Vaasan yliopisto
UNIVERSITY OF VAASA

Sohaib Saeed

Intelligent Stress Estimation in Hydraulic Crane Structures Using Deep Sequential Models

School of Technology and Innovations
Master's Thesis
Sustainable Autonomous Systems

Vaasa 2025

UNIVERSITY OF VAASA			
School of Technology and Innovations			
Author:	Sohaib Saeed		
Title of the thesis:	Intelligent Stress Estimation in Hydraulic Crane Structures Using Deep Sequential Models		
Degree:	Master of Sustainable Autonomous Systems		
Discipline:	School of Technology and Innovations		
Supervisor:	Muhammed Elmusrati		
Evaluator:	Petri Välisuo		
Instructor	Victor Zhidchenko		
Year:	2025	**Pages:**	61

ABSTRACT:

The accurate estimation of the structural stress in the hydraulically crane system is very important to ensure the safety and reliability, as it leads to integration of Real time Control. This study focuses specifically on the data driven approach where the stress prediction is carried out using the deep sequence learning, trained on the simulated operational data from the Flexible Hydraulic Crane Model. The mechanical system under investigation, a Flexible Multibody representation of PATU 655 crane, was subjected to different dynamic simulations where real time sensor signals such as joint torques, joint forces, accelerations were recorded across various payload and motion scenarios. To learn the complex temporal patterns governing the stress evolution, two deep learning architectures were developed and compared. The first was the CNN+BiLSTM Model, where convolutional layers act as feature extractors and bidirectional LSTMs models learned the pattern of temporal dependencies. The second model in comparison was taken as Transformer based sequence model, utilizing the positional encoders and attention mechanisms to capture long range relationships in the multivariate input sequences. The key point to be noted was that both models were trained to predict the stress profile of movement of the crane directly from the sequence of the mechanical and kinematic features. Results showed that both architectures achieved strong performance in the unseen test cases. The CNN+BiLSTM model reached a test MAE of 0.0348 abd MSE of 0.0032, while transformer outperformed it with a significantly lower MAE of 0.0054 and MSE of 0.0000687. This thesis demonstrates that the deep learning models can serve as the intelligent surrogates for physical simulation, enabling fast and accurate stress estimation in real time applications. The findings are clear pathway for the future application of Artificial Intelligence based structural monitoring and intelligent control system in Flexible multibody systems.

KEYWORDS: Deep learning, Transformer Networks, Neural Networks, CNN, LSTM, Hydraulic Crane, Surrogate Modelling, Real-Time stress Estimation, Structural Health, Data Driven Simulation, Sequential Sensor Data, Time Series forecasting.

Contents

Figures

Tables

Abbreviations

CNN	Convolutional Neural Network
BiLSTM	Bidirectional Long Short-Term Memory
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Network
MAP	Mean Squared Error
MAE	Mean Absolute Error
MBD	Multibody Dynamics
FEM	Finite Element Method
FFRF	Floating Frame of Reference
MBS	Multibody System
MPa	Megapascal
Exudyn	Multibody simulation software used in this research
ReLU	Rectified Linear Unit
MHA	Multi-Head Attention
RTF	Real-Time Factor
SDIST	Sensor-measured Distance
SVEL	Sensor-measured Velocity
AACC	Angular Acceleration
SMC	Sliding Mode Control
DNN	Deep Neural Network
MLP	Multi-Layer Perceptron
RMSE	Root Mean Square Error
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning

1 Introduction

In modern mechanical engineering systems, specifically, involving articulated structures such as cranes and complex manipulators, accurate stress prediction plays a key role in ensuring structural reliability, safety, and performance. Traditional approaches for stress estimation such as the well-known Finite Elements Method (FEM), although highly accurate, are computationally expensive and often unsuitable for real-time applications. FEM models require complex meshing, boundary conditions tuning, and iterative solvers which are then subjected to dynamic systems under varying conditions such as load and motion parameters leading to significant time costs. As systems become more complex, and the operating environment becomes dynamic there is a growing demand for fast, adaptive, and intelligent integrated alternatives for physics-based solvers.

1.1 Background and Motivation

Machine learning (ML), particularly deep learning (DL), has shown state-of-the-art potential in an approximation of complex mappings from high dimensional inputs to output spaces. When coupled with simulation environments, capable of producing realistic physical behavior, ML models can be trained to infer hidden or unmeasured physical quantities such as stress even for unseen scenarios. This approach helps to shift the paradigm from physics-based numerical solutions to data-driven learning therefore reducing the computational overhead while maintaining high prediction fidelity.

In this context, the need arises for predictive models, capable of learning from sensor-driven sequential data collected during simulated crane operations. The model developed in this idea would enable rapid estimation of stress responses based on the observable motion-related inputs, supporting the real-time application area for mechanical systems.

1.2 Problem Statement

The dynamic behavior of hydraulically actuated crane systems involves complex stress profiles influenced by multiple interacting factors including payload, angular velocities, actuator movements, and inertial effects. Conventional methods fall short when required to provide immediate insights into stress levels under continuously changing configurations. Therefore, the main problem arises with accurate and efficient estimation of the dynamic stress responses in hydraulic-actuated flexible systems. While having an adequate number of sensors and data storage techniques available for both physical and simulated systems, the utilization of the data remains unsolved. Hence, the goal of the study is to leverage advanced state-of-the-art machine learning architectures, specifically BiLSTM-CNN and Transformer models, to predict stress values based on input time series data reflecting real-time system states. The challenge lies not only in achieving the high prediction accuracy of the stress profiles but also in ensuring the generalization of the models to new, unseen crane motions considering different configurations.

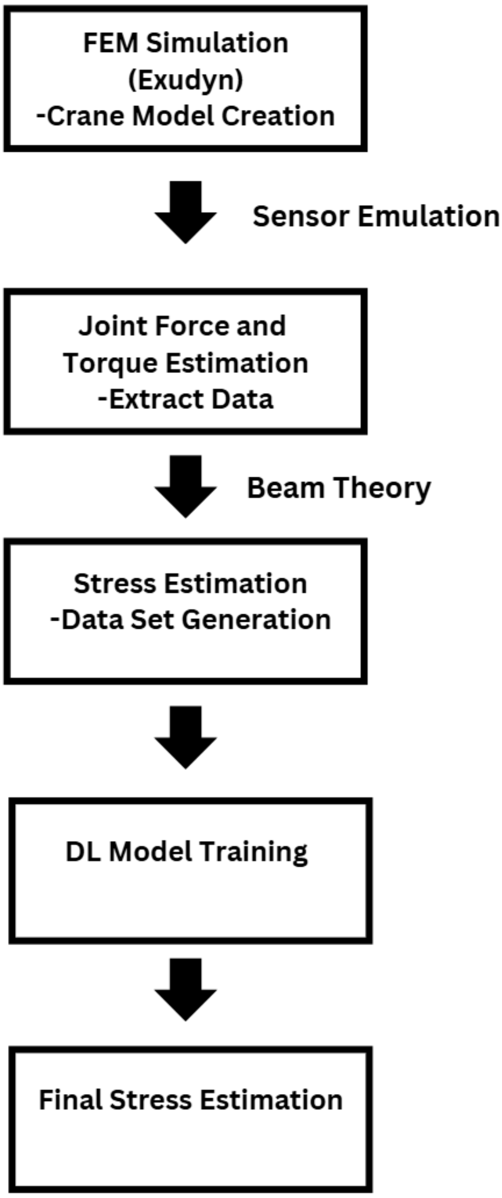

Figure 1: Problem Formulation, (Stress Estimation proposed method)

1.3 Objectives of Study

To address the identified problem, the study sets the key objectives as:

- To construct high-fidelity flexible multibody simulation data of the PATU 655 crane system using a dynamic modeling framework capable of real-time sensor emulation (joint forces, acceleration, torques, etc.).

- To simulate diverse crane motion scenarios to generate sequential datasets capturing time series input features and associated stress responses.
- To design and implement 2 different distinct deep learning architectures, Convolutional Bidirectional Long Short Term Memory model for sequential pattern extraction and long-term memory learning. And a transformer-based model leveraging attention mechanisms for sequence wide learning and interpretability.
- To cross-evaluate the performance of both the models, specifically the ability to generalize to unseen test motions.
- To demonstrate the capability of the trained models to predict stress trajectories from arbitrary crane movements, supporting real-time stress estimation applications.

1.4 Research Questions

Dynamic stress prediction and machine learning haven't been explored that much as of now, taking account of this application, this study aims to investigate several research questions such as, whether deep learning models can accurately estimate stress in dynamic actuated complex heavy machinery by utilizing the sequential data derived from various sensors. The second question focuses on model performance, stating a clear difference between two distinct deep learning algorithms. Lastly, it focuses on the model's ability to handle unseen test scenarios, demonstrating the accuracies of models and generalization when predicting stress across varying crane operations. These all-research questions further pave a bridge towards the future insights of the research leading to real-time adaptivity of systems in the context of different applications, such as intelligent control and monitoring the health of heavy machinery for a given period.

1.5 Contributions of the Work

The introduction of the simulation-based framework to synthetically generate high-quality time series data using sensor-integrated crane motion scenarios can lead to digital twin technology. The idea can work as the testbed for learning-based structural predictions. The novelty of the system lies in the idea that this research can be applied in different heavy machinery systems subject to the changes in the constraints and can lead to the prediction of the stress in different motion trajectories and hence adaptability to different applications such as real-time control etc. Utilization of the Finite Element modeling is considered one of the most authentic and accurate methods in mechanical engineering systems. However, it is costly when the need for the iterations for different motion scenarios is required and therefore it could take much longer to generate the accurate data for different cases. But, considering the approach being implemented in this thesis, the model generalizes well and the need for the FEM is eliminated introducing fast and accurate modeling mechanical systems.

1.6 Scope and Limitations

The scope of this thesis is limited to the simulation-driven investigation using the PATU 665 crane model. All the data considered for training and testing was obtained from the Exudyn-based simulations with real-time sensor emulations. The models are trained on features such as actuator strokes, joint angular velocities, body-framed acceleration, forces, and torques with varying load conditions. The study doesn't extend to the physical hardware validation or implementation in live crane operations. The results are therefore constrained by the fidelity of the simulation environment and the assumptions embedded within the multibody model. Also, the model may require retraining and tuning when applied to other crane, or heavy machinery structures (manipulators) under substantially different load conditions and parameters. Generalization across broader mechanical and robotics systems is not within the current study's scope.

1.7 Structure of Thesis

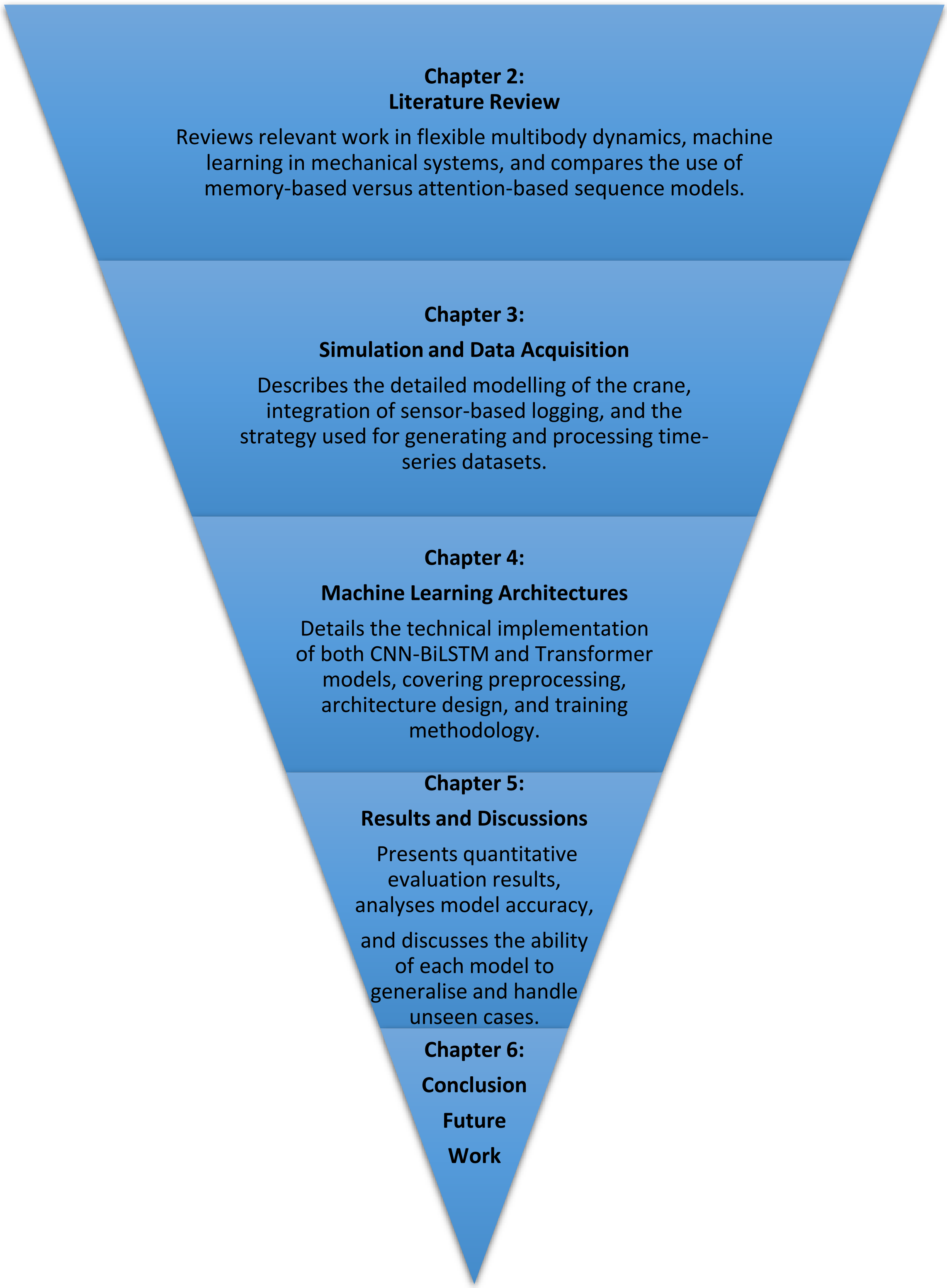

Figure 2: Thesis Layout, Chapters, and Contents Covered

2 Literature Review

Accurate stress prediction in mechanical systems is an important factor for ensuring structural integrity as well as operational safety. This chapter provides insights into different methods and techniques implemented in the context of machine learning and specifically artificial intelligence for ensuring structural integrity. Artificial intelligence and mechanical engineering are two different perspectives of science and a combination of both technologies can pave the way for more easier way to solve complex problems. It has been found in recent research that the intersection of both technologies such as Multibody dynamics and Machine Learning opens new possibilities and solutions.

