Proceedings of the International Conference on Industrial Engineering and Operations Management

Publisher: IEOM Society International, USA DOI: 10.46254/AP07.20260491
Published: March 25, 2026

Design Knowledge-Powered Graph Attention Network (GAT) for Performance Prediction of Signal Transmission Devices

Liuyang Shan and Francisco A. Fernandez De Castro Bobadilla
School of Mechanical Engineering
Georgia Institute of Technology, Atlanta, GA, USA
lshan8@gatech.edu

Shu Wang
TE Connectivity, Middletown, Pennsylvania, USA

Roger J. Jiao
School of Mechanical Engineering
Georgia Institute of Technology, Atlanta, GA, USA

Abstract

In simulation-driven product development, evaluating the signal response behavior of signal transmission devices often depends on repeated electromagnetic simulations, which become increasingly time-consuming as design complexity grows. This creates a need for data-driven methods that can support faster and more efficient early-stage design evaluation. To address this challenge, this paper proposes a design knowledge–powered Graph Attention Network (GAT) framework for predicting frequency-domain signal response intensity from product design representations. The proposed approach transforms CAD models into hierarchical knowledge graphs that capture assembly structures and geometric dependencies and uses these graph-based representations together with simulation results to train a predictive model for multi-curve response estimation. The framework enables efficient and structure-aware prediction of signal response behavior, providing a practical surrogate modeling solution for accelerating design exploration.

Keywords
Knowledge Graph, Design Model, Graph Attention Network, Sign Transmission Device, Performance Prediction

1. Introduction
In simulation-driven product development, evaluating geometry–performance relationships rely on time consuming simulations that generate response curves across operating conditions. As design complexity increases, the cost of obtaining these performance evaluations becomes a limiting factor in efficiently assessing candidate geometries. Consequently, enabling fast and reliable prediction of performance responses from design representations has become critical for supporting efficient engineering workflows (Zheng et al. 2025).

To address this challenge, this work proposes a design knowledge driven GAT pipeline designed to predict product performance responses directly from geometric representations. Historical three-dimensional designs are parameterized into standardized structural representations that capture geometric attributes and assembly relationships. These representations are paired with corresponding simulated electromagnetic characteristic response results obtained from prior evaluations, typically represented as response curves across relevant operating conditions, which is the structured dataset used to train a GAT that learns the relationship between geometric configurations and their associated performance curves. Once trained, the model can estimate the expected response behavior of newly generated designs without requiring full simulation or experimental testing.

By providing predictions of physical property curves, the proposed framework enables rapid evaluation of candidate geometries during early design stages. Engineers can use these predictions to identify promising design directions or detect potential performance issues before committing computational resources to detailed simulations.

While there remains room for further improvement in predictive accuracy, the current results already demonstrate the effectiveness and potential of the proposed model. Moreover, the GAT-based approach provides a substantial computational advantage over traditional simulation methods, enabling rapid evaluation of design candidates. In this way, predictive modeling of geometry-to-performance relationships serves as a practical tool for accelerating design exploration while maintaining reliable performance assessment.

2. Related Work

2.1 Hierarchical CAD Knowledge Graph Representation

Engineering design models contain heterogeneous information across multiple structural levels, including assembly hierarchies, component attributes, and geometric relationships. Representing this information in a unified structure is essential for enabling downstream analysis and data-driven modeling. Knowledge graphs have emerged as an effective representation framework for complex engineering systems by organizing entities and relationships into graph structures that support reasoning, retrieval, and learning. In product design research, knowledge graphs have been used to represent design-specific entities and semantic relationships, allowing machine learning models to operate directly on structured design knowledge. For example, design-oriented knowledge graph frameworks model engineering concepts as entities connected through semantic relations, enabling heterogeneous design information to be integrated into a unified relational structure while preserving contextual dependencies (Liu et al. 2022).

Recent studies have extended this idea by constructing knowledge graphs directly from CAD model repositories. In these approaches, the hierarchical structure of product assemblies is converted into graph representations where assemblies, subassemblies, and parts are represented as nodes and their hierarchical relationships as edges. Such assembly hierarchy graphs provide a natural representation of product architecture and design dependencies. Large-scale CAD repositories have been transformed into knowledge graphs by extracting assembly trees and encoding relationships between assemblies, subassemblies, and parts as graph edges (Bharadwaj et al. 2022). Similar graph-based representations have also been used to integrate heterogeneous engineering data from multiple modeling environments. In manufacturing systems, CAD models, process information, and manufacturing knowledge can be organized within a unified graph structure, often through layered architectures that separate graph schema definitions from instance-level engineering data (Li et al. 2017).

Beyond hierarchical structures, knowledge graphs can represent geometric dependencies between CAD entities. CAD models contain references among geometric elements such as edges, surfaces, and solid features that collectively define the topology of the design. Representing these references as graph edges enables geometric relationships to be encoded explicitly within the graph structure, supporting multi-scale representations that connect assembly-level structures with fine-grained geometric entities (Li et al. 2017).

Building on these developments, this work represents signal transmission device designs as hierarchical knowledge graphs that integrate assembly structures and geometric relationships extracted from CAD models. Component entities capture assembly-level hierarchy, while geometric entities describe fine-grained topology. The resulting multi-scale graph representation preserves structural dependencies of the signal transmission device and provides a structured interface for graph neural network models used to predict signal intensity performance.

