# Process-Aware Reward Modeling for Automated Construction Planning


**Akarsth Kumar Singh and Shang-Hsien Hsieh**

Department of Civil Engineering, National Taiwan University, Taiwan
akarsth@caece.net , shhsieh@ntu.edu.tw



**Abstract –**
**Construction planning remains error-prone because existing automated approaches primarily assess final schedules and lack mechanisms to evaluate intermediate planning decisions. As a result, early-stage errors in task decomposition, sequencing, or duration estimation can propagate unnoticed across planning stages. This study proposes a process-aware framework for automated construction planning that embeds planning-time supervision through a construction-specialized Process Reward Model (PRM). The framework integrates sequential multi-stage plan generation with rubric-based Small Language Model (SLM)-as-a-Judge evaluation to generate step-level quality signals, which are used to train the PRM to predict the quality of partial planning states. Experiments on real-world building construction data show that the PRM achieves 78% agreement with human-verified evaluations while reducing evaluation cost by approximately 50× and inference time by more than 5× compared to SLM-as-a-Judge assessment. The results indicate that process-level reward modeling provides scalable and efficient quality assurance for structured planning elements such as work breakdown consistency and resource feasibility, while serving as a pre-screening mechanism for complex dependency reasoning.**



**Keywords –**
**Automated Construction Planning, Process Reward Model, SLM-as-a-Judge, Process-Level Evaluation, Large Language Model**


## 1 Introduction

Construction projects worldwide continue to experience significant delays and cost overruns, largely due to inadequate planning and scheduling, limited resource availability, and inefficient decision-making processes [1][2]. Developing a reliable construction schedule requires identifying activities from extensive contract documents and transforming them into structured planning elements. This process is typically performed manually, making it time-consuming, inconsistent, and susceptible to human error [3][4]. Although conventional project management software supports schedule generation, it provides little capability to assess planning quality during the progressive decomposition of complex construction scopes into executable work packages.

Recent advances in construction informatics have explored the use of large language models (LLM) and natural language processing (NLP) for activity extraction from textual documents [3][5], knowledge graphs for adaptive work packaging and reasoning [6], and machine learning techniques for schedule enhancement and optimization [7]. While these approaches improve automation and efficiency, they primarily focus on extracting information or validating final schedules. They do not explicitly assess intermediate planning decisions, including task decomposition, dependency formulation, and duration estimation.

In parallel, process reward models (PRMs) have recently shown strong potential in improving reasoning performance in domains such as mathematics [8], financial analysis [9], and question-answering systems [10]. Unlike conventional reward mechanisms that evaluate only final outputs, PRMs assess the quality of intermediate reasoning steps. Domain-specific PRMs, such as Fin-PRM, further demonstrate that embedding expert knowledge and procedural constraints leads to substantial performance gains [9]. Despite this progress, no PRM framework has been developed for construction planning, a domain characterized by sequential, interdependent decisions that must satisfy constructability rules, resource constraints, and contractual requirements.

This gap is particularly critical because construction planning errors tend to propagate across multiple stages, including work breakdown structure development, task sequencing, duration estimation, and resource allocation. General-purpose language models lack the domain knowledge required to reliably evaluate intermediate planning states against industry practices and project-specific constraints [11][12]. As a result, automated planning systems remain vulnerable to infeasible or low-quality schedules.

To address this limitation, this study proposes a process-aware framework for automated construction planning that integrates sequential plan generation, domain-calibrated LLM-as-a-Judge evaluation, and Process Reward Model (PRM) training. Construction planning is decomposed into four stages—work breakdown structure generation, task relationship identification, duration estimation, and resource allocation—evaluated using construction-specific rubrics to produce step-level quality signals that guide generation. Validation on real-world building projects shows that the approach (i) provides construction-specialized process-level supervision for planning, (ii) enables planning-time quality assurance rather than post-hoc validation, and (iii) supports transferable planning knowledge across similar projects. The study further introduces a hybrid evaluation framework in which the PRM performs planning-time screening of structured planning logic, while complex dependency reasoning is selectively validated using higher-fidelity evaluators.

