Advanced Frameworks for Large Language Models in CAD Education: Evaluating and Enhancing Academic Problem Solving

Xiang Li[1], Ling Chen[2], Guang Feng[3], Sara McMains[4]

[1]Research Institute of Advanced Materials, CISRI Group, xli@berkeley.edu
[2]China Railway Siyuan Survey and Design Group, hust_chenling@outlook.com
[3]Research Institute of Advanced Materials, CISRI Group, hb0102fengguang@163.com
[4]UC Berkeley, mcmains@berkeley.edu

Corresponding author: Xiang Li, xli@berkeley.edu

Abstract. Large Language Models (LLMs) have rapidly advanced research and reshaped educational strategies across numerous fields. However, their proficiency in specialized, interdisciplinary domains such as Computer-Aided Design (CAD) is not yet fully understood. This paper systematically evaluates five state-of-the-art LLMs on CAD academic problems spanning conceptual understanding, mathematical theory, coding, and comprehensive application-based tasks. While LLMs outperform graduate students on foundational questions, they struggle with comprehensive problems requiring multi-step reasoning and domain-specific contextualization. To address these deficits, we propose two tailored frameworks based on knowledge refinement and Chain-of-Thought (CoT) prompting, that significantly boost LLM performance from around 36 to over 90 points on these complex tasks. These findings highlight the untapped potential of LLMs as powerful tools in CAD education, automating routine assignments and supporting deeper learning. They also underscore the importance of carefully designed curricula and pedagogical strategies that balance automated solutions with rigorous conceptual mastery, ensuring students acquire both technical proficiency and core disciplinary understanding.

Keywords: large language models (LLMs), computer-aided design (CAD), CAD education
DOI: https://doi.org/10.14733/cadaps.2026.398-410

1 INTRODUCTION

The rapid advancement of Large Language Models (LLMs) has significantly accelerated innovation in scientific research, technology development, and data analysis [8, 9], while concurrently reshaping established educational paradigms. These models excel in solving academic problems by employing sophisticated reasoning capabilities and an extensive knowledge base. Consequently, LLMs have emerged as valuable tools in education [2, 4], providing insights, generating solutions, and facilitating personalized learning pathways in disciplines that range from the humanities to complex scientific domains [1, 10]. Despite their demonstrated potential, the application of LLMs to highly specialized and interdisciplinary subjects remains underexplored.

Computer-Aided Design (CAD) exemplifies such a specialized domain. Its inherent complexity arises from the integration of diverse knowledge areas, including mathematics, algorithms, data structures, and manufacturing processes. Academic problems in CAD often extend the boundaries of conventional learning, demanding rigorous problem-solving skills and a deep understanding of both theoretical and practical aspects [12]. As such, CAD tasks represent a particularly challenging benchmark for both students and LLMs. Yet, to our knowledge, no systematic evaluation has been undertaken to assess the proficiency of LLMs in tackling CAD academic problems.

This paper addresses this gap by making a two-fold contribution. First, we present a systematic evaluation of five state-of-the-art LLMs on real-world CAD academic problems. These problems are carefully categorized into four areas: conceptual understanding, mathematical theory, coding/algorithmic skills, and comprehensive application-based problems. This categorization not only reflects the multifaceted nature of CAD but also serves to assess both the foundational and integrative problem-solving capabilities required in this domain.

Second, our study identifies specific challenges that LLMs face, particularly with comprehensive application-based problems that require multi-step reasoning and domain-specific contextualization. In response, we introduce two tailored, two-phase frameworks: the knowledge refinement framework and the Chain-of-Thought (CoT) framework, which enhance LLM performance on these complex tasks. These frameworks aims to mitigate the observed deficits and improve the overall reliability of LLM-based problem solving in specialized educational settings.

In this research, we aim to answer several key questions: How effectively do current LLMs perform on CAD academic problems across different domains? What are the primary challenges that these models encounter, especially in tasks requiring integrated reasoning? And, to what extent can targeted frameworks improve their performance in addressing these challenges? By exploring these questions, our study seeks to provide a deeper understanding of both the capabilities and limitations of LLMs within the specialized context of CAD education, while also offering insights that could inform future curriculum design and pedagogical strategies.

