A DEEP LEARNING APPROACH TO A RAPID AND ACCURATE SURROGATE MODELLING OF FINITE ELEMENT ANALYSIS

Vedant Singh
Department of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
vhsingh@andrew.cmu.edu

Arushi Jain
Department of Mechanical Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
arushij@andrew.cmu.edu

Abstract

The assessment of mechanical properties such as stress and strain is crucial in mechanical engineering for analyzing complex structures, particularly in biomedical engineering. Traditional methods such as Finite Element Analysis (FEA) are integral but suffer from high computational demands that make the process lengthy and impractical for timely decision-making. This work introduces a significant advancement with the integration of Deep Learning-based Finite Element Analysis (DLFEA). DLFEA combines deep learning with traditional FEA to create efficient surrogate models that rapidly predict complex simulation outcomes, enabling quicker evaluations of new designs. Specifically, we apply DLFEA to the design and analysis of Bioprosthetic Heart Valves (BHVs), a common heart valve replacement that is prone to fatigue failure and difficult to assess directly from medical images. By employing a Graphical Neural Network that learns the biomechanics of valve deformation directly from past simulations, the DLFEA framework not only expedites the analysis process but also ensures high accuracy, effectively predicting deformations and coaptation areas of heart valves with remarkable precision. This novel approach promises substantial improvements in the feasibility and efficiency of applying FE methods in critical biomedical applications, potentially extending to other types of BHVs to enhance patient care and support personalized valve design.

1 Introduction

Semilunar valves, crucial for cardiac function, regulate blood flow by directing it during systole and preventing backflow during diastole. Valvular heart disease affects over 2.5% of the U.S. population, often leading to stenosis due to leaflet calcification or regurgitation from insufficient closure. These conditions frequently necessitate interventions such as valve repair or replacement to avert heart failure or death. Each year in the U.S., about 90,000 prosthetic valves are implanted, with bioprosthetic heart valves (BHVs) being favored for their superior hemodynamic performance compared to mechanical valves. However, BHVs are prone to fatigue failure, generally lasting only 10-15 years. Accurately predicting their lifespan from medical imaging is essential but challenging, underscoring the need for timely replacements to enhance patient outcomes. Furthermore, valve performance metrics are vital for physicians to make informed decisions regarding valve replacement, thus preventing premature surgeries. Computational analysis of heart valves serves as a crucial tool for clinicians, offering insights into the origins of valvular diseases and assisting in therapeutic decisions, especially when linked to stress concentrations in valve leaflets. Optimizing these computational analyses is critical for timely and precise clinical decisions. This process, by providing a deeper

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

understanding of the stresses and strains under various physiological conditions, aids in predicting durability and failure modes, supports the design of more effective and durable valve replacements, and facilitates a personalized approach to treatment, ultimately improving patient outcomes and quality of life.

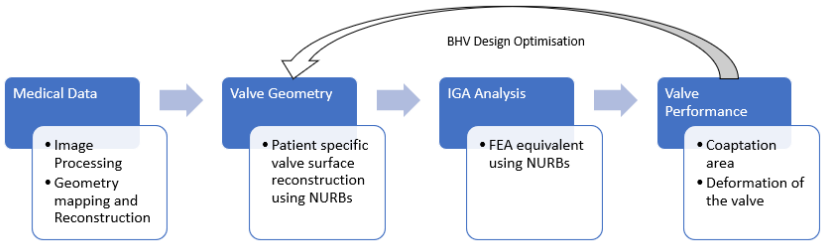


Figure 1: Design process of a BHV (Bio Prosthetic Heart Valve)

While these traditional analysis frameworks (as highlighted in Fig. 1) are useful, they often involve substantial computational overhead. Deep learning presents a viable, fast alternative to computational analysis, particularly Isogeometric Analysis (IGA), to accelerate the design and analysis processes for bioprosthetic valves. As a major machine learning paradigm, deep learning has demonstrated transformative potential in various fields of science and engineering, bolstered by recent algorithmic advances and the availability of extensive data and high-performance computing resources like GPUs. In this work, we leverage the advancements in deep learning to address the computational challenges in heart valve analysis and to enhance valve design decision-making. Specifically, we introduce a deep-learning-based graphical autoencoder architecture (DLFEA) that predicts analysis output directly from the heart valve geometry. This model acts as a surrogate, expediting analysis by utilizing data from previous simulations. As a proof of concept, we focus on predicting the final deformed closed shape of the heart valve and the coaptation area—a critical measure for surgeons and a key performance metric in both the design and diagnosis of heart valves, which has been the subject of numerous studies. This innovative approach not only speeds up the analysis but also enhances the precision with which these vital metrics are predicted, streamlining the development and assessment of bioprosthetic valves. [1]

2 Background and Related Work

2.1 Finite Element Analysis

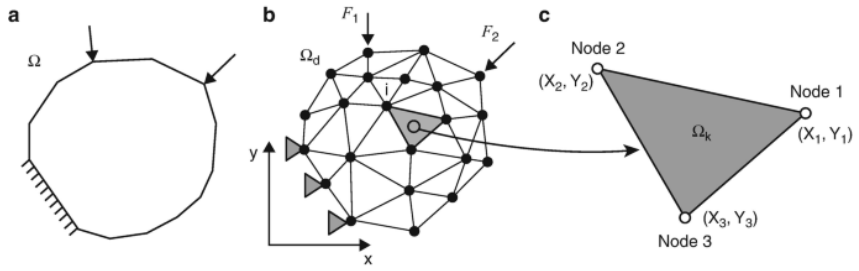


Figure 2: Discretization in FEA

Finite Element Analysis (FEA) is a critical computational technique in engineering for modeling complex structures by subdividing them into smaller, finite elements. Each element's stiffness matrix,

[1] Division of work: Vedant Singh contributed to the development of the GCN architecture, conducted literature review, and compiled and visualized the report. Arushi Jain contributed to the development of the MeshGraphNet architecture, conducted literature review, and also compiled and visualized the report.

