AI in Construction Management: Preparedness and Potential

(article starts on next page)

AI in Construction Management: Preparedness and Potential

Hai Ha Vu, Mohammed Rauf, Rasmus Rempling

Chalmers University of Technology, Gothenburg, Sweden

Mats Granath

University of Gothenburg, Gothenburg, Sweden

Erik Ulvås

NCC Sweden AB, Gothenburg, Sweden

Contact: vuha@chalmers.se,

Abstract

To address the constant challenges related to project delays and low productivity in the construction industry, this paper explores the opportunity of integrating machine learning based predictive models to improve decision-making in construction project management. In collaboration with NCC, a case study of the Ingelkärr–Stenkullen transmission line project was conducted to develop a hybrid forecasting model that combines Monte Carlo simulations with neural–network–based machine learning. The initial results showed high predictive accuracy (R^2=0.92) and updates weekly with the actual progress, enabling adaptive learning. The proposed framework shows strong potential to transform industry practices by significantly improving risk forecasting, optimizing resource management, and increasing responsiveness to uncertainty, thereby offering a pathway to more efficient and resilient project management in construction.

Keywords: Monte Carlo simulation; machine learning; artificial intelligence; coefficient of determination; progress forecasting; transmission line construction; float management; data-driven decision support; GPU acceleration; hybrid forecasting model.

1 Introduction

External factors such as economic uncertainties and regulatory pressures, as well as internal factors including complex project scopes and tight deadlines, contribute to the persistent challenges in the construction industry. These issues are interrelated, creating an industry characterized by a highly interdependent network of risk factors. Traditionally, project managers have relied on intuition, experience, and conventional management methods. However, these approaches often fall short in addressing modern project complexities, leaving managers struggling to meet deadlines and stay within budget. As a result, the industry has suffered from consistently low productivity levels, with construction seemingly rank among the least productive sectors worldwide [1].

1.1 Problem statement

Since these challenges arise from diverse sources, an ideal solution should involve analytical methods such as Monte Carlo simulations and deep neural networks, which provide systematic approaches to quantifying and analyzing uncertainties. However, applying these mathematical models in practice has proven difficult. Although the construction industry generates vast amounts of data, ranging from images to economic records, such data requires tailored processing techniques. The absence of standardization across data types further complicates their integration and comprehensive analysis.

1.2 Objective and scope

This research combines both technical development and practical relevance [2]. From a technical perspective, the study develops and validates a machine learning model capable of forecasting project progress and detecting trends that may indicate emerging risks or delays. From a practical perspective, the work incorporates interviews and collaborative sessions with construction professionals to align the technical solution with real-world needs. To ground the research in practice, a case study of the Ingelkärr–Stenkullen transmission line project was conducted. This provided access to actual project data for model development, while interviews with on- and off-site project managers offered valuable insights into their views on artificial intelligence and its potential role in construction management.

2 Background and Theory

As previously mentioned, the construction industry relies heavily on traditional methods like Gantt charts and critical path methods. While these approaches remain widely used due to their simplicity and visual clarity, they have notable limitations **[3]**. Gantt charts provide accessible overviews but struggle with complexity and dynamic updates whereas the Critical Path Method (CPM) offers more rigorous scheduling yet is often hindered by inconsistent training and lack of standardization **[4]**.

2.1 Machine Learning in construction

The implementation of Machine learning in the construction industry is still at a nascent stage; this means it is unable to fully adapt and benefit from the vast amount of data generated from a construction project. Additionally, concern remains regarding the safety of adopting new technologies in an industry where data privacy and security hold significant weight, alongside questions of industry readiness, worker acceptance and cost [5]. These challenges call for substantial resources and investment, not only on the technical side but also in supporting and training workers.

2.2 Theory

The theory behind the project is based on a combination of a mathematical model and machine learning (ML). ML is a branch of AI that allows systems to recognize patterns, make predictions, or solve complex problems by learning from data, rather than relying on explicit programming for every different scenario.