2 EVALUATION SETUP

2.1 Academic CAD Questions

The test questions were collected from real-world academic CAD problems aimed at graduate students, encompassing homework and exam challenges from institutions such as UC Berkeley, University of Cincinnati, Tianjin University, and the China Iron and Steel Research Institute. These questions span a broad range of topics and necessitate advanced problem-solving skills across multiple domains. They are organized into four distinct categories, each designed to evaluate different facets of CAD knowledge and reasoning:

1. **Conceptual Understanding:** This section consists of 30 multiple-choice questions and 30 Q&A problems, designed to evaluate how well students comprehend representations, foundational geometric theories, and computational geometry in CAD contexts. It assesses how well students grasp core principles and concepts before applying them to problem-solving.

> **Example Question – Conceptual Understanding**
>
> Consider a CAD system that uses boundary representation (B-rep) for modeling solids. Which of the following issues is most commonly associated with numerical precision errors during B-rep operations?

(a) Self-intersecting surfaces (b) Non-manifold edges
(c) Gaps and overlaps between faces (d) Excessive polygon counts

2. **Math Problems:** This section consists of 30 multiple-choice questions and 30 Q&A problems, designed to assess analytical skills and quantitative reasoning. It focuses on the mathematical aspects of geometric concepts, algorithmic complexity, and constraint solving within CAD.

Example Question – Math Problem

Consider a 4×4 homogeneous transformation matrix used in 3D CAD systems to represent affine transformations. Which of the following properties is guaranteed to be preserved by any affine transformation?

(a) Euclidean distance between any two points

(b) The ratio of distances between points lying on the same line

(c) Angles between vectors

(d) Orthogonality of coordinate axes

3. **Coding/Algorithm:** This section consists of 30 multiple-choice questions and 30 Q&A problems, designed to assess hands-on coding proficiency and algorithmic thinking. It encompasses mesh generation, constraint resolution, CAD software scripting, and related data structures, testing the ability to design, implement, and optimize algorithms relevant to CAD tasks.

Example Question – Coding/Algorithm

Develop a program using the Quickhull algorithm to compute the convex hull of a set of 2D points. Your program should take as input a list of points (each represented as an (x, y) tuple) and output the vertices of the convex hull in clockwise order, starting from the leftmost point. Use the following set of points as your test case:

$$\{(0,0),(2,1),(1,3),(3,3),(4,0),(2,2)\}$$

What is the output of your program (i.e., the list of convex hull vertices) for this test case?

4. **Comprehensive Application-Based Problems:** This section consists of 20 Q&A problems (no multiple-choice questions), designed to evaluate a comprehensive integration of mathematical theory, programming, conceptual reasoning, and applied knowledge in complex, real-world CAD scenarios. It further examines multi-step problem-solving, paper-reading skills, and scenario interpretation.

One example of a comprehensive application-based problem involves studying and interpreting the C-space (configuration-space) approach to tool-path generation for computer numerical control (CNC) machining, as described in [3]. In this exercise, participants are required to replicate the algorithm and compute the inverse tool-offset surface from real-world mechanical parts. (See Section 5 for a detailed case study presenting the complete question, associated prompts, and LLM-generated answers.) Additional examples encompass the integrated application of CAD techniques in diverse engineering contexts, such as additive manufacturing [6] and computational materials science [5, 7].

In our evaluation, each multiple-choice question is weighted as one-third of a Q&A problem to balance the grading scale. All questions are framed to yield quantitative or qualitative answers that can be objectively graded. They are presented exclusively in text form without figures or charts, ensuring compatibility with language-only models.

2.2 LLM Model Selection

In this evaluation, we employed five state-of-the-art large language models (LLMs):

- **OpenAI o1**
- **OpenAI 4o**
- **Anthropic Claude 3.5 Sonnet**
- **Deepseek R1-Lite**
- **Alibaba Cloud Qwen2.5**

These models were selected based on their diverse architectures, performance characteristics, and accessibility to both graduate students and CAD researchers. They have been widely adopted in various practical contexts and have generated substantial user feedback. Together, these factors ensure that our evaluation offers a comprehensive assessment of current LLM capabilities in solving academic CAD problems across four categories.

All models are evaluated using their versions as of January 2025, which captures the latest advancements in LLM technology. Furthermore, since some of the selected models offer integrated web search functionality (for example, OpenAI 4o) while others do not, we uniformly disabled web search features across all models. This approach ensures that every model is assessed solely on its internal knowledge and reasoning capabilities, thereby establishing uniform evaluation conditions.