$\mathbf{K}_e$, is derived using the formula:

$$\mathbf{K}_e = \int_V \mathbf{B}^T \mathbf{D} \mathbf{B} \, dV$$

where $\mathbf{B}$ is the strain-displacement matrix, $\mathbf{D}$ the material stiffness matrix, and V the volume of integration. These matrices are assembled into the global stiffness matrix $\mathbf{K}$, reflecting node connectivity. The primary FEA equation,

$$\mathbf{K}\mathbf{u} = \mathbf{F}$$

where $\mathbf{F}$ is the external load vector and $\mathbf{u}$ the nodal displacement vector, is solved using methods like Gaussian elimination to determine displacements. After computing displacements, stresses are calculated through the stress-displacement relationship:

$$\boldsymbol{\sigma} = \mathbf{D}\mathbf{B}\mathbf{u}$$

This leverages $\mathbf{D}$ and $\mathbf{B}$ to convert nodal displacements into stresses, offering insights into structural deformation under specific load conditions, critical for assessing structural integrity.

Despite its utility, FEA has limitations such as high computational demands, especially for detailed models, the need for specialized skills, and time-consuming setup and optimization processes. These challenges highlight the potential benefits of incorporating advanced computational methods like deep learning to improve the efficiency and accessibility of FEA in engineering applications.

2.2 Non Uniform Rational B-Spline Surface (NURBS) and IsoGeometric Analysis (IGA)

Non-Uniform Rational B-Splines (NURBS) are a sophisticated mathematical representation employed extensively in computer-aided design (CAD), manufacturing (CAM), and engineering (CAE). They utilize a combination of non-uniform knot vectors, rational coefficients, and B-Spline basis functions to offer unparalleled control over complex geometries with high precision. Mathematically, a NURBS curve is defined as:

$$C(u) = \frac{\sum_{i=0}^{n} N_{i,p}(u) w_i P_i}{\sum_{i=0}^{n} N_{i,p}(u) w_i}$$

where P_i are the control points, w_i are the weights associated with these control points, $N_{i,p}(u)$ are the B-Spline basis functions of degree p, and u is the parameter along the curve. The non-uniformity of the knot vector allows for more flexibility in shape control, making NURBS ideal for modeling everything from simple geometric forms to complex free-form shapes. This capability is crucial in industries such as automotive, aerospace, and animation, where precision in digital modeling translates directly to real-world outcomes.

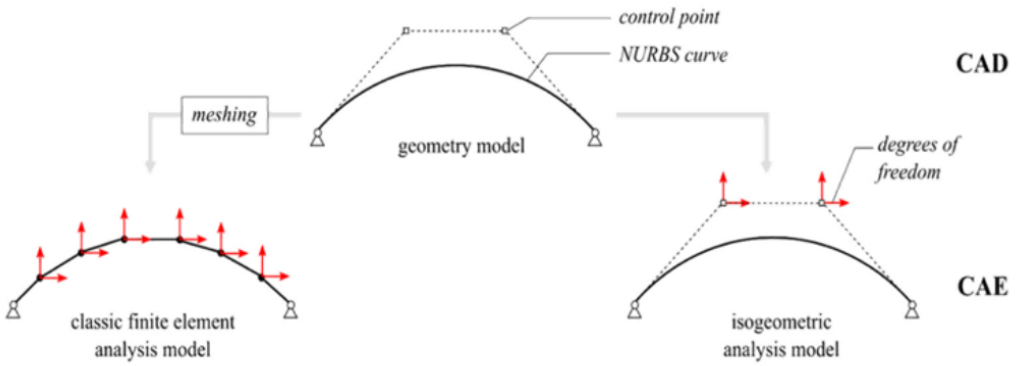


Figure 3: Difference between FEA and IGA discretization

Isogeometric Analysis (IGA) represents an advanced computational approach that integrates the robust, precise capabilities of Non-Uniform Rational B-Splines (NURBS) within the framework of Finite Element Analysis (FEA). Conceptualized to create a seamless bridge between Computer-Aided Design (CAD) and FEA, IGA employs NURBS not only to delineate geometric properties but also as basis functions in the analysis phase. This dual application of NURBS enables a smooth transition from design to analysis without the need for traditional mesh refinement, thereby preserving the geometric integrity and improving both the accuracy and efficiency of simulations. In essence, IGA can be considered as the FEA equivalent using NURBS.

The mathematical foundation of IGA involves utilizing NURBS-defined geometry directly within the integral formulations of FEA, ensuring higher-order continuity across elements. This methodological integration provides IGA with the capability to handle complex simulations more effectively than traditional FEA, particularly in scenarios where precise modeling of stresses and deformations is critical. Despite its advantages, IGA does come with its own set of challenges. Although it eliminates the need for meshing, IGA is still computationally intensive. The computation can take an exponential amount of time, especially as the complexity of the geometry and the precision of the analysis increase. This can limit its practical application in scenarios where computational resources are a constraint.

Consequently, while IGA is increasingly preferred in advanced engineering and scientific research for its continuity and computational stability, its extensive computational demands highlight the need for further optimization and development. These advancements are crucial for enhancing the feasibility of using IGA in a broader range of applications requiring intricate detail and high fidelity in analysis.

2.3 DLFEA : Deep learning in computational mechanics

Deep Learning for Finite Element Analysis (DLFEA) is a powerful framework that seeks to reduce the computational overhead associated with traditional FEA while preserving its ability to provide detailed insights into the mechanical behaviors of complex systems. Surrogate models serve as an efficient approximation technique that can significantly speed up the simulation process.