# 2 Related Work

## 2.1 Automated Construction Planning

Construction planning remains largely dependent on manual expert judgment and semi-structured representations [4]. A three-decade review showed that although automated methods for scope definition, WBS generation, and sequencing exist, most projects still rely on manual workflows due to inflexible knowledge representations, dependence on manually maintained templates, limited learning from historical data, and weak integration between planning and optimization [7]. Earlier rule-based and optimization approaches, including genetic algorithms, require predefined task networks, limiting generalizability [13]. BIM-based 4D planning remains labor-intensive to maintain [14], while knowledge graph approaches improve semantic work packaging but rely on explicitly constructed graphs and do not infer implicit constructability logic from documents [6].

## 2.2 Large Language Models in Construction

NLP and LLMs have been applied to construction documents with high accuracy in activity and procedural information extraction [3][5][18], but these methods focus on retrieval rather than complete schedule generation. Learning-based approaches have modeled task dependencies and improved scheduling performance [7][12]. Multi-LLM pipelines with SLM-as-a-judge evaluation have also been proposed [11]; however, supervision is limited to synthetic WBS and construction description generation and post-hoc evaluation of final outputs, leaving intermediate planning stages unsupervised.

## 2.3 Process Reward Models

PRMs evaluate intermediate reasoning steps rather than only final outputs and have demonstrated notable performance gains in structured domains. Trajectory-aware PRMs improve mathematical reasoning by incorporating step-level supervision [8], while domain-specialized models such as Fin-PRM enhance financial reasoning through procedural and factual verification [9]. Generative PRMs further strengthen step-wise evaluation using chain-of-thought verification with limited supervision [16][10].Despite these advances, existing PRMs assume well-defined correctness criteria and structured reasoning spaces. Construction planning, by contrast, involves constraint satisfaction across constructability logic, resource feasibility, document alignment, and industry standards, which cannot be directly addressed using current PRM formulations.

## 2.4 LLM-as-a-Judge Evaluation

LLM-as-a-judge approaches enable rubric-based, domain-specific evaluation beyond generic accuracy metrics and correlate strongly with expert judgment when customized criteria are applied [17]. In construction contexts, rubric-guided evaluation has been used to assess planning completeness, hierarchical consistency, and task relationship correctness, including SLM-as-a-judge assessment [11][18]. However, these approaches operate post-hoc, requiring additional inference passes after plan generation, introducing computational overhead and preventing real-time feedback. In contrast, the proposed framework embeds a construction-specialized PRM directly within the planning loop, enabling planning-time guidance of intermediate decisions and mitigating error propagation across planning stages.

## 2.5 Research Gap

Despite advances in LLM-based planning, construction planning lacks a process-level supervision framework that validates intermediate decisions during generation. Existing approaches emphasize final-output evaluation, allowing decomposition and sequencing errors to propagate unchecked. Current PRMs are limited to domains with explicit correctness criteria, restricting their applicability to construction planning. This study addresses this gap by introducing a construction-specialized process reward modeling framework that incorporates explicit evaluation of intermediate planning decisions and integrates quality guidance throughout the reasoning process.

