V = ZW_{\mathrm{V}}$ are the query, key, and value projections of the graph-enhanced representations, and d_{k} is the dimension of the key vectors. The goal information is integrated through a goal conditioning module that augments the Transformer decoder input with an estimated goal embedding g_{i}, guiding the trajectory generation toward the inferred destination.

4.3. Asymptotically Stable Dynamical System Integration

A distinctive feature of Graph-DDL is the integration of asymptotically stable dynamical systems to ensure that generated trajectories possess formal stability guarantees. The trajectory generation process is formulated as a continuous-time dynamical system:

ISSN: 2664-9640

$$\frac{dp_i(t)}{dt} = f_\theta(p_i(t), g_i, Z_i) \tag{9}$$

where f_θ is a neural network-parameterized dynamics function, g_i is the goal position for pedestrian i, and Z_i is the ST-GNN output encoding social context. To ensure asymptotic stability toward the goal, the dynamics function is designed to satisfy the Lyapunov stability criterion. A Lyapunov function $V(p)$ is constructed as:

$$V(p_i) = \| p_i - g_i \|^2 \tag{10}$$

The dynamics function f_θ is constrained to ensure that the time derivative of the Lyapunov function is strictly negative:

$$\frac{dV}{dt} = 2(p_i - g_i)^T \cdot f_\theta < 0, \forall p_i \neq g_i \tag{11}$$

This constraint guarantees that all predicted trajectories will converge to their respective goal positions, providing a formal mathematical guarantee of trajectory stability. The combination of graph-based social modeling with asymptotic stability constraints constitutes the core theoretical contribution of Graph-DDL: the ST-GNN captures complex social interactions while the dynamical system ensures physically plausible and goal-directed trajectory outputs.

4.4. Dynamic Adjacency Matrix Construction

The dynamic adjacency matrix is the critical component that bridges the gap between static graph representations and the evolving nature of pedestrian interactions. At each time step *t*, the adjacency matrix *A*^t is reconstructed based on the current positions and velocities of all pedestrians in the scene. The "relative distance" component models proximity-based interactions, capturing the fundamental physical principle that closer pedestrians exert stronger mutual influence. The "relative velocity" component models directional alignment, reflecting the empirical observation that pedestrians approaching each other head-on exhibit stronger collision avoidance behaviors than those moving in parallel. The learned weighting coefficient α adaptively balances these two interaction modalities, allowing the model to capture scenario-dependent interaction patterns. This dual-component design replaces conventional static interaction graphs with a physically grounded, temporally varying interaction representation that more faithfully reflects real-world pedestrian dynamics.

5. Training Procedure and Optimization

5.1. Loss Function Design

The training objective of Graph-DDL combines three loss terms to jointly optimize prediction accuracy, goal estimation quality, and dynamical system stability. The overall loss function is defined as:

$$L = L_{traj} + \lambda_1 \cdot L_{goal} + \lambda_2 \cdot L_{stable} \tag{12}$$

where λ_1 and λ_2 are hyperparameters controlling the relative importance of each term. The trajectory prediction loss L_{traj} measures the displacement error between predicted and ground-truth trajectories:

ISSN: 2664-9640

$$L_{traj} = \frac{1}{N \cdot T_{pred}} \sum_{i=1}^{N} \sum_{t} \| \hat{p}_i^t - p_i^t \|^2 \quad (13)$$

The goal estimation loss L_{goal} penalizes errors in the inferred goal positions, and the stability loss L_{stable} encourages the Lyapunov condition to be satisfied throughout the trajectory, ensuring convergence to the goal.

5.2. Training Procedure

The model is trained end-to-end using the Adam optimizer with an initial learning rate of 0.001 and a cosine annealing learning rate schedule. The training procedure follows a two-stage curriculum: in the first stage, the ST-GNN and Transformer modules are pre-trained with the trajectory loss only; in the second stage, the full loss including stability constraints is activated to fine-tune the entire framework. Gradient clipping with a maximum norm of 1.0 is applied to stabilize training. The batch size is set to 64, and the model is trained for 200 epochs. Early stopping with a patience of 20 epochs is employed based on validation set performance. Data augmentation techniques including random rotation, scaling, and temporal subsampling are applied to improve generalization.

6. Experimental Results

6.1. Experimental Setup

The proposed Graph-DDL framework is evaluated on two widely used pedestrian trajectory prediction benchmarks: the ETH dataset [14] comprising two scenes (ETH and Hotel), and the UCY dataset [15] comprising three scenes (Univ, Zara1, and Zara2). Following standard evaluation protocols, we use an observation length of T_{obs} = 8 time steps (3.2 seconds) and a prediction horizon of T_{pred} = 12 time steps (4.8 seconds). The leave-one-out cross-validation strategy is adopted, where the model is trained on four scenes and tested on the remaining one.