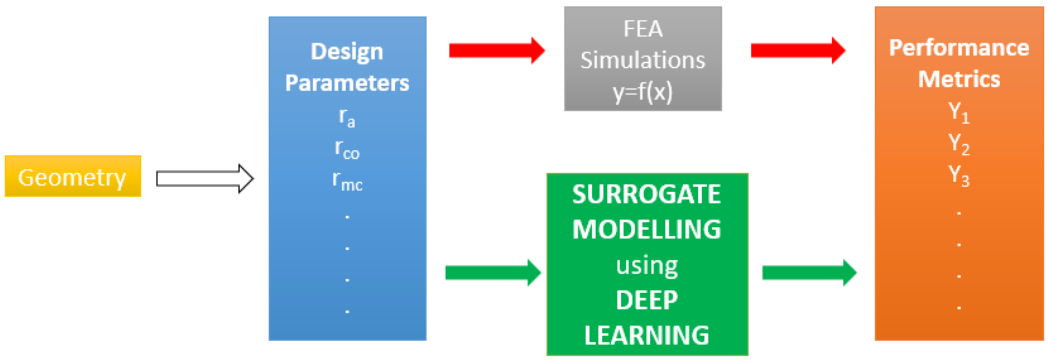


Figure 4: Traditional approach versus DLFEA approach

Surrogate modeling for biomedical applications is explored in multiple studies. In one such study by Liang et al. (Reference [18]), a deep learning model quickly predicts aortic wall stress from thoracic aorta geometries, surpassing the speed of traditional Finite Element Analysis (FEA) methods. The model, trained on 729 distinct aorta shapes derived from patient data using Statistical Shape Modeling (SSM), incorporates an Encoder-Decoder architecture to translate complex geometrical data into stress distributions. Despite its speed and precision, the model's simplification of aortic geometries and exclusion of some material properties highlight areas for improvement.

Another study by Aditya Balu et al. ([2]) utilizes a deep learning approach to streamline the analysis of heart valves. Employing Non-Uniform Rational B-Splines (NURBS) and Isogeometric Analysis (IGA), this study conducts over 90,000 simulations to refine valve design, demonstrating the utility

of a NURBS-aware Convolutional Encoder-Decoder framework for enhancing valve design. The model successfully handles complex interactions and deformations of valve leaflets, though it also faces challenges in fully capturing the intricate details of aortic structures due to simplifications in model geometries and assumptions.

In a different application, a study referenced as [11] applies Convolutional Neural Networks (CNNs) to predict stress fields on cantilever beams, using a dataset of stress images under various loading conditions. The architecture, featuring symmetrical downsampling and upsampling coupled with SEResNet modules, effectively predicts stress distributions, though it struggles with the extension to 3D analysis due to architectural and computational limits.

Each study underscores the potential and current limitations of using deep learning for complex biomedical simulations, pointing towards future directions that might include more accurate geometric representations and enhanced computational techniques.

3 Dataset

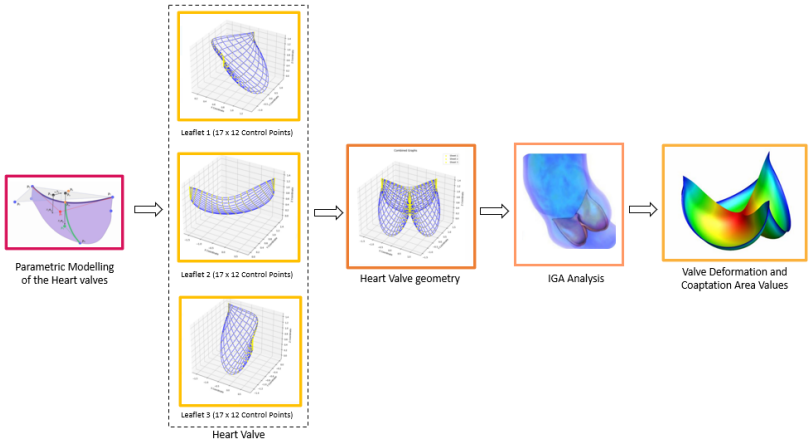


Figure 5: Data Generation

The dataset utilized in our study was sourced from Aditya Balu et al's research presented in [2]. They crafted their dataset for modeling heart valve dynamics by adjusting parameters outlined in the table 1. Each parametric model underwent IGA analysis using Kirchoff-Love shell theory and the Lee-Sacks material model. The analysis included a methodical increase in transvalvular pressure across 100 incremental time steps, maintained under steady conditions with high damping to ensure system stability. This process aimed to derive accurate measurements of coaptation areas and deformations, considering biological boundary conditions such as valve thickness and aortic heart pressure. In total, 18,668 simulations were conducted, and the data distribution is detailed in the table 2.

Parameter	Min. Value	Max. Value	Number of Values
Free Edge Curve Parameter (cm)	0.05	0.45	3
Belly Curve Parameter (cm)	0.2	1.4	7
Height of the Free Edge (cm)	-0.1	0.5	4
Thickness (mm)	0.186	0.427	5
c_0 (kPa)	54.084	81.130	3
c_1 (kPa)	10.628	15.942	3
c_2	30.554	45.826	3

Table 1: Parameters used for generating the reference geometries and material parameters required for training

Dataset	Percentage
Training Dataset	60%
Testing Dataset	20%
Validation Dataset	20%

Table 2: Dataset Distribution

4 Baseline Selection and Implementation

We have adopted the methodology from Aditya et al. [2] as a baseline for our project. Their work, which utilizes a CNN framework, focuses on predicting the coaptation area and deformations of heart valves from data described in the previous section. Their deep learning approach captures the effects of various loads and boundary conditions on valve deformation. Specifically, their model learns to predict interactions between valve leaflets, an aspect typically handled by complex contact algorithms in traditional finite element analysis.