3 FRAMEWORKS FOR EFFECTIVE LLM PROBLEM-SOLVING

To address the challenges in solving academic problems in CAD and enhance the problem-solving capabilities of LLMs, we employed three distinct approaches: the naive approach, the knowledge refinement framework, and the Chain-of-Thought (CoT) framework (Fig. 1). While the naive approach serves as a baseline, the two advanced frameworks were to improve LLM performance, particularly for comprehensive, multi-domain problems.

In the ***naive approach***, the raw question and associated reference materials, such as academic papers or book chapters, are directly provided to the LLM. The model processes these inputs using its internal reasoning capabilities to generate an answer. However, this method often struggles with CAD problems due to the dense and complex nature of reference materials. CAD-related sources frequently contain intricate mathematical derivations, domain-specific terminology, and theoretical formulations that are difficult for LLMs to parse and understand effectively. Consequently, the naive approach often leads to suboptimal performance, especially on problems that require interdisciplinary integration and precise comprehension of technical details.

The ***knowledge refinement framework*** mitigates the limitations of the naive approach by preprocessing the input to make it more focused and digestible for the LLM. Instead of directly feeding raw references, we manually extract and condense the most relevant definitions and theoretical insights from the source materials. These refined inputs are then provided alongside the original question, enabling the LLM to focus on essential knowledge without being overwhelmed by extraneous details. By simplifying verbose and technical

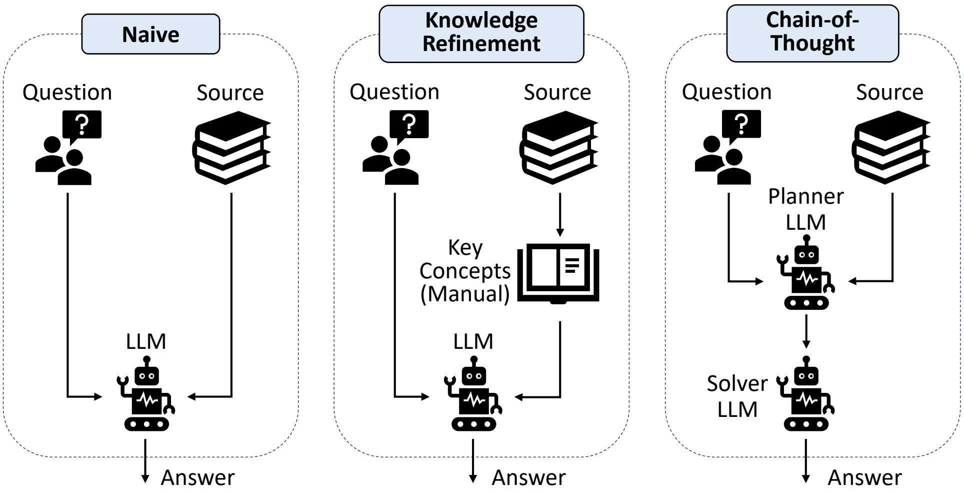

Figure 1: Framework structures for solving CAD academic problems with LLMs: naive, knowledge refinement, and Chain-of-Thought.

reference materials, the knowledge refinement framework improves the model's ability to reason through complex problems and deliver accurate solutions, even for challenging CAD-related tasks.

The ***Chain-of-Thought (CoT) framework*** adopts the concept of CoT prompting, which emphasizes breaking down complex problems into smaller, logically connected reasoning steps to enhance problem-solving performance [11]. Building on this concept, our framework introduces a structured, multi-phase process specifically tailored for solving comprehensive problems. This framework consists of two distinct phases:

1. **Planning:** In this phase, a "Planner LLM" analyzes the raw question and associated references to design a logical, step-by-step plan for solving the problem. This phase decomposes the problem into manageable components, ensuring clarity and direction in the problem-solving process.

2. **Solving:** In this phase, the plan generated by the Planner LLM is passed to a "Solver LLM," which executes the outlined steps to generate the final solution.

The effectiveness of Chain-of-Thought reasoning has been widely demonstrated in improving LLM performance, particularly for tasks requiring multi-step reasoning and logical decomposition. By separating the planning and execution processes, this framework leverages the complementary strengths of both models, with strategic reasoning in the planner and precise implementation in the solver.