2.2.1 Monte Carlo

To operationalize the mathematical part of this framework, we use Monte Carlo methods. Within this framework, the simulations of the mathematical models are done by the Monte Carlo methods, which are numerical techniques that rely on randomness and statistical sampling [6]. Monte Carlo approaches can be described by integrals of the general form:

$$G = \int g(x)f(x)dx$$

Here, $f(x)$ represents a probability distribution (probability density function), satisfying

$$\int f(x)dx = 1$$

and $g(x)$ is a quantity for which the expected value should be calculated.

Monte Carlo methods approximate such integrals by repeatedly generating random samples from the distribution $f(x)$ and computing the average value of the function $g(x)$ at these sample points. Formally, the Monte Carlo approximation of the integral is expressed as:

$$G \approx \frac{1}{N}\sum_{i=1}^{N} g(x_i)$$

where each x_i is independently drawn from the distribution $f(x)$. As the number of samples N increases, the estimated average becomes progressively closer to the true value of the integral due to the law of large numbers [7].

2.2.2 Artificial Neural Networks

Artificial Neural Networks (ANNs) are computational models inspired by how biological neural networks in the brain process information. They are built from layers of interconnected processing units called neurons, which pass information along through weighted connections [7]. These weights determine the strength of the connections and play a key role in how the network learns and updates its internal structure based on input data. Together, the Monte Carlo simulations and ANNs form the project's hybrid framework.

3 Methodology

3.1 Data collection and preprocessing

The initial phase of a construction project involves developing a project schedule in a Gantt chart. This schedule details all tasks required for project completion along with attributes such as task dependencies, durations, and assigned resources. Throughout the project, the site manager continuously updates the Gantt chart to reflect real-time progress. Although the update frequency varies between projects, it is ideally conducted on a weekly basis. For this study, the final PowerProject file from the Ingelkärr- Stenkullen project was collected and used as validation data for the program.

This study is a single-project case study due to data-access constraints typical in industry settings. Rather than relying on large databases of comparable historical projects, which are often unavailable or difficult to obtain due to data laws, the proposed hybrid framework is designed to function in data-scarce environments by learning from Monte Carlo–generated trajectories and incrementally assimilating project-specific weekly updates.

The file contained all updates recorded throughout the project's duration. The PowerProject file was systematically processed and key attributes associated with each task were extracted and exported to an Excel (xlsx) file. The extracted attributes included:

- Power Post
- Duration
- Start Date
- End Date
- Predecessors
- Successors
- Progress per week

Before analysis, the extracted data were preprocessed to ensure consistency and avoid computational errors. Over-tasks, which were only categorical headers, were removed to retain only actionable tasks. Inconsistent or incomplete task dependencies were corrected to align sequences with the actual workflow. Missing end dates and incomplete progress values were updated using project records and logical inferences. These steps ensured accurate, clean, and well-structured data for the hybrid model.

3.2 Data pipeline

The data pipeline began with an Excel file. After cleaning the dataset, durations were standardized to hours, and unique identifiers were assigned. A dependency graph was then built to capture sequencing and float days. Each week, a cut-off date was applied, allowing real progress up to that point to be integrated into the dataset, while the remaining tasks were prepared for simulation. This ensured a continuously updated and structured dataset ready for forecasting.

3.3 Hybrid Framework

In this study, a hybrid modeling framework was developed, integrating a mathematical simulation with a ML model to produce a predictive timeline for construction project management. The approach consists of two main components: a Monte Carlo simulation and a deep learning model, combined into a hybrid predictive model.

In the hybrid predictive model, the Monte Carlo simulation serves as the foundation for the model. It uses project parameters, such as task durations and dependencies, to generate a range of possible project outcomes through stochastic sampling. Following this, an ML model complements the predictions by learning from the simulated project outcomes and ranking them based on probability.

At project start before any real progress has been observed the machine learning ML component is trained exclusively on trajectories generated by the Monte Carlo simulation. At this stage, the model has no project-specific performance evidence and therefore cannot conclude which trajectories are more likely, it mainly captures the central tendency of the simulated distribution.