## 3 Methodology

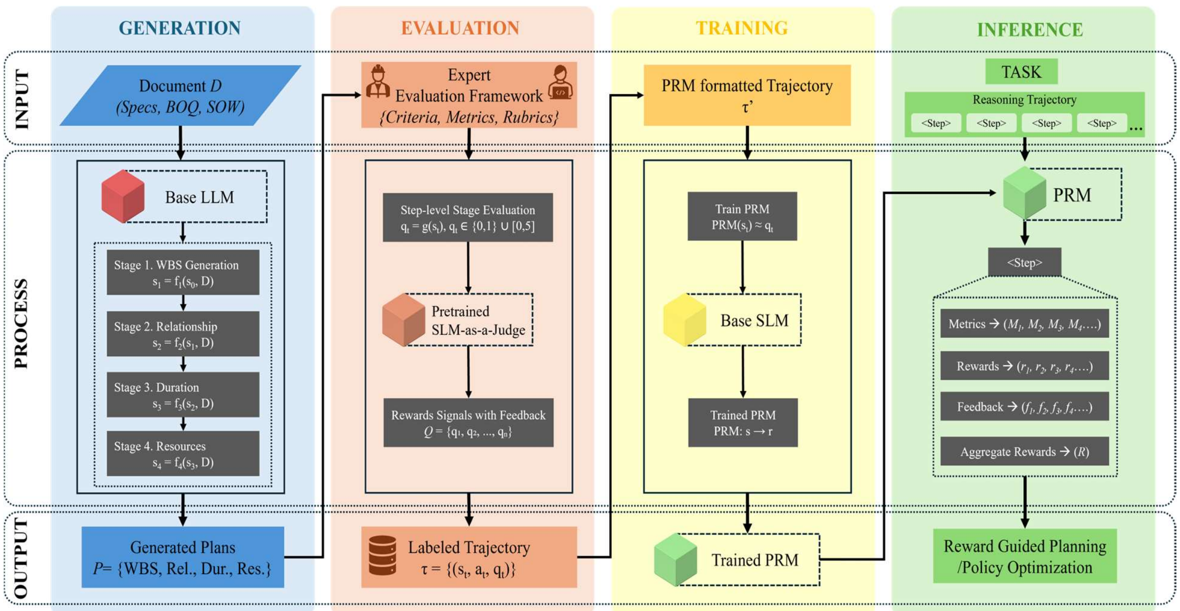


Figure 1. Overview of Methodology

This study proposes, as shown in Fig. 1, a process-aware framework for automated construction planning that integrates sequential plan generation, intermediate quality evaluation, and learning-based guidance. The framework operates in three stages: (1) sequential construction planning generation, (2) intermediate evaluation using a large language model acting as a judge, and (3) PRM training. Construction inputs are represented as the document set $D$ defined in Eq. (1), while the generated construction plan $P$ is defined in Eq. (2). Stage-level quality signals produced during planning are denoted by $Q$ in Eq. (3). The PRM is formulated as a mapping from partial planning states to predicted quality rewards, as shown in Eq. (4).

$$D = \{\text{specifications,BOQ,scope}\} \tag{1}$$

$$P = \{WBS, Relationship, \text{Duration,Resources}\} \tag{2}$$

$$Q = \{q_1, q_2, \dots, q_n\} \tag{3}$$

$$\text{PRM}: s \rightarrow r \tag{4}$$

### 3.1 Sequential Construction Planning

Construction planning is formulated as a sequence of interdependent decision stages reflecting standard project workflows. Rather than generating a complete schedule in a single step, the framework incrementally transforms a partial planning state $s_t$ into a refined state $s_{t+1}$ using a stage-specific planning operator, as expressed in Eq. (5).

$$s_{t+1} = f_t(s_t, D) \tag{5}$$

This sequential formulation exposes intermediate planning states and enables targeted evaluation and correction across work breakdown formulation, task relationship reasoning, duration estimation, and resource allocation stages.

### 3.2 Intermediate Evaluation via SLM-as-a-Judge

Intermediate planning states are evaluated using a domain-calibrated LLM acting as a judge. For each planning stage t, the evaluator assigns quality signals using two instruments: binary scores $q_t \in \{0,1\}$ for structural validation checks (e.g., schema compliance, field presence), and 5-point Likert scores $q_t \in \{1,\dots,5\}$ for reasoning dimensions admitting partial credit. The stage-level signal is the mean across all applicable metrics within a step, with Likert scores linearly rescaled to [0,1], yielding a normalized composite reward $q_t \in [0,1]$ as defined in Eq. (6).

$$q_t = \mathrm{g}(s_t), \mathrm{q_t} \in \{0,1\} \text{ or } [0,5] \tag{6}$$

Each scale point is anchored to a domain-specific behavioral descriptor; for example, for Sequence Logic Explained: 1 = no sequencing rationale provided; 5 = explicit constructability justification for every task link (e.g., formwork must precede concrete pour as it defines the structural mould). In addition to the scalar quality signal, the evaluator generates structured textual feedback identifying logical inconsistencies, feasibility issues, and misalignment with the source construction context/documents; this feedback is used for analysis and interpretation but does not constitute the reward signal.