2.2 Graph Neural Networks for Structure to Signal Response Modeling

Signal response intensity evaluation is a crucial step in the design of high-speed signal transmission devices and interconnect systems. Conventional analysis typically relies on full-wave electromagnetic simulation and circuit-level transient analysis, which become computationally expensive as geometric complexity and operating bandwidth increase. To accelerate design iterations, machine learning-based surrogate modeling has been increasingly explored.

Early studies focused on learning mappings from parameterized design variables to signal response intensity indicators such as eye height and eye width using regression models and deep neural networks (Lu et al. 2018). These works demonstrated potential of deep learning methods. However, most existing approaches rely on low-dimensional manually defined parameters and assume fixed structural configurations, limiting their ability to generalize across geometrically diverse designs.

Recent research has moved toward graph-based modeling to better capture structural relationships in electronic systems. Graph neural networks (GNNs) have been applied to model circuit topologies and device interconnections for high-speed link surrogate modeling (Li et al. 2023, A). In these methods, circuits are represented as graphs where nodes correspond to devices or ports and edges represent electrical connectivity, allowing topology-aware feature extraction before performance prediction. Similar graph-based approaches have also been used in electronic reliability analysis, where circuit netlists are converted into graphs and GNNs are used to predict delay degradation under process variations (Alrahis et al. 2022). Beyond circuit modeling, GNNs have demonstrated strong capability in learning physical responses of complex engineering systems. Graph-based surrogate models have been used to predict structural dynamic responses with significantly reduced computational cost compared with numerical simulations (Li, Wang et al. 2023, B). In parallel, survey studies on artificial intelligence methods for signal response prediction indicate that most existing approaches rely on feedforward, convolutional, or recurrent neural networks and face challenges in handling high-dimensional structural variations and limited training data (Shan et al. 2023). Meanwhile recent work has also explored graph constructions derived directly from geometric discretization of CAD models. Mesh-based graph representations combined with GNNs have been used to predict engineering performance metrics, providing more expressive structural features than low-dimensional parameter vectors (Park et al. 2024). However, mesh-level graphs mainly encode local geometric adjacency and often lack explicit modeling of assembly hierarchy or semantic part–subpart relationships.

Overall, prior studies demonstrate the effectiveness of machine learning and GNN-based surrogate modeling for electronic and physical systems. Nevertheless, most approaches rely on simplified parameter inputs or circuit-level connectivity graphs. The use of hierarchical graph representations constructed from CAD-derived multi-scale design information for predicting signal response intensity characteristics of complex signal transmission device remains largely unexplored, motivating the design knowledge modeling powered GAT framework proposed in this work.

3. System Analysis and Design

To enable data-driven prediction of signal response intensity behavior from signal transmission device designs, an end-to-end processing pipeline is developed to transform engineering design models into graph representations suitable for graph neural network learning. The system integrates CAD data extraction, knowledge graph construction, graph dataset generation, and graph neural network training into a unified workflow, bridging the gap between device design models and machine learning-based signal response intensity prediction.

The overall workflow of the system is illustrated in Figure 1, which describes the transformation from signal transmission device design models to graph-based learning datasets. The process begins with device models created in the Ansys Electronics Desktop (AEDT) environment. Each device design is associated with a corresponding electromagnetic simulation signal response intensity output file. In the proposed framework, each design instance is processed independently, establishing a one-to-one correspondence between the device model and its signal response intensity data.

Figure 2 shows the overall workflow for the pipeline for model training. The first stage of the pipeline focuses on extracting structural and geometric information from the AEDT design environment. After obtaining these two sources of structural information, the AEDT component hierarchy is integrated with the geometric entity definitions derived from the STEP model, which is processed in Design Export function part. Then based on the merged representation, a knowledge graph is constructed by Knowledge Graph Converter to encode the structural relationships within the signal transmission device model. To prepare the data for graph neural network learning, the Model Trainer will convert exported CSV files into PyTorch Geometric (PyG) graph objects. With all prepared design modeling knowledge graph and simulation results, a Graph Attention Network (GAT) model will be trained and perform signal response intensity prediction within the Model Trainer part. The data processing and training details will be discussed in sections 4, 5 and 6.

