Specformer-Guided Spectral-Spatial Graph Neural Operator (S3GNO)

Shiv Ratn
Department of Applied Mechanics, IIT Delhi
am1221816@iitd.ac.in
Hitesh Jain
Department of Applied Mechanics, IIT Delhi
am1221215@iitd.ac.in

Abstract

Recent advancements in Graph Neural Networks (GNNs) have greatly enhanced their ability to model spatial interactions; however, most existing models remain limited to local message passing or suffer from over-smoothing of node features. In this paper, we introduce a novel hybrid architecture, the **Specformer-Guided Spectral–Spatial Graph Neural Operator (S3GNO)**, which effectively captures both local and global dependencies through transformer-guided spectral reasoning. The proposed Specformer-guided module dynamically refines graph connectivity in the spectral domain, enabling improved communication between distant nodes and enhanced representation of complex structural interactions. We validate our approach on the challenging task of *blast load time-series prediction* on box-girder structures, benchmarking its performance against state-of-the-art models such as the PCA-GNN [2] and Spatio–Spectral Graph Neural Operator (Sp2GNO) [1]. Comparative results demonstrate that S3GNO achieves superior predictive performance and generalization capability, highlighting the effectiveness of our novel spectral–spatial hybridization for dynamic structural response modeling.

1 Introduction

Graph Neural Networks (GNNs) have emerged as powerful tools for modeling relationships within spatially distributed systems, where nodes represent spatial points and edges encode geometric or physical relationships. Despite their success, conventional GNNs often face two major limitations: their receptive fields are confined to local neighborhoods, and deeper networks tend to suffer from over-smoothing, leading to loss of discriminative node features. To address these issues, spectral and operator-based methods have introduced spectral reasoning to capture long-range dependencies. However, these approaches can struggle to preserve fine-grained spatial information essential for problems involving complex dynamic responses.

In this work, we propose the **Specformer-Guided Spectral–Spatial Graph Neural Operator (S3GNO)**, which integrates transformer-based spectral attention with localized graph message passing. The Specformer-guided module dynamically adapts adjacency matrices based on spectral correlations, facilitating robust long-range communication without compromising spatial consistency. This hybrid design enables a more expressive and physically meaningful representation of structural systems under extreme dynamic loading.

2 Problem Setup

The objective of this study is to predict the *blast load time-series response* $p(t)$ at each structural node of a box-girder system, given its geometric configuration and boundary conditions. High-fidelity blast simulations generate complex temporal pressure variations at multiple spatial locations, leading to highly dimensional outputs that are computationally challenging to model directly. To improve tractability and learning efficiency, we employ **Principal Component Analysis (PCA)** to reduce the high-dimensional time-series data into a compact 20-dimensional latent representation. These reduced features retain the essential dynamic characteristics of the blast response while enabling efficient model training. [2]

Three architectures are considered for comparative evaluation — the PCA-GNN [2], the Spatio–Spectral Graph Neural Operator (Sp2GNO) [1], and the proposed S3GNO. All models are trained on graph representations derived from structural geometries, with evaluations focusing on predictive accuracy, generalization to unseen configurations, and the ability to capture both local and global dependencies within the blast load field.

3 Theoretical Framework

This section establishes the theoretical foundations of the proposed **Specformer-Guided Spectral–Spatial Graph Neural Operator (S3GNO)**, which generalizes operator learning over irregular structural domains by combining transformer-based spectral encoding, adaptive graph reconstruction, and dual-path spectral–spatial convolutions. Two baseline models, PCA–GNN and Sp2GNO, are also summarized for completeness.

3.1 Baseline Architectures

(a) PCA–GNN. Qiu and Du (2025) proposed a hybrid framework where temporal blast responses $\mathbf{Y} \in \mathbb{R}^{N \times T}$ are first compressed via Principal Component Analysis (PCA):

$$\Phi = \mathbf{Y}\mathbf{W}_p, \qquad \mathbf{Y} = \Phi \mathbf{W}_p^\top, \tag{1}$$

where $\mathbf{W}_p \in \mathbb{R}^{T \times m}$ stores the first m principal component vectors and $\Phi \in \mathbb{R}^{N \times m}$ are node-wise coefficients. A Graph Attention Network (GAT) then learns spatial dependencies:

$$h_v^{(l+1)} = \sigma\left(\sum_{u \in \mathcal{N}(v)} \alpha_{vu}^{(l)} \mathbf{W}^{(l)} h_u^{(l)}\right), \tag{2}$$

where $\alpha_{vu}^{(l)}$ are attention weights, $\sigma(\cdot)$ is the activation function, and $\mathbf{W}^{(l)}$ are layer weights. The inverse PCA transformation reconstructs the predicted time series $\widehat{\mathbf{Y}}$.