2.1 Stress Prediction in Mechanical Systems

Heavy machinery is often subjected to high loads and different kinds of varying situations. This creates complex problems that mainly lead to the failure of the machines and machines age faster than they should. Keeping account of this problem, the stress in the machine is a very important factor. Stress prediction in heavy machinery is vital because it can provide information about the critical stress locations and configurations. Accurately predicting the stress of the machines can help in optimizing machine maintenance and therefore this can open new opportunities for also solving physics-based numerical problems. Shabana et al. (214) introduced a method for predicting dynamic stresses using flexible multibody system simulations for tracked upper structure of vehicles. They developed the method considering the nonlinear floating frame of reference formulation intermediating finite element coordinate system with a general continuum mechanics approach to develop accurate kinematic equations for the formulation of strains. The tracked vehicle system specifically the model of the upper structure of bulldozer was developed using Computer Aided Design. To calculate the dynamic stresses, the flexible structure of the tracked vehicle was modeled utilizing Mindlin Shell Finite Elements. The authors considered the idea of dynamic coupling between rigid body movement and elastic body deformations for stress calculations. It was found that this model is particularly relevant for systems like hydraulic cranes, where components experience varying loads and motions. By integrating flexible multibody dynamics, engineers can simulate the real-time behavior of such systems more efficiently than with the traditional FEM alone.

2.2 Flexible Multibody Dynamics and Simulation Tools

Flexible Multibody Dynamics (FMD) is a modeling approach that captures the dynamic behavior of the system with interconnected flexible and rigid behaviors. Several tools in the study can help in the interpretation and integration of the Multibody dynamics of the complex systems. Some of the important tools include MSC Adams, Modelica/Dymola, MATLAB Simulink, Chrono Engine, MBDyn, and Exudyn. In this study, Exudyn has been taken into consideration due to open-source compilation and implementation of automation tools and integration with Machine learning and Deep learning models.
Hashemi et al. (2023) explored the integration of machine learning with multi-body dynamics to enhance control and reduce cognitive load. The study demonstrated that the data drive models can effectively predict how the system behaves, offering the idea of real-time control applications. It was found that the combination of Machine learning and FMD can lead to more adaptive and intelligent control systems for hydraulic cranes leading to improvement in both performance and safety.

2.3 Machine Learning in Prediction and Forecasting Patterns

Artificial Intelligence has emerged as a very powerful tool for structural health monitoring, and forecasting, enabling the prediction of the system responses based on sensor data. Bolandi et al. (2023) developed Neuro-DynaStress, a deep-learning model that was specifically designed to predict dynamic stress distribution in the structural components. The model was able to leverage the finite element simulations to train a neural network capable of real-time stress prediction. The authors also stated that their approach offered a computationally efficient alternative to traditional methods. The approach can also be adapted to real-time stress monitoring applications as the idea was to prevent structural failures and enhance operational safety.

It is often very common to combine more than one machine learning model to integrate it in such a way that it performs better, a study by Zhao et al. (2024) proposed a stacking ensemble learning model for the rapid prediction of fatigue life in crane structures. By integrating multiple regression models, their approach achieved higher accuracy compared to individual models, highlighting the potential of ensemble methods in different mechanical applications. Their method involves the construction of a sample dataset with lifting load and trolley running positions as inputs and fatigue life cycle times as output. The stacking ensemble model combines gradient boosting, ridge regression, extra trees, and linear models to predict fatigue life accurately.

2.4 Deep Learning Architectures for Time Series Sequential Data

Time series data, characterized by sequential observations over time, are prevalent in various domains, such as robotics and mechanical engineering. In the context of hydraulic crane systems, time series data emerge from sensor readings capturing dynamic behaviors such as joint angles, velocities, and applied forces. Analyzing and modeling such data require architectures capable of capturing temporal dependencies and complex patterns.

Deep learning has revolutionized time series analysis by providing models that can automatically learn hierarchical representations for raw data. Unlike traditional methods, deep learning models do not necessitate manual feature engineering, making them particularly suitable for complex and high-dimensional time series data.

2.4.1 Convolutional Neural Networks

Convolutional Neural networks (CNNs) are a class of deep learning models that utilize convolutional layers to extract features from input data. While CNNs gained prominence in image processing tasks, their applicability extends to one-dimensional time series data, where they can effectively capture local temporal patterns. A CNN model typically consists of convolutional layers that apply filters (kernels) that slide over the input data to produce feature maps. In time series, one-dimensional convolution is used to capture temporal patterns. They also consist of Activation functions, pooling layers, and fully connected layers. Where the activation functions are used to introduce the nonlinearities into the model. The rectified linear unit also known as ReLU is most of the time used as an activation function in the CNNs. Pooling layers reduce the dimensionality of feature maps, aiding in generalization and reducing the computational load. Fully connected layers serve as the final layers that map extracted features to the output. It has been found that CNN can be a big player in the game of machine learning because it allows parallel computation therefore leading to a faster training time. Hybridization is also a very important factor that is facilitated by CNN models. This means CNN can be combined with other machine learning/deep learning to improve the feature extractions and better training. An example of the hybrid model is presented by Kim et al. (2025), who developed a CNN-BiLSTM framework for financial systemic risk prediction. Their model utilized CNN layers to capture temporal dependencies. This architecture demonstrated superior performance in predicting systemic risk, highlighting the efficacy of combining CNNs with recurrent layers.

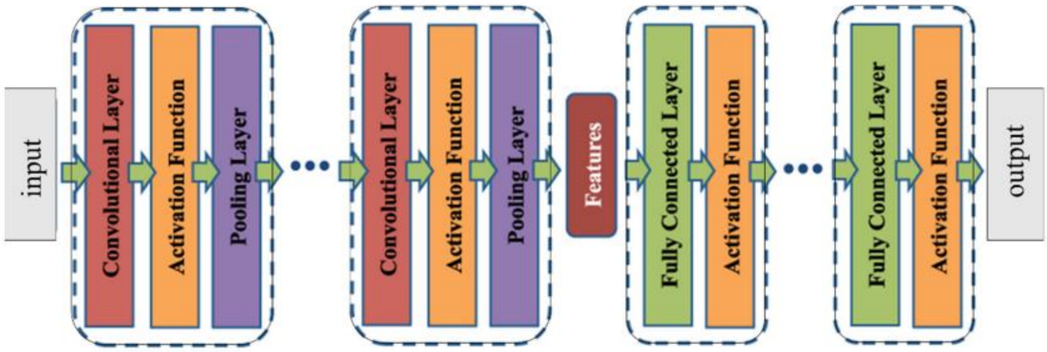

Figure 3: General Architecture of Convolutional Neural Network (Bhatnagar & Gill, 2020)

2.4.2 Long Short-Term Memory Models

Long short-term memory (LSTM) networks are a specialized form of recurrent neural networks (RNNs) designed to model sequential data by capturing long-range dependencies. It has been a known problem in the RNNs about the vanishing or exploding gradients, LSTMs utilize a unique architecture comprising memory cells and gating mechanisms. LSTMs consist of input gates and forget gates, that regulate the flow of information. This enables the model to retain the important information and forget the unnecessary information.

Considering the mechanical systems, LSTM has a long history of solving complex problems related to the prediction of dynamic behaviors based on sequential sensor data. For instance, a study by Zhang et al. (2020) introduced a physics-informed multi-LSTM network for metamodeling nonlinear structural systems. The work included the physical laws incorporated into the loss functions, and the model was able to capture accurately latent system nonlinearity even with limited training data., demonstrating the potential of LSTMs in modeling complex mechanical behaviors.

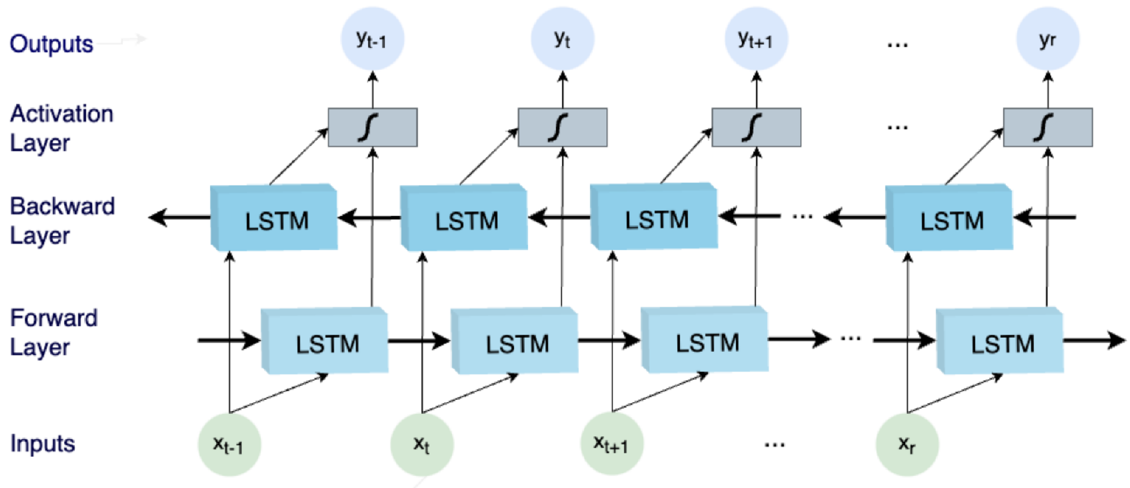

Figure 4: General Architecture of BiLSTM (Pavlatos et al., 2023)

Hybrid modeling is a very important way to combine the computational efficiency of two different algorithms. Another study by Liu et al. (2024) proposed a hybrid model combining LSTM and transformer architectures to enhance time series prediction accuracy. Their approach demonstrated improved performance in capturing both short-term fluctuations and long-term trends in dynamic systems, highlighting the efficacy of integrating LSTM networks in stress prediction tasks.

2.4.3 Attention Transformers-based Models

Transformer models, introduced by Vaswani et al. in 2017, have completely changed sequence modeling by introducing self-attention mechanisms to capture global dependencies within data sequences. In comparison to RNNs, Transformers can process input data in parallel, which enhances computational efficiency and enables the modeling of long-ranged interactions without considering the limitations of sequential processing.

This is to be noted that in time series forecasting, Transformer architectures have demonstrated superior performance in capturing complex temporal patterns. For instance, a study by Liu et al. (2024) developed a time series prediction model based on the fusion of Transformer and LSTM algorithms. They stated that their model effectively captured both local and global temporal dependencies, overcoming traditional approaches in numerous forecasting tasks.

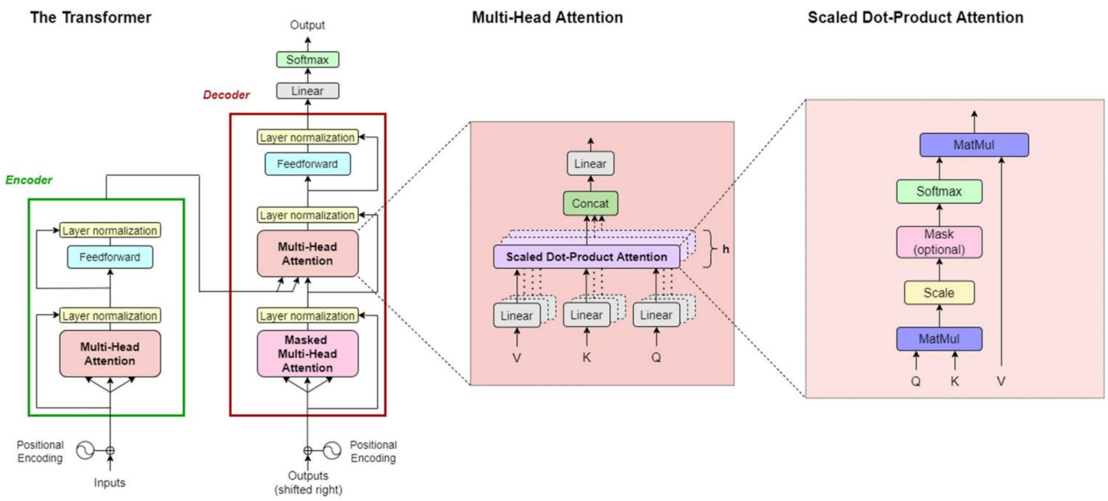

Figure 5: The Transformer architecture and the attention mechanisms it uses in detail. (Left) The Transformer with one encoder-decoder stack. (Soydaner, 2022)

Also, a study by Wang et al. (2024) introduced a customized dual-Transformer framework for remaining useful life (RUL) prediction of mechanical systems. By considering the degraded state of the system, their model achieved accurate predictions, showcasing the potential of Transformer-based architectures in modeling complex mechanical systems. To determine the uncertain change time, a contrastive Transformer network is designed to learn the representation discrepancy of the operation state. In addition, a versatile

Transformer network is developed to learn multi-term dependencies for RUL prediction beyond the state change point. Finally, a self-built IR experiment and IMS-bearing datasets are used to validate the effectiveness and superiority of the proposed method.