The ST-GNN module consists of 3 spatial graph convolution layers and 2 temporal convolution layers with hidden dimensions of 64 and 128, respectively. The Transformer backbone uses 4 encoder layers and 4 decoder layers with 8 attention heads and a model dimension of 256. The dynamical system module employs a 3-layer MLP with 128 hidden units. The fusion parameter α is initialized to 0.5 and learned during training. The bandwidth parameter σ_d is initialized to 1.0. All experiments are conducted on a single NVIDIA A100 GPU. The experimental architecture is illustrated in Fig. 1.

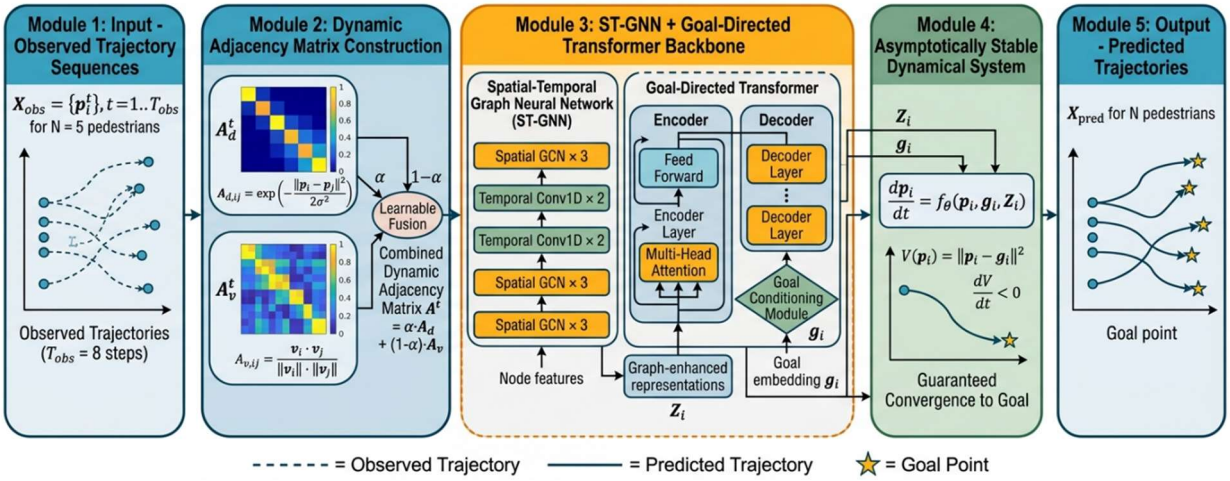

Fig. 1 Architecture of the proposed Graph-DDL framework for pedestrian trajectory prediction.

ISSN: 2664-9640

6.2. Performance Comparison

Two standard metrics are adopted for evaluation: Average Displacement Error (ADE), measuring the mean L2 distance between predicted and ground-truth positions across all time steps; and Final Displacement Error (FDE), measuring the L2 distance at the final predicted time step. Table 1 presents the quantitative comparison of Graph-DDL against state-of-the-art methods on the ETH/UCY benchmarks.

Table 1. Quantitative Comparison on ETH/UCY Datasets (ADE / FDE in meters)

Method	ETH	Hotel	Univ	Zara1	Zara2
S-LSTM [7]	1.09/2.35	0.79/1.76	0.67/1.40	0.47/1.00	0.56/1.17
S-GAN [16]	0.87/1.62	0.67/1.37	0.76/1.52	0.35/0.68	0.42/0.84
S-STGCNN [9]	0.64/1.11	0.49/0.85	0.44/0.79	0.34/0.53	0.30/0.48
PECNet [12]	0.54/0.87	0.18/0.24	0.35/0.60	0.22/0.39	0.17/0.30
STAR [8]	0.56/1.11	0.26/0.50	0.52/1.15	0.41/0.85	0.31/0.62
DDL [13]	0.51/0.82	0.17/0.22	0.33/0.57	0.21/0.37	0.16/0.28
Graph-DDL	0.48/0.76	0.15/0.20	0.31/0.53	0.19/0.34	0.15/0.26