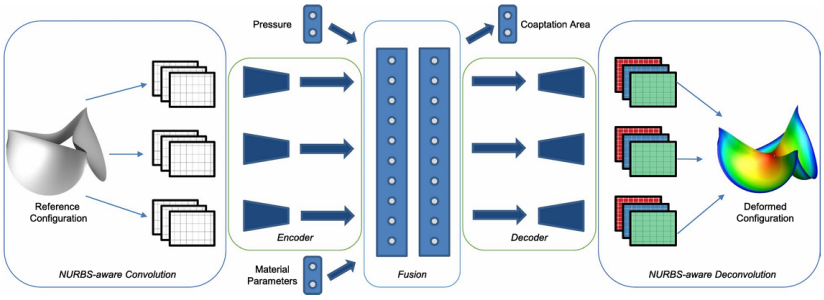


Figure 6: NURBs aware CNN

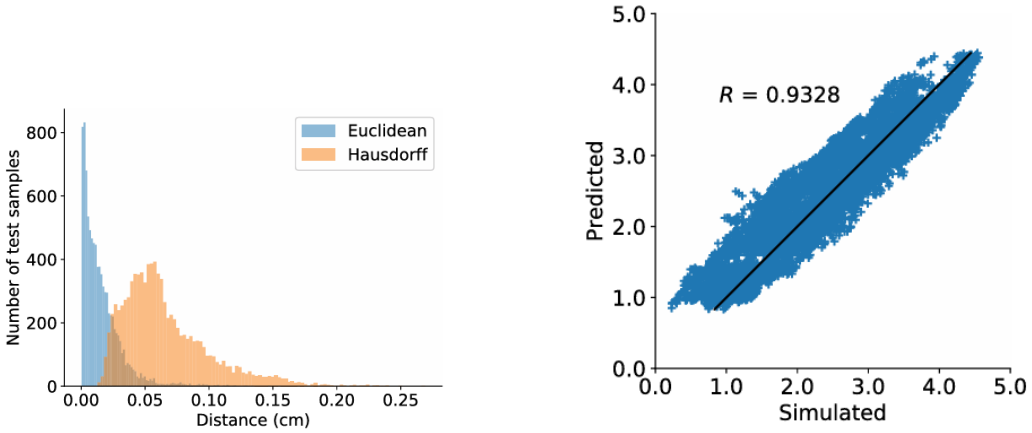
Aditya et al.'s model processes 3D Euclidean space geometries, applying a CNN that learns to predict deformed structures under given loads and conditions. Their data representation includes NURBS-aware convolutions directly on the control mesh, enhancing the ability to maintain geometric detail without information loss. The model architecture combines a convolutional autoencoder with a unique code layer, optimizing the balance between data compression and detail preservation. The innovative approach of using NURBS-aware convolutions allows for precise geometry handling during the learning process, a significant enhancement over traditional methods that lose spatial information. Their model is described in table 3.

The DLFEA framework demonstrates high accuracy in predicting valve deformations. These predictions were evaluated using three metrics: Euclidean distance, Hausdorff distance, and Procrustes matching, discussed in the coming sections. The average Euclidean distance between the predicted and simulated valve deformations was found to be 0.0649 cm, notably small given that the average diameter of a heart valve is approximately 2.3 cm. Figure 7(a) presents histograms of both Euclidean and Hausdorff distances, illustrating the comparative analysis between predicted and actual deformed valve geometries.

Additionally, the DLFEA framework proves effective in predicting key performance metrics of the valve, particularly the coaptation area. The root mean squared error (RMSE) for coaptation area predictions was 0.1167 cm^2, with a high correlation coefficient of 0.9328 with actual measurements, as shown in Figure 7(b).

Common Training Parameters	Architecture Block	Hyperparameters
Batch Size: 512 Optimizer: Adam	Encoder	Convolutional layer 1 (3 filters with size 5x5) Convolutional layer 2 (8 filters with size 3x3) Convolutional layer 3 (16 filters with size 3x3) Convolutional layer 4 (32 filters with size 3x3)
	Repetitions of scalars	Pressure: 20 Thickness: 20
	Code layer	Fully connected layer 1 (64 neurons) Fully connected layer 2 (48 neurons)
	Decoder	Deconvolution start size (8x9x4) Deconvolution layer 1 (8 filters with size 5x5) Deconvolution layer 2 (6 filters with size 3x3) Deconvolution layer 3 (6 filters with size 3x3) Deconvolution layer 4 (4 filters with size 3x3) Deconvolution layer 5 (3 filters with size 1x1)

Table 3: Baseline implemented architecture



(a) Euclidean and Hausdorff distance

(b) RMSE score

Figure 7: Evaluation Metrics

5 Implemented Extension

Despite these advancements, the model confronts several challenges that could limit its effectiveness. Firstly, the model involves increased training complexity and may oversimplify the intricate geometries of aortic structures. Additionally, the dataset used is uniform but non-homogeneous; the spacing between each control point varies, which could affect the model's accuracy.

The model's reliance on Convolutional Neural Networks (CNNs) introduces further limitations:

- Topology Awareness: CNNs struggle to inherently respect or utilize complex topologies, which is critical for accurate anatomical modeling.
- Data Rasterization: The approach converts data into a grid format, which can result in the loss of spatial hierarchies and relationships, crucial for detailed medical imaging.
- Feature Propagation: In CNNs, the propagation of features is limited to immediate grid neighbors. This propagation primarily operates through a fixed mechanism defined by the convolutional filters and pooling layers, potentially overlooking broader contextual information.

These challenges highlight the need for more sophisticated approaches that can handle the spatial and topological complexity of medical datasets more effectively.

Our study leverages Graph Neural Networks (GNNs) to address the challenges of modeling complex geometries in Finite Element Analysis (FEA), particularly for intricate structures such as the aorta and bioprosthetic heart valves (BHV). Traditional methods have explored GNN applications in 2D and simple 3D constructs; however, our research extends these applications to more sophisticated geometries that present unique challenges not fully explored in existing studies.