2.5 Digital Twins and Real-Time Stress Estimation

In mechanical systems, digital twin technology has become a transforming approach to real-time monitoring and predictive analysis of complex systems. A digital twin is a virtual representation of a physical system that uses real-time sensor data, incorporating computational models and integrating machine learning algorithms to simulate, predict, and optimize performance. These methods enable proactive maintenance, operational efficiency, and safety.

Digital twins in the case of hydraulic cranes prove to be an advantage in terms of continuous health insights regarding the structure and the stress distribution generated on it. For example, Moi et al. (2020) developed a digital twin for a knuckle boom crane by using real-time strain gauge data and finite element analysis to monitor stress levels during operation. This approach allowed them to accurately estimate stress at critical points and improve the reliability and safety of the crane.

Pan et al. (2024) also proposed a digital twin method for real-time stress prediction using surrogate modeling. Using numerical simulations and machine learning algorithms, their model was able to predict structural stress in real-time and in three dimensions under different conditions. The latter point suggests that the potential of digital twins is in real-time structural health monitoring and decision-making.

Also, Digital twins are integrated with advanced computational frameworks to predict such failures. For instance, researchers in Nature Communications (2025) discuss the DeepONet framework which combines deep learning and operator learning to predict complex physical phenomena in real-time. The presented approach illustrates that employing digital twins as a viable approach to stress estimation in dynamically changing systems is indeed feasible.

Hence, these studies demonstrate the effectiveness of digital twin technology in performing real-time stress estimation, predictive maintenance, and operational optimization in hydraulic crane systems. Digital twins are a critical tool to improve the safety, reliability, and efficiency of heavy machinery operations by leveraging real-time data and advanced modeling techniques.

2.6 Summary and Research Gaps

The review of existing research reveals that significant progress has been made in integrating simulation-based approaches with artificial intelligence for mechanical stress prediction. However, several limitations still restrict the practical application of these methods in dynamic, real-time systems such as hydraulically actuated cranes.

One of the primary issues in the literature is the over-reliance on traditional physics-based simulations, particularly Finite Element Methods (FEM), which, while accurate, are computationally intensive. This makes them unsuitable for real-time decision-making or rapid feedback applications. Flexible multibody dynamics have emerged as a more efficient alternative, yet it still requires re-simulation for each new loading or motion condition, creating a bottleneck in scalability and adaptability.

Another limitation lies in the use of machine learning for stress estimation, where many studies train models on highly constrained or static datasets. These models often focus on fatigue life prediction or predefined load scenarios and do not capture the full variability and complexity of crane operations in real-world environments. As a result, their ability to generalize to unseen conditions is limited, and the models fail to adapt when operational parameters change.

The use of deep learning methods such as CNNs, LSTMs, and Transformers introduces new possibilities for learning from sequential sensor data. However, most research in this space is still at an early stage for mechanical applications. Existing deep models are either applied to different domains like finance or robotics or are not optimized for the temporal and structural patterns relevant to stress behavior. They often lack comprehensive integration with high-resolution actuation data such as joint forces, torques, and accelerations over time. Moreover, deep learning models in the current literature typically operate as black boxes, offering limited interpretability and control feedback mechanisms.

Considering these challenges, this research introduces a methodology that specifically tackles these unresolved issues. It leverages deep sequential models trained on multivariate time-series data generated from the detailed simulation of a flexible hydraulic crane model. The simulation replicates real-world operating conditions with varying payloads and actuator inputs, ensuring that the models are trained on complex, realistic data.

Instead of relying on single-point predictions or static models, the proposed framework uses temporal learning networks—specifically CNN-BiLSTM and Transformer architectures—that are capable of learning long-range dependencies in stress evolution. These architectures are designed to generalize to unseen motion patterns, making them suitable for predictive control in variable conditions.

The study proposes a stress-aware predictive surrogate system that can be integrated into a digital twin environment for hydraulic cranes. This system can estimate stress at critical joints in near real-time using live actuation data, offering a faster and more adaptive alternative to conventional methods.

3 Simulation Framework and Data Acquisition

This chapter presents the overall simulation framework that was used to model and analyze a hydraulically actuated crane system. The main purpose of the simulation is to generate realistic data that represents the dynamic behavior of the crane under different loading and control conditions. This data is later used for stress estimation and deep learning applications.

The simulation was developed using Exudyn, which is an open-source Python-based multibody dynamics software. Exudyn allows the modeling of both rigid and flexible components, making it well-suited for simulating complex mechanical systems like cranes. Its modular structure supports custom force elements, flexible body dynamics (through reduced-order models), and user-defined control functions.

Figure 6: PATU 655 Crane under observation for proposed approach (Kotta, 2021)

The Patu 655 crane was modeled in Exudyn using a modular approach, where each mechanical component was defined as either a rigid body or a flexible beam, depending on its role and deformation characteristics. The most critical parts, such as the Lift Boom and Tilt Boom, were modeled using Flexible Body FFRF (Floating Frame of Reference Formulation) to capture their bending and dynamic deflection under load.

3.1 Overview of the Simulation Environment

The investigation relies on dynamic simulations that combine multibody system theory and flexible body mechanics. These simulations provide a controlled environment where forces, displacements, and stress development can be observed with high temporal resolution. The simulation includes all key components of the crane system:

- Pillar (Base support)
- Lift Boom
- Tilt Boom
- Extension Arm
- Hydraulic Actuators (Lift and Tilt)
- Joint Constraints (Revolute and Generic Joints)
- Sensors (Joint forces, torques, accelerations)

The primary goal of the environment was to simulate the realistic actuation behavior of the crane, track internal joint reactions, and compute beam stress values using theoretical and sensor-based approaches.

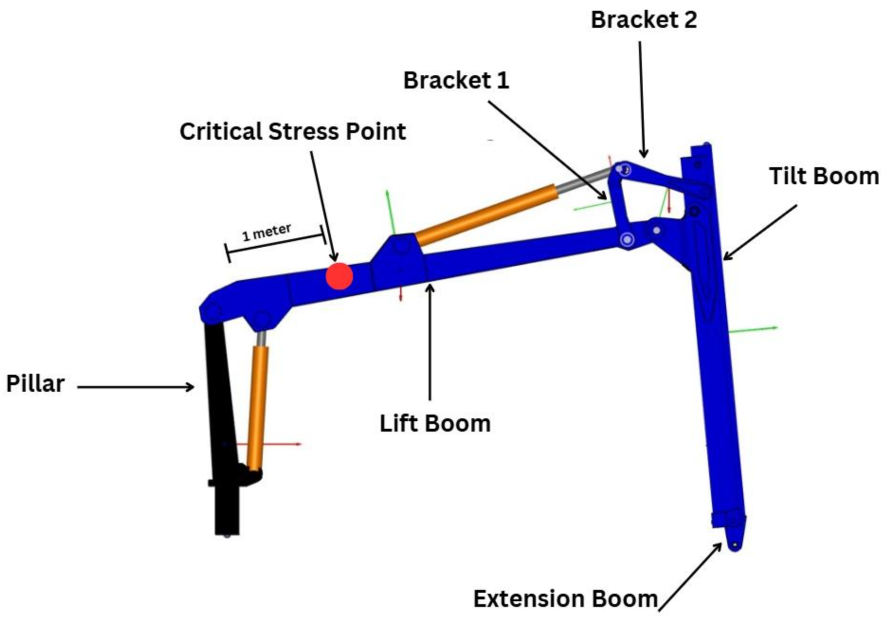

Figure 7: Simulation Model of PATU crane in Exudyn (Khadim et al., 2022), (Kostiainen, 2024)

The simulation is based on the principles of multibody dynamics, which treat the crane system as a combination of rigid and flexible components connected by joints. Each part is defined with its own mass and inertia properties, and the system obeys Newton–Euler equations for motion and equilibrium. The simulation framework includes both translational and rotational motions, as well as deformations in the flexible members.

The key idea behind the simulation model is to solve the system of differential-algebraic equations (DAEs), which can be written in general form as:

$$M(q)\ddot{q} + C(q,\dot{q}) + K(q) = Fext(t)$$

Where, q is the vector of generalized coordinates (positions and angles), $M(q)$ is the mass matrix, $\ddot{q}$ is the acceleration vector, $C(q,\dot{q})$ represents damping and Coriolis forces, $K(q)$ is the stiffness-related internal force vector, $Fext(t)$ is the external force vector (including actuator and payload forces).

For the flexible components like the lift boom and tilt boom, the model uses a reduced-order representation based on modal reduction from Finite Element Analysis (FEA). This approach simplifies the system while still capturing its flexible behavior. The flexible body theory uses the floating frame of reference formulation (FFRF), which separates rigid body motion from local deformations.

Table 1 below shows the main components of the crane and their theoretical modeling approach:

Table 1: System Components and Theoretical Representations

Component	**Type**	**Model Approach**	**Governing Theory**
Lift Boom	Flexible	FFRF + Modal Reduction	Euler–Bernoulli Beam Theory + Modal Superposition
Tilt Boom	Flexible	FFRF + Modal Reduction	Euler–Bernoulli Beam Theory
Pillar	Rigid	Classical Rigid Body	Newton–Euler Dynamics
Hydraulic Cylinder	Actuator	Force Input based on Pressure-Displacement	Fluid Power and Control Theory
Joints	Constraints	Revolute/Fixed Joint Constraint Equations	Constraint-based Motion Coupling
Payload	Point Mass	Added Mass at Boom End	Newton's Second Law

These components are assembled using constraint equations to define joint connections. Each joint adds kinematic constraints which reduce the system's degrees of freedom and couple the motion of adjacent bodies. The reaction forces at these joints are crucial in computing internal stresses.

3.2 Crane System Modelling in Exudyn

The crane used in this simulation consists of multiple interacting parts. It includes a base pillar, a lift boom, a tilt boom, and two hydraulic cylinders. These parts are connected through joints and actuated using hydraulic pressure. The motion of the crane is driven by external control signals (e.g., joystick angles and valve pressure) that control the cylinder extensions.

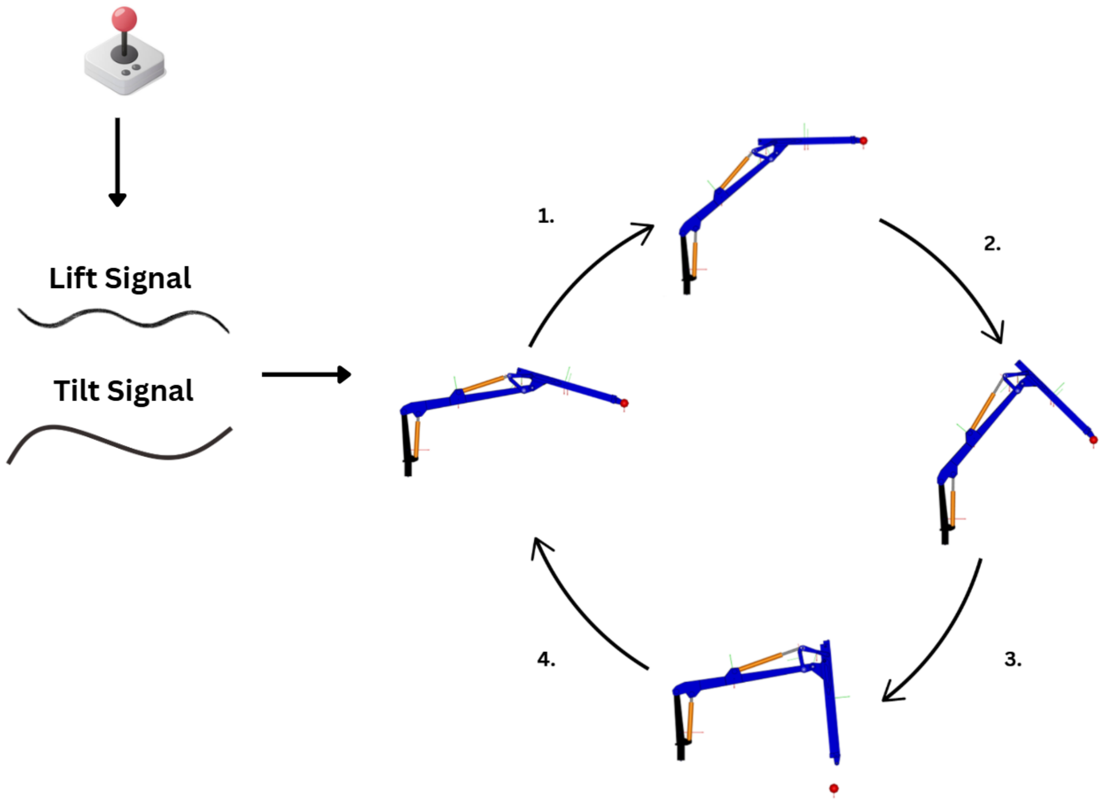

Figure 8: Simulation of Crane in Different Cycles (Different Lift and Tilt Boom Signals Provided)

To represent the mechanical behavior of the crane, a multibody dynamics formulation is employed. This treats each part as either a rigid or flexible body. The flexible parts are modeled using the Euler–Bernoulli beam theory, which assumes that plane cross-sections remain plane and perpendicular to the neutral axis during bending. For axial and bending stress analysis, the standard beam theory equations are used:

- **Axial stress**:

$$\sigma_{axial} = \frac{F}{A}$$

- **Bending stress**:

$$\sigma_{bending} = \frac{M.y}{I}$$

Where F is the axial force, A is the cross-sectional area, M is the bending moment, y is the distance from the neutral axis, and I is the second moment of area.

