arXiv:2601.01774v1 [cs.AI] 5 Jan 2026

大语言模型能解工程方程吗？直接预测与求解器辅助方法的系统比较

SaiVarun Kodathala研发 Sports Vision, Inc. 明尼苏达州明尼通卡
varun@sportsvision.ai

Rakesh Vunnam
研发 Vizworld, Inc. 明尼苏达州明尼通卡
rakesh@vizworld.ai

摘要

从流体力学中的摩擦因子计算到轨道位置确定，需要迭代数值求解的超越方程在工程实践中无处不在。我们系统评估大语言模型能否通过直接数值预测求解这些方程，抑或结合大语言模型符号运算与经典迭代求解器的混合架构更为有效。我们在涵盖七个工程领域的100个问题上测试了六种最先进模型（GPT-5.1、GPT-5.2、Gemini-3-Flash、Gemini-2.5-Lite、Claude-Sonnet-4.5、Claude-Opus-4.5），比较直接预测与求解器辅助计算：前者由大语言模型建立控制方程并提供初始条件，后者则通过牛顿-拉夫森迭代进行数值求解。各模型的直接预测平均相对误差为0.765至1.262，而求解器辅助计算的误差为0.225至0.301，误差降低幅度达67.9%至81.8%。分领域分析显示，电子学领域的提升尤为显著（93.1%），这归因于指数方程敏感性；相比之下，流体力学领域的提升较为有限（7.2%），因为大语言模型展现出有效的模式识别能力。这些发现表明，当代大语言模型擅长符号运算和领域知识检索，却难以胜任精度要求高的迭代算术；因此，与其将其作为独立计算引擎，不如将其用作经典数值求解器的智能接口，以发挥其最佳效用。

关键词 大语言模型 · 数值方法 · 牛顿-拉夫森法 · 超越方程 · 工程计算 · 混合人工智能系统

1 引言

大语言模型（LLM）的快速发展推动了从自然语言处理到代码生成等各个领域的变革 [Brown等人，2020年，OpenAI， 2023]。这些模型展现出卓越的数学推理能力，在标准化考试中取得近乎完美的准确率，并在研究生级科学问题上表现出精妙的问题求解策略 [Schiappa等人， 2025]。随着这些能力不断增强，一个关键问题随之浮现：大语言模型能否作为直接计算引擎，解决工程应用中的数值问题？抑或存在一种更有效的架构范式，能够将其融入科学计算工作流？

这一问题在超越方程的语境下显得尤为紧迫。超越方程是问题中的一个基本类别，广泛存在于工程实践中。与可通过代数运算求解的多项式方程不同，超越方程涉及指数函数、对数函数或三角函数，因而无法获得闭式解析解[Burden和Faires，2010，Press等人， 2007]。考虑支配行星运动的开普勒著名方程：

$$M = E - e \sin E \tag{1}$$

给定平近点角 M 和偏心率 e，求偏近点角 E 需要采用迭代数值方法，这是天文学家自17 世纪以来一直面对的现实。类似的超越方程还出现在流体力学中的 Colebrook-White 摩擦因子方程、电子学中的二极管电路分析、热力学中的范德瓦耳斯真实气体行为，以及现代工程学中许多其他关键领域。

三个多世纪以来，工程师一直通过经典迭代方法处理这些问题，其中牛顿-拉夫森迭代凭借其二次收敛特性，成为计算工作的主力。然而，这一传统流程设置了诸多障碍：工程师必须将文字问题描述转换为数学形式，实现数值算法，选择适当的初始条件，并管理收敛行为。这一工作流程同时要求具备领域知识和数值方法方面的专业能力，从而限制了非专业人士的使用。

大语言模型在科学文献上的广泛训练催生了应对这些挑战的两种相互竞争的构想。第一种构想将大语言模型定位为端到端的问题求解器：给定自然语言问题描述，模型可以直接生成数值解，利用从数十亿个科学文本词元中学习到的模式。这种方法有望使数值计算更加普及：工程师可以以对话方式描述问题，并立即获得答案，而无需实现算法或构建方程。

第二种构想提出了一种混合架构，承认语言模型与经典算法各自互补的优势。在这种架构中，大语言模型充当智能接口：解析自然语言，检索领域知识，构建控制方程，并提供初始条件。随后，经典求解器负责精度要求高的迭代算术。这种分解体现了一个基本假设：大语言模型擅长语义理解和符号运算，而数值方法则通过经过充分验证的数学理论提供有保障的精度。