The two advanced frameworks (knowledge refinement and Chain-of-Thought) were applied to address the shortcomings of the naive approach when solving comprehensive application-based problems, which require a seamless integration of mathematical reasoning, programming, conceptual understanding, and practical application. The refined inputs provided by the knowledge refinement framework and the structured reasoning in the CoT framework offered significant enhancements to LLMs' ability to effectively tackle challenges that were otherwise difficult to solve, improving their problem-solving capabilities across complex academic tasks.

4 RESULTS

Fig. 2 illustrates the evaluation results of the naive approach applied to academic problems across four categories: conceptual understanding, math, coding/algorithm, and comprehensive application-based problems.

These results were obtained from tests conducted on both LLMs and a baseline of 5 graduate students majoring in CAD-related disciplines. To simulate an authentic test situation, these graduate students were provided with all the required background knowledge one week in advance and given one week to study before the assessment. The findings indicate that, even with the naive approach, LLMs outperform graduate students in conceptual understanding, math, and coding/algorithm problems, achieving an average score of 96.5 compared to 73.7 points for students. This demonstrates the strong foundational and technical capabilities of LLMs in processing and reasoning within these domains.

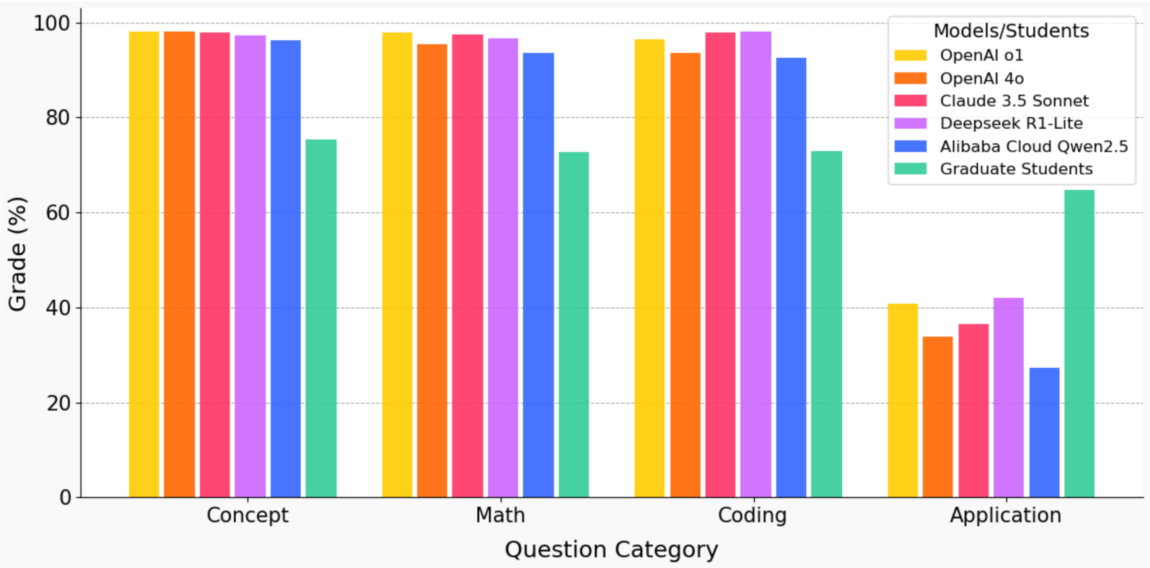

Figure 2: Performance on CAD academic problems across LLMs and graduate students.

However, for comprehensive application-based problems, LLMs perform poorly when using the naive approach, with an average score of 36.1 points, compared to 64.8 points achieved by graduate students. These results suggest that the naive approach is insufficient for solving problems that require a seamless integration of reasoning, programming, and real-world application. In contrast, the use of our frameworks, namely the knowledge refinement framework and the Chain-of-Thought (CoT) framework, results in significantly improved performance (as shown in Table 1). The top three LLMs (OpenAI o1, Anthropic Claude 3.5 Sonnet, and Deepseek R1-Lite) achieve average scores of 94.1 and 96.5 points using the knowledge refinement and CoT frameworks, respectively. These results highlight the ability of LLMs to logically understand and solve complex problems when guided by refined inputs or structured reasoning processes.