The total stress at a point can be estimated by the superposition of axial and bending stress:

$$\sigma_{total} = \sigma_{axial} \pm \sigma_{bending}$$

In this study, the stress is computed at a point of 1 metre along the lift boom (to evaluate stress away from the joint, which was found to be having highest stress point in the FEM simulations).

This point was selected to understand both local and distributed stress patterns.

The hydraulic actuation is modeled as an input force applied between two points. The actuator force in general is defined by:

$$F_{hydraulic} = p \cdot A$$

Where p is the input pressure, and A is the effective piston area. This force drives the motion of the connected bodies and contributes to internal joint forces. As the actuator extends or retracts, the corresponding parts of the crane rotate about their joints.

Joint reaction forces and torques are computed as part of the dynamic solution. These values are then used to estimate internal stresses using classical mechanics, assuming a beam section model for the flexible components. The joint forces also help identify stress-critical scenarios that can be used for further machine learning-based stress estimation models. Table 2 outlines the input and output signals used in the simulation framework:

Table 2: Theoretical Inputs and Outputs in the Crane Model

Category	**Variable**	**Description**	**Units**
Input	Joystick Angle	Lift and tilt direction control	radians
Input	Pump Pressure	Pressure supplied by the pump to the system	Pascals (Pa)
Output	Joint Force	Reaction forces at boom connections	Newtons (N)
Output	Joint Torque	Rotational constraint torque	Newton-meters

Output	Actuator Displacement	Piston movement under pressure	Millimetres (mm)
Output	Bending Stress	Estimated using beam theory	Megapascals (MPa)

3.3 Sensor Emulation and Data Logging

In any physical mechanical system, especially in mobile cranes, sensors are essential for detecting forces, torques, displacements, velocities, and system stresses. In this simulated environment, these sensors are not physical devices but are mathematically modeled and emulated using governing physics-based equations. The process of sensor emulation ensures that simulation output can replicate what real sensors would measure during operation.

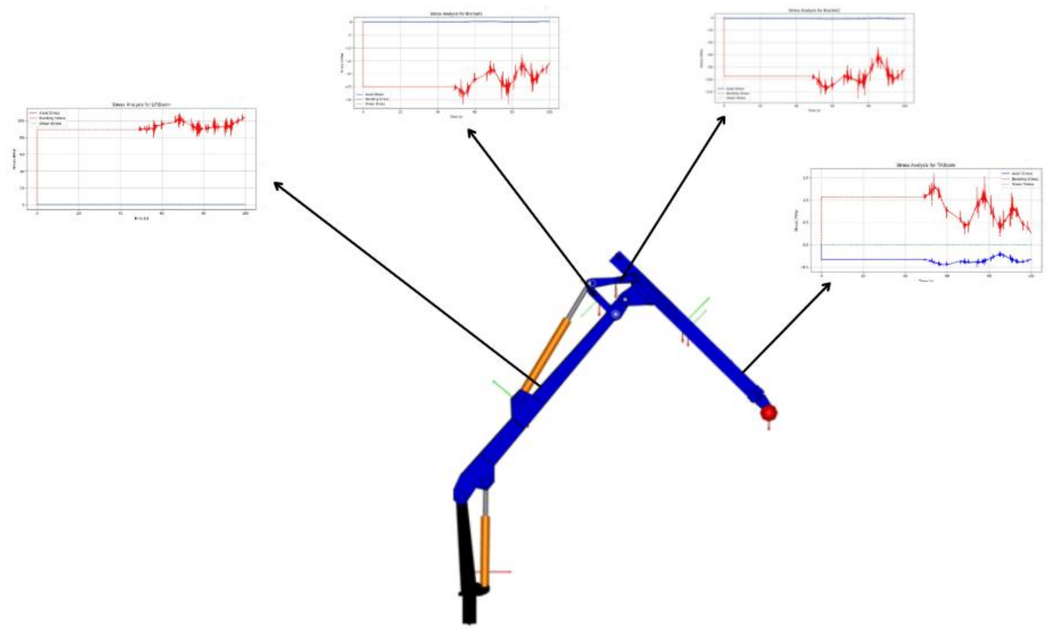

Figure 9: Computation of Stress utilizing beam theory and data from sensors (Force, torque joint sensor emulation)

These virtual sensors allow for detailed evaluation of internal forces and mechanical response, and they play a key role in developing intelligent control strategies and stress estimation algorithms.

3.3.1 Reaction Force and Torque Detection at Joints

Joints in a crane structure, such as revolute or prismatic joints, impose motion constraints between connected bodies. As a result, reaction forces and torques are generated to enforce these constraints. These reactions are not added manually; instead, they arise naturally through the equations of motion based on multibody dynamics.

From theory, the system of equations for a constrained multibody system includes the term:

$$G(q)^T\lambda$$

Where $G(q)$ is the Jacobian matrix of constraint equations, a mapping from configuration space to constraint space. λ is the vector of Lagrange multipliers, representing the internal forces (or torques) needed to satisfy each constraint.

When solving the full dynamic system, the simulation computes λ at each time step. These values are then used to recover the reaction forces acting at specific joints. These theoretical forces and torques are recorded as time-series data, allowing downstream analysis such as stress computation or fatigue estimation.

For instance, if a revolute joint connects the lift boom to the base, the reaction force vector $\vec{F}_{joint}(t)$ corresponds to the internal constraint force ensuring rotational movement without translation. These are critical for computing beam loading at adjacent locations.

3.3.2 Actuator Displacement and Velocity Emulation

Hydraulic actuators connect different segments of the crane and apply movement based on internal fluid pressure. In the simulation, we do not model the fluid mechanics in full detail; instead, we emulate actuator behavior using theoretical distance and force relations.

The actuator displacement is the time-varying distance between the actuator’s base and rod ends, calculated as:

$$\Delta x(t) = \| \vec{r}_{rod}(t) - \vec{r}_{base}(t)) \|$$

Where $\vec{r}_{rod}(t)$ and $\vec{r}_{base}(t)$ are the spatial positions of the actuator connection points at time t. This emulates a position sensor or LVDT in an actual hydraulic cylinder.

From displacement, velocity is derived through numerical differentiation:

$$v(t) = \frac{d}{dt}\Delta x(t)$$

In this study, a double-acting hydraulic cylinder is used to drive the structural motion. In such a configuration, hydraulic pressure is applied to both the piston and rod chambers, enabling bidirectional movement of the actuator. The resulting axial force transmitted to the structure is calculated based on the differential pressure across the piston chambers. The hydraulic force output is computed as:

$$F_{act}(t) = A_p \cdot p_A(t) - A_r . p_B(t)$$

Where A_p is the full piston area on the piston side, A_r is the reduced area on the rod side (accounting for the piston rod), $p_A(t)$ and $p_B(t)$ represent the time-varying internal pressures in the piston and rod chambers respectively, and $F_{act}(t)$ is the net axial force exerted by the actuator. This force is then utilized in the global force vector within the system's equations of motion, contributing to the dynamic response of the multibody structure.

3.3.3 Beam Stress Monitoring at Critical Locations

One of the key outcomes of this simulation is to estimate stress levels in the crane's structural members. Especially in flexible beams such as the lift boom, both axial force and bending moments contribute to internal stress.

At any cross-section along the beam (especially near joints or actuator connections), the total longitudinal stress is calculated using the classical beam theory:

$$\sigma_{total}(t) = \frac{F_{axial}(t)}{A} \pm \frac{M_z(t). y}{I_z}$$

Where $F_{axial}(t)$is the axial reaction force at that section, $M_z(t)$ is the bending moment at that location (often derived from joint torques), A is the cross-sectional area of the beam, y is the distance from the neutral axis, I_z is the second moment of inertia of the section. The term $\frac{F}{A}$ captures axial tension/compression, while $\frac{M.y}{I_z}$ reflects tensile or compressive stress due to bending. This stress is logged at specific locations, for instance, in the considered case, 1 meter from the lift boom joint, where bending effects are known to peak under load.

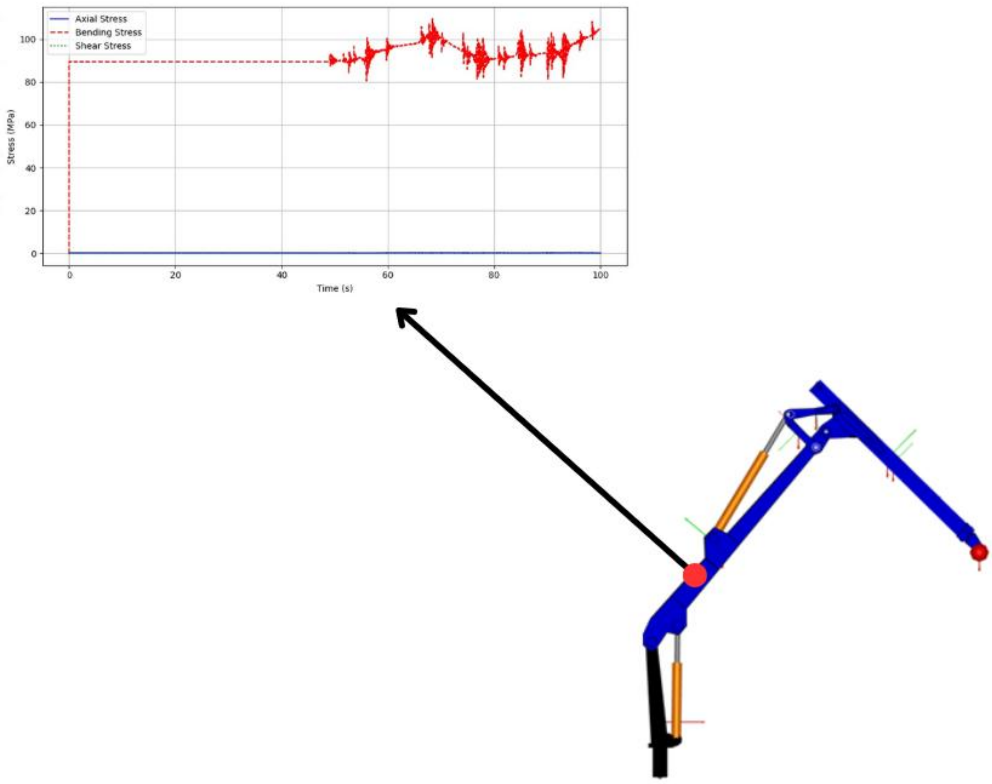

Figure 10: Computation of Stress at a critical location (shown with a red circle on the lift boom of the crane)

3.4 Simulation Parameters

To simulate realistic structural and dynamic behavior, the system must be governed by mathematically consistent and physically accurate parameters. The crane model is based on the generalized multibody system equations, which describe both rigid and flexible body dynamics, joint constraints, and externally applied forces.

3.4.1 Governing Multibody Dynamics Equations

The dynamic behavior of the crane system is modeled using a second-order differential-algebraic system of equations:

$$M(q)\ddot{q} + C(q,\dot{q}) + K(q) = F_{ext}(t) + G(q)^T\lambda$$

Each term represents a specific physical phenomenon, Mass matrix M(q) represents the inertia of all bodies, and includes contributions from rigid body masses and flexible body modal masses. Damping term $C(q,\dot{q})$ accounts for structural damping, fluid resistance, and energy dissipation. Stiffness forces $K(q)$ arise only in flexible components, describing elastic resistance based on Young's modulus and geometry. External force vector $F_{ext}(t)$ Includes all applied forces such as gravity, actuator forces, and payload weight (varying). Constraint Jacobian $G(q)$ and Lagrange multipliers λ ensure the kinematic constraints of joints are maintained throughout the motion.

The numerical solver integrates these equations over time, updating all degrees of freedom and constraint forces. The dynamic response — displacements, joint forces, and internal stresses — emerges from solving this coupled system.

A. Mass Matrix – M(q)

Rigid Body's mass and inertia are represented by 6×6 matrices. Translational and rotational inertia are placed along the diagonal. Flexible Body's mass matrix is reduced using modal projection. For a beam, it is:

$$M_{flex} = \int_0^L \rho A \Phi^T(x)\Phi(x)\,dx$$

Where $\Phi(x)$ are the shape functions or mode shapes derived from FEM.

B. Stiffness and Damping – $\boldsymbol{K(q)}, \boldsymbol{C(q,\dot{q})}$

The stiffness matrix captures the elastic resistance:

$$K = \int_0^l EI\left(\frac{d^2\Phi(x)}{dx^2}\right)^T\left(\frac{d^2\Phi(x)}{dx^2}\right)dx$$

Damping is modeled using Rayleigh damping, a combination of mass and stiffness:

$$C = \alpha M + \beta K$$

Values of α and β are chosen based on expected damping behavior (typically calibrated with experimental data).

C. Constraint Jacobian – G(q)

Each joint imposes constraints $\Phi(q) = 0$, reducing the degrees of freedom. For a revolute joint, five constraints are enforced — 3 translational and 2 rotational — so:

$$G(q) = \frac{\partial \Phi(q)}{\partial q}$$

This Jacobian is used to enforce constraints using Lagrange multipliers, which simultaneously compute the required joint forces.