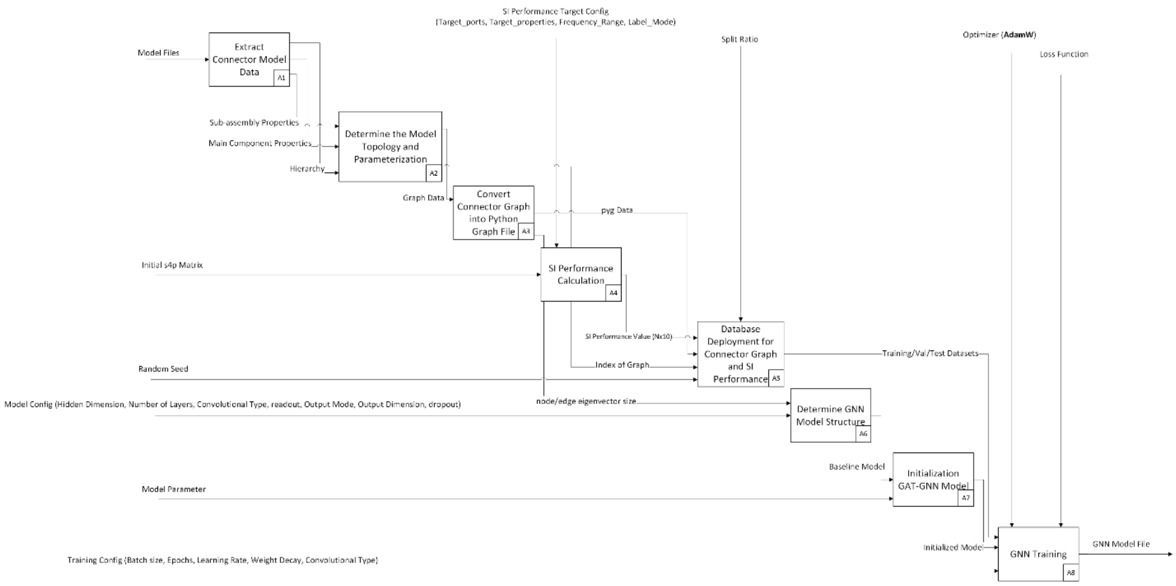

Figure. 1 IDEF0-based functional decomposition of the proposed system for signal transmission device modeling and prediction.

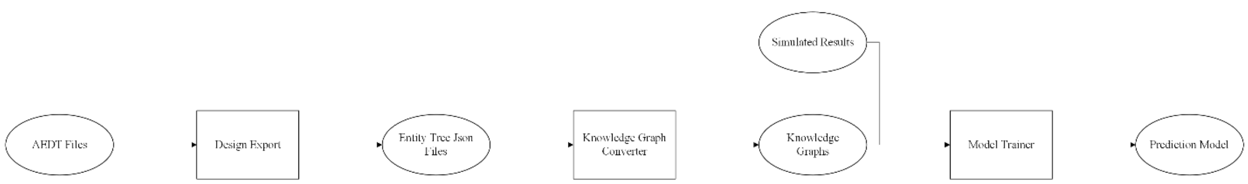

Figure.2 Overall workflow of the proposed data-driven pipeline for graph neural network training.

4. CAD-Based Knowledge Graph Parameterization of Signal Transmission Device Designs

To enable graph-based modeling of signal transmission device designs, structural information from CAD models must be converted into representations that preserve assembly hierarchy and geometric relationships. In this work, designs created in Ansys Electronics Desktop (AEDT), an example shown in Figure 3, are transformed into hierarchical knowledge graphs capturing the multi-scale structure of the device. Knowledge graphs provide a unified representation of engineering entities and their relationships, allowing heterogeneous design information to be integrated into a relational structure suitable for downstream learning tasks (Liu et al. 2022).

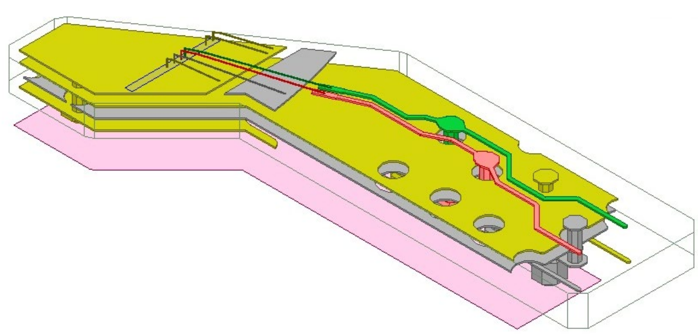

Figure 3. An illustrative example of an AEDT-based device model. This model is for visualization purposes only and is not part of the experimental dataset.

The parameterization pipeline converts the CAD model through intermediate representations before producing the final knowledge graph, as illustrated in Figure 4. Device designs are stored as AEDT project files containing assembly organization and geometric definitions. The geometry is exported to STEP format using the PyAEDT interface, which provides a standardized description of geometric entities, parameters, and reference relationships. In parallel, the assembly hierarchy, shown in Figure 5, is extracted from the AEDT project to capture parent–child relationships among components.

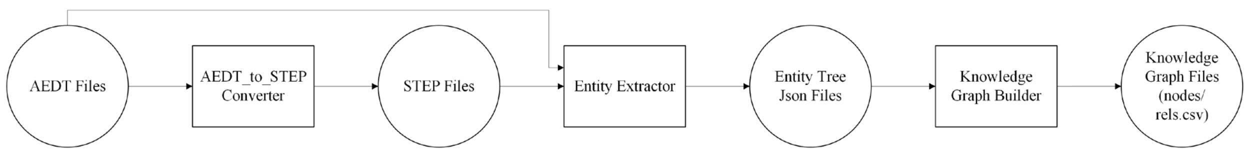

Figure. 4 Pipeline for knowledge graph generation from AEDT design files. The process includes geometry conversion, entity extraction, and graph construction, transforming raw design data into a structured knowledge graph representation.

Figure 5. Partial view of the extracted assembly hierarchy from AEDT, showing structured component and material information in a JSON-like format.

The STEP representation is processed through an entity extraction stage in which geometric entities are parsed into structured records containing entity identifiers, types, parameter attributes, and reference links. These entities are organized into intermediate entity-tree JSON structures where each node maintains references to its dependent entities. This representation preserves geometric dependency relationships while remaining independent of the original CAD format.