Once weekly Real progress data becomes available, the Monte Carlo simulation is rerun conditioned on the observed progress up to the chosen cutoff week. The ML model is then updated using these observations and used to assign a relative plausibility score (ranking) to candidate trajectories, selecting those most consistent with the real-world progress history.

Integrating the real-world project progress each week increased the model accuracy. This hybrid framework combined the strengths of probabilistic simulation and ML pattern learning. While the Monte Carlo simulation generated a diverse range of potential timelines, the ML model processed these results to identify the most probable progress paths based on real-world project performance. Together, these components form a hybrid predictive model that serves as a forecasting framework, adapting to new information and continually improving overtime over time.

3.4 Monte Carlo component

A construction project comprises multiple interdependent tasks, each with distinct characteristics. Project progress is measured as the percentage of tasks completed; when every task is finished, the project reaches 100% completion. The mathematical modeling of a construction project begins with interpreting the project as a network of interdependent tasks. Each task is defined by its baseline duration, number of predecessors, and available float days.

Monte Carlo simulations were then applied by perturbing task durations using a Beta distribution with user-defined parameters. This allowed each task to vary realistically, becoming shorter or longer than planned. Sampling bounds were set using both a variability factor relative to the baseline and an absolute adjustment range, ensuring realistic duration intervals. Delays and accelerations were propagated through task dependencies, with float days acting as buffers for small deviations and larger overruns cascading to successors.

Each simulation iteration reconstructed a feasible schedule by enforcing sequencing and precedence constraints. The outputs included a time-series of cumulative project progress and a task-level dataset containing simulated durations, adjusted start and finish times, float values, and dependency information for further analysis.

The dataset comprised approximately 1,000 actionable tasks with complex dependencies, and experiments were run with 100 Monte Carlo iterations to generate a substantial synthetic training data for the learning component.

3.5 Machine Learning component

The machine learning model integrated simulation outputs and real-world project data to predict and rank likely completion paths. The architecture was based on a Multi-Layer Perceptron (MLP) enhanced with a Multi-Head Self-Attention mechanism. Task identifiers were converted into embeddings, which were combined with input features such as simulated durations, predecessors, and float days. These representations passed through fully connected layers with ReLU activation and dropout for regularization, while the attention layer captured dependencies and interactions between tasks within each simulation.

Training used iteration-aware batching, where each batch contained all tasks from a single simulation run, preserving task interdependencies. The dataset was split into training and validation sets across different iterations, and the model was optimized using Huber loss with the AdamW optimizer, guided by a learning rate scheduler and early stopping.

After initial training on simulation data, the model was updated weekly with new real-world progress, ensuring adaptability. It ranked simulated project paths by comparing predicted and simulated outcomes, highlighting the most realistic ones for analysis and visualization.

3.6 GPU acceleration

Initially, simulations ran on CPU libraries, but profiling revealed bottlenecks. To mitigate the bottlenecks, it was decided to mitigate critical routines to a GPU with CuPy, including preprocessing, Monte-Carlo simulations, scheduling, and ML training using CuPy and PyTorch. This significantly improved performance and scalability.

3.7 Design & visualisation

The model was designed as both a visual and analytical tool, providing managers with a clearer view of potential project outcomes under varying conditions. Feedback from interviews emphasized the need for clear and accessible visuals to support all stakeholders. Due to time constraints, the final demonstration was produced in Canva to illustrate the potential visualization of the result. Future development could explore various approaches to create user-friendly interfaces.

4 Results

4.1 Interview insights

The interviews conducted focused on three main perspectives: The project owner, on-site managers and off-site managers. Overall, most participants expressed a positive and optimistic outlook on AI and are generally eager to implement AI models into the construction projects. At the same time, several concerns were raised. Interviewees highlighted that AI might struggle to account for external factors such as weather conditions and regulatory constraints. On-site managers also emphasized on the importance of human judgment and expertise in the decision-making process, cautioning against over-reliance on automated predictions.

Another challenge identified was the generational gap and knowledge gap in the workforce, with older employees potentially feeling less confident in adopting new technologies.