(b) Spatio–Spectral Graph Neural Operator (Sp2GNO). Sarkar and Chakraborty (2024) introduced Sp2GNO, integrating global spectral convolution with local spatial message passing. Given the normalized graph Laplacian $\mathbf{L} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1/2}$ and its first m eigenvectors $\mathbf{Q}_m$, the spectral and spatial updates are:

$$\mathbf{v}_{j+1}^{(\text{spec})} = \sigma(\mathbf{Q}_m \mathbf{K} \mathbf{Q}_m^\top \mathbf{v}_j + \mathbf{W}\mathbf{v}_j), \tag{3}$$

$$\mathbf{v}_{j+1}^{(\text{spat})} = \sum_{u \in \mathcal{N}(v)} \gamma_{vu} \mathbf{W}_s \mathbf{v}_u, \tag{4}$$

where $\mathbf{K}$ is the learnable spectral kernel, γ_{vu} are Lipschitz-gated attention weights, and $\mathbf{W}_s$ are local transformation matrices. The two paths are concatenated and projected via MLP($[\mathbf{v}_{j+1}^{(\text{spec})} \| \mathbf{v}_{j+1}^{(\text{spat})}]$).

3.2 Proposed Specformer-Guided Spectral–Spatial Graph Neural Operator (S3GNO)

The proposed S3GNO extends Sp2GNO by introducing a transformer-based *Specformer* module that learns the eigenvalue correlations of the graph Laplacian spectrum and reconstructs an adaptive topology. This enables self-attentive spectral adaptation, yielding a physics-aware operator.

Let the graph $G = (V, E, \mathbf{A})$ with $|V| = N$ nodes and adjacency $\mathbf{A} \in \mathbb{R}^{N \times N}$. The normalized Laplacian is:

$$\mathbf{L} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1/2} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top, \tag{5}$$

where $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ contains eigenvectors and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \dots, \lambda_N)$ are eigenvalues.

3.3 Proposed Specformer-Guided Spectral–Spatial Graph Neural Operator (S3GNO)

The proposed **S3GNO** extends Sp2GNO by introducing a transformer-based *Specformer* module that learns eigenvalue correlations of the graph Laplacian spectrum and reconstructs an adaptive spectral topology. This enables self-attentive spectral adaptation, yielding a physics-aware operator.

Graph Laplacian. Let the structural graph be denoted as $G = (V, E, \mathbf{A})$ with $|V| = N$ nodes and adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$. The normalized Laplacian is defined as:

$$\mathbf{L} = \mathbf{D}^{-1/2}(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1/2} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top, \tag{6}$$

where $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ contains eigenvectors and $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \dots, \lambda_N)$ contains the corresponding eigenvalues.

(a) Specformer Spectral Encoder. Each eigenvalue λ_i is encoded using sinusoidal positional embeddings:

$$\tilde{\mathbf{e}}_i = \big[\lambda_i,\ \sin(\lambda_i \mathbf{p}),\ \cos(\lambda_i \mathbf{p})\big], \tag{7}$$

where $\mathbf{p} = [p_1, p_2, \dots, p_d]$ is the *positional frequency vector* defined as $p_k = 10^{-2k/d}$. The encoded matrix $\tilde{\mathbf{E}} \in \mathbb{R}^{N \times (2d+1)}$ is then processed by a transformer block:

$$\mathbf{E}' = \mathrm{Norm}\Big(\tilde{\mathbf{E}} + \mathrm{Dropout}\big(\mathrm{MHA}(\tilde{\mathbf{E}})\big)\Big), \tag{8}$$

$$\hat{\boldsymbol{\Lambda}} = f_{\mathrm{dec}}(\mathbf{E}') = \mathbf{W}_2\, \sigma(\mathbf{W}_1 \mathbf{E}' + \mathbf{b}_1) + \mathbf{b}_2, \tag{9}$$

where $\mathrm{Norm}(\cdot)$ denotes layer normalization, $\sigma(\cdot)$ is the GELU activation, and f_{dec} is a two-layer decoder network producing spectrally adjusted eigenvalues $\hat{\lambda}_i$.