The extracted entities are then integrated with the assembly hierarchy obtained from AEDT. Component names from the AEDT hierarchy are matched to STEP entities through string correspondence in entity parameters. Entities associated with each component are grouped, and internally referenced entities are removed to retain root entities representing primary geometric definitions. Reference relationships are then recursively traversed from these roots to construct reference trees that capture the geometric dependency structure of the CAD model. This process merges assembly hierarchy and geometric relationships into a unified intermediate representation as shown in Figure 6.

```
[
  {
    "name": "NET1",
    "material": "copper",
    "step_entities": [
      {
        "id": 7875,
        "type": "MANIFOLD_SOLID_BREP",
        "editable": true,
        "refs": [
          7874
        ],
        "parameters": [
          "NET1"
        ]
      },
      {
        "id": 7898,
        "type": "DESCRIPTIVE_REPRESENTATION_ITEM",
        "editable": true,
        "refs": [],
        "parameters": [
          "AEDT_EntityName_V1",
          "NET1"
        ]
      }
    ],
    "reference_trees": [
      {
        "id": 7875,
        "type": "MANIFOLD_SOLID_BREP",
        "editable": true,
        "parameters": [
          "NET1"
        ],
        "children": [
          {
            "id": 7874,
            "type": "CLOSED_SHELL",
            "editable": true,
            "parameters": [],
            "children": [
              {
                "id": 4608,
                "type": "ADVANCED_FACE",
                "editable": true,
                "parameters": [
                  true
                ],
                "children": [
                  {
                    "id": 4596,
                    "type": "PLANE",
                    "editable": true,
                    "parameters": [],
                    "children": [
                      {
```


Figure 6. Example branch of the extracted entity tree, illustrating the hierarchical organization of components and their associated attributes.

The resulting knowledge graph is exported as node and relationship CSV tables, providing a standardized format for graph databases and machine learning workflows. The node table stores attributes Component (blue) representing elements of the AEDT assembly hierarchy and StepEntity (red) nodes representing geometric entities from the STEP model, while the relationship table records 3 relationship types: HAS_CHILD for assembly hierarchy, MATCHED linking components to their associated STEP entities, and REFERS_TO representing geometric reference dependencies. Example of a knowledge graph is shown in Figure 7. Through this pipeline, signal transmission device CAD models are systematically converted into graph representations encoding assembly hierarchy and geometric dependencies.

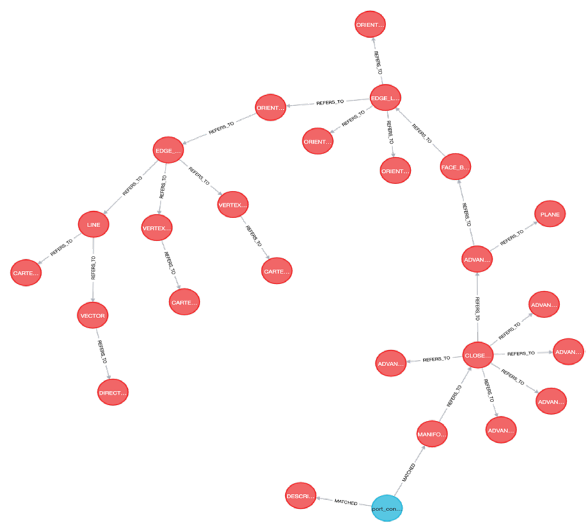

Figure 7. Partial visualization of the constructed knowledge graph, illustrating entities and their relationships; only a subset is shown for clarity.

5. Graph Neural Network-Based Signal Intensity Curve Prediction

After converting the knowledge graph representation of the signal transportation device into a PyTorch Geometric (PyG) format, the signal intensity modeling task is formulated as a graph-to-multi-curve regression problem. Each device instance is represented as a graph $G = (X, E, A)$, where X denotes node attributes derived from CAD entities, E encodes structural relationships among entities, and A represents types edge attributes. The objective of the model is to learn a mapping $f(G) \rightarrow Y$, where $Y \in \mathbb{R}^{N \times K}$ denotes the frequency-domain signal intensity response (unit: dB) matrix, N is the number of frequency sampling points and K is the number of predicted curves in the corresponding curve group.

Graph neural networks provide a natural framework for modeling structure–response relationships in complex engineering systems. Prior studies have demonstrated the effectiveness of graph-based surrogate modeling for high-speed interconnects and circuit-level performance prediction (Li et al. 2023, A). Unlike approaches relying on manually defined parameter vectors (Li et al. 2023, A), the proposed method learns directly from hierarchical CAD-derived graphs, allowing structural dependencies to be captured through message passing.

To better model these dependencies, a GAT encoder is adopted (Veličković et al. 2018). GAT uses masked self-attention to compute adaptive importance weights between nodes and their neighbors, enabling structure-aware feature aggregation without spectral graph convolution. In hierarchical CAD graphs, different geometric entities contribute unequally to signal intensity curves; attention mechanisms allow the model to emphasize structurally influential components during message passing. Multi-head attention further stabilizes representation learning (Veličković et al. 2018). In the implemented architecture, stacked GAT layers perform graph message passing, and the resulting node embeddings are aggregated using a permutation-invariant readout to produce a global device representation.