Despite these reservations, there was a consensus among all interviewees that AI should serve as a complementary tool rather than a replacement for human expertise, providing a basis for team discussions by offering data-driven forecasts and feedback to support decision-making.

4.2 Forecast distributions and path refinement

The technical results illustrate how the hybrid predictive model improves project forecasting. Figure 1 illustrates the baseline Monte Carlo simulation, which generates a broad distribution of possible completion paths compared to the planned schedule. Figure 2 illustrates how the ANN refines these outcomes by highlighting the most probable trajectories, thereby significantly narrowing the completion window. As new weekly progress data are incorporated, forecasts continue to tighten and align closely with observed project performance, as illustrated in Figure 3.

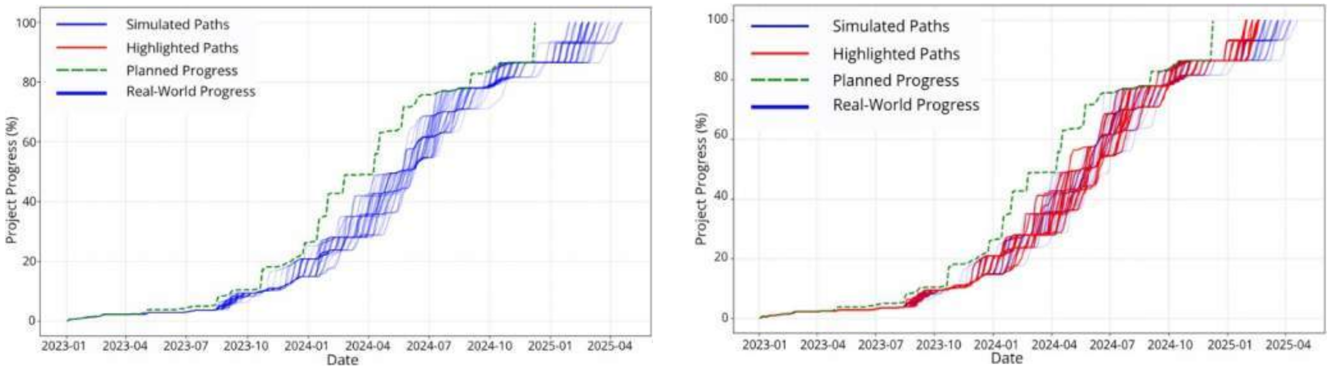

Figure 2. Prior-only forecast at project start (before any observed progress). Both subplots show the same Monte Carlo trajectories. Left: Monte Carlo only. Right: ML-ranked/selected trajectories from the same set. Blue: simulated paths; green dashed: planned schedule; red: ML-selected paths. Axes: time vs. project completion (%).

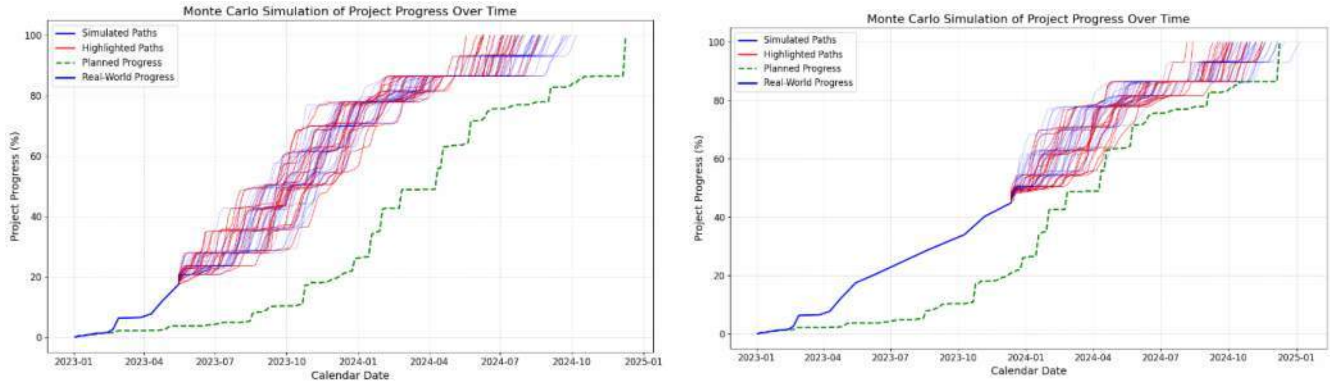

Figure 1. Model prediction and real-world progress at three successive update points (Week 22 and Week 49). Each plot shows Monte Carlo–simulated progress trajectories, with the ML-selected most probable paths highlighted in red.