Although the Chain-of-Thought framework achieves marginally higher scores in our evaluations, each approach offers distinct advantages and disadvantages. The knowledge refinement framework requires researchers to manually extract and define critical terminologies from technical literature, a process accessible even to those with limited expertise in LLMs. In contrast, the CoT framework employs a specialized planner LLM to perform this task automatically, but it demands a higher level of understanding in selecting high-performance LLMs and in prompt engineering to guide multi-step reasoning effectively. Moreover, for researchers with sufficient LLM background knowledge, the CoT framework is expected to provide superior performance because it not only identifies critical terminologies but also supports deeper analysis and synthesis of the remaining background material.

Our results indicate that all tested LLMs provide better results when guided by advanced frameworks com-

Framework	o1	4o	Claude3.5	R1-Lite	Qwen2.5
Naive Approach	40.7	33.8	36.6	42.1	27.2
Knowledge Refinement	92.5	70.7	93.8	95.9	53.5
Chain-of-Thought	95.1	84.6	97.3	97.1	59.3

Table 1: Scores of different LLM models on comprehensive application-based CAD problems across the naive approach, knowledge refinement framework, and CoT framework. The CoT framework results were obtained by using OpenAI o1 as the planner and these LLMs as solvers.

pared to the naive approach on comprehensive application-based CAD problems. However, since these models are developed by different organizations, not every LLM demonstrates uniformly good overall performance (for example, Qwen2.5 demonstrates relatively lower performance). Therefore, the careful selection of a high-performance LLM is crucial for achieving optimal performance across all tasks. Given the rapid evolution of LLM technology, continuous evaluation and appropriate model selection remain essential for maintaining and enhancing performance across diverse applications.

These findings suggest that the naive approach is effective for solving most CAD academic problems, including conceptual, mathematical, and coding challenges. However, it is not suitable for handling complex application-based problems. The two advanced frameworks, by contrast, demonstrate that with moderate effort in knowledge extraction or structured reasoning, LLMs can overcome these limitations and deliver robust solutions for comprehensive challenges.

5 CASE STUDY

In this section, we present a representative case study of a comprehensive application-based problem designed to demonstrate the efficacy of our advanced frameworks in enhancing LLM performance on complex CAD tasks. This problem, which employs both the knowledge refinement framework and the Chain-of-Thought framework, is among the most challenging scenarios that most LLMs struggle to solve. To facilitate clarity, we present the input prompts and the corresponding LLM-generated responses in distinct text boxes, with the prompts displayed in blue and the answers in green. In some text boxes, we also provide explanatory notes at the end. These notes are formatted in italics and prefixed with "Note:" to supply additional context for the reader of this paper. These notes are intended solely as commentary and do not constitute any part of the actual interaction between the LLMs and the provided prompts.

In addition to the prompt, we provide essential background materials that serve as the preparatory source for solving this problem, mirroring the resources typically provided in an academic setting. This comprehensive application-based problem involves studying and interpreting the C-space (configuration-space) approach to tool-path generation for computer numerical control (CNC) machining, as described in [3]. The task requires replication of the algorithm and computation of the inverse tool-offset surface from real-world mechanical parts. To support this task, we supply the complete manuscript text of [3] as background information and a starter code file, `generate_ito.m`, to the LLM's input. These materials provide the necessary foundation for the LLM to tackle the problem.

5.1 Naive Approach

Prompt – Original Problem Description

Write a Python function to generate the inverse tool-offset (ITO) z-map for a ball-end mill. Please implement your function in the file `generate_ito.m` provided in the attachment. The function should have the format:

```
function [ito_surf] = generate_ito(dsurf_fname, spacing, diameter)
```

where `dsurf_fname` is the file name of the text file containing the z-map matrix for the input design surface, `spacing` is the spacing between the z-map points in both the x and y directions, `diameter` is the diameter of the ball-end mill, and `ito_surf` is the output z-map matrix for the ITO surface (of the same size as the input matrix). Please use the center of the hemisphere of the ball end as the CL point, as described in the paper.

Hint: As an intermediate step, you may want to construct a z-map of the end of the inverted tool using the specified spacing, with the center entry equal to the tool radius (so that the CL point is at $z = 0$). The other values in this z-map will correspond to (x, y) positions within the tool footprint, which can be calculated using trigonometry.

[Attachments] Paper manuscript [3] and `generate_ito.m` are provided as attachments.

Note: In later text boxes, the content of this prompt will be represented by [Original Problem Description] and [Attachments] so as to avoid redundancy in our paper.