但哪种愿景更有利于工程实践？答案将深刻影响我们如何将人工智能融入科学计算、塑造工具开发，以及分配计算资源。为系统地探讨这一问题，我们设计了一项全面的实证研究，对这两种范式进行比较，涵盖横跨七个工程领域、经过精心构建的100个超越方程。评估涉及六种代表当前语言模型能力前沿的最先进模型：OpenAI 的 GPT-5.1 和 GPT-5.2，它们在 AIME 2025 数学竞赛中达到 100% 的准确率 [Aciano, 2025]；谷歌的 Gemini-3-Flash 和 Gemini-2.5-Lite，其中 Gemini-3-Flash 在研究生水平的 GPQA Diamond 基准测试中得分为 91.9% [Vellum AI, 2025]；以及 Anthropic 的 Claude-Sonnet-4.5 和 Claude-Opus-4.5，它们着重于数学语境中的事实准确性 [R&D World, 2025]。

我们的实验设计仔细控制了混杂因素。所有模型都接收了相同的自然语言问题描述，以及旨在引导其给出适合任务的回答的标准化系统提示词。模型以温度 1.0 运行，代表其默认推理配置。问题经过专门构造，以避免记忆效应，同时体现真实的工程场景。对于求解器辅助方法，我们实现了标准化的牛顿-拉夫森求解器，并采用收敛阈值 $|error| < 0.0001$，以确保所有评估中的数值精度一致。

本文其余部分安排如下。第 2 节回顾有关大语言模型数学能力、数值方法和混合架构的相关研究。第 3 节介绍基准数据集构建方法论。第 4 节详述实验设计、提示策略和算法实现。第 5 节全面呈现整体性能、特定领域模式以及查询级失败模式方面的结果。第 6 节讨论对工程实践和未来研究方向的启示。第 7 节作出总结。

2 相关研究

2.1 大语言模型中的数学推理

大语言模型的数学能力已得到广泛研究，研究结果既展现了令人瞩目的成就，也揭示了系统性局限。早期基准（如 GSM8K（小学数学））显示出快速进步，当代模型在基础算术和多步推理方面已达到近乎完美的准确率 [Schiappa et al., 2025]。然而，更深入的研究揭示了基准测试表现与稳健的数学理解之间存在令人担忧的差距。

GSM-Symbolic 基准生成的问题在结构上与 GSM8K 完全相同，但表层特征各异。研究发现，在问题描述中加入无关条款后，最先进的大语言模型的表现最高会下降 65% [Apple MachineLearning Research, 2025]。这种脆弱性表明，模型严重依赖表层模式匹配，而非真正的数学推理；对于需要稳健问题解决能力的应用而言，这是一个令人担忧的发现。

Zhang 等人提出了 APBench，这是一个针对高级天体动力学问题的基准，这些问题需要领域知识、因果推理和多步数学运算 [Zhang et al., 2025]。尽管模型展现出对物理原理的基本理解，但在需要精确数值解的问题上表现仍然有限。思维链提示改善了推理结构，却未能解决数值精度方面的局限，凸显出推理能力与计算准确性之间持续存在的脱节。

加州大学伯克利分校近期针对研究生级别基于证明的问题开展的研究表明，大语言模型虽然能够通过模式匹配解决以答案为导向的问题，但仍无法完成需要形式演绎推理的复杂基于证明的任务 [加州大学伯克利分校电气工程与计算机科学系, 2025]。答案生成能力与真正推理能力之间的这种区别，对于数值计算应用具有深远影响，因为在此类应用中，正确性无法仅通过模式匹配来验证。

2.2 当代模型的性能特征

GPT-5.1 和 GPT-5.2 代表了 OpenAI 的最新一代模型，在 AIME 2025 中达到 100% 的准确率，在 GPQA Diamond 中达到 90% [Aciano，2025年，Effortless Academic， 2025]。在扩展推理模式下，GPT-5.2 产生的事实性错误比之前的版本减少了 78%。然而，批判性分析揭示了持续存在的局限性 [AI Fire，2025年，Furze， 2025]。幻觉率从 1.8% 降至 1.4%，但模型仍会生成“自信地错误”的数值结果，即答案看似精确，实际上却存在严重错误。值得注意的是，基础预训练模型的校准效果优于后训练版本，这表明对齐可能会无意中降低数值可靠性。