To better align the optimization objective with the physical characteristics of frequency-domain response curves, a modified curve-aware loss function is adopted. Conventional regression losses such as L1 or L2 primarily penalize point-wise differences and often fail to capture resonance behaviors in frequency responses. In contrast, the proposed loss incorporates additional regularization terms derived from first- and second-order finite differences along the frequency axis. Let $y \in \mathbb{R}^T$ denote the ground-truth curve and $\hat{y} \in \mathbb{R}^T$ denote the predicted curve, where T is the number of total predicted points. The overall loss function is defined as

$$Loss_{total} = L_1 + \lambda_{smooth} \cdot L_{smooth} + \lambda_{osc} \cdot L_{osc} + \lambda_{res\,slope} \cdot L_{res\,slope} + \lambda_{res\,value} \cdot L_{res\,value} \tag{1}$$

The first term is a point-wise regression loss defined as the mean absolute error over all predicted samples:

$$L_1 = \frac{1}{T}\sum_{t=1}^{T}|\widehat{y_t} - y_t| \tag{2}$$

To capture curve shape characteristics, first- and second-order finite differences are computed along the frequency axis

$$\Delta y_t = y_{t+1} - y_t,\ \Delta^2 y_t = \Delta y_{t+1} - \Delta y_t \tag{3}$$

A smoothness mask is used to identify locally smooth regions of the ground-truth curve:

$$M_{smooth}(t) = \begin{cases} 1, |\Delta^2 y_t| < 2 \\ 0, otherwise \end{cases} \tag{4}$$

The smoothness loss penalizes excessive curvature in these regions:

$$L_{smooth} = \frac{1}{|M_{smooth}|}\sum_t \left(M_{smooth(t)}(\Delta^2 \widehat{y_t})^2\right) \tag{5}$$

To suppress artificial oscillations, a slope sign representation is computed

$$s_t = \tanh(10 \cdot \Delta\widehat{y_t}) \tag{6}$$

and abrupt sign changes are penalized as

$$L_{osc} = \frac{1}{|M_{smooth}|}\sum_t (M_{smooth}(t)(S_{t+1} - S_t)^2) \tag{7}$$

To capture resonance behaviors characterized by sharp local variations (Helszajn et al. 1983), two additional penalties are introduced. The resonance slope loss measures the difference between predicted and true slope magnitudes in high-gradient regions

$$L_{res\,stop} = \frac{1}{|M_{res\,slope}|}\sum_t \left(M_{res\,slope}(t)\big||\Delta\widehat{y_t}| - |\Delta y_t|\big|\right) \tag{8}$$

where $M_{res}(t)$ indicates resonance regions determined by a threshold on $|\Delta y_t|$. The resonance magnitude loss further emphasizes accurate prediction of response values near resonance points through a dilated resonance mask

$$L_{res\ value} = \frac{1}{|M_{res\ val}|} \sum_t (M_{res\ val}(t) |\hat{y}_t - y_t|) \tag{9}$$

The overall objective combines these components using tunable weighting coefficients to balance numerical accuracy, smooth spectral behavior, and accurate resonance modeling. Detailed training configurations are provided in the case study section.

6. Case Study of Signal Response Intensity Prediction in Signal Transmission Device Product

6.1 Signal Intensity Prediction for a Four-Port Signal Transmission Device

To evaluate the practical applicability of the proposed framework, a case study is conducted using a four-port high-performance signal transportation device family commonly used in advanced electronic systems. Figure 8 shows an example of a signal transmission device's signal response intensity curve from simulation.

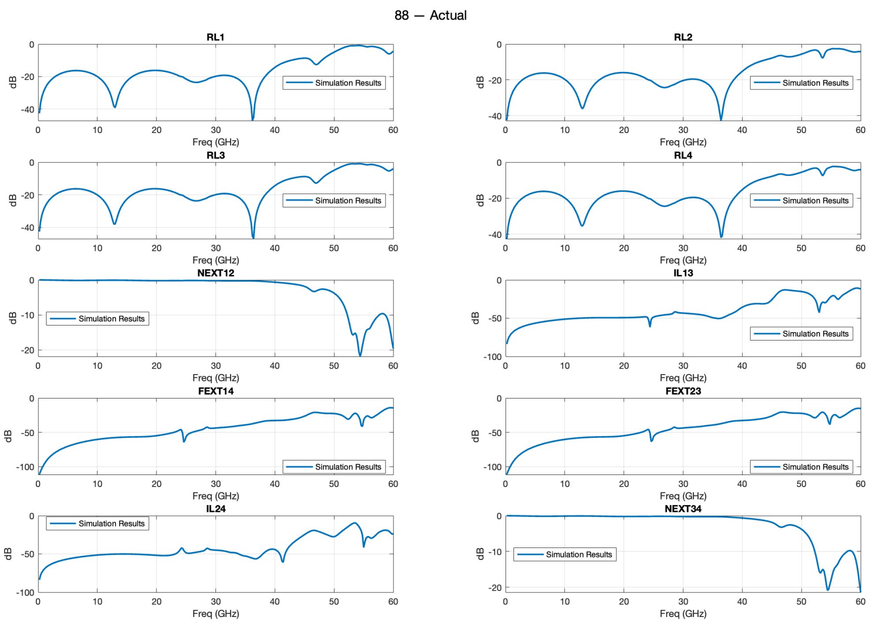

Figure 8. Simulated signal intensity curves of a representative four-port signal transportation device.

In this study, the prediction problem is formulated as a multi-curve dependent regression task. The signal intensity response of the device is represented by several characteristic curves associated with interactions among the four ports. (Piger et al. 2009) Specifically, four predictive models are trained to estimate different groups of curves: the first model predicts the four curves RL1, RL2, RL3, and RL4, the second model predicts FEXT14 and FEXT23, the third model predicts NEXT12 and NEXT34, and the fourth model predicts IL13 and IL24. Each model focuses on learning the structural factors that influence a specific category of signal intensity behavior (Borchani et al. 2015).