4.3 Validation metrics & GPU-acceleration

Validation was conducted using a temporal holdout strategy: model components were calibrated using earlier progress updates, and performance was evaluated on unseen project data from later stages of execution. This provides an objective within-project assessment of generalization over time.

In figure 3 a small number of pronounced outliers are observed (high predicted values paired with 0-day recorded values). These are treated as anomalous cases in the current prototype; future work may address them through additional data auditing, task categorization, and model refinement.

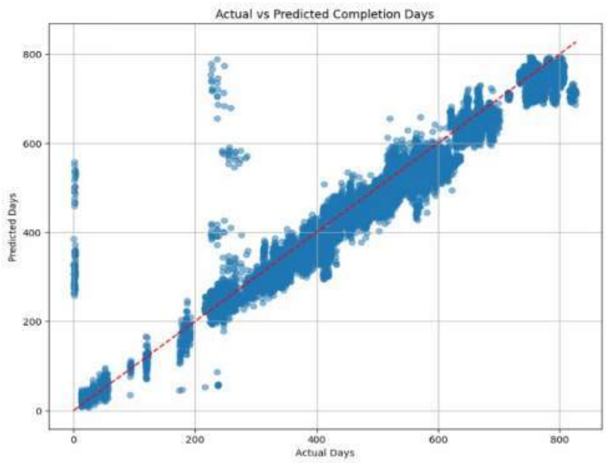

Figure 3. Actual versus predicted task completion times. The red dashed line represents perfect prediction zalignment, with points close to the line indicating high model accuracy.

Computational efficiency was critical: GPU acceleration reduced runtime for 100 iterations from 33 hours on CPU to only 5 minutes (~396×), enabling scenario testing at near real-time speed. Model accuracy is summarized in Figure 4, with R^2 = 0.92, MAE = 683 hours (28 days), RMSE = 1,124 hours (47 days), and a mean error of 11%. These results confirm the feasibility and practical value of hybrid forecasting for large construction projects.

Table 1. Validation metrics

Metrics [units]	Results
RMSE [Hours]	1 124.38
R^2 [Hours]	0.9203
RMSE [Days]	46.85
R^2 [Days]	0.9203
Mean Percent Error [Days]	11.01%

4.4 Design results

The design integrates Monte Carlo simulations with an Artificial Neural Network in a unified framework. A preprocessing pipeline ensures clean task data, while the Monte Carlo module generates possible project paths. The ANN then learns from these paths and weekly updates, refining forecasts over time. This setup enables the model to adapt continuously, providing project managers with scenario analysis and predictive insights.

5 Discussion

5.1 Data management

This project highlighted both the opportunities and challenges when it comes to implementing AI and machine learning in construction project management. Data management was the first challenge due to the fragmented and inconsistent nature of construction data. It requires the development of a partly automated data pipeline to manage preprocessing and update efficiently. This pipeline, designed for generalizability, not only improved efficiency but also laid a foundation for future reuse, reducing the time needed to develop similar models in other projects.

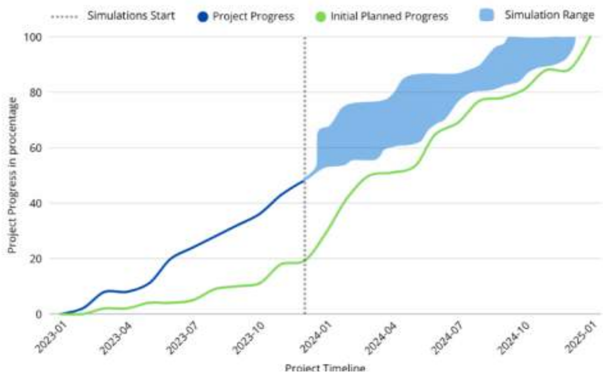

Figure 4. Final visualization showing actual progress vs. planned schedule, with the simulation range and ML-selected most probable outcomes.