### 3.3 Process Reward Model Learning

Outputs from the evaluation stage are aggregated into labeled planning trajectories, where each trajectory captures the partial planning state, the corresponding planning action, and the assigned quality signal, as formalized in Eq. (7).

$$\tau = \{(s_t, a_t, q_t)\}_{t=1}^{T} \tag{7}$$

The PRM is trained to approximate the evaluation function by predicting the quality signal associated with a given partial planning state, as shown in Eq. (8).

$$\mathrm{PRM}(s_t) \approx q_t \tag{8}$$

During planning, the PRM guides decision-making by scoring candidate actions and selecting the action expected to yield the highest-quality planning outcome, as expressed in Eq. (9).

$$a_t^* = \arg\max_{a \in A_t} \mathrm{PRM}(s_t \oplus a) \tag{9}$$

Critically, this selection mechanism relies on ordinal ranking rather than precise cardinal scoring. The PRM is not required to predict absolute quality values, but only to consistently rank candidate actions such that an action $a_1$ leading to a higher-quality planning state is preferred over $a_2$. This distinction between relative comparison and absolute evaluation is central to the effectiveness of PRM-guided construction planning. Unlike SLM-as-a-Judge evaluation, which is applied after plan generation, the PRM is queried during the planning process to score candidate actions and guide selection toward higher-quality intermediate states. This guidance mechanism enables early suppression of low-quality planning paths and improves robustness and interpretability.

## 4 Experimental Setup

### 4.1 Dataset Construction

The experimental dataset is derived from a real-world high-rise residential building project comprising a reinforced concrete (RCC) framed structure of 22 floors above ground, with more than 2,500 scheduled activities spanning structural works, all internal and external architectural finishing, and full mechanical, electrical, plumbing, and fire protection (MEPF) systems. The majority of these activities represent repetitions of the same work types across different floors and locations. To avoid redundancy and prevent overfitting the PRM on structurally identical samples, 243 unique activity types were selected, capturing the minimum sufficient set of distinct planning scenarios across all domains. Selected activities were manually reviewed for completeness, clarity, and representativeness, producing a high-quality seed dataset for generation and evaluation.

### 4.2 Sequential Planning Generation

For each activity, synthetic Bill of Quantities (BOQ) detailed descriptions were generated following the procedure established in prior work [11]. These BOQ descriptions serve as the primary input documents for the multi-stage planning generation process.

At each planning stage, the system generates step-wise reasoning traces structured as ⟨Thought, Analysis, Decision⟩ triplets, followed by a stage-specific structured output that conforms to a predefined schema. This explicit reasoning decomposition exposes intermediate decision-making processes and enables targeted evaluation of planning logic at each stage. The complete generation workflow is formalized in Algorithm 1.

**Algorithm 1: Multi-Stage BOQ Processing with Structured Reasoning**

**Input**: Project data D, BOQ items B, prompt set Π, environment E

**Output**: Structured representations per BOQ item

```
1: Initialize LLM model and client using E
2: Load BOQ items B and project context Cproj from P

3: for each b ∈ B do                                   ▷ Item level
4:     Cb ← ComposeContext(b, C_proj)
5:     x ← ⊥                                           ▷ Initial state
6:     for each stage s∈⟨WBS, REL, DUR, RES⟩ do        ▷ Stage level
7:     T ← GenerateReasoning (s, b, x, Cb)             ▷ Thought
8:     A ← Derive Intermediate (T, s, Cb)              ▷ Analysis
9:     x ← Select Structured Output (A, s)             ▷ Decision
10:        Store (b, s, x)
11:    end for
12: end for
```

### 4.3 Step-Wise Evaluation with SLM-as-a-Judge

Each intermediate reasoning step produced during planning is evaluated using a rubric-based assessment framework calibrated for construction planning tasks. The evaluation combines binary validation metrics (0 or 1) with fine-grained Likert-scale scores (0–5). The framework quantifies reasoning quality across four dimensions applied in Algorithm 2 (lines 8–10 per reasoning step; lines 12–14 for structured output validation): (1) constructability logic via Likert metrics such as Sequence Logic Explained and Lag Time Justification; (2) domain knowledge correctness via binary checks verifying that task names, relationship types (FS/SS/FF/SF), and resource selections conform to industry norms; (3) hierarchical consistency via metrics such as L1–L7 Reasoning Shown and 100% Rule

Verified; and (4) document alignment via traceability checks linking extracted activities and quantities to the input BOQ.

Evaluation is conducted at three levels: (1) step-level assessment of individual ⟨Thought, Analysis, Decision⟩ triplets, (2) validation of stage-specific structured outputs against schema constraints, and (3) cross-stage consistency verification. For each evaluation instance, the judge model returns both a numerical score and explanatory feedback identifying specific strengths or deficiencies in the reasoning trace. The complete evaluation procedure is formalized in Algorithm 2.