D. External Forces – $\boldsymbol{F_{ext}(t)}$

The external force term combines:

- **Gravitational force**:

$$F_g = m \cdot g$$

- **Hydraulic actuator force**:

$$F_{act}(t) = A_p \cdot p_A(t) - A_r . p_B(t)$$

Applied torques or moment loads, if present. These are computed dynamically and injected into the global force vector during each time step.

3.4.2 System Configuration Parameters

Table 3: Physical and Material Constants of Crane Simulation Model

Parameter	Symbol	Value	Unit
Material Density	ρ	7850	kg/m^3
Elastic Modulus (Steel)	E	210×10^9	Pa
Cross-sectional Area	A	Based on design	m^2
Moment of Inertia	I	Based on geometry	m^4
Gravitational Acceleration	g	9.81	m/s^2
Rayleigh Damping Coeffs	α, β	0.01, 0.001	—

Note: While payload mass can vary across simulation runs, it is not fixed in this framework and is treated as a modifiable external parameter.

3.5 Data Preprocessing and Storage

The data-driven stress prediction framework required precise generation, transformation, and organization of simulation data obtained from the flexible multibody crane model described earlier. The goal was to produce structured time-series datasets reflecting the dynamic response of the crane under different motion profiles, which could then be used for training predictive algorithms. This section outlines the simulation-driven data acquisition process, the types of physical quantities collected, and the transformations performed to prepare them for machine learning tasks.

3.5.1 Simulation-based Data Generation

Initial datasets were generated using realistic input signals derived from joystick-controlled crane operations. These signals corresponded to the angular commands for both the lift and tilt arms of the crane. The joystick values were scaled and interpolated into time-continuous angle references, which were then dynamically converted into equivalent hydraulic actuator displacements. This mapping reflected the kinematic relationship between joint rotations and the linear extensions of the associated hydraulic cylinders. The actuator displacements were then applied to the simulation model as time-varying boundary conditions.

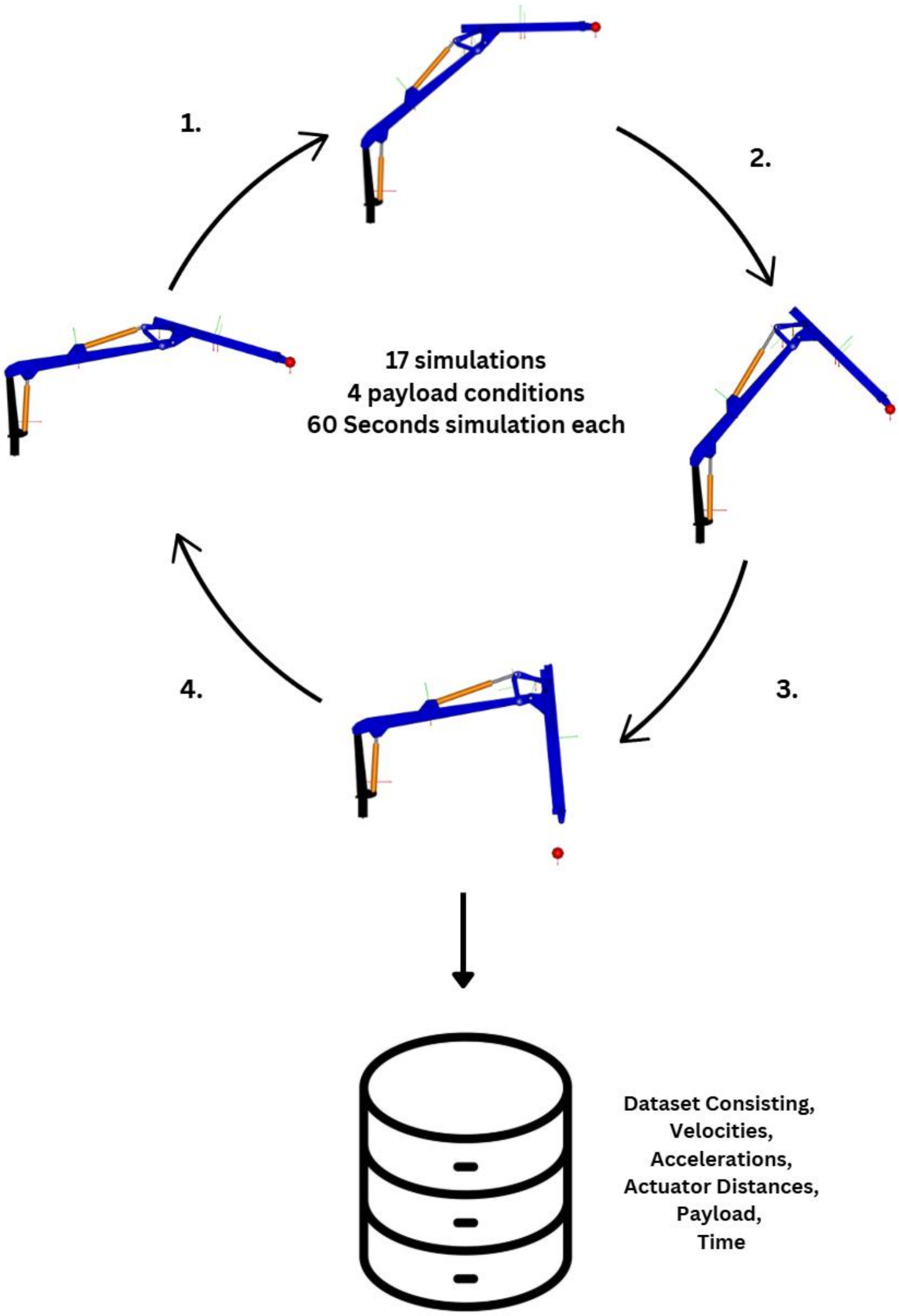

Figure 11: Dataset Generation using Simulations

In addition to joystick-driven scenarios, further simulations were performed by applying randomized or scripted sequences of angular velocities, displacements, and accelerations. This was done to artificially expand the diversity of operational states, especially under less frequent or more extreme loadings. These simulated profiles generated a wide spectrum of structural responses, especially in terms of stress distribution and joint dynamics.

Each simulation run produced a set of mechanical variables, stored as time series. These included quantities as actuator velocities $v1(t), and\ v2(t)$ corresponding to the lift and tilt actuators. Actuator displacements $x_1(t), x_2(t)$ are measured as the instantaneous length between base and rod ends. Angular accelerations $\alpha_1(t),\ and\ \alpha_2(t)$, were measured along the vertical axis of the joint points. Structural stress $\sigma(t)$, computed at a critical section located 1 meter from the base joint of the lift boom. Applied load (mass) values, used in selected simulations to examine payload influence. All simulations used a fixed time step (0.005 seconds), with signals sampled at high frequency to ensure temporal smoothness and physical realism.

3.5.2 Data Characteristics and Physical Relevance

The stress profiles obtained from the simulation exhibited significant physical variation. For joystick-driven cases, the stress response followed smoother trajectories, with distinguishable rising and falling edges. However, in randomly actuated sequences, stress signals showed oscillatory behavior, including sharp peaks and rapid transients. These phenomena mirrored the structural oscillations expected under sudden actuator movement or high angular acceleration.

The presence of such variations was important not only for capturing the system's physical complexity but also for providing rich learning targets during model training. The stress values were notably sensitive to combinations of actuator displacement speed, angular acceleration, and payload configuration. Table 4 below summarises the key physical quantities included in each time step of the dataset.

Table 4: Simulation Signal Features and Descriptions

Symbol	**Quantity**	**Description**
t	Time	Simulation time in seconds (60 seconds)
m	Payload	Mass applied at crane tip (180, 280,300,480 kg)

$v1(t)$	Velocity (Lift Actuator)	Axial velocity of lift cylinder (derived from displacement)
$v2(t)$	Velocity (Tilt Actuator)	The axial velocity of the tilt cylinder
$x_1(t)$	Displacement (Lift Actuator)	Length between base and rod markers on lift cylinder
$x_2(t)$	Displacement (Tilt Actuator)	Length between base and rod markers on tilt cylinder
$\alpha_1(t)$	Angular Acceleration (Lift)	Vertical angular acceleration at the lift joint
$\alpha_2(t)$	Angular Acceleration (Tilt)	Vertical angular acceleration at tilt joint
$\sigma(t)$	Stress at 1m on Lift Boom	Axial stress computed using beam theory at 1m from the base joint

3.5.3 Normalization and Dataset Structuring

Once all simulations were completed, the raw time-series data was compiled into structured files, each containing several thousand time steps of input and output variables. To ensure numerical stability during model training and allow the algorithms to focus on patterns rather than scale, all signals were normalized using min-max scaling. Each input variable was scaled independently to a range between 0 and 1 using global extrema computed across the full simulation dataset.

Each time-series file was then segmented into overlapping sequences of a fixed length (100 time steps per segment), representing a sliding window of crane motion history. These sequences became the input to the stress prediction models, with the corresponding stress value at the end of each sequence used as the target output.

Different simulation files were assigned to training, validation, and test sets. Care was taken to avoid data leakage by ensuring that the validation and test files were not part of the training pool.

4 Machine Learning Models for Stress Prediction

4.1 Introduction to Machine Learning Approaches

In the context of the hydraulic simulation of Exudyn, machine learning enables the approximation of complex nonlinear mappings between sensor-based observations and underlying stress states. Given the multivariate and temporal nature of simulation data from the crane system, deep sequential models are well-suited to extract both spatial features (from individual time steps) and temporal dependencies (across sequences).

This chapter focuses on two advanced neural architectures used for modeling the mapping:

$$Input\ Features \rightarrow Predicted\ Stress$$

Each input sequence consists of time-ordered vectors:

$$X = \left\{x^{(t)}, x^{(t+1)}, \ldots., x^{(t+L-1)}\right\}, x^{(i)} \in \mathbb{R}^8$$

Where L=100 is the sequence length and each feature vector $x^{(i)}$ contains x_1 as Time t, x_2 Payload x_3, x_4 Actuator velocities (v_1, v_2), x_5, x_6 Actuator displacements (x_1, x_2), x_7, x_8 Angular accelerations (a_1, a_2). The target output for each sequence is the corresponding axial stress $\sigma(t + L)$ at the critical point on the lift boom.

4.2 CNN-biLSTM Model Architecture:

The CNN-BiLSTM model architecture combines spatial filtering with temporal memory. The model can be represented as a composite nonlinear function.

$$\hat{\sigma}(t + L) = f_{CNN-BILSTM}\,(X)$$

A 1D convolutional layer is applied to extract local patterns from the input signal:

$$Z_c[i,j] \ = \phi\left(\sum_{k=0}^{K-1} W_{j,k}\ .x^{(i+k)} + b_j\right), j = 1, \ldots., F$$

Where Z_c is the convolutional output of shape, $W_{j,k}$ learnable filter weights, F is several filters (64), K is kernel size (3), $and\ \phi$ is the ReLU activation function.

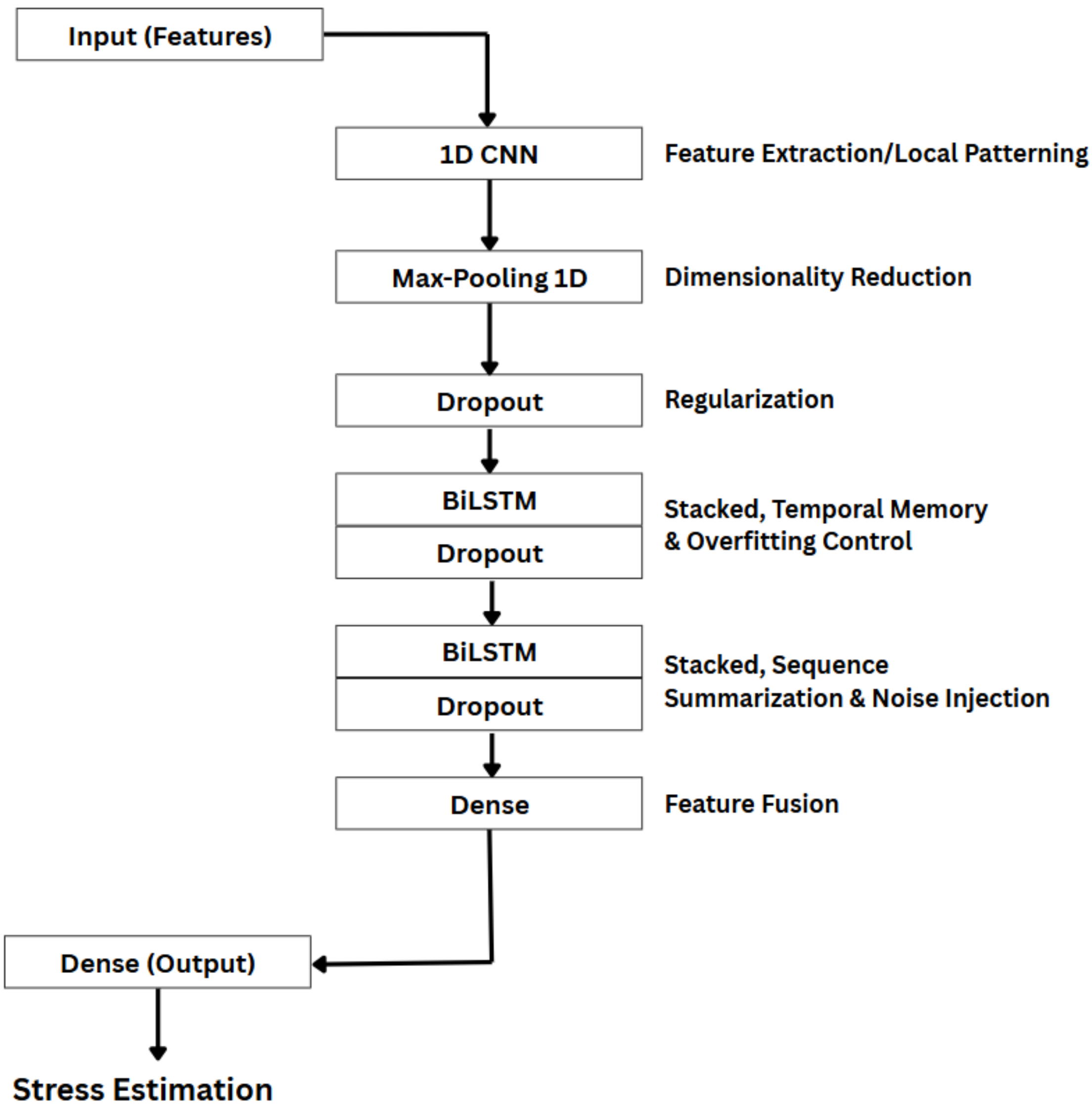

Figure 12: Overview of CNN-BiLSTM Model implemented for Stress Prediction

A max-pooling operation reduces the dimensionality:

$$Z_p[i,j] = \max_{l=0}^{P-1} Z_c\,[i+l, j]$$

Where P is the pooling size. The pooled feature sequence is passed through a BiLSTM block:

$$\overrightarrow{h_t} = LSTM\left(Z_{p[t]}, \overrightarrow{h_{t-1}}\right), \quad \overleftarrow{h_t} = LSTM(Z_p[t], \overleftarrow{h_{t-1}})$$

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}]$$

This captures both forward and backward temporal dependencies. Two stacked BiLSTM layers allow hierarchical temporal abstraction.