5.2 Hybrid model efficiency

By combining Monte Carlo simulations, which generate a broad range of technically feasible project paths, with a machine learning model that refines these outputs through weekly updates, the approach adapts to evolving project conditions. This reduces reliance on extensive historical datasets, which are often unavailable or incomparable across projects. Initial results demonstrated high predictive accuracy (R2= 0.92), suggesting that the approach can deliver reliable forecasts that complement traditional management strategies.

With this being said, the effectiveness of the model remains highly dependent on the quality and consistency of input data. Unstructured and lack of data standardization will continue to be a hurdle for the adoption of new technologies in the construction industry. Addressing these issues will require a significant investment in data management across the industry. While results indicate strong predictive performance within the studied project, evaluating transferability across multiple projects and organizations is a key next step for future work.

5.3 Role and expectations of AI

While AI and ML hold great potential for project management, their effectiveness depends on the quality and quantity of data used for training. Unrealistic expectations can lead to overconfidence, and any misalignment between model predictions and professional judgment may be perceived as failure. Instead, such discrepancies should be viewed as opportunities to reflect, refine and explore alternative perspectives.

5.4 Future improvements

For future improvement, there are two immediate steps that can be taken: develop a user-friendly interface and implement an automated data cleaning and structure module. These approaches would minimize manual intervention and facilitate quicker onboarding of new users.

Several other highly impactful improvements can be made, such as implementing CPM to quickly pinpoint the most critical tasks, including resource allocation, workforce management and budget balancing.

To address concerns from interviewees, integration of external factors like seasonal variations and weather conditions would further enhance the accuracy and relevance of predictions.

The potentially most significant impact of further development would be to predict and analyse cost and time alterations due to corrective and additional work, both upstream and downstream.

6 Conclusion

This paper presents a machine learning model for forecasting project progress and identifying trends, utilising real-world data from the Ingelkärr-Stenkullen transmission line project. Initial results are promising, indicating that the construction industry can significantly benefit from adopting new technologies. However, model performance remains highly dependent on the quality and consistency of input data, highlighting the need for more structured data practices. Beyond technical development, interviews and sessions with industry professionals revealed a generally positive view of AI, provided that data inputs are transparent and supported by beginner-friendly user interfaces. By combining these aspects, the thesis demonstrates how AI and ML can be integrated as a supportive tool that enhances, rather than replaces, current project management practices.

7 Bibliography

[1] R. Fulford and C. Standing, "Construction industry productivity and the potential for collaborative practice," *International Journal of Project Management,* vol. 32, no. 2, p. 315–326, Jun. 2023.

[2] M. Rauf and H. H. Vu, "AI in Construction Management: Preparedness and Potential: A case study on implementing a predictive machine learning framework for construction project scheduling [Master's thesis]," Chalmers University of Technology, Gothenburg, 2025.

[3] J. M. Wilson, "Gantt charts: A centenary appreciation," *European Journal of Operational Research,* vol. 149, no. 2, p. 430–437, May 2003.

[4] K.-h. Kim, S.-y. Kim and J.-j. Kim, "CPM Bar Chart Technique for Construction Scheduling," *Proceedings of the 7th International Conference on Construction Applications of Virtual Reality,* p. 1118–1125, 2005.

[5] T. D. Akinosho, L. O. Oyedele, M. Bilal, O. A. Anuoluwapo, M. D. Delgado, O. O. Akinade and A. A. Ahmed, "Deep learning in the construction industry: A review of present status and future innovations," *Journal of Building Engineering,* vol. 32, p. 101827, Sep. 2020.

[6] M. H. Kalos and P. A. Whitlock, Monte Carlo methods, John Wiley & Sons, 2008.

[7] B. Mehlig, Machine Learning with Neural Networks, Cambridge University Press, 2021.