**Algorithm 2: Multi-Stage Reasoning Evaluation with SLM-as-a-Judge**

**Input**: Stage outputs S, BOQ data B, metrics M, stage config Q, judge J
**Output**: Quality scores R per stage per sample

```
Load metrics M and quality config Q
for each stage σ ∈ ⟨WBS, REL, DUR, RES⟩ do
    Load stage outputs Sσ and BOQ data Bσ
    for each sample si ∈ S σ do
        Ci ← BuildContext (B σ , σ )
        for each reasoning step rj ∈ si do
            Tj ← Merge (rj.thought, rj.analysis, rj.decision)
            for each metric mk ∈ M[ σ ][rj] do
                R[ σ ][si][rj][mk] ← J.Evaluate (mk, Q[ σ ][rj], Ci, Tj)
            end for
        end for
        for each validation metric mv ∈ M[ σ ][val] do
            R[ σ ][si][val][mv] ← J.Evaluate(mv, Q[ σ ][val], Ci, si.output)
        end for
        PersistResults(R[σ][si])
    end for
end for
return R
```

The resulting evaluation dataset consists of labeled planning trajectories with dense, step-wise reward signals. These data support supervised training of PRM that estimate intermediate planning quality and guide future generations toward higher-quality, logically consistent, and feasible construction plans.

## 4.4 PRM Model Training

The labeled planning trajectories generated by the SLM-as-a-Judge evaluation are used to train the PRM. Each trajectory consists of partial planning states, corresponding ⟨Thought, Analysis, Decision⟩ reasoning steps, and judge-assigned quality scores. The PRM is trained to predict quality signals from partial planning states by minimizing the error between predicted rewards and judge-provided scores. Inputs to the model include the construction context and BOQ descriptions, the planning stage identifier, the accumulated planning state, and the current reasoning step. The model outputs scalar reward predictions aligned with the evaluation scale, using binary values (0 or 1) for validation metrics and discrete scores (0–5) for Likert-scale assessments.

Table 1. PRM training configuration

| Item | Value |
|---|---|
| Model | Qwen2.5-1.5B-Instruct |
| Adaptation | LoRA (r=16, α=16, dropout=0) |
| Seq. length | 2048 |
| Optimizer | AdamW (lr=2e-4) |
| Batch / Epochs | 2 / 3 |
| Loss | MSE |

The PRM is fine-tuned using parameter-efficient adaptation. Key training hyperparameters, including LoRA configuration, learning rate, batch size, and optimization settings, are summarized in Table 1. This setup enables efficient training while preserving the base model's general reasoning capabilities.

# 5 Results

## 5.1 Evaluation Setup

The PRM was evaluated against SLM-as-a-Judge reference scores across 251 matched samples spanning four construction planning stages: work breakdown structure generation, task relationship identification, duration estimation, and resource allocation. The SLM-as-a-Judge framework was used to generate reference scores, which were subsequently verified by the authors to ensure consistency and reliability. This combined automated and manual validation establishes a robust ground truth for PRM performance evaluation.

## 5.2 Overall Performance

Table 2 reports the overall agreement between PRM predictions and the manually verified SLM-as-a-Judge scores. The PRM achieved 78% accuracy with near-zero mean bias (−0.05), indicating it reliably captures relative quality differences between planning candidates without systematic over- or under-estimation. The moderate correlation ($r = 0.515$) confirms the model ranks alternatives consistently with human judgment.

For reward-guided generation (Eq. 9), this ranking capability is more critical than absolute scoring accuracy: the PRM need only identify that candidate A is superior to candidate B, not assigning objectively correct scores to either.

Table 2. Overall performance metrics

| Metric | Value |
|---|---:|
| Samples (n) | 251 |
| Accuracy | 78% |
| Pearson correlation (r) | 0.515 |
| MAE | 0.887 |
| RMSE | 1.159 |
| Within ±1.0 | 65.7% |
| Mean bias | −0.05 |

The minimal bias demonstrates the PRM fulfills this comparative function, enabling effective guidance away from low-quality planning paths during iterative generation.