The final hidden state is passed through dense layers:

$$\tilde{h} = \Phi(W_1 h_t + b_1), \qquad \hat{\sigma}(t+L) = W_2 \tilde{h} + b_2$$

Where $\tilde{h}$ is a dense intermediate representation and $\hat{\sigma}(t+L)$ is the predicted stress value.

Table 5: Model Summary Table CNN-biLSTM

Layer	**Output Shape**	**Description**
Input	(100, 8)	Sequence of multivariate inputs
Conv1D	(100, 64)	Local pattern extractor
MaxPooling1D	(50, 64)	Downsampling temporal resolution
BiLSTM (Layer 1)	(50, 128)	Forward + backward memory
BiLSTM (Layer 2)	(50, 64)	Further temporal encoding
Dense + Dropout	(32)	Nonlinear compression
Output Dense	(1)	Final stress prediction

The model architecture stated above in the table proved effective in capturing both rapid actuator movements and gradual stress accumulations, as indicated by low test-set mean absolute error (MAE) during evaluation. In the next section, the transformer-based architecture is presented as an alternative model capable of attention-guided temporal weighting.

4.3 Transformer Model Architecture

The transformer model architecture relies on self-attention mechanisms to model long-range temporal dependencies. Unlike recurrent models, it does not process sequences sequentially; instead, it attends to all positions simultaneously. The stress prediction function is denoted as:

$$\hat{\sigma}(t+L) = f_{Transformer}(X)$$

Since transformers lack recurrence, positional encoding "PE" is added to preserve the order of timesteps. For each timestep $i \in [0, L-1]$ and dimension d, the encoding is defined as:

$$PE(i, 2k) = \sin\left(\frac{i}{10000^{\frac{2k}{d}}}\right), \; PE(i, 2k+1) = \cos\left(\frac{i}{10000^{\frac{2k}{d}}}\right)$$

The input sequence becomes:

$$Z_0 = X + PE$$

For each head h, the input is projected into query, key, and value spaces:

$$Q = Z_0 W_Q, K = Z_0 W_K, V = Z_0 W_V$$

The scaled attention output is computed as:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{dk}}\right)V$$

Multiple heads allow the model to capture diverse temporal relations:

$$MHA(Z_0) = [head_1, \ldots, head_h]W_O$$

Where $W_Q, W_K, W_V, and\ W_O$ are learnable projection matrices and d_k is the key dimension. After layer normalization and dropout, the attention output is passed through a two-layered feed-forward network:

$$Z_{FFN} = \phi(Z_{att}W_1 + b_1)W_2 + b_2$$

Where ϕ is the ReLU activation. The output tensor is pooled across time using global average pooling:

$$z_{pooled} = \frac{1}{L}\sum_{i=1}^{L} Z_{FFN}[i]$$

This vector is passed through a final regression head:

$$\hat{\sigma}(t + L) = W_r z_{pooled} + b_r$$

Table 6: Model Summary Table Attention Transformer

Layer	Output Shape	Description
Input + PE	(100, 8)	Time series with positional info
Conv1D	(100, 64)	Local temporal filtering
Transformer Block	(100, 64)	Multi-head attention + FFN
GlobalAvgPooling	(64)	Temporal aggregation

Dense + Dropout	(32)	Feature compression
Output Dense	(1)	Scalar stress prediction

The transformer's self-attention layers enabled the model to identify stress-relevant events across long time windows, including oscillatory dynamics or delayed actuator effects.

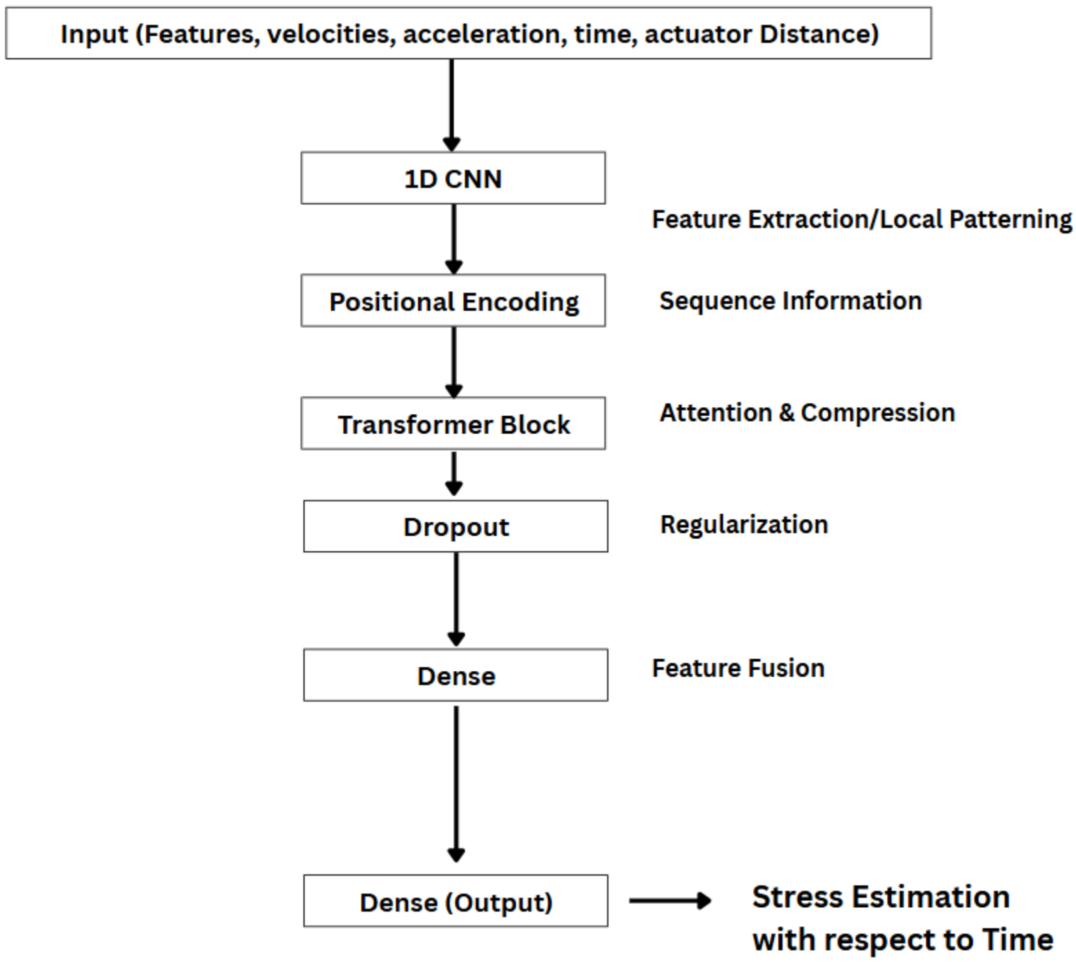

Figure 13: General Architecture of Proposed Transformers Model for Stress Estimation

Its attention-guided weighting contrasts with LSTM’s hidden-state memory approach, offering competitive accuracy under varying payload and motion conditions.

4.4 Model Training and Validation

The models were trained on sequential input data to map actuator and sensor signals to predicted stress values. The training process involved supervised learning with mean squared error (MSE) as the loss function:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(\hat{\sigma}_i - \sigma_i)^2$$

Where σ_i is the ground truth stress from the simulation, and $\hat{\sigma}_i$ is the predicted stress by the model, N is the number of training sequences.

The Adam optimizer was used to minimize the loss, with a learning rate $\eta = 1 \times 10^{-4}$ for the transformer and 5×10^{-4} for the CNN-BiLSTM model. Early stopping was applied with a patience of 5 epochs to prevent overfitting.

Validation loss was tracked on a separate validation set, generated from a simulation run with a payload of 280 kg. The input-output structure of the model training pipeline is summarised below.

4.4.1 Flow of Model Data

Each sample in the dataset consists of Input Sequence $X \in \mathbb{R}^{100\times 8}$: 100 timesteps of actuator/sensor signals, Output Target $\sigma \in \mathbb{R}$: Single stress value corresponding to the 101^{st} timestep. The preprocessing pipeline includes Feature Normalisation as

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}, for\ each\ x \in X$$

The Sequence Construction was carried out as:

$$X_i = \{x^i, x^{i+1}, \ldots, x^{i+99}\}, y_i = \sigma^{i+100}$$

Dataset Partitioning was carried out with the formulation as:

$$Training\ set, Validation\ set, Test\ set \subset \{(X_i, y_i)\}^{T=100}{}_{i=1}$$

The table 7 is illustrated in the following schematic:

Table 7: Flow of Methods, Schematic Followed

Stage	**Input**	**Output**
Simulation Logs	Raw actuator and stress data	.csv time series per simulation

Preprocessing	Raw signals	Normalized, windowed sequences
Training Loop	Batches of X_i, y_i	Updated model weights
Evaluation	Test sequences	Predicted stress + error metrics

This structured pipeline allowed consistent stress prediction across varied input conditions and crane motions.

4.5 Comparative Analysis

The performance of the two models was compared based on the following metrics:

Mean Absolute Error (MAE):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{\sigma}_i - \sigma_i|$$

Root Mean Squared Error (RMSE):

$$MAE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{\sigma}_i - \sigma_i)^2}$$

Prediction Oscillation Capture (POC) was assessed qualitatively by comparing predicted stress curves with ground truth curves, especially during fast actuator reversals or large angular accelerations.

5 Results and Discussions

5.1 Experimental Setup

To investigate real-time stress prediction in a hydraulically actuated crane, a total of 17 detailed simulations were conducted using the Exudyn multibody simulation platform. Each simulation represented a distinct payload condition ranging from low to high values and lasted 60 seconds in duration. These simulations incorporated randomized lift and tilt joystick control signals to emulate realistic and diverse crane operations. As a result, the dataset captured a wide variety of motion patterns, dynamic loading conditions, and stress responses over time.

From each simulation, a time-series dataset was extracted comprising eight input features: TIME, angular velocities, actuator displacements, and vertical accelerations. In parallel, axial stress was computed at a fixed location—1 meter from the base of the lift boom—using joint reaction forces and beam theory.

For model training, 11 simulations were used to build the training set, and one simulation (payload 280 kg) was used as a validation set. The remaining 5 simulations, each with different motion characteristics and payload configurations, were held out as an unseen test set. All sequences were formed using a sliding window of 100-time steps, and both input and output values were scaled using Min-Max normalization. The final dataset consisted of tens of thousands of sequence samples representing a rich and diverse crane stress behavior across varying operating conditions. This experimental setup ensured that the models were exposed to general crane dynamics during training while being tested on entirely unseen crane motions. Such a split enabled a realistic evaluation of the models' generalization capacity under practical working conditions.

5.2 Performance Analysis of CNN-BiLSTM Model

The CNN+BiLSTM model architecture consisted of a Conv1D layer for local pattern extraction, followed by MaxPooling, two Bidirectional LSTM layers for sequence learning, and dense layers for regression. The model was trained using the Adam optimizer with a learning rate of 5e-4, and early stopping was applied to prevent overfitting.

Quantitatively, the model achieved a test MAE of 0.0348 and an MSE of 0.0032. The inference time for predicting 28.2 seconds of simulated motion was measured at 15.041 seconds. In smoother regions of the simulation, particularly during consistent lifting or tilting motion with minor velocity changes, the predicted stress closely follows the ground truth. The trend of rising and falling stress in these intervals is well preserved, which validates the model's understanding of motion-to-stress translation.