GNNs compute over graphs, treating entities and their relationships as nodes and edges, respectively. This approach is ideal for the FE domain where mesh irregularities and variable resolutions are crucial for representing complex physical phenomena accurately. Unlike traditional grid-based methods, GNNs can handle non-Euclidean data and operate directly on graph structures, effectively capturing complex relationships and features among nodes. This capability makes them particularly suitable for simulations where maintaining the original spatial relationships and topology of data is vital.

Advantages of GNNs in FEA include:

- Preservation of Geometry and Topology: GNNs maintain the true structure of each node and edge, essential for simulations and deformations that depend on accurate physical properties of the mesh.
- Local Feature Learning: By operating directly on nodes and their immediate neighbors, GNNs learn localized features without transforming input data, thus preserving the geometric and topological integrity of the mesh.
- Iterative Feature Propagation: GNNs iteratively propagate node features across the graph, enabling nodes to accumulate information from an expanding neighborhood. This property is crucial for mesh structures where both local and global information significantly influence simulation accuracy.

Through this research, we aim to bridge the gap in current knowledge by applying GNNs to highly complex geometrical frameworks, thereby enhancing our understanding of their capabilities in capturing the nuanced details inherent in sophisticated structures.

5.1 Model Description

In this study, we adopt an undirected, homogeneous, and static graph architecture, leveraging an Encoder-Decoder framework inspired by the surrogate models previously examined. The architecture is delineated into three primary components:

1. **Encoder:** The Encoder is responsible for generating node and edge embeddings from the graph's initial features.
2. **Processor:** Serving as the core of the architecture, the Processor executes the Graph Neural Network's (GNN) message passing, aggregation, and update mechanisms. It processes the graph, now enriched with features from the Encoder, through a GNN pipeline comprising message generation, aggregation, and node updates for a predetermined number of layers.
3. **Decoder:** The Decoder interprets the node updates post-processing to ultimately yield the stress values for any given model.

   The pipeline is depicted below wherein the 3 dimensional NURBS control points is the inpt to the Neural Network model(DLFEA) and the output is the coaptation area of the heart valve and the individual deformed coordinates of the heart valve. To effectively utilize Graph Neural Networks (GNNs) in Finite Element Analysis (FEA) simulation models, it's necessary to convert the mesh nodes and edges into a graph data structure. The conversion process is detailed in the table 4 provided. In this project, we have implemented two different architectures for GNN's which are described in the following section.

Figure 8: Model Description

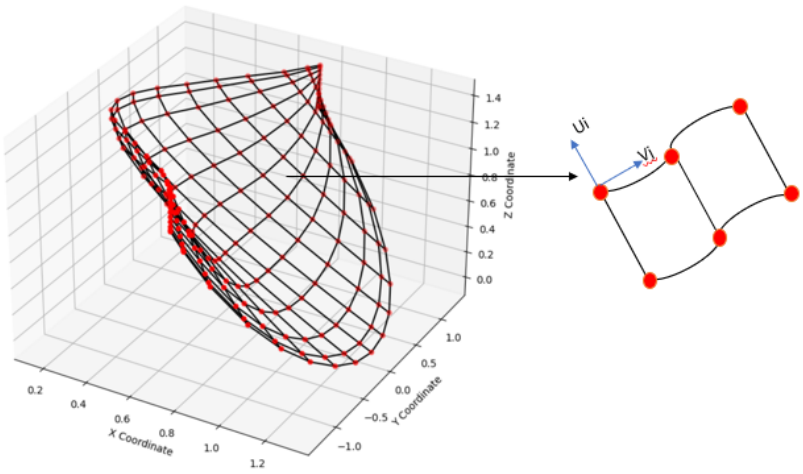


Figure 9: Graph formation from dataset

Features	Details	Count
Node Inputs	Mesh control Points	204
Node Attributes	Spatial coordinates of Control Points of undeformed leaflet	3
Graph Attributes	Aortic Pressure and Material Properties	2
Edge Index	Connectivity of Graph (COO)	379
Edge Attributes	3D positional vector between connecting nodes	379
Node Labels	Spatial coordinates of Control Points of deformed leaflet	204
Graph Labels	Coaptation Area	1

Table 4: Features and Details of the Graph Neural Network Model

5.1.1 Graphical Convolution Network

GCNs leverage the concept of graph convolution, an operation that updates a node's feature by aggregating features from its neighbors and itself. This process effectively captures the local graph structure and node features simultaneously.

Feature Aggregation: Each node in the graph aggregates features from its neighbors. This aggregation may involve simple operations such as summing up or averaging these features. However, it can also include more sophisticated aggregations that apply different weights to different neighbors, allowing for a nuanced integration of the local neighborhood information.

Transformation: The aggregated features are then combined with the node's own features. This combined feature vector typically undergoes a linear transformation, often represented as a multiplication with a weight matrix, followed by a non-linear activation function such

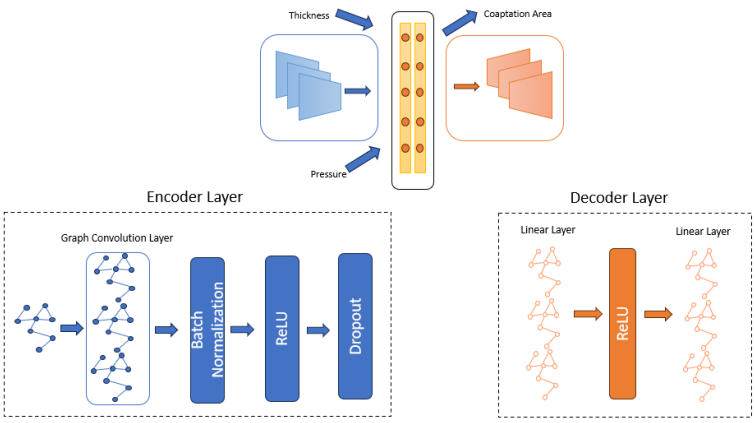


Figure 10: GCN Architecture

as ReLU. This step is crucial for transforming the aggregated information into a useful form for further processing.