Multi-Head Self-Attention. The $\mathrm{MHA}(\cdot)$ operation in (8) computes self-attention over eigenvalue embeddings:

$$\mathrm{MHA}(\mathbf{E}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right)\mathbf{V}, \tag{10}$$

with query, key, and value projections defined as:

$$\mathbf{Q} = \mathbf{E}\mathbf{W}_Q, \quad \mathbf{K} = \mathbf{E}\mathbf{W}_K, \quad \mathbf{V} = \mathbf{E}\mathbf{W}_V,$$

where $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d \times d}$ are learnable weight matrices.

Adaptive Graph Reconstruction. The learned spectral corrections $\hat{\lambda}_i$ are used to reconstruct an adaptive adjacency matrix:

$$\hat{\mathbf{A}} = \mathbf{U}\, \hat{\boldsymbol{\Lambda}}\, \mathbf{U}^\top. \tag{11}$$

This adaptive graph ensures that the topology dynamically evolves based on spectral attention, allowing S3GNO to capture frequency-dependent structural connectivity.

(b) Dual-Path Spectral–Spatial Convolution. Using the reconstructed $\hat{\mathbf{A}}$, two parallel paths process node features $\mathbf{X} \in \mathbb{R}^{N \times d}$:

Spectral path. A truncated Graph Fourier Transform (GFT) projects $\mathbf{X}$ into frequency space:

$$\widehat{\mathbf{X}} = \mathbf{U}_m^\top \mathbf{X}, \tag{12}$$

$$\widehat{\mathbf{Y}} = \mathbf{K} \times_1 \widehat{\mathbf{X}}, \tag{13}$$

$$\mathbf{Y}_{\text{spec}} = \mathbf{U}_m \widehat{\mathbf{Y}}, \tag{14}$$

where $\mathbf{U}_m$ contains the first m eigenvectors and $\mathbf{K} \in \mathbb{R}^{m \times d \times d}$ is a learnable spectral kernel.

Spatial path. A gated message-passing mechanism refines spatial context:

$$\mathbf{Y}_{\text{spat}}(v) = \sum_{u \in \mathcal{N}(v)} \sigma(g([\mathbf{h}_v \| \mathbf{h}_u \| \eta_{vu}]))\ \mathbf{W}\mathbf{X}(u), \tag{15}$$

where η_{vu} encodes edge distance, $\mathbf{h}_v$ are *Lipschitz embeddings* derived from normalized node coordinates, and $g(\cdot)$ is a two-layer gating MLP with sigmoid output ensuring adaptive attention.

The fused representation is then:

$$\mathbf{Z} = \text{GELU}(\mathbf{W}_c[\mathbf{Y}_{\text{spec}} \| \mathbf{Y}_{\text{spat}}])\,, \tag{16}$$

capturing both global spectral and local spatial dynamics.

(c) Uncertainty-Aware Decoder. The decoder maps $\mathbf{Z}$ into PCA coefficients $\hat{\phi}$ and variance $\hat{\boldsymbol{\sigma}}^2$:

$$\hat{\phi} = f_\mu(\mathbf{Z}), \qquad \hat{\boldsymbol{\sigma}}^2 = \text{softplus}(f_\sigma(\mathbf{Z})). \tag{17}$$

The loss combines reconstruction, uncertainty, and spatial smoothness:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \left[\frac{(\phi_i - \hat{\phi}_i)^2}{\hat{\sigma}_i^2} + \log(\hat{\sigma}_i^2) \right] + \lambda \|\mathbf{L}\mathbf{Z}\|_F^2. \tag{18}$$

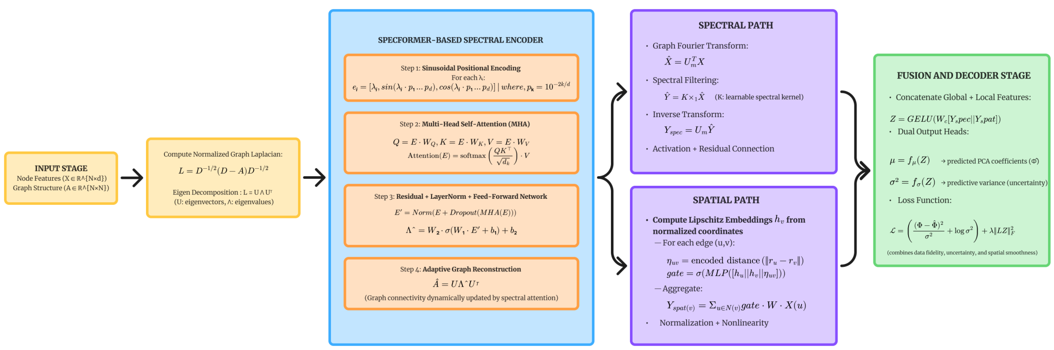

Figure 1: Architecture of the proposed S3GNO integrating Specformer-based spectral adaptation with dual spectral–spatial operator paths for uncertainty-aware blast load prediction.