In the naive approach, the original problem description is provided to the LLMs along with the most fundamental support materials, which include the complete paper manuscript [3] and the starter code file `generate_ito.m`. This straightforward input method does not incorporate any advanced prompt engineering or additional contextual guidance. The evaluation results, summarized in Table 2, reveal that none of the selected LLMs generated code that correctly solves the problem. In several cases, the generated code was not even executable. These outcomes underscore the limitations of the naive approach in addressing complex CAD problem-solving tasks and highlight the need for advanced frameworks to better harness LLM capabilities.

Models	**Code Executability**	**Correctness**
OpenAI o1	×	–
OpenAI 4o	×	–
Claude 3.5 Sonnet	✓	×
Deepseek R1-Lite	✓	×
Qwen2.5	✓	×

Table 2: Naive approach evaluation results for the case study

5.2 Knowledge Refinement Framework

Although the paper manuscript was provided as part of the support materials in the naive approach, its extensive and complex information hindered LLMs from identifying critical details required to solve the problem. In the knowledge refinement framework, we address this limitation by supplementing the original problem description and fundamental attachments with additional, targeted information considered essential by CAD researchers. In particular, we provide a precise definition of the ITO surface (excerpted from [3]), a key concept for solving the problem. This refined input enables the LLMs to focus on the most relevant knowledge, thereby improving their overall performance on the task.

Prompt – Knowledge Refinement Framework

I will provide you with a problem:

[Original Problem Description]
[Attachments]

I will also supply some background information:

The ITO-surface is defined as the envelope of the cutter swept volumes obtained by sweeping the inverse cutter over the entire surface while keeping its reference point on the surface (i.e., using the inverse offset method to generate the ITO-surface). The "inverse" tool refers to a virtual cutter that has the same shape as the actual cutting tool but is oriented in the opposite direction; it is used to compute the compensation surface of the original design surface.

Please answer the problem and write Python code.

The results of applying the knowledge refinement framework to this case study are presented in Table XX. Our findings reveal that three of the evaluated LLMs benefit significantly from the refined input, successfully generating executable and correct code solutions. These results underscore the value of targeted knowledge enhancement in enabling LLMs to extract critical information and effectively address complex CAD problem-solving tasks.

Models	Code Executability	Correctness	Execution Time
OpenAI o1	$\times$	–	–
OpenAI 4o	$\times$	–	–
Claude 3.5 Sonnet	✓	✓	0.11s
Deepseek R1-Lite	✓	✓	0.04s
Qwen2.5	✓	✓	21.7s

Table 3: Knowledge refinement framework evaluation results for the case study

5.3 Chain-of-Thought Framework

In the Chain-of-Thought (CoT) framework, instead of relying on externally refined knowledge, the LLM is prompted to plan its strategy in advance through a dedicated planner LLM. This approach utilizes the inherent reasoning capabilities of the planner to decompose the problem into manageable sub-tasks and to outline a coherent solution strategy. Recognizing the advanced reasoning abilities of OpenAI o1, which is renowned for its robust chain-of-thought capabilities, we designate it as the planner LLM. The planner generates a detailed strategy that is subsequently employed by the solver LLM to produce the final answer. This method facilitates a more systematic and strategic approach to problem-solving, ultimately enhancing the accuracy and reliability of the solution.

Prompt – CoT Framework – Planning Phase

I will provide you with a problem:

[Original Problem Description]
[Attachments]

Next, please reason through the problem step-by-step and present your strategy along with pseudocode to solve it. Please provide sufficient explanation for your response.

LLM Response – Planner LLM

In order to generate the ITO z-map for a ball-end mill, we need to consider the envelope formed by sweeping the inverse cutter across the entire design surface. The detailed steps for solving the problem are as follows:

1. **Read the Design Surface Z-Map:** Load the z-map matrix for the design surface from the file `dsurf_fname`.

2. **Generate the Inverse Tool Z-Map:**

   - Calculate the radius of the ball-end mill as $r = \text{diameter}/2$.
   - Based on the specified spacing and the dimensions of the design surface, generate the grid coordinates for the tool's z-map.
   - Use trigonometric functions to compute the z-value for each grid point using $z = \sqrt{r^2 - x^2 - y^2}$. For grid points where $x^2 + y^2 > r^2$, assign a value of NaN.