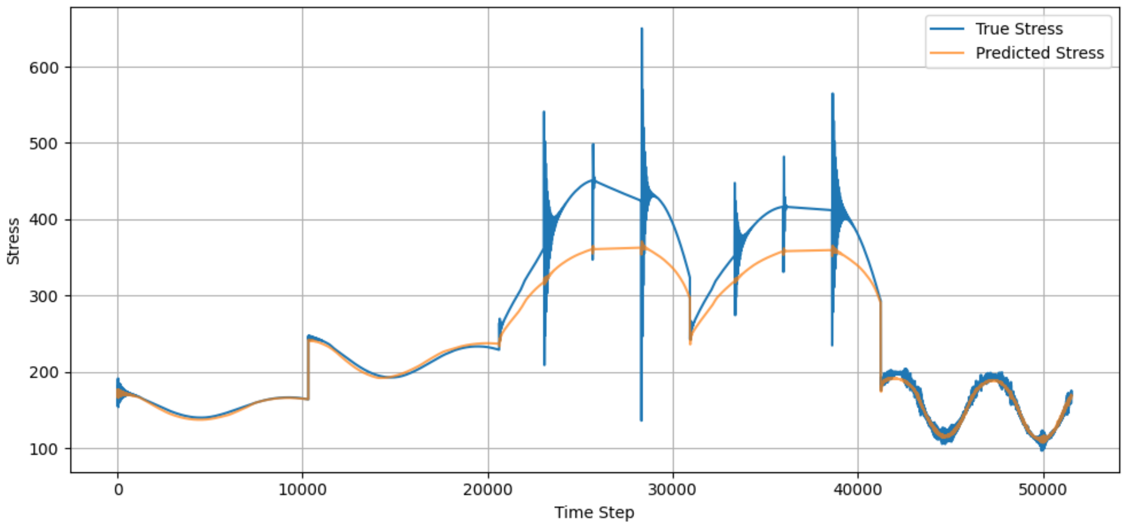

Figure 14: True Stress vs. Predicted Stress for CNN-BiLSTM Model

The total inference time was computed as:

$$T_{Total} = t_{end} - t_{start} = 15.041 \; seconds$$

Simulation Duration was computed as:

$$T_{sim} = N \times \Delta t = 28271 \; \times 0.001 = 28.271 \; seconds$$

Where N was taken as the number of prediction sequences and Δt as 1 millisecond (sampling interval time for the unseen test case taken). The time per sequence was calculated as:

$$T_{per_{SEQ}} = \frac{T_{Total}}{N} = \frac{15.041}{28271} = 0.000532\ seconds$$

Also, for the scenario of adaptability to the real-time applications, the real-time factor was concluded, calculated as:

$$RTF = \frac{T_{sim}}{T_{Total}} = \frac{28.271}{15.041} = 1.88$$

This indicates the model runs 1.88 times faster than real-time, which confirms its suitability for real-time deployment in control loops and embedded systems. The model's performance can be evaluated by comparing the predicted stress curve (green) to the actual stress curve obtained from Exudyn simulations (yellow).

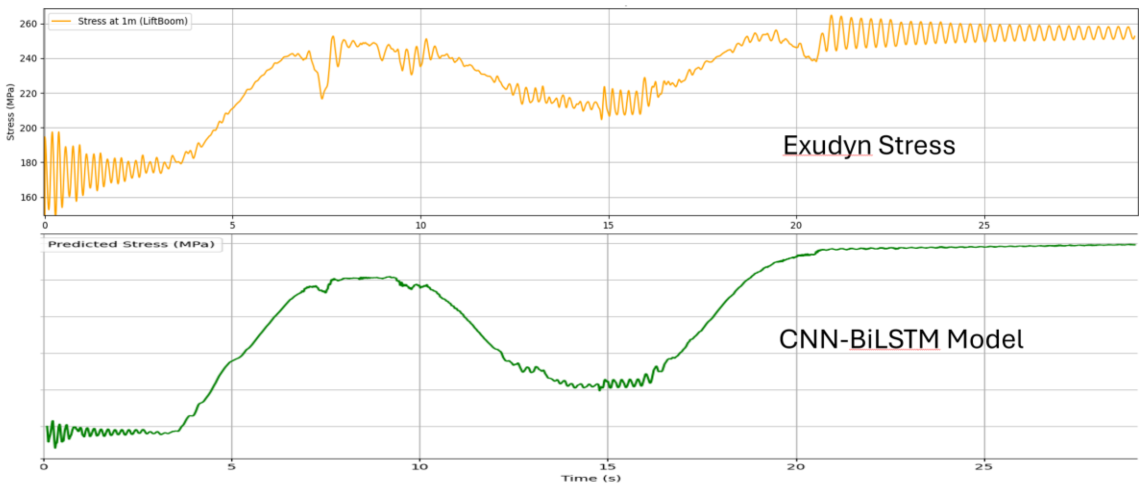

Figure 15: Comparison of Stress Profile, Yellow referring to Exudyn Simulation stress plot, and Green referring to Estimated stress plot (Same input parameters, and unseen scenario for DL)

However, in high-dynamic scenarios—particularly around the 22s to 28s mark—where crane actuators exhibit rapid transitions or decelerations, the model struggles to replicate the abrupt changes in the stress profile. For example, the real stress curve shows several sharp spikes and valleys, likely driven by combined tilt and lift acceleration effects. In contrast, the LSTM-based prediction smooths out these features, resulting in a

noticeable underestimation of local maxima and minima. This can be attributed to the memory-based nature of LSTM layers, which tend to average over sequential states.
Despite this limitation, the predicted signal still provides a useful qualitative view of the crane's structural response. Importantly, the model succeeds in preserving the overall curvature and timing of stress shifts, even if it does not perfectly match the magnitude. This is critical in trend-based stress monitoring, where the interest lies more in the shape of the stress trajectory and its synchronization with control actions than in precise stress values.
It could be justified from all stated above that the CNN+BiLSTM model presents a highly efficient method for estimating the stress profile in real-time. It is well suited for low-risk applications where the goal is to monitor general system behavior or anticipate long-term fatigue trends, but less appropriate for failure prediction or critical safety monitoring where stress peaks must be precisely captured.

5.3 Performance Analysis of Attention Transformer Model

The Attention Transformer model was developed to overcome limitations associated with recurrent architectures by leveraging self-attention for context-aware learning. The network begins with a Conv1D layer to extract local motion features, followed by a positional encoding layer that retains temporal information. The core Transformer block includes multi-head attention to weighing time steps according to their relevance, followed by a feed-forward layer and dropout regularisation.

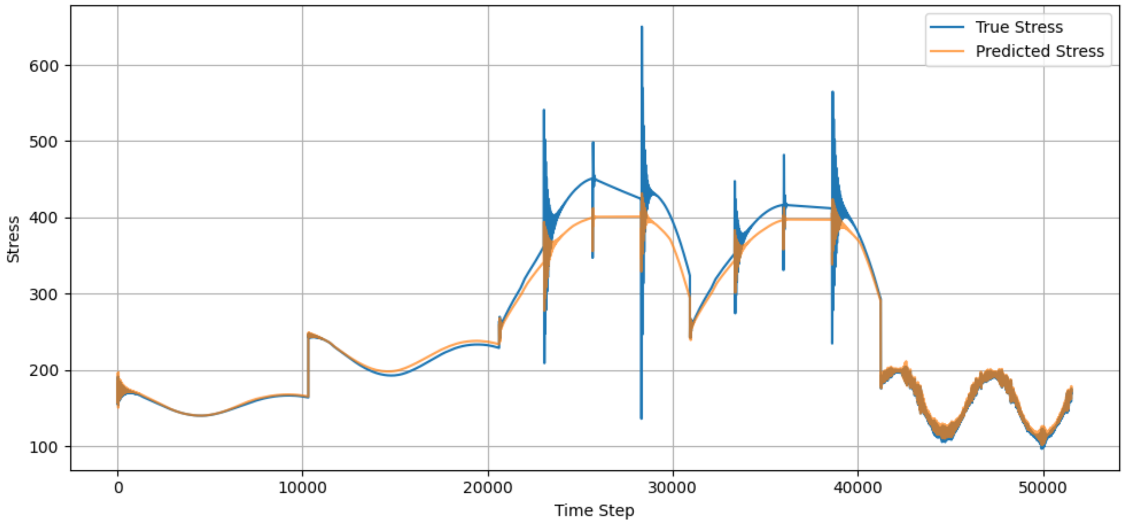

Figure 16: True Stress vs. Predicted Stress for Transformer Model

Quantitatively, the performance for the computation time was calculated using different factors considered. Such include,

The total inference time was computed as:

$$T_{Total} = t_{end} - t_{start} = 25.2281\ seconds$$

Simulation Duration was computed as:

$$T_{sim} = N \times \Delta t = 28271\ \times 0.001 = 28.271\ seconds$$

Where N was taken as the number of prediction sequences and Δt as 1 millisecond (sampling interval time for the unseen test case taken). The time per sequence was calculated as:

$$T_{per_{SEQ}} = \frac{T_{Total}}{N} = \frac{25.2281}{28271} = 0.000892\ seconds$$

Also, for the scenario of adaptability to the real-time applications, the real-time factor was concluded, calculated as:

$$RTF = \frac{T_{sim}}{T_{Total}} = \frac{28.271}{25.2281} = 1.12$$

Though slower than the BiLSTM model, the Transformer still exceeds real-time performance, making it suitable for practical applications. The Transformer's prediction offers closer alignment with the Exudyn-generated stress signal, particularly in high-dynamic regions where LSTM predictions deteriorate.

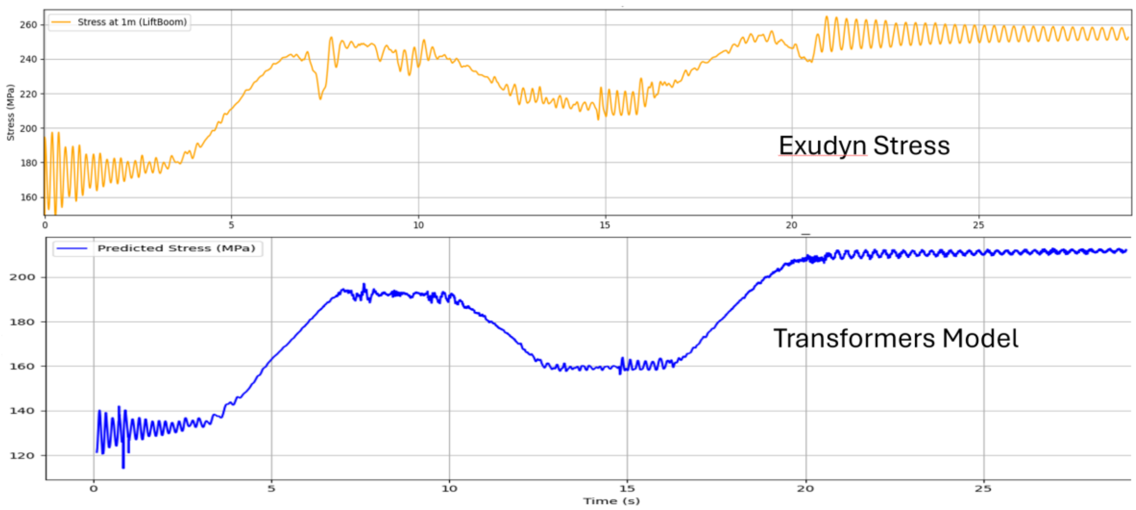

Figure 17: Comparison of Stress Profile, Yellow referring to Exudyn Simulation stress plot, and Blue referring to Estimated stress plot (Same input parameters, and unseen scenario for DL)

From 22s onward, the real stress signal exhibits multiple sharp peaks and reversals caused by sudden decelerations and direction changes in lift and tilt actuators. The Transformer can follow these transitions more closely, preserving the shape, timing, and partial amplitude of stress spikes. This improvement is a result of the self-attention mechanism, which allows the model to dynamically attend to impactful time steps—those linked to strong motion variations. Unlike LSTM, which processes time sequentially and averages memory, the Transformer compares all positions simultaneously, retaining sharp local changes while understanding the global context. Although it still slightly underestimates stress amplitude at extreme peaks, the phase alignment and waveform fidelity make it a stronger candidate for structural monitoring systems where catching fast, short-term stress events is essential. The Transformer model demonstrates superior accuracy and responsiveness to dynamic crane behavior, making it ideal for safety-critical tasks like overload detection or adaptive control. It balances prediction precision with acceptable computational load, operating at 1.12x real-time speed. This makes it an excellent choice for digital twin implementations and intelligent maintenance strategies that depend on timely and reliable stress forecasting.

5.4 Comparative Evaluation of Models

To evaluate the relative performance of the two deep learning models; CNN+BiLSTM and Transformer, a detailed comparison was conducted using both quantitative metrics and qualitative behavior. Each model was assessed in terms of prediction accuracy, ability to follow stress profile trends, responsiveness to rapid dynamic changes, and computational efficiency. The table below summarises the core results:

Table 8: Comparative Analysis of Deep Learning Algorithms Proposed

Model	MAE	MAP	Inference Time (s)	Time per Sequence (s)	Simulated Duration (s)	Real-Time Factor
CNN+BiLSTM	0.0348	0.0032	15.041	0.000532	28.271	1.88×
Transformer	0.0054	0.0000687	25.2281	0.000892	28.271	1.12×

The Transformer outperformed the CNN+BiLSTM model across all accuracy metrics. Its mean squared error was two orders of magnitude lower, demonstrating exceptional precision in following stress curves, including high-frequency oscillations and sudden peaks. The Transformer maintained stronger phase alignment with the true stress signal, which is critical in scenarios involving predictive maintenance or overload detection.