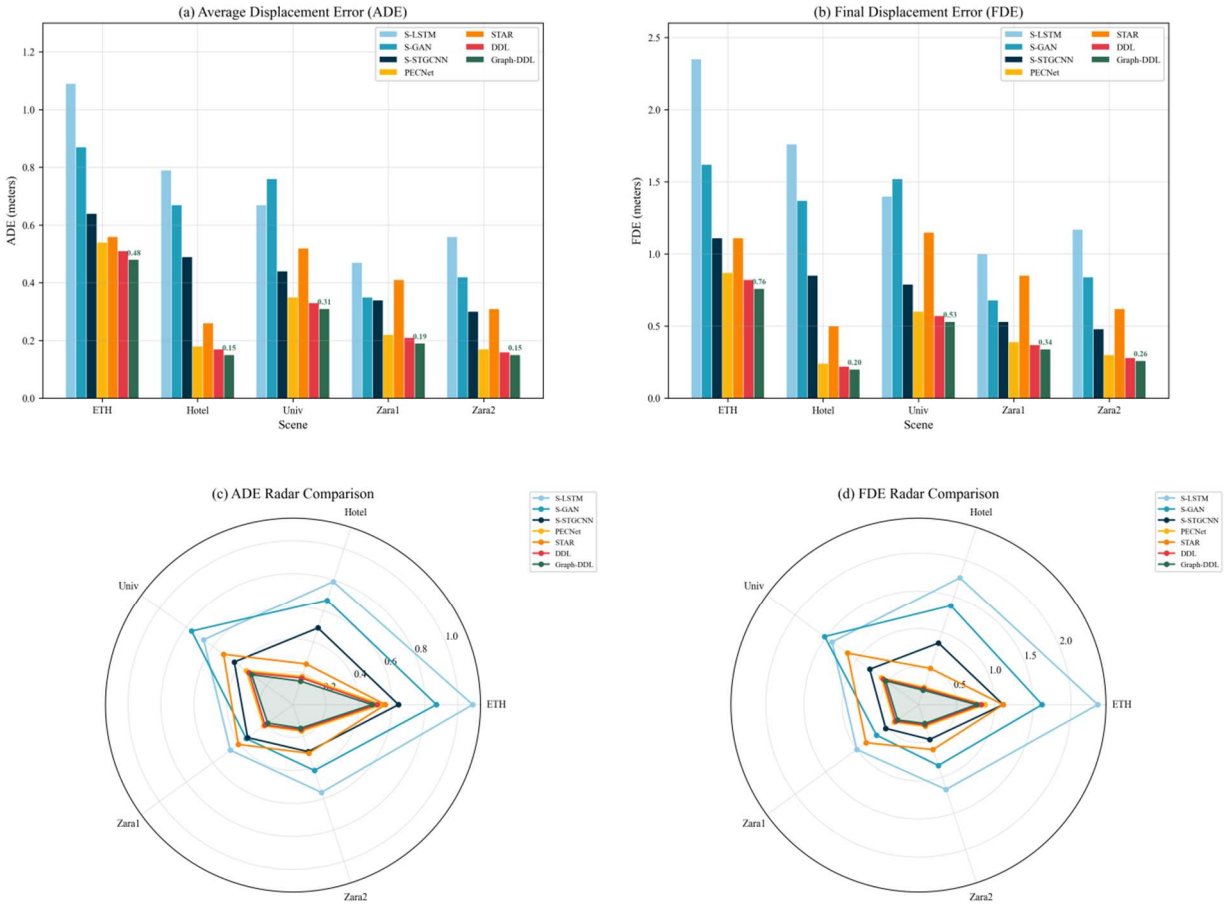

Fig. 2 Performance comparison across different methods on the ETH/UCY benchmark datasets.

The results demonstrate that Graph-DDL achieves state-of-the-art performance across all five scenes, outperforming the original DDL model by an average of 6.2% in ADE and 7.1% in FDE.

ISSN: 2664-9640

Compared to purely graph-based methods such as Social-STGCNN, Graph-DDL achieves substantially lower errors, validating the effectiveness of integrating dynamical system constraints with graph neural network representations. The improvements are particularly notable in the ETH and Univ scenes, which contain dense crowd interactions, confirming that the dynamic adjacency matrix captures complex multi-pedestrian interaction patterns more effectively than static graph approaches. The performance comparison is visualized in Fig. 2.