Layer Stacking: Multiple graph convolution layers can be stacked on top of each other, enabling the network to learn more complex features at higher "distances". Each layer's output becomes the input to the next layer. This structure allows the model to progressively incorporate information from increasingly larger neighborhoods in the graph, enhancing its ability to capture wide-ranging contextual information.

The architecture used in this project is as defined in the figure 11.

5.1.2 Mesh Graph Nets

MeshgraphNets is a specialized neural network architecture tailored for processing mesh-based data in Finite Element Analysis (FEA). This architecture enhances traditional graph neural network approaches to handle the complexities associated with mesh data, crucial for simulations involving physical phenomena like stress and strain distributions. Key components are:

Feature Aggregation and Transformation: Similar to traditional GNNs, MeshgraphNets aggregate features from neighboring nodes and edges. However, these operations are adapted to respect the geometric properties of the mesh. Aggregated features undergo transformations through layers that may include nonlinear activation functions to capture complex relationships within the data.

Layer Stacking: Multiple layers of graph convolutions allow MeshgraphNets to integrate local and global information, capturing dependencies across larger regions of the mesh. This is essential for accurately predicting phenomena that depend on widespread interactions within the material or structure. The architecture used for the project is described in figure 12

Model	Hidden Layers	Hyperparameters
Encoder	GCNConv Layer(3->8) BatchNorm1d Dropout GCNConv Layer(8->16) BatchNorm1d Dropout GCNConv Layer(16->32) BatchNorm1d Dropout	Batch Normalization(8,16,32) Dropout(0.25)
Graph Decoder	7 x {Linear Layer, Activation Layer}	Activation Layer(ReLU)
Node Decoder	2 x {Linear Layer(32->64), Activation Layer} Dropout 1 x {Linear Layer(64->64), Activation Layer} BatchNorm1d Residual Connection 1 x {Linear Layer(64->64), Activation Layer} Dropout 1 x {Linear Layer(64->32), Activation Layer} 1 x {Linear Layer(32->32), Activation Layer} Residual Connection 1 x {Linear Layer(32->32), Activation Layer} 1 x {Linear Layer(32->16), Activation Layer} Linear Layer(16->16) Residual Connection 1 x {Linear Layer(16->8), Activation Layer} 1 x {Linear Layer(8->8), Activation Layer} Linear Layer(8->3)	

Figure 11: GCN Architecture

Part	Model	Hidden Layers	Hyper Parameters
MeshGraphNets	Node Encoder	Linear(3->10) Activation Layer Linear(10->10) LayerNorm	Activation Layer(ReLU) Normalization(Layer Normalization)
	Edge Encoder	Linear(3->10) Activation Layer Linear(10->10) LayerNorm	Activation Layer(ReLU) Normalization(Layer Normalization)
	Node Decoder	Linear(10->10) Activation	Activation Layer(ReLU)
Processor Layer	Edge Message Passing	Linear(3*in_channels, out_channels) Activation Layer Linear(out_channels, out_channels) LayerNorm	Activation Layer(ReLU) Normalization(Layer Normalization)
	Node Message Passing	Linear2*in_channels, out_channels) Activation Layer Linear(out_channels, out_channels) LayerNorm	Activation Layer(ReLU) Normalization(Layer Normalization)

Figure 12: MeshGraphNet Architecture

Loss Function

Adjusted boundary conditions influence the deformation of the BHV simulations. Although the fixed node theoretically exhibits zero deformation, attaining absolute zero deformation with precision poses numerical challenges. Additionally, when the back-propagation algorithm strives for this ideal, nodes with non-zero deformation experience interference, complicating the learning of overall deformation. To address fixed boundary conditions, we mitigate this issue by incorporating the true deformations into the loss weighting mechanism.

$$\mathrm{L} = \sum_{k \in \mathcal{D}} (u_{\text{pred},k} - u_{\text{true},k})^2 * \frac{1}{|\mathcal{D}|}$$

Results

The metrics used to evaluate the baseline model are chosen to enable a clear comparison between the performances of the models. The metrics used are as follows:

(a) **R2 Score:** An R-Squared value shows how well the model predicts the outcome of the dependent variable. R-Squared values range from 0 to 1. An R-Squared value of 0 means that the model explains or predicts 0% of the relationship between the dependent and independent variables. This metric evaluates the coaptation area predicted by the models. The R2 scores recorded are as follows:

- Baseline model: 0.9328
- Graph Convolution Network: 0.9492
- MeshGraphNets: 0.9589

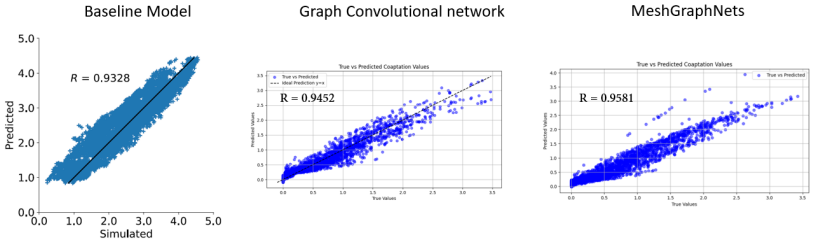


Figure 13: R2 scores of different architectures

(b) **Distance Metrics:** These metrics evaluate the predictions of deformations:

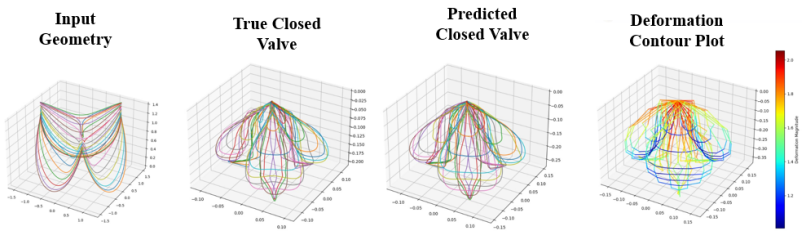
- **Euclidean Distance:** The Euclidean distance between two points in Euclidean space is the length of the line segment between them. It is calculated using the Pythagorean theorem, and is sometimes referred to as the Pythagorean distance.
- **Hausdorff Distance:** This metric measures how far two subsets of a metric space are from each other.
- **Procrustes Distance:** The Procrustes distance is a measure of dissimilarity between shapes based on Procrustes analysis. It identifies the best shape-preserving Euclidean transformation (consisting of rotation, reflection, scaling, and translation) between two shapes.