6.2 Graph Dataset Construction

To illustrate how the parameterization framework converts signal transmission device designs into graph representations, 217 devices are used in the case study. Each device is represented by an AEDT design model and their corresponding simulation results, which generates 217 knowledge graphs to represent device designs.

After constructing the knowledge graph representation for each device, the graph data are converted into graph datasets suitable for graph neural network training. The node and relationship tables exported from the knowledge graph are transformed into PyTorch Geometric (PyG) graph objects through a schema-aware conversion module, enabling direct processing by GAT. (Veličković et al. 2017)

Specifically, the node table and relationship table stored in CSV format are first loaded and parsed to reconstruct the graph structure. Each node is mapped to a numerical index, and its attributes are converted into a feature vector representation. Node features include component attributes, entity types, material properties, and parameter values extracted from the CAD model. Continuous attributes are normalized while categorical attributes are encoded using one-hot representations. In addition, parameter values extracted from the STEP entities are represented using a dual-channel encoding that distinguishes numerical values from missing or placeholder entries. Through this process, each node is represented by a fixed-dimensional feature vector.

The relationship table is then converted into the graph edge structure. Each relationship is represented as a directed edge between two nodes, and the relationship types are encoded as edge features. In the current graph schema, three relationship types are used: HAS_CHILD, MATCHED, and REFERS_TO, which respectively represent assembly hierarchy, component–entity correspondence, and geometric reference relationships. These relationships are transformed into edge indices and edge feature vectors in the PyG graph representation.

After the conversion process, each device is represented as a graph object $G = (X, E)$, where X denotes the node deature matrix and E denotes the graph connectivity defined by edge indices and edge features. The resulting graph representation preserves both the hierarchical assembly structure and the geometric dependency relationships extracted from the CAD model.

The dataset used in this study contains 217 device instances, each represented by a knowledge graph and a corresponding simulation result file. The dataset is divided into 153 training samples, 32 validation samples, and 32 test samples. Each graph instance corresponds to a single device model and is paired with its associated frequency-domain simulation data. Due to the detailed geometric representation extracted from the CAD model, the resulting graphs are relatively large, containing approximately 20,000–30,000 nodes and a comparable number of edges per graph on average. Table 1 contains all detailed information for generated dataset.

The target outputs for each graph are derived from the corresponding simulation results and represented as multiple signal intensity curves describing interactions among the four ports of the device. Specifically, the prediction targets include the curves RL1–RL4, FEXT14–23, NEXT12-34, and IL13-24, each sampled across the frequency range of interest. These graph representations together with their corresponding curve responses form the supervised dataset used to train and evaluate the proposed graph neural network model (Table 1).

Table 1. Dataset Summary Table

Property	Value
Number of devices	217
Training Samples	152
Validation Samples	32
Testing Samples	32
Average nodes/edges per graph	20-30k
Node feature dimension	47
Edge types	[HAS_CHILD, MATCHED, REFERS_TO]
Prediction targets	[RL1, RL2, RL3, RL4, FEXT14, FEXT23, NEXT12, NEXT34, IL13, IL24]

6.3 GNN Training Setup

To train the proposed graph neural network models, the graph dataset described in the previous section is used together with the corresponding signal intensity curves obtained from electromagnetic simulations. The training process follows a supervised learning setup in which each knowledge graph instance is paired with its corresponding multi-curve response data.

Since the signal intensity responses represent different interaction behaviors among the four ports of the device, the prediction task is decomposed into four model groups, each responsible for a specific category of curves. The first model, RL_Model, is trained to predict the four curves RL1, RL2, RL3, and RL4, which correspond to the responses associated with individual ports. The second model, FEXT_Model, focuses on the curves FEXT14 and FEXT23, while the third model, NEXT_Model, predicts NEXT12 and NEXT34. The fourth model, IL_Model, is trained to estimate the coupling curves IL13 and IL24. Each model therefore learns the relationship between the graph representation of the device and a specific subset of signal intensity curves. This decomposition allows the models to focus (Evgeniou, et al. 2004) on curve groups with similar structural characteristics while reducing the learning complexity associated with predicting all curves simultaneously

During training, the loss function parameters are configured as follows: the smoothness weight , $\lambda_{smooth} = 3 \times 10^{-4}$, the oscillation factor is set to $\lambda_{osc} = 10^{-3}$, the slope-based resonance factor is set to $\lambda_{resonance\,slope} = 10^{-3}$, and the value-based resonance factor is set to $\lambda_{resonance\,value} = 2.5 \times 10^{-3}$. The thresholds controlling the detection of smooth regions and resonance regions are set to 2.0 and 5.0, respectively, with a scaling factor $\alpha = 10$.

The graph neural network model is trained using mini-batch stochastic optimization. The batch size is set to 6, and the training is conducted for 1500 epochs using the Adam optimizer with a learning rate of $LR = 10^{-3}$ and weight decay of $WD = 10^{-4}$. To ensure reproducibility, the random seed is fixed at 36. Network architecture employs a GAT backbone with eight message-passing layers, a hidden feature dimension of 288, and four attention heads per layer. Dropout with a rate of 0.1 is applied during training to mitigate overfitting. The node feature dimension is fixed at 47 to maintain consistent graph representations across all devices, and an attention-based readout mechanism is used to generate the graph-level representation for prediction. Model selection is performed based on validation performance, and the model achieving the lowest validation loss is used for evaluation on the validation set.