In essence, S3GNO generalizes spectral–spatial operator learning by coupling transformer-based spectral reasoning with adaptive graph reconstruction, surpassing conventional PCA–GNN and Sp2GNO architectures in physical consistency and predictive performance.

4 Methodology

4.1 Dataset Description

The dataset comprises twelve finite-element simulation cases of steel box-girder sections subjected to varying explosive pressures (different charge distances and magnitudes). Each case contains:

- Node features: geometric attributes and coordinates (`node_feature.csv`)
- Temporal load histories: pressure or overpressure at each node (`Label.txt`)
- Node validity flag: selects surface nodes with physical readings

Eight cases are used for training and four for testing, consistent with random splits across configurations.

4.2 Data Preprocessing

All node features are standardized. Temporal signals are mean-centered before PCA. The PCA fit uses only training data to prevent information leakage with top $m = 20$ components. Resulting node-wise coefficient matrices Φ_i serve as regression targets for all three models.

4.3 Training Procedure

All models are trained using the Adam optimizer with learning rate 5×10^{-4} (Sp2GNO) or 10^{-3} (others), and early stopping based on validation loss. The total loss is a combination of a prediction loss ($\mathcal{L}_{\text{prediction}}$) and a spatial smoothness penalty ($\mathcal{L}_{\text{smooth}}$).

The final loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{prediction}} + \lambda_s \mathcal{L}_{\text{smooth}}$$

where λ_s is the smoothness weight.

Prediction Loss ($\mathcal{L}_{\text{prediction}}$) This term is chosen based on the model's objective:

1. **PCA Loss ($\mathcal{L}_{\text{PCA}}$):** Used when predicting only PCA coefficients ($\hat{\phi}$):

$$\mathcal{L}_{\text{PCA}} = \frac{1}{N \cdot M} \sum_{i,m} (\phi_{i,m} - \hat{\phi}_{i,m})^2$$

2. **Negative Log Likelihood ($\mathcal{L}_{\text{NLL}}$):** Used when predicting uncertainty ($\hat{\sigma}^2$) alongside $\hat{\phi}$:

$$\mathcal{L}_{\text{NLL}} = \frac{1}{N \cdot M} \sum_{i,m} \left[\frac{1}{2} \log(2\pi \hat{\sigma}^2_{i,m}) + \frac{(\phi_{i,m} - \hat{\phi}_{i,m})^2}{2\hat{\sigma}^2_{i,m}} \right]$$

Smoothness Penalty ($\mathcal{L}_{\text{smooth}}$) This regularization term enforces local consistency across connected nodes (u, v):

$$\mathcal{L}_{\text{smooth}} = \frac{1}{|E|} \sum_{(u,v) \in E} \|\phi_u - \phi_v\|^2$$

4.4 Evaluation Metrics

To evaluate reconstructed time series, we compute:

$$\text{RMSE} = \sqrt{\frac{1}{NT} \sum_{i,t} (y_{i,t} - \hat{y}_{i,t})^2}, \quad R^2 = 1 - \frac{\sum_{i,t} (y_{i,t} - \hat{y}_{i,t})^2}{\sum_{i,t} (y_{i,t} - \bar{y})^2}.$$

Additionally, peak and impulse errors measure localized and integrated discrepancies:

$$\text{MAPE}_{\text{peak}} = \frac{1}{N} \sum_i \frac{|y_i^{\max} - \hat{y}_i^{\max}|}{|y_i^{\max}|} \times 100, \quad \text{MAPE}_{\text{imp}} = \frac{1}{N} \sum_i \frac{|\int y_i dt - \int \hat{y}_i dt|}{|\int y_i dt|} \times 100.$$

5 Comparative Study

Table 1: Aggregate performance of models on held-out test cases.