The table below shows that both the Graph Convolution Network and MeshGraphNets outperform the CNN model in terms of predicting the deformed coordinates. The reason for

Table 5: Comparison of Distance Metrics by Model

Metric	CNN	GCN	MeshGraphNets
Euclidean Distance	0.0173	0.0160	0.0148
Hausdorff Distance	0.0830	0.0772	0.0712
Procrustes Distance	0.0021	0.012	0.0099

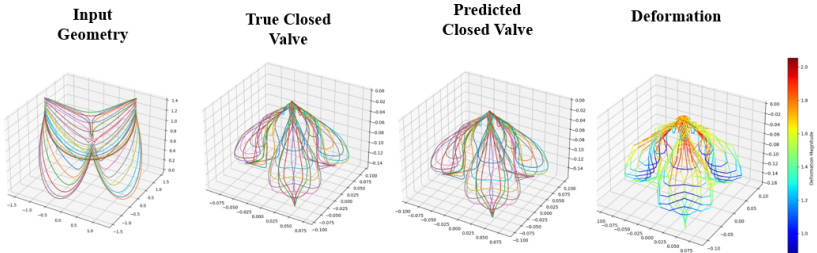
Procrustus Distance not being better than the baseline model is due to the lack of data augmentation. Procrutus Distance depends on data augmentation which was not present in the Graph Convolution Network model and the MeshGraphNets model



Predicted Coaptation Area	2.2782 cm^2
True Coaptation Area	2.5731 cm^2

GCN Results

Figure 14: GCN Results



Predicted Coaptation Area	0.9425 cm^2
True Coaptation Area	0.9621 cm^2

MeshGraphNets Results

Figure 15: MeshGraphNet Results

Discussion

Graph Neural Networks excel in processing 3-dimensional data where spatial features are of utmost importance. MeshGraphNets, a specific type of Graph Neural Network, likely outperform traditional Graph Convolutional Networks (GCNs) in tasks such as Finite Element Analysis (FEA) due to several key factors:

- **Spatial Awareness:** MeshGraphNets are explicitly designed to handle mesh data, which involves complex spatial relationships that are critical for accurate physical predictions. Unlike traditional methods, these networks maintain the integrity of the spatial structure throughout the analysis, ensuring that the relationships between elements are not distorted or lost. This ability is crucial in applications like FEA where the precise arrangement of elements can significantly impact the outcome of simulations.
- **Inter-Node Connectivity:** Traditional Convolutional Neural Networks (CNNs) effectively capture individual node features but often fail to preserve the intricate relationships between nodes. MeshGraphNets, on the other hand, excel in this aspect by incorporating edges and their attributes prominently within their architecture. This approach allows them to define the type and significance of connections between nodes explicitly, facilitating superior capture and integration of relational information which is pivotal for generating more accurate and reliable model predictions.
- **Model Tuning and Outliers:** While MeshGraphNets offer significant improvements in handling spatial data, they are not without challenges. There have been instances where the predictions included outliers, indicating that the model requires fine-tuning. This suggests that certain case-specific scenarios are overlooked during training, necessitating further refinement of the model to improve its accuracy and robustness. Addressing these issues is crucial for deploying these networks in critical applications where precision is paramount.

The input data for MeshGraphNets, consisting of the three-dimensional coordinates of an open heart valve, and the labels, represented by the coordinates of the closed valve, are foundational for accurately modeling valve deformations. Sensitivity to changes in this input data is critical; discrepancies or inaccuracies can significantly impair the model's ability to learn the transformation from an open to a closed state. This limitation would hinder the model's capacity to capture essential geometric details, leading to suboptimal design adjustments. Such inaccuracies ultimately result in incorrect predictions of the coaptation area, which is vital for ensuring the valve's effective operation once implanted.

Integrating this understanding with the predictive capabilities of MeshGraphNets enhances the design and optimization of heart valves. By accurately simulating the deformations of heart valves under various physiological conditions, MeshGraphNets enable engineers and designers to foresee how a valve might behave once implanted. This foresight is invaluable, as it allows for the refinement of valve design to maximize coaptation and overall valve efficiency before actual production and surgical implantation. Consequently, the integration of MeshGraphNets into the valve design process leads to more reliable, efficient, and patient-specific heart valve solutions, significantly enhancing surgical outcomes and patient well-being.

Future Work

The future scope of the project entails the following:

(a) **Stress and Strain Analysis:** Future studies should aim to quantify stress and strain on bio prosthetic heart valves under varied physiological conditions to improve predictions of valve durability and performance.

(b) **Fatigue Life Prediction:** Developing predictive models for the fatigue life of prosthetic valves can aid in optimizing the timing of valve replacements, enhancing patient outcomes.

(c) **Design Parameters for Manufacturing:** Research could extend to evaluate manufacturing design parameters that optimize the structural integrity and cost-efficiency of heart valves.

(d) **Valve Opening Studies:** Incorporating studies on the valve opening mechanics will provide insights into potential functional failures and improve valve designs to mitigate such risks.

(e) **Dynamic Valve Simulation:** Implementing dynamic simulations of heart valve operations under varying cardiac conditions can lead to better-designed prosthetic valves, tailored to patient-specific needs.