6.4 Prediction Results

To evaluate the effectiveness of the proposed framework, the trained models are tested on the held-out test dataset consisting of 32 device instances. Each test sample contains a knowledge graph derived from the CAD model and the corresponding simulation-derived signal intensity curves. All curves are sampled across the frequency range of 0–60 GHz with a 0.2 GHz resolution, resulting in 300 sampling points per curve. Hence the four models mentioned above should have following output sizes: RL_Model: 300×4 ; FEXT_Model: 300×2 ; NEXT_Model: 300×2 ; IL_Model: 300×2.

Figure 9 presents a representative example comparing the predicted curves with the simulation results for one device in the test set. The figure shows the prediction results for the ten signal intensity curves, including RL1–RL4, FEXT14–23, NEXT12-34, and IL13–24. As illustrated in the figure, the predicted curves closely follow the overall trends of the simulation curves throughout the frequency range. In particular, the model successfully captures both the global variations of the curves and the local resonance behaviors that appear in specific frequency regions.

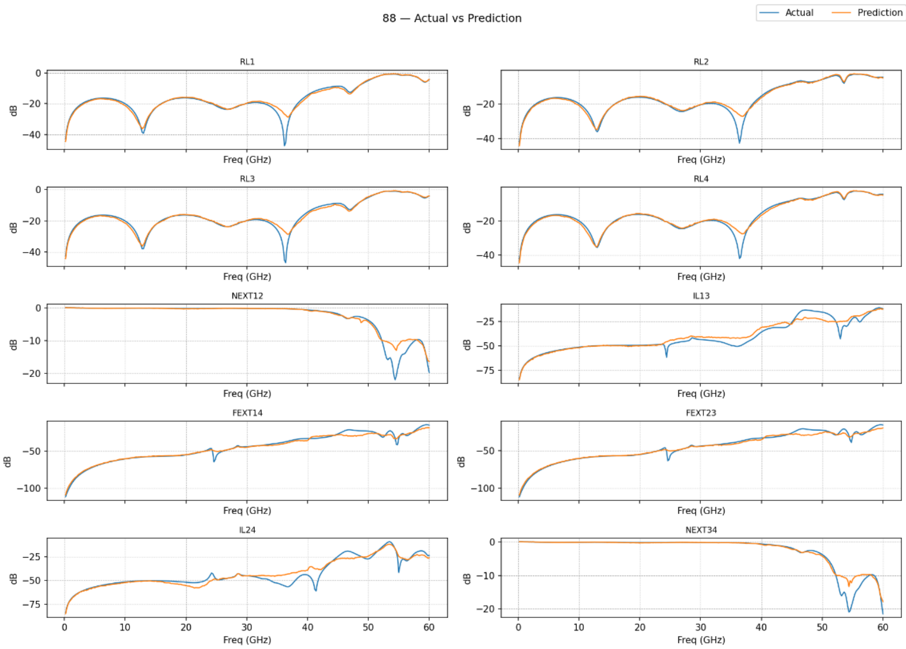

Figure 9. Comparison between simulation results and model predictions for signal intensity curves of a representative device in the test dataset. (Each curve is sampled from 0–60 GHz with 300 frequency points.)

To quantitatively evaluate prediction performance, the Mean Absolute Relative Error (MARE), shown in equation 10, is computed for each curve across the test dataset.

$$MARE = \sum \frac{|prediction - actual|}{|actual|} \qquad (10)$$

The results are summarized in Table 1. For the RL-related curves (RL1–RL4), the average MARE values are 12.75%, 12.37%, 12.69% and 12.07%. The curves predicted by the FEXT model achieve MARE of 7.62% and 7.6% for FEXT14 and FEXT23. The NEXT model MARE yield to 27.4% and 26.7% for NEXT12 and NEXT34. For the IL curves (IL13 and IL24), MAREs reach 9.18% for both IL13 and IL24.

7. Conclusion

This study addresses a key limitation in signal transmission device development workflows: the dependence of design iteration speed on repeated high-cost electromagnetic simulations. A design knowledge modeling powered GAT framework was developed in which signal transmission device CAD models are converted into hierarchical knowledge graphs and used as inputs to GAT for early-stage signal response intensity prediction. By linking assembly hierarchy from AEDT with geometric dependencies extracted from STEP representations, the proposed approach encodes performance-relevant structural information in a relational format suitable for graph-based learning.

Case study results demonstrate the feasibility of predicting frequency-domain signal intensity responses for four-port signal transportation devices. On a held-out test set, the model achieved MARE of 12.45% for RL, 7.61% for FEXT, and 9.18% for IL, while maintaining reasonable accuracy for the more challenging NEXT group at 27.08%. Predicted curves captured both global response trends and localized resonance behavior across the 0–60 GHz range. These findings indicate that hierarchical graph representations can produce surrogate predictions of signal response intensity from CAD structures in certain cases, enabling rapid preliminary screening of design alternatives and reducing reliance on repeated full-wave simulations during engineering design exploration.

References

Alrahis, L., Knechtel, J., Klemme, F., Amrouch, H. and Sinanoglu, O., GNN4REL: Graph neural networks for predicting circuit reliability degradation, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 11, pp. 3826–3837, 2022.