谷歌的 Gemini 架构展现出卓越的推理性能。Gemini-3-Flash 在 GPQA Diamond 上得分为 91.9%，在 Deep Think 模式下得分为 93.8% [Vellum AI, 2025年, Vertu, 2025]。在 MathArena Apex 中，针对以往人工智能系统无法解决的问题设计的前沿题目上，Gemini-3 的得分为 23.4%，比竞争模型提升了 20 倍。尽管取得了这些成就，这些架构在精确的迭代数值计算方面仍面临挑战。

Anthropic 的 Claude 模型通过采取保守策略来强调事实准确性 [R&D World, 2025]。这种方法或许能降低产生幻觉的风险，但也可能导致数值输出过于保守。近期关于推理时扩展的研究表明，延长推理时间可以提升抽象问题求解能力，却无法从根本上满足迭代运算对精度的要求 [MIT News, 2025], , 这表明其局限性源于架构，而非单纯的计算能力。

2.3 超越方程的经典数值方法

牛顿-拉夫森法于 17 世纪得到系统阐述，至今仍是求解超越方程的主流方法[Burden和Faires, 2010, Press等人，2007]。给定函数 $f(x)$ 和初始猜测 x_0，该方法通过迭代逐步逼近解：

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{2}$$

在适当条件下，该方法表现出二次收敛，每次迭代后正确数字的位数大致翻倍。对于初始猜测合理的良态问题，牛顿-拉夫森法因其高效而成为首选方法。

超越方程广泛存在于工程领域。管道湍流中用于计算摩擦因子的 Colebrook-White 方程：

$$\frac{1}{\sqrt{f}} = -2.0 \log_{10}\left(\frac{\epsilon/D}{3.7} + \frac{2.51}{Re\sqrt{f}}\right) \tag{3}$$

由于 f 在对数项中同时出现在等式两边，因此需要通过迭代求解。真实气体的范德瓦耳斯方程：

$$\left(P + \frac{a}{V^2}\right)(V - b) = RT \tag{4}$$

会产生关于体积 V 的三次方程，需要进行数值求解。将肖克利定律与基尔霍夫电压定律结合起来的二极管方程：

$$V_s = V_D + R \cdot I_s \left(e^{V_D/V_T} - 1\right) \tag{5}$$

具有指数敏感性，因此近似解并不可靠。

经典方法提供了机器学习方法所缺乏的数学保证。收敛定理确立了牛顿-拉夫森法成功的条件，而误差界则量化了经过给定迭代次数后的精度

。这些基础使得我们能够认证解是否满足精度要求，这对安全关键应用至关重要。

2.4 经典算法与人工智能混合架构

结合经典算法与机器学习的混合系统已在各个计算领域涌现。在量子计算领域，量子-经典混合架构利用量子处理器执行特定子程序，同时由经典计算机负责编码、优化和后处理 [《国际创新科学与研究技术期刊》, 2024, UChicago Voices, 2025]。在计算化学领域，基于量子模拟训练的深度网络能够快速预测性质，同时保持精度 [Endev Solutions, 2025]。
然而，用于大语言模型辅助数值方程求解的混合架构仍鲜有研究。现有工作侧重于代码生成或结果解释，而非系统地将数值问题求解拆分为大语言模型符号运算与经典迭代。

2.5 研究空白与贡献

此前尚无研究在相同问题上，针对多个模型系统地比较直接大语言模型计算与大语言模型辅助经典求解。本文对六个大语言模型求解 100 个超越方程进行了系统评估，实证展示了混合架构带来的显著误差降低，开展了领域特定分析，揭示了模式识别何时有效、何时迭代不可或缺，并为如何在数值工作流程中优化大语言模型部署提供了实践指导。

3 数据集构建

我们构建了一个包含 100 个超越方程的基准数据集，涵盖七个工程学领域，可在 https://huggingface.co/datasets/sportsvision/Engineering-Numerical-Methods-Benchmark 公开获取。每个问题都包含一个描述真实工程场景的自然语言查询，以及经过验证的真实数值解。

3.1 领域选择与数学基础

领域选择依据三个标准：超越方程的普遍程度、数学结构的多样性（指数、三角、对数），以及实际工程重要性。选定的七个领域涵盖了专业工程实践中出现超越方程的主要领域。

流体力学（16 个问题）。Colebrook-White 方程是工程学中最常见的超越方程之一。对于管道湍流，达西摩擦因子 f 隐式取决于雷诺数 Re和相对粗糙度 ϵ/D:

$$\frac{1}{\sqrt{f}} = -2.0\log_{10}\left(\frac{\epsilon/D}{3.7} + \frac{2.51}{Re\sqrt{f}}\right) \tag{6}$$