## 5.3 Stage-wise Performance Analysis

Figure 2 presents a radar comparison of PRM and SLM-as-a-Judge scores across the four planning stages. The PRM shows strongest agreement for WBS generation and resource allocation, with correlations exceeding 0.5 in both stages. In WBS generation, the model performs reliably on scope interpretation and hierarchical structuring, indicating effective evaluation of structured decomposition tasks. Similarly, strong agreement in resource allocation reflects consistent assessment of labor crew composition and equipment selection.

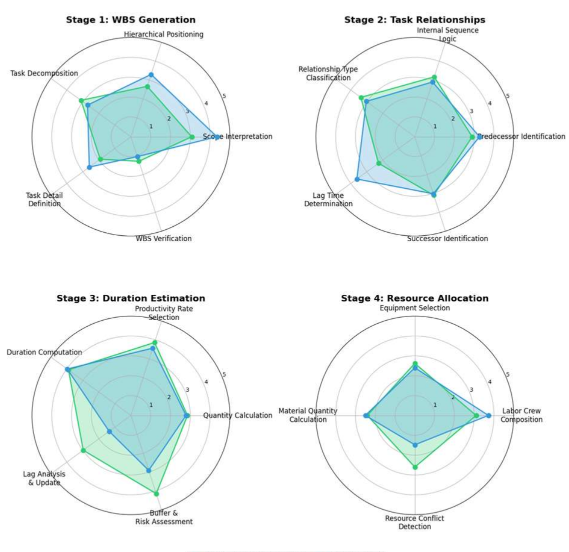


Figure 2. Stage-wise comparison of PRM and SLM-as-a-Judge evaluation scores.

In contrast, task relationship evaluation exhibits the weakest agreement, with near-zero correlation, highlighting challenges in assessing complex task relationship logic, predecessor–successor reasoning, and sequencing rationale. This stage requires multi-step causal reasoning and implicit constructability knowledge, which remains difficult for the current PRM formulation. Duration estimation shows moderate correlation, with the PRM exhibiting conservative scoring tendencies relative to the SLM baseline, particularly for buffer and risk-related considerations.

## 5.4 Cost–Accuracy and Cost–Time Trade-Offs

The accuracy–cost relationship between PRM and SLM-as-a-Judge is shown in Figure 3. While SLM-as-a-Judge achieves perfect agreement with the verified ground truth, it incurs substantially higher per-sample cost. In contrast, the PRM achieves 78% accuracy at a cost of approximately \$0.003 per sample, representing an approximate 50× cost reduction. Cost estimates are based on online GPU pricing for an RTX 3090 (24 GB VRAM, 128 GB RAM) configuration.

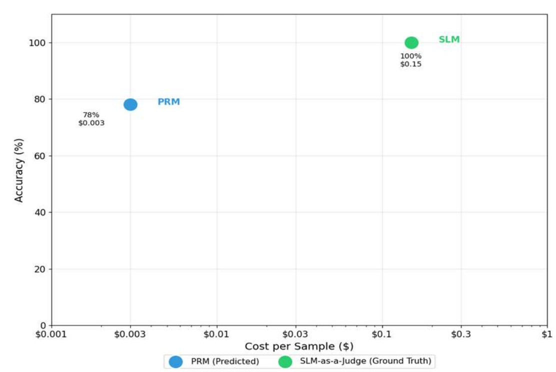


Figure 3. Accuracy–cost trade-off between PRM and SLM-as-a-Judge evaluation

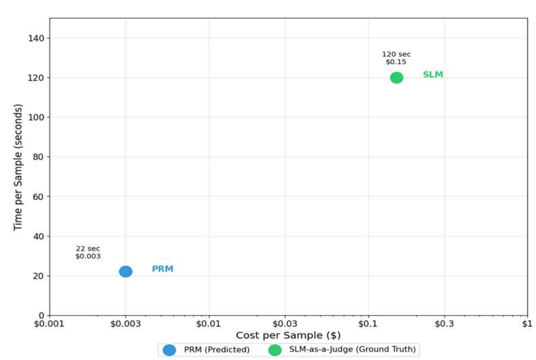


Figure 4. Cost–time efficiency comparison of PRM and SLM-as-a-Judge evaluation

Computational efficiency is compared in Figure 4. PRM inference requires approximately 22 seconds per sample, compared to 120 seconds for SLM-as-a-Judge, resulting in a 5.5× speed-up and an order-of-magnitude reduction in total evaluation cost under the same hardware assumptions. These results demonstrate the practical advantages of PRM for scalable construction planning assessment.