Bharadwaj, A. G. and Starly, B., Knowledge graph construction for product designs from large CAD model repositories, *Advanced Engineering Informatics*, vol. 53, pp. 101680, 2022.

Borchani, H., Varando, G., Bielza, C. and Larrañaga, P., A survey on multi-output regression, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 5, no. 5, pp. 216–233, 2015.

Evgeniou, T. and Pontil, M., Regularized multi-task learning, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 109–117, Seattle, WA, USA, August 2004.

Helszajn, J. and Sharp, J., Resonant frequencies, Q-factor, and susceptance slope parameter of waveguide circulators using weakly magnetized open resonators, *IEEE Transactions on Microwave Theory and Techniques*, vol. 31, no. 6, pp. 434–441, 1983.

Li, Q., Wang, Z., Li, L., Hao, H., Chen, W. and Shao, Y., Machine learning prediction of structural dynamic responses using graph neural networks, *Computers & Structures*, vol. 289, pp. 107188, 2023, B.

Li, X., Zhang, S., Huang, R., Huang, B., Xu, C. and Kuang, B., Structured modeling of heterogeneous CAM model based on process knowledge graph, *The International Journal of Advanced Manufacturing Technology*, vol. 96, no. 9, pp. 4173–4193, 2018.

Li, Z., Li, X. C., Wu, Z. M., Zhu, Y. and Mao, J. F., Surrogate modeling of high-speed links based on GNN and RNN for signal integrity applications, *IEEE Transactions on Microwave Theory and Techniques*, vol. 71, no. 9, pp. 3784–3796, 2023, A.

Liu, A., Zhang, D., Wang, Y. and Xu, X., Knowledge graph with machine learning for product design, *CIRP Annals*, vol. 71, no. 1, pp. 117–120, 2022.

Lu, T., Sun, J., Wu, K. and Yang, Z., High-speed channel modeling with machine learning methods for signal integrity analysis, *IEEE Transactions on Electromagnetic Compatibility*, vol. 60, no. 6, pp. 1957–1964, 2018.

Park, J. and Kang, N., BMO-GNN: Bayesian mesh optimization for graph neural networks to enhance engineering performance prediction, *Journal of Computational Design and Engineering*, vol. 11, no. 6, pp. 260–271, 2024.

Piger, J., Econometric models and methods for forecasting recessions, *Journal of Economic Surveys*, vol. 23, no. 3, pp. 498–527, 2009.

Shan, G., Li, G., Wang, Y., Xing, C., Zheng, Y. and Yang, Y., Application and prospect of artificial intelligence methods in signal integrity prediction and optimization of microsystems, *Micromachines*, vol. 14, no. 2, pp. 344, 2023.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P. and Bengio, Y., Graph attention networks, *arXiv preprint arXiv:1710.10903*, 2017.

Zheng, K., Zhong, Y., Su, X., Leng, J., Liu, Q. and Chen, X., Towards agentic smart design: An industrial large model-driven human-in-the-loop agentic workflow for geometric modelling, *Applied Soft Computing*, vol. 185, pp. 113920, 2025.

Biographies

Liuyang Shan is a Mechanical Engineering student at the Georgia Institute of Technology in Atlanta, GA, USA. He has participated in multiple real-world industrial projects focused on AI-powered manufacturing and automation processes. His work centers on applying data-driven methods to enhance engineering workflows and system efficiency. Additional information can be found on his LinkedIn profile: https://www.linkedin.com/in/liuyang-shan-898526335/

Francisco Fernandez de Castro Bobadilla is a Master's student in Mechanical Engineering at the Georgia Institute of Technology. Prior to pursuing his graduate studies, he worked as an Automation Engineer at TE Connectivity, where he gained hands-on experience in industrial automation. His research interests include mechatronics and AI-powered manufacturing and automation processes. More information about his projects and professional trajectory can be found on his LinkedIn profile: https://www.linkedin.com/in/francisco-fernandez-de-castro-1b8a23267

Shu Wang is a applied machine learning engineer at TE Connectivity. His expertise spans in industrial AI and data analytics, engineering design decision-making, human-automation interaction, and operations optimization. He holds a Ph.D. in Mechanical Engineering from Georgia Tech, along with dual master's degrees in Mechanical Engineering and Electrical and Computer Engineering. He earned his bachelor's degree in Electrical engineering from Wuhan University. More info about his research: https://scholar.google.com/citations?user=_R1ldZkAAAAJ&hl=en

Roger Jiao is the editor-in-chief of Journal of Engineering Design and an associate professor of mechanical and industrial engineering at Georgia Tech, USA. Prior to joining the School of Mechanical Engineering at Georgia Tech in December 2008, he was an Assistant Professor and then Associate Professor in the School of Mechanical and Aerospace Engineering at Nanyang Technological University, Singapore. Before his career in Singapore, he was a Visiting Scholar in the Department of Industrial Engineering and Engineering Management at Hong Kong University of Science and Technology from 1998 to 1999. From 1993 to 1994, he was a Lecturer of Industrial Engineering in the School of Management at Tianjin University, China, and from 1988 to 1990, he worked as an Associate Lecturer in the Department of Industrial Design at Tianjin University of Science and Technology, China. More info about his research: https://scholar.google.com/citations?user=9yikEHAAAAAJ&hl=en&oi=ao.