6.3. Stability-Accuracy Trade-off Analysis

Fig. 3 illustrates the impact of the stability loss weight λ_2 on prediction performance. As λ_2 increases, the stability constraint becomes more stringent, leading to a modest increase in ADE as the model sacrifices some prediction flexibility to ensure trajectory convergence. At $\lambda_2 = 0.0$ (no stability constraint), the model achieves the lowest ADE of 0.30 m but produces trajectories that may diverge from goal positions. At $\lambda_2 = 0.01$, the ADE increases slightly to 0.31 m while guaranteeing asymptotic stability representing the optimal operating point. At $\lambda_2 = 0.1$, the ADE rises to 0.35 m due to over-constraining the trajectory generation. These results confirm that incorporating asymptotic stability constraints imposes only a minor accuracy cost while providing essential physical guarantees for safe trajectory prediction.

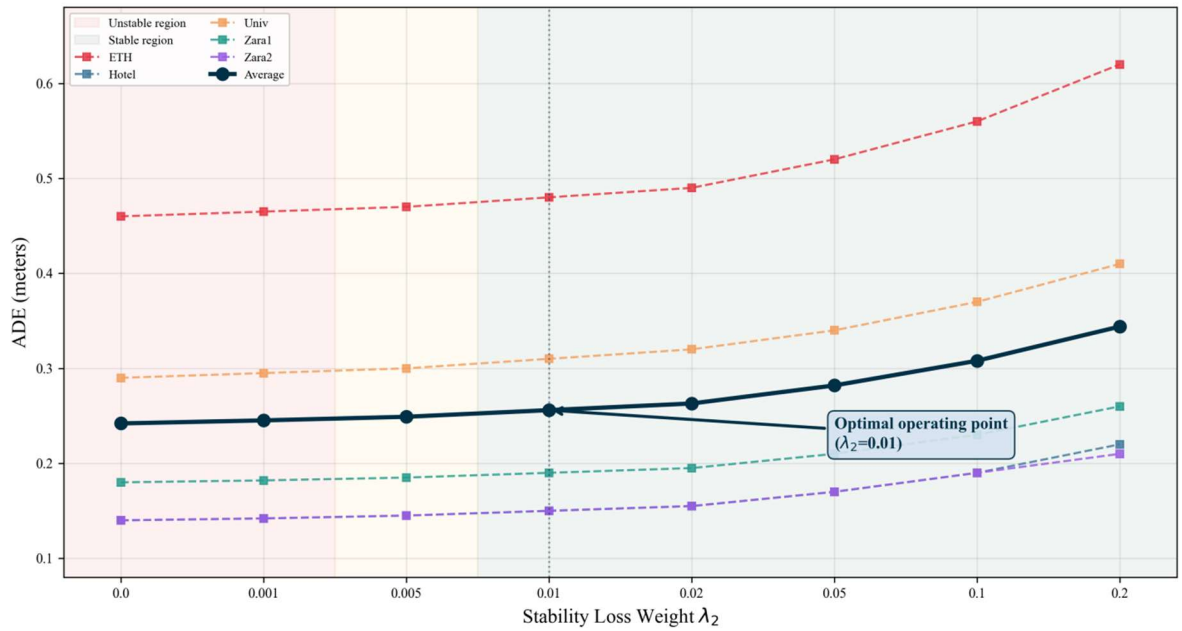

Fig. 3 Impact of stability loss weight λ_2 on prediction accuracy (ADE) across the ETH/UCY datasets.

6.4. Ablation Study

Table 2. Ablation Study Results (Average ADE/FDE across ETH/UCY)

Model Variant	ADE (m)	FDE (m)	Stable
Baseline Transformer	0.42	0.78	No
DDL (Transformer + DS)	0.33	0.57	Yes
Graph-only (Trans + GNN)	0.35	0.62	No
Graph-DDL (Full)	0.31	0.53	Yes

To isolate the contribution of each component, ablation experiments are conducted on all five ETH/UCY scenes. Four variants are compared: (a) Baseline Transformer, (b) Transformer with

ISSN: 2664-9640

dynamical system (DDL), (c) Transformer with ST-GNN but without dynamical system (Graph-only), and (d) full Graph-DDL model. Results are presented in Table 2. All experiments are repeated 5 times with different random seeds, and mean values are reported.

The results reveal several important findings. First, the addition of the dynamical system module (DDL vs. Baseline) yields a 21.4% improvement in ADE, demonstrating the effectiveness of stability-constrained trajectory generation. Second, incorporating the ST-GNN module (Graph-only vs. Baseline) provides a 16.7% ADE reduction, confirming the value of explicit graph-based interaction modeling. Third, the full Graph-DDL model achieves the best performance overall, with a 26.2% ADE improvement over the baseline, indicating that the graph-based interaction modeling and dynamical system constraints provide complementary benefits.