## 6 Discussion

The results indicate a deployment-aware performance profile rather than a fundamental limitation of the proposed approach. The Process Reward Model (PRM) demonstrates strong agreement for hierarchical planning tasks, including work breakdown structure generation and resource allocation, while exhibiting substantially weaker performance for task relationship evaluation. This divergence reflects differences in underlying reasoning requirements across planning stages.

Task relationship evaluation exposes a fundamental representational boundary of text-based PRMs. While linguistic representations support local and hierarchical reasoning, they do not explicitly encode the global network structure required to enforce precedence, concurrency, and constructability constraints across an activity graph. Multiple dependency configurations may appear locally plausible in text, but only a subset satisfies global feasibility. The near-zero correlation therefore reflects inconsistent scoring of locally valid but globally distinct dependency configurations due to missing structural information, rather than random model behavior.

This stage-wise performance profile motivates a tiered evaluation strategy in which the PRM is deployed as a first-stage screening and guidance mechanism rather than a final certification authority. The PRM's strength lies in comparative ranking of planning candidates based on hierarchical consistency and feasibility, enabling early filtering of low-quality plans. More complex dependency reasoning, which requires explicit topological constraints, can then be addressed using higher-fidelity evaluation methods capable of enforcing global network logic. In this role, the PRM enables focused application of detailed validation within large-scale automated planning pipelines.

Critically, within the iterative planning process of Algorithm 1, absolute correctness of intermediate evaluations is less important than relative improvement across candidate actions. The PRM guides selection by estimating which candidate is likely to yield a higher-quality planning state, rather than certifying global schedule correctness. Even when exact agreement on task relationships is limited, the PRM fulfills its role by consistently identifying inferior candidates and steering generation away from infeasible reasoning paths. This reliance on ordinal ranking rather than precise cardinal scoring, together with minimal systematic bias, explains why the PRM remains effective as a reward-guided planning signal despite imperfect dependency grading.

The PRM further supports this role by requiring substantially fewer computational resources and enabling significantly faster inference than SLM-as-a-Judge evaluation under identical hardware assumptions. This efficiency allows quality assessment to be embedded directly into the planning loop, supporting early suppression of low-quality reasoning paths and iterative refinement of candidate plans.

These findings directly inform future research directions. Improving task relationship evaluation will require incorporating graph-structured representations of task networks and explicit constructability constraints into the PRM input space. Additional opportunities include integrating PRM-based guidance with rule-based schedulability validators and applying active learning strategies to expand supervision in regions where PRM and high-fidelity evaluations diverge. Together, these directions point toward hybrid evaluation architectures that combine linguistic reasoning, structured representations, and deterministic constraints for robust automated construction planning.

## 7 Conclusion

This study introduces a process-aware framework for automated construction planning that embeds construction-specialized supervision through a Process Reward Model. The primary contribution lies in enabling planning-time quality assessment across intermediate planning stages, including work breakdown structure generation, task relationship formulation, duration estimation, and resource allocation, thereby supporting early identification and correction of planning errors before they propagate.

The framework's key contribution is the explicit alignment of evaluation strategy with representational requirements of different planning tasks. By recognizing that hierarchical decomposition and topological sequencing demand different forms of supervision, the proposed approach positions the PRM as an effective first-stage screening mechanism within a tiered evaluation architecture, while reserving dependency validation for methods capable of enforcing global network constraints.

Beyond immediate performance outcomes, this work clarifies both the strengths and boundaries of text-based intermediate supervision in construction planning and demonstrates how these boundaries can inform system design. Future work will focus on developing graph-augmented PRM architectures, expanding evaluation across diverse construction domains, incorporating active learning strategies, and integrating learned rewards with deterministic schedulability validation in hybrid planning frameworks and extending the framework to support multi-domain schedule integration and project-level duration computation. Collectively, these directions advance the practical realization of reliable, domain-aware, AI-assisted construction planning systems.

## Acknowledgements

The authors thank the National Science and Technology Council, Taiwan, for supporting this research through research grant Nos. NSTC 114-2119-M-002-034.