The CNN+BiLSTM model, on the other hand, achieved much faster inference time and lower computational load. It is approximately 40% faster than the Transformer, with a real-time factor of 1.88×, making it more suitable for edge-based or resource-constrained deployments. However, this speed comes at the cost of smoothing out high-frequency fluctuations, which limits its utility in safety-critical or failure-sensitive use cases. From a functional perspective, both models successfully captured the underlying trend of stress evolution in response to crane movement inputs. They both preserved the relationship between control signals and resulting stress patterns. The key difference lies in how sharply and accurately they responded to transient changes. The

Transformer, due to its attention mechanism, was better equipped to handle non-linear and irregular dynamics caused by abrupt joystick commands or payload shocks.

This can be concluded from the study that the Transformer is the preferred model when accuracy and detailed tracking of structural response are required, particularly in digital twin or supervisory control frameworks. The CNN+BiLSTM remains a strong alternative where speed and simplicity are more critical than capturing every stress peak with high fidelity.

5.5 Summary of Findings

This study demonstrated the viability of using deep learning models to estimate structural stress in a hydraulically actuated crane. By leveraging simulated datasets generated from 4 different payload conditions under varying motion commands (17 simulations), two models—CNN+BiLSTM and Transformer—were trained and evaluated. The CNN+BiLSTM model was effective at capturing general stress trends and exhibited low computational latency, achieving nearly twice the speed of real-time simulation. It performed reliably in low to moderately dynamic scenarios but struggled to capture abrupt changes in stress values. Conversely, the Transformer model delivered high-fidelity stress predictions with strong temporal alignment and responsiveness to fast-changing inputs, albeit with a slightly longer inference time. Both models were able to generalize to unseen crane motion data, proving their potential as surrogate stress estimators for flexible-body simulations. The choice between the two depends on the application requirements—whether the goal is to detect trends quickly or to capture detailed structural behavior precisely. However, in the case of justification, the Transformer model provides the most accurate and robust stress estimation, while the CNN+BiLSTM model offers a lightweight, deployable alternative for real-time monitoring.

6 Conclusion and Future Work

6.1 Conclusion

In this thesis, we have explored whether data-driven models could replicate and predict the dynamic stress behavior of a flexible hydraulic crane system. The work is motivated by a practical need to monitor structural response in real-time, not through computationally expensive physical models, but via efficient machine learning surrogates. The core objective was not to compute exact stress values but to understand and model how stress evolves under various crane motion conditions. The idea began by designing a controlled simulation environment that mimicked real-world actuation signals and mechanical responses. From this foundation, a rich dataset was generated that captured a diverse range of operational scenarios and payload combinations. By translating these physical dynamics into structured sequences, the study created a platform suitable for training deep learning models capable of temporal understanding. Research of a comparative exploration of two model architectures was taken into consideration: a CNN+BiLSTM and an attention-based Transformer. These models represented distinct approaches—one prioritizing efficiency and temporal memory, the other favoring global pattern recognition through attention mechanisms. Through rigorous testing on unseen crane motion sequences, both models proved capable of tracking the temporal profile of structural stress, confirming that neural sequence learners can indeed serve as viable proxies for mechanical behavior. The broader implication of this work lies in how it reimagines structural monitoring. Rather than waiting for data from physical sensors or running high-fidelity simulations, a trained neural model can now anticipate how a structure might behave based on its inputs. This creates opportunities for intelligent, responsive crane systems that can pre-emptively adapt to mechanical stress, enhance safety, and reduce maintenance costs. Ultimately, the thesis demonstrated that artificial intelligence can be seamlessly integrated into engineering workflows, not to replace physical models, but to extend their reach and operational usefulness. The CNN+BiLSTM model demonstrated good generalization and captured the overall trend of the stress profile effectively. In contrast, the Transformer model achieved significantly higher prediction accuracy. It maintained better alignment with the Exudyn-generated ground truth, particularly in segments where actuator commands caused abrupt changes in stress. The Transformer preserved the shape and timing of high-frequency components in the stress profile more effectively than the LSTM model. Both models were able to generalize to motion sequences they had never seen before, confirming their robustness. The results demonstrate that deep sequential models can be successfully used as lightweight, real-time surrogates for computationally intensive stress calculations in multibody simulations. This conclusion supports the idea that stress estimation models can be integrated into digital twin frameworks for predictive maintenance and real-time control.

6.2 Contributions Revisited

This work contributed to a simulation-to-learning pipeline for intelligent stress estimation in a dynamic mechanical system. A multibody simulation framework was employed to create a diverse dataset under 4 different payload and motion scenarios, enabling robust model training. Two different neural architectures—CNN+BiLSTM and Transformer—were implemented and rigorously evaluated. In addition to validating prediction accuracy, the study measured the real-time performance of both models, demonstrating their applicability in embedded systems and control frameworks. This dual evaluation of fidelity and efficiency sets the foundation for using these models in both offline analysis and real-time decision-making systems.

6.3 Limitations of Research

Despite the promising outcomes, the research faced several limitations. All training and evaluation data was derived solely from simulations; no experimental data was incorporated, which limits external validity. The model was applied only to a specific crane design (PATU 655), making it unclear how well it would generalize to other structural systems. Furthermore, the simulated data did not account for sensor noise, mechanical wear, or environmental conditions that often influence real-world measurements. Stress was also computed at a single location on the crane, while actual structural analysis often requires distributed sensing to understand full-system behavior.

6.4 Future Works

Future work should focus on bridging the gap between simulation and deployment by incorporating real-world crane sensor data into the training process. Extending the dataset to cover various crane geometries and control scenarios would enhance model robustness. Introducing uncertainty quantification would provide confidence estimates alongside predictions, improving reliability in critical applications. The potential of hybrid models—combining physics-informed constraints with deep learning—also offers a path forward for enhancing generalisability and trust. Ultimately, embedding the developed models into a digital twin platform could enable predictive maintenance, live condition monitoring, and smart control in real industrial environments.

References

1. Bhatnagar, S.; Gill, L.; Ghosh, B. Drone Image Segmentation Using Machine and Deep Learning for Mapping Raised Bog Vegetation Communities. *Remote Sens.* **2020**, *12*, 2602. https://doi.org/10.3390/rs12162602
2. Bolandi, H., Ahmadi, F., & Farzi, A. (2023). Neuro-DynaStress: Predicting dynamic stress distributions in structural components. *Engineering Structures, 287*, 115334. https://doi.org/10.1016/j.engstruct.2023.115334
3. Hashemi, A., Grzegorz., Mikkola, & McPhee. (2023). Multibody dynamics and control using machine learning. *Journal of Mechanical Science and Technology, 37*(1), 57–68. https://doi.org/10.1007/s12206-022-1259-3
4. Jiang., Zhenguo, Roselyn., Amir., Levent. (2020). StressGAN: A generative deep learning model for 2D stress distribution prediction. *Computers, Materials & Continua, 64*(2), 813–829. https://doi.org/10.32604/cmc.2020.010844
5. Cheng, Xu, Chen, Wang, Lin, Liu. (2025). A deep learning framework integrating CNN and BiLSTM for financial systemic risk prediction. *Expert Systems with Applications, 234*, 120234. https://doi.org/10.1016/j.eswa.2024.120234
6. Q. Khadim et al., "State Estimation in a Hydraulically Actuated Log Crane Using Unscented Kalman Filter," in IEEE Access, vol. 10, pp. 62863-62878, 2022, doi: 10.1109/ACCESS.2022.3179591.
7. K. J. Jensen, M. K. Ebbesen, and M. R. Hansen, “Online Deflection Compensation of a Flexible Hydraulic Loader Crane Using Neural Networks and Pressure Feedback,” *Robotics*, vol. 11, no. 1, p. 34, Mar. 2022, doi: 10.3390/robotics11010034.
8. Kotta, J. (2021). *Modeling of PATU-655 loader with a combination of Mevea and Simulink using Functional Mock-Up Interface* (Master’s thesis, LUT University, LUT School of Energy Systems). LUT Mechanical Engineering.
9. Li, Q.; Fan, W.; Huang, M.; Jin, H.; Zhang, J.; Ma, J. Machine Learning-Based Prediction of Dynamic Responses of a Tower Crane under Strong Coastal Winds. *J. Mar. Sci. Eng.* **2023**, *11*, 803. https://doi.org/10.3390/jmse11040803

10. Shi, J., Wang, S., Qu, P. *et al.* Time series prediction model using LSTM-Transformer neural network for mine water inflow. *Sci Rep* **14**, 18284 (2024). https://doi.org/10.1038/s41598-024-69418-z
11. Manzl, P., Alexander, H., Khadim Q., & Gerstmayr, J. (2024). SLIDE: A machine-learning-based method for forced dynamic response estimation of multibody systems. *Mechanism and Machine Theory, 190*, 105215. https://doi.org/10.1016/j.mechmachtheory.2023.105215
12. Pavlatos, C.; Makris, E.; Fotis, G.; Vita, V.; Mladenov, V. Enhancing Electrical Load Prediction Using a Bidirectional LSTM Neural Network. *Electronics* **2023**, *12*, 4652. https://doi.org/10.3390/electronics12224652
13. Sasaki T, Shabana AA. Prediction of dynamic stresses using flexible multibody system algorithms: Application to tracked vehicle upper structure. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics*. 2014;229(2):177-192. doi:10.1177/1464419314557300
14. Soydaner, D. Attention mechanism in neural networks: where it comes and where it goes. *Neural Comput & Applic* **34**, 13371–13385 (2022). https://doi.org/10.1007/s00521-022-07366-3
15. Yang, X., Gao, J., Wang, P., Li, Wang, & Li. (2024). Digital twin-based stress prediction for autonomous grasping of underwater robots. *IEEE Transactions on Industrial Informatics, 20*(1), 66–75. https://doi.org/10.1109/TII.2023.3296721
16. Zhao, J., Dong, Q., Xu, G. *et al.* A fast prediction method of fatigue life for crane structure based on Stacking ensemble learning model. *J. Eng. Appl. Sci.* **71**, 207 (2024). https://doi.org/10.1186/s44147-024-00545-0
17. Armendia, M., Cugnon, F., Berglind, L., Ozturk, E., Gil, G., & Selmi, J. (2019). Evaluation of machine tool digital twin for machining operations in an industrial environment. Procedia CIRP, 82, 231–236.
18. Auricht, M., & Stark, R. (2015). MINARGUS: Test tool for User Experience measurement and parameter modification within ADAS simulation. Procedia CIRP, 36, 83–88.

19. Autiosalo, J., Vepsalainen, J., Viitala, R., & Tammi, K. (2020). A Feature-Based Framework for Structuring Industrial Digital Twins. IEEE Access, 8, 1193-1208. Article 8887161. https://doi.org/10.1109/ACCESS.2019.2950507
20. Balachandar, S., & Chinnaiyan, R. (2019). Reliable digital twin for connected footballers. International Conference on Computer Networks and Communication Technologies, (pp. 185–191).
21. Donoghue, I. D., Hannola, L. H., & Papinniemi, J. (2017). Product lifecycle management framework for business transformation. 24th International Conference on Production Research (ICPR 2017), 14. Poznan, Poland.
22. Donoghue, I., Hannola, L., & Mikkola, A. (2019). The Value of Digital Twins and IoT-Based Services in Creating Lifecycle Value in B2B Manufacturing Companies. 2019 Portland International Conference on Management of Engineering and Technology (PICMET), (pp. 1-6).
23. Jaiswal, Suraj, Pasi Korkealaakso, Rafael Åman, Jussi Sopanen, and Aki Mikkola. 2019. "Deformable terrain model for the real-time multibody simulation of a tractor with a hydraulically driven front-loader." IEEE Access (IEEE) 7: 172694-172708.
24. A. Avello, J. M. Jiménez, E. Bayo, and J. G. de Jalón, ‘‘A simple and highly parallelizable method for real-time dynamic simulation based on velocity transformation,’’ Comput. Methods Appl. Mech. Eng., vol. 107, no. 3, pp. 313–339, 1993.
25. J. Watton, Fluid Power Systems: Modeling, Simulation, Analog and Microcomputer Control. Cambridge, U.K.: Prentice-Hall, 1989.
26. S. Jaiswal, M. I. Islam, L. Hannola, J. Sopanen, and A. Mikkola, ‘‘Gamification procedure based on real-time multibody simulation,’’ Int. Rev. Model. Simul., vol. 11, no. 5, pp. 259–266, 2018.
27. J. Rahikainen, A. Mikkola, J. Sopanen, and J. Gerstmayr, ‘‘Combined semi-recursive formulation and lumped fluid method for monolithic simulation of multibody and hydraulic dynamics,’’ Multibody Syst. Dyn., vol. 44, no. 3, pp. 293–311, 2018.

28. J. Cuadrado, D. Dopico, M. Gonzalez, and M. A. Naya, ‘‘A combined penalty and recursive real-time formulation for multibody dynamics,’’ J. Mech. Des., vol. 126, no. 4, pp. 602–608, 2004.
29. J. T. Cook, L. Ray, and J. Lever, ‘‘Multi-body dynamics model of a tracked vehicle using a towing winch for optimal mobility control and terrain identification,’’ in Proc. ASME Dyn. Syst. Control Conf., Minneapolis, MN, USA, 2016, pp. 1–8.
30. Z. Zou, X. Pang, and J. Chen, ‘‘Comprehensive theoretical digging performance analysis for hydraulic excavator using convex polytope method,’’ Multibody Syst. Dyn., vol. 47, no. 2, pp. 137–164, 2019.
31. A. Nicolini, F. Mocera, and A. Somà, ‘‘Multibody simulation of a tracked vehicle with deformable ground contact model,’’ Proc. Inst. Mech. Eng. K, J. Multi-Body Dyn., vol. 233, no. 1, pp. 152–162, 2018.
32. M. G. Bekker, Introduction to Terrain-Vehicle Systems. Ann Arbor, MI, USA: Univ. of Michigan Press, 1969.