Table 3. Computational Cost Comparison

Method	Params (M)	Train (s/epoch)	Infer (ms)
Social-LSTM	0.34	18.5	2.1
Social-STGCNN	7.60	4.2	1.8
DDL	3.25	12.8	3.5
Graph-DDL	5.12	15.6	4.1

Table 3 presents the computational cost comparison. Graph-DDL has 5.12M parameters, which is larger than DDL (3.25M) due to the additional ST-GNN module, but the increase is moderate. The per-epoch training time of 15.6 seconds and inference time of 4.1 ms per sample are well within the real-time requirements of autonomous driving applications, confirming the practical feasibility of the proposed framework.

7. Conclusion and Future Work

This paper presents Graph-DDL, a novel pedestrian trajectory prediction framework that integrates Spatial-Temporal Graph Neural Networks with goal-directed asymptotically stable dynamical systems. By designing a dual-component dynamic adjacency matrix based on relative distance proximity and relative velocity alignment, the framework captures physically grounded pedestrian interaction patterns that evolve dynamically over time. The integration of asymptotic stability constraints ensures that all predicted trajectories converge to well-defined goal positions, providing formal mathematical guarantees and physical interpretability.

Experimental results on the ETH/UCY benchmark datasets demonstrate that Graph-DDL achieves state-of-the-art performance with an average ADE of 0.256 m and FDE of 0.418 m, outperforming the original DDL model by 6.2% in ADE and 7.1% in FDE. The ablation study confirms that both the graph-based interaction module and the dynamical system constraints contribute complementary improvements to prediction quality. The computational overhead introduced by the ST-GNN module remains acceptable for real-time applications.

Several directions for future work merit investigation. First, the current framework has been validated primarily on 2D pedestrian trajectory datasets; extension to 3D environments and heterogeneous traffic scenarios (e.g., involving vehicles, cyclists, and pedestrians simultaneously) remains an open challenge. Second, the dynamic adjacency matrix currently relies on pairwise relationships; incorporating higher-order group interactions through hypergraph neural networks may further improve crowd modeling fidelity. Third, integration with real-world perception pipelines (e.g., object detection and tracking systems) and deployment on edge computing platforms for autonomous vehicles represent important steps

ISSN: 2664-9640

toward practical application. Finally, exploring the incorporation of scene context information such as static obstacles, road boundaries, and semantic map features into the graph construction process may enhance prediction accuracy in structured environments.

References

[1] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," Int. J. Robot. Res., vol. 39, no. 8, pp. 895–935, 2020.

[2] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in Proc. IEEE 12th Int. Conf. Comput. Vis. (ICCV), 2009, pp. 261–268.

[3] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," Comput. Graph. Forum, vol. 26, no. 3, pp. 655–664, 2007.

[4] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," Phys. Rev. E, vol. 51, no. 5, pp. 4282–4286, 1995.

[5] R. Mehran, A. Oyama, and M. Shah, "Abnormal crowd behavior detection using social force model," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2009, pp. 935–942.

[6] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2016, pp. 961–971.

[7] C. Yu, X. Ma, J. Ren, H. Zhao, and S. Yi, "Spatio-temporal graph transformer networks for pedestrian trajectory prediction," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 507–523.

[8] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," in Proc. Adv. Neural Inf. Process. Syst. (NeurIPS), 2017, pp. 5998–6008.

[9] A. Mohamed, K. Qian, M. Elhoseiny, and C. Claudel, "Social-STGCNN: A social spatio-temporal graph convolutional neural network for human trajectory prediction," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2020, pp. 14412–14420.

[10] T. Mangalam, H. Girase, S. Aber, and J. Malik, "It is not the journey but the destination: Endpoint conditioned trajectory prediction," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 759–776.

[11] C. Wang, Y. Wang, M. Xu, and D. J. Crandall, "SSDL: A stable and scalable deep learning framework for trajectory prediction," IEEE Trans. Intell. Transp. Syst., vol. 24, no. 4, pp. 3896–3908, 2023.

[12] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social GAN: Socially acceptable trajectories with generative adversarial networks," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), 2018, pp. 2255–2264.

[13] Y. Yuan and K. Kitani, "DLow: Diversifying latent flows for diverse human motion prediction," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 346–364.

[14] J. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in Proc. Eur. Conf. Comput. Vis. (ECCV), 2020, pp. 683–700.

[15] Y. Shi, P. Tao, T. Fernando, et al., "Trajectory prediction with graph-based dual-frequency guidance," in Proc. AAAI Conf. Artif. Intell., 2023, pp. 11910-11918.