Conclusion

We have introduced a deep-learning framework designed to predict the deformation biomechanics of heart valves, which are not readily ascertainable through medical imaging and typically necessitate significant computational expertise and expense to assess. Our methodology has proven effective in learning the complex deformation biomechanics of heart valves across varying geometries, material properties, and boundary conditions. This capability renders the framework immediately beneficial for the parametric design of bioprosthetic heart valves (BHVs). By facilitating a rapid decision support system, this approach can accelerate the development of personalized heart valve designs that offer improved fit and performance, ultimately enhancing patient care.

6 GitHub Repository

This document references a GitHub repository for additional resources and codes used in our study. The codes and datasets used in our study can be accessed at the following GitHub repository: https://github.com/vhsingh/IDL-Final.

References

[1] Gulakala, R., Markert, B., & Stoffel, M. (2023, March 1). Graph Neural Network enhanced Finite Element modelling. *Proceedings in Applied Mathematics & Mechanics*. https://doi.org/10.1002/pamm.202200306

[2] Balu, A., Nallagonda, S., Xu, F., Krishnamurthy, A., Hsu, M., & Sarkar, S. (2019, December 6). A Deep Learning Framework for Design and Analysis of Surgical Bioprosthetic Heart Valves. *Scientific Reports*. https://doi.org/10.1038/s41598-019-54707-9

[3] Jiang, C., & Chen, N. Z. (2023, August 1). Graph Neural Networks (GNNs) based accelerated numerical simulation. *Engineering Applications of Artificial Intelligence*. https://doi.org/10.1016/j.engappai.2023.106370

[4] Jin, Z., Zheng, B., Kim, C., & Gu, G. X. (2023, November 22). Leveraging graph neural networks and neural operator techniques for high-fidelity mesh-based physics simulations. *APL Machine Learning*. https://doi.org/10.1063/5.0167014

[5] Wang, M., Furnary, A. P., Li, H. F., & Grunkemeier, G. L. (2017, September 1). Bioprosthetic Aortic Valve Durability: A Meta-Regression of Published Studies. *The Annals of Thoracic Surgery*. https://doi.org/10.1016/j.athoracsur.2017.02.011

[6] Liang, L., & Sun, B. K. H. (2019, November 8). A Proof of Concept Study of Using Machine-Learning in Artificial Aortic Valve Design: From Leaflet Design to Stress Analysis. *Bioengineering*. https://doi.org/10.3390/bioengineering6040104

[7] Wu, M., Zakerzadeh, R., Kamensky, D., Kiendl, J., Sacks, M. S., & Hsu, M. (2018, June 1). An anisotropic constitutive model for immersogeometric fluid–structure interaction analysis of bioprosthetic heart valves. *Journal of Biomechanics*. https://doi.org/10.1016/j.jbiomech.2018.04.012

[8] Russ, J. B., Li, R. L., Herschman, A. R., Waisman, H., Vedula, V., Kysar, J. W., & Kalfa, D. (2021, November 1). Design optimization of a cardiovascular stent with application to a balloon expandable prosthetic heart valve. *Materials & Design*. https://doi.org/10.1016/j.matdes.2021.109977

[9] Morganti, S., Auricchio, F., Benson, D. J., Gambarin, F. I., Hartmann, S., Hughes, T. J., & Reali, A. (2015, February 1). Patient-specific isogeometric structural analysis of aortic valve closure. *Computer Methods in Applied Mechanics and Engineering*. https://doi.org/10.1016/j.cma.2014.10.010

[10] Liang, L., Liu, M., Martin, C., Elefteriades, J. A., & Sun, W. (2017, April 6). A machine learning approach to investigate the relationship between shape features and numerically predicted risk of ascending aortic aneurysm. *Biomechanics and Modeling in Mechanobiology*. https://doi.org/10.1007/s10237-017-0903-9

[11] Nie, Z., Jiang, H., & Kara, L. B. (2019, September 10). Stress Field Prediction in Cantilevered Structures Using Convolutional Neural Networks. *Journal of Computing and Information Science in Engineering*. https://doi.org/10.1115/1.4044097

[12] Gulakala, R., Markert, B., & Stoffel, M. (2023, March 1). Graph Neural Network enhanced Finite Element modelling. Proceedings in Applied Mathematics & Mechanics. https://doi.org/10.1002/pamm.202200306

[13] Salehi, Y., & Giannacopoulos, D. (2021, September 9). PhysGNN: A Physics-Driven Graph Neural Network Based Model for Predicting Soft Tissue Deformation in Image-Guided Neurosurgery. arXiv.org. https://arxiv.org/abs/2109.04352

[14] Jin, Z., Zheng, B., Kim, C., & Gu, G. X. (2023, November 22). Leveraging graph neural networks and neural operator techniques for high-fidelity mesh-based physics simulations. APL Machine Learning. https://doi.org/10.1063/5.0167014

[15]Jiang, C., & Chen, N. Z. (2023, August 1). Graph Neural Networks (GNNs) based accelerated numerical simulation. Engineering Applications of Artificial Intelligence. https://doi.org/10.1016/j.engappai.2023.106370

[16] Balu, A., Nallagonda, S., Xu, F., Krishnamurthy, A., Hsu, M., & Sarkar, S. (2019, December 6). A Deep Learning Framework for Design and Analysis of Surgical Bioprosthetic Heart Valves. Scientific Reports. https://doi.org/10.1038/s41598-019-54707-9

[17] Gulakala, R., Markert, B., & Stoffel, M. (2023, March 1). Graph Neural Network enhanced Finite Element modelling. Proceedings in Applied Mathematics & Mechanics. https://doi.org/10.1002/pamm.202200306

[18]: Liang L, Liu M, Martin C, Sun W. 2018 A deep learning approach to estimate stress distribution: a fast and accurate surrogate of finite-element analysis. J. R. Soc. Interface 15: 20170844. http://dx.doi.org/10.1098/rsif.2017.0844