3. **Generate the ITO Surface Z-Map:** Overlay the inverse tool's z-map onto each corresponding point of the design surface to compute the ITO surface height. Specifically, at each grid point, the ITO surface height is determined by subtracting the tool's z-map value from the design surface's height. This method simulates the sweeping of the inverse cutter over the design surface through a morphological dilation operation, thereby generating the ITO surface z-map.

Note: In later text boxes, the content of this plan/strategy will be represented by [Strategy] so as to avoid redundancy in our paper.

Prompt – CoT Framework – Solving Phase

I will provide you with a problem:

[Original Problem Description]
[Attachments]

I will now provide my problem-solving strategy:

[Strategy]

Based on this strategy and your understanding, please provide the Python code that solves the problem.

The results of applying the CoT framework to this case study are presented in Table 4. Our evaluation indicates that most LLMs were able to solve the problem correctly and efficiently under the CoT framework. Compared to the naive approach, the distinct planning and solving phases enabled by the CoT framework allowed the models to generate accurate and executable code with improved efficiency.

Models	Code Executability	Correctness	Execution Time
OpenAI o1	✓	✓	0.04s
OpenAI 4o	×	–	–
Claude 3.5 Sonnet	✓	✓	0.04s
Deepseek R1-Lite	✓	✓	0.04s
Qwen2.5	×	–	–

Table 4: Chain-of-Thought framework evaluation results for the case study

In summary, our case study clearly demonstrates that the two advanced frameworks significantly enhance problem-solving capabilities. Although not all LLM models generate correct solutions, largely depending on the performance of each vendor's model, the transition from complete failure in the naive approach to successful resolution in most cases suggests that, with appropriate model selection and the integration of advanced frameworks, there is a strong likelihood that even the most challenging CAD problems can be successfully resolved.

6 IMPLICATIONS FOR CAD EDUCATION

The integration of LLMs into CAD education has significant implications. By leveraging frameworks such as knowledge refinement and CoT, students and educators can solve complex academic problems more efficiently, transforming how CAD is taught and learned. However, the ease with which students can rely on LLMs to solve academic problems raises critical questions about the role of education. It becomes essential to reconsider teaching methods to ensure that students develop a deep understanding of CAD concepts, rather than relying solely on LLM outputs for completing problem sets. Balancing the use of LLMs as tools for enhancing learning while maintaining rigorous academic engagement is a challenge that educators must address to avoid superficial learning and encourage meaningful intellectual growth.

7 FUTURE WORK

With ongoing rapid advances in commercial LLMs and prompting methods, we anticipate that in the near future these models will tackle CAD problems more accurately and support CAD education more effectively. Beyond these model gains, and building on our evaluation and the two frameworks presented above, we suggest the following areas for future work:

1. **Longitudinal Pedagogical Studies.** Integrating our frameworks into real classroom or online courses and measuring student outcomes (problem-solving accuracy, conceptual retention, and engagement) over a semester to assess long-term impact.

2. **Multimodal CAD Problem Solving.** Extending our frameworks to handle diagrams, 3D geometry files, and CAD screenshots by developing vision-language modules that enable LLMs to parse and reason about graphical content alongside text.

3. **Domain-Specific Fine-Tuning.** Fine-tuning open-source LLMs on CAD-focused corpora (manuals, research articles, forum discussions) and comparing their performance and efficiency to our prompt-engineering approaches on the same academic tasks.

4. **Cross-Disciplinary Transfer.** Evaluating whether our knowledge refinement and CoT frameworks generalize to other technical domains by applying them to interdisciplinary problem sets in fields such as finite element analysis, control systems, or computational materials science.

8 CONCLUSIONS

In this work, we have compiled a comprehensive set of CAD academic problems and conducted a rigorous evaluation of multiple state-of-the-art LLMs.

Our systematic evaluation shows that LLMs outperform graduate students on foundational CAD tasks, including conceptual understanding, mathematical problem solving, and coding. However, these models encounter considerable challenges when solving comprehensive application-based problems that require the integration of theory, implementation, and contextual understanding. To address these challenges, we developed two advanced frameworks. The knowledge refinement framework supplements the original input with targeted background information, and the Chain-of-Thought (CoT) framework directs the model to plan its strategy using a structured, multi-phase reasoning process. The application of these frameworks increased performance on comprehensive application-based problems from an average score of 36 points to over 90 points.