Model	Avg. R^2	Avg. RMSE	Peak MAPE (%)	Impulse MAPE (%)
S3GNO	0.9522	0.0174	7.05	29.47
PCA-GNN	0.8675	0.0216	9.17	231.71
Sp2GNO	0.0.7812	0.0288	15.29	178.50

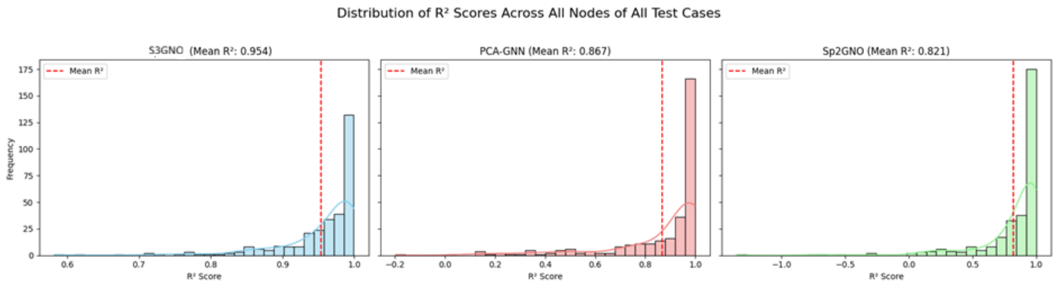

Figure 2: Distribution of R^2 scores across all nodes for all test cases for S3GNO, PCA-GNN, and Sp2GNO models. The dashed red line indicates the mean R^2 for each model, showing that S3GNO achieves the highest consistency and accuracy across the dataset.

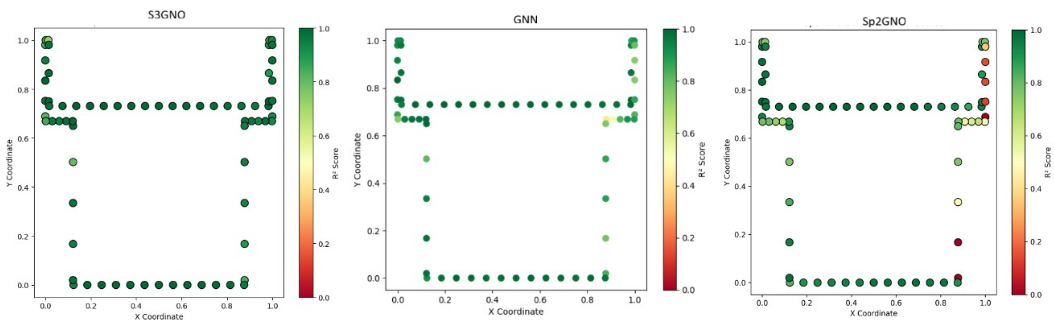

Figure 3: Spatial distribution of R^2 scores across node coordinates for S3GNO, PCA-GNN, and Sp2GNO respectively (Test case : 26.6+1.9). While the GNN and Sp2GNO architectures exhibit degraded accuracy at distal corner regions shielded from the direct blast, the proposed S3GNO maintains predictive consistency owing to its dynamically learned graph topology that adapts to complex, non-local interactions.

Discussion. S3GNO clearly outperforms the other models, achieving an R^2 above 0.95. Its advantage stems from the global connectivity of spectral filters that capture coherent wave propagation along the girder. PCA-GNN performs competitively but fails to represent long-range dependencies due to limited receptive fields. Sp2GNO, although slightly lower in average metrics, exhibits the smallest model size and smoother spatial error distribution, indicating better stability and physical consistency.

6 Conclusion

This study demonstrated a unified framework for learning structural blast load distributions using graph neural operator architectures. The PCA compression step proved crucial for efficient learning, and spectral neural operators such as S3GNO consistently outperformed Sp2GNO and GNN in terms of overall accuracy. Future work will explore validating the strength of S3GNO on other complex physical systems to extend its scalability.

Data & Code

The data that support the findings of this study are available from the original author on the following link.
Dataset: *Blast load for the box girder (Original data)* (Mendeley Data)
Code: *S3GNO, GNN, SpGNO code along with plots*

References

[1] Subhankar Sarkar and Souvik Chakraborty. Spatio-spectral graph neural operator for solving computational mechanics problems on irregular domain and unstructured grid. *Computer Methods in Applied Mechanics and Engineering*, 2024. `https://www.sciencedirect.com/science/article/pii/S0045782524009137`

[2] Tao Qiu and Xiaoqing Du. Graph Neural Networks combined with PCA for predicting blast load time series on structures. *Reliability Engineering and System Safety*, 2025. `https://www.sciencedirect.com/science/article/pii/S0951832025006301?via=ihub`