## References

[1] M. O. Sanni-Anibire, R. Mohamad Zin, and S. O. Olatunji. Causes of delay in the global construction industry: a meta-analytical review. International Journal of Construction Management, 22(8):1395–1407, 2022.

[2] Y. J. T. Zidane and B. Andersen. The Top 10 Universal Delay Factors in Construction Projects. International Journal of Managing Projects in Business, 11(3):650–672, 2018.

[3] F. U. Hassan and T. Le. Extraction of Activities Information from Construction Contracts Using Natural Language Processing (NLP) Methods to Support Scheduling. In Construction Research Congress 2022: Computer Applications, Automation, and Data Analytics - Selected Papers from Construction Research Congress 2022, volume 2-B, pages 773–781, 2022.

[4] Y. Hong, H. Xie, G. Bhumbra, and I. Brilakis. Improving the accuracy of schedule information communication between humans and data. Advanced Engineering Informatics, 53:101645, 2022.

[5] R. Ren and J. Zhang. Semantic Rule-Based Construction Procedural Information Extraction to Guide Jobsite Sensing and Monitoring. Journal of Computing in Civil Engineering, 35(6):04021026, 2021.

[6] X. Li, C. Wu, Z. Yang, Y. Guo, and R. Jiang. Knowledge graph-enabled adaptive work packaging approach in modular construction. Knowledge-Based Systems, 260:110115, 2023.

[7] F. Amer, Y. Jung, and M. Golparvar-Fard. Construction schedule augmentation with implicit dependency constraints and automated generation of lookahead plan revisions. Automation in Construction, 152:104896, 2023.

[8] J. Zou, L. Yang, J. Gu, J. Qiu, K. Shen, J. He, and M. Wang. ReasonFlux-PRM: Trajectory-Aware PRMs for Long Chain-of-Thought Reasoning in LLMs. arXiv preprint arXiv:2506.18896, 2025.

[9] Y. Zhou, S. Jiang, J. Zhu, J. Li, L. Guo, F. Chen, and C. Zhang. Fin-PRM: A Domain-Specialized Process Reward Model for Financial Reasoning in Large Language Models. arXiv preprint arXiv:2508.15202, 2025.

[10] S. She, J. Liu, Y. Liu, J. Chen, X. Huang, and S. Huang. R-PRM: Reasoning-Driven Process Reward Modeling. In Proceedings of EMNLP 2025, pages 13449–13462, 2025.

[11] A. K. Singh and S. H. Hsieh. Multi-LLM-based augmentation and synthetic data generation of construction schedules and task descriptions with SLM-as-a-judge assessment. Advanced Engineering Informatics, 69:103825, 2026.

[12] Y. Zhang and X. Yang. CONSTRUCTA: Automating Commercial Construction Schedules in Fabrication Facilities with Large Language Models. arXiv preprint arXiv:2502.12066, 2025.

[13] V. Faghihi, A. Nejat, K. F. Reinschmidt, and J. H. Kang. Automation in construction scheduling: a review of the literature. International Journal of Advanced Manufacturing Technology, 81(9-12):1845–1856, 2015.

[14] A. Pal, J. J. Lin, F. Amer, S. Hsieh, and M. Golparvar Fard. Automatic mapping of schedule activities and reality models for tracking construction progress. Engineering, Construction and Architectural Management, pages 1–27, 2025.

[15] M. Khalifa, R. Agarwal, L. Logeswaran, J. Kim, H. Peng, M. Lee, H. Lee, and L. Wang. Process Reward Models That Think. arXiv preprint arXiv:2504.16828, 2025.

[16] S. Kim, J. Shin, Y. Cho, J. Jang, S. Longpre, H. Lee, S. Yun, S. Shin, S. Kim, J. Thorne, and M. Seo. Prometheus: Inducing Fine-Grained Evaluation Capability in Language Models. In 12th International Conference on Learning Representations, ICLR 2024, pages 1–37, 2024.

[17] F. U. Hassan, T. Le, and C. Le. Automated Approach for Digitalizing Scope of Work Requirements to Support Contract Management. Journal of Construction Engineering and Management, 149(4), 2023.

[18] X. Zhao, K. Yeoh, and D. K. H. Chua. Extracting Construction knowledge from Project Schedules by using Natural Language Processing. In Lecture Notes in Mechanical Engineering, pages 371–381. Springer, 2020.