These findings not only reveal the significant yet underexploited potential of LLMs to transform CAD education, but also emphasize the urgent need for curricula and instructional strategies that combine the benefits of automated problem solving with the cultivation of deep, robust conceptual understanding. Such strategies are essential for educators to prevent superficial learning and foster meaningful intellectual growth.

ORCID

Xiang Li, http://orcid.org/0000-0001-5936-8457
Ling Chen, http://orcid.org/0009-0004-1086-8222
Guang Feng, http://orcid.org/0000-0002-1500-2282
Sara McMains, http://orcid.org/0000-0002-7152-9409

REFERENCES

[1] Abd-Alrazaq, A.; AlSaad, R.; Alhuwail, D.; Ahmed, A.; Healy, P.M.; Latifi, S.; Aziz, S.; Damseh, R.; Alrazak, S.A.; Sheikh, J.; et al.: Large language models in medical education: opportunities, challenges, and future directions. JMIR Medical Education, 9(1), e48291, 2023. http://doi.org/10.2196/48291.

[2] Alqahtani, T.; Badreldin, H.A.; Alrashed, M.; Alshaya, A.I.; Alghamdi, S.S.; Bin Saleh, K.; Alowais, S.A.; Alshaya, O.A.; Rahman, I.; Al Yami, M.S.; et al.: The emergent role of artificial intelligence, natural learning processing, and large language models in higher education and research. Research in Social and Administrative Pharmacy, 19(8), 1236–1242, 2023. http://doi.org/10.1016/j.sapharm.2023.05.016.

[3] Choi, B.K.; Kim, D.H.; Jerard, R.B.: C-space approach to tool-path generation for die and mould machining. Computer-Aided Design, 29(9), 657–669, 1997. http://doi.org/10.1016/S0010-4485(97)00012-2.

[4] Kasneci, E.; Seßler, K.; Küchemann, S.; Bannert, M.; Dementieva, D.; Fischer, F.; Gasser, U.; Groh, G.; Günnemann, S.; Hüllermeier, E.; et al.: ChatGPT for good? On opportunities and challenges of large language models for education. Learning and Individual Differences, 103, 102274, 2023. http://doi.org/10.1016/j.lindif.2023.102274.

[5] Li, X.; Cui, L.; Li, J.; Chen, Y.; Han, W.; Shonkwiler, S.; McMains, S.: Automation of intercept method for grain size measurement: A topological skeleton approach. Materials & Design, 224, 111358, 2022. http://doi.org/10.1016/j.matdes.2022.111358.

[6] Li, X.; McMains, S.: A Voronoi diagram approach for detecting defects in 3D printed fiber-reinforced polymers from microscope images. Computational Visual Media, 9(1), 41–56, 2023. http://doi.org/10.1007/s41095-021-0265-1.

[7] Li, X.; Shonkwiler, S.; McMains, S.: Detection of resin-rich areas for statistical analysis of fiber-reinforced polymer composites. Composites Part B: Engineering, 225, 109252, 2021. http://doi.org/10.1016/j.compositesb.2021.109252.

[8] Lv, H.; Niu, Z.; Han, W.; Li, X.: Can GPT embeddings enhance visual exploration of literature datasets? A case study on isostatic pressing research. Journal of Visualization, 27(6), 1213–1226, 2024. http://doi.org/10.1007/s12650-024-01010-z.

[9] Thirunavukarasu, A.J.; Ting, D.S.J.; Elangovan, K.; Gutierrez, L.; Tan, T.F.; Ting, D.S.W.: Large language models in medicine. Nature Medicine, 29(8), 1930–1940, 2023. http://doi.org/10.1038/s41591-023-02448-8.

[10] Tsai, M.L.; Ong, C.W.; Chen, C.L.: Exploring the use of large language models (LLMs) in chemical engineering education: Building core course problem models with chat-GPT. Education for Chemical Engineers, 44, 71–95, 2023. http://doi.org/10.1016/j.ece.2023.05.001.

[11] Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; et al.: Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824–24837, 2022.

[12] Ye, X.; Peng, W.; Chen, Z.; Cai, Y.Y.: Today's students, tomorrow's engineers: an industrial perspective on CAD education. Computer-Aided Design, 36(14), 1451–1460, 2004. http://doi.org/10.1016/j.cad.2003.11.006.