摩擦因子在对数表达式的两侧均出现，因此需要进行迭代求解。我们的问题涵盖雷诺数从 10^4 到 10^7 、相对粗糙度从 10^{-6} 到 10^{-2}的范围，覆盖了实际的管流状态。
轨道力学（16 个问题）。开普勒方程将椭圆轨道中的平近点角 M 与偏近点角 E 联系起来:

$$M = E - e\sin E \tag{7}$$

其中， e 表示偏心率。这个经典的超越方程必须通过迭代求解，才能确定给定时刻的航天器位置。问题涵盖的偏心率范围从 0.1（接近圆形）到 0.9（高度椭圆），平近点角范围从 0 到 π 弧度。

电子学（16 个问题）。二极管电路分析需要同时求解肖克利二极管方程和基尔霍夫电压定律。对于电压源为 V_s、串联电阻为 R、二极管饱和电流为 I_s、热电压为 V_T的电路，二极管电压 V_D 满足:

$$V_s = V_D + R \cdot I_s\left(e^{V_D/V_T} - 1\right) \tag{8}$$

指数项会造成极端敏感性，使微小的电压误差呈指数级放大，因此这对近似方法尤其具有挑战性。问题中的电源电压（1 至 10V）、电阻（100 至 10,000Ω）和饱和电流（10^{-15} 至 10^{-12}A）各不相同。

热力学（16 个问题）。范德瓦耳斯方程通过考虑分子体积和分子间作用力来修正理想气体的行为：

$$\left(P + \frac{a}{V^2}\right)(V - b) = RT \tag{9}$$

其中 a 和 b 是特定物质的常数。在给定压力 P 和温度 T 下求解摩尔体积 V，会得到一个需要迭代求解的三次方程。问题涉及氮气、二氧化碳和甲烷，压力范围为 10 至 100 标准大气压，温度范围为 200 至 500 K。

传热（11 个问题）。普朗克辐射定律描述了黑体光谱辐亮度：

$$B_\lambda(T) = \frac{2hc^2}{\lambda^5}\frac{1}{e^{hc/\lambda k_B T} - 1} \tag{10}$$

根据峰值波长 λ_{max} 反求温度 T 需要进行数值求解。问题涉及的波长范围为 0.5 至 10 微米，对应的温度范围为 300 至 5800 K。

结构工程（13 个问题）。兰伯特W函数定义为 $f(W) = We^W$ 的反函数，应用于柱屈曲、缆索动力学和时滞微分方程。相关问题要求求解以下形式的方程：

$$xe^x = k \tag{11}$$

针对从 0.1 到 100 不等的各种 k 值，代表不同的物理场景。

化学工程（12 个问题）。Brunauer-Emmett-Teller（BET）等温线描述了多层气体吸附：

$$\frac{P}{V(P_0 - P)} = \frac{1}{V_m C} + \frac{C - 1}{V_m C}\frac{P}{P_0} \tag{12}$$

根据实验数据确定单层容量 V_m 需要进行迭代拟合。问题涉及 BET 常数（10 至 1000）和相对压力（0.05 至 0.35）。

3.2 自然语言问题表述

问题以自然语言表述，不提供明确的方程，这反映了真实的工程工作流：问题最初通常以对话形式描述，而非采用数学符号。代表性示例如下：

流体力学：“我正在为一栋商业建筑设计供水系统，需要计算光滑管中的水头损失。管道直径为 0.15 米，水流的雷诺数为 100,000。光滑管的绝对粗糙度为 0.000045 米。此流动条件下的达西摩擦因子是多少？”

轨道力学：“我正在计算一颗卫星在地球周围椭圆轨道上的位置。该轨道的偏心率为 0.6，在某一特定时刻，平近点角为 0.6 弧度。我需要求出以弧度表示的偏近点角，以确定卫星的位置。”

电子学：“我正在分析一个简单的二极管电路，它由一个 5 伏直流电压源、一个 1000 欧姆电阻器和一个硅二极管串联而成。该二极管的饱和电流为 1×10^{-14} 安培，室温下的热电压为 0.026 伏。二极管两端的电压是多少伏？”

这种表述有多重用途：它用于测试大语言模型从文字描述中提取相关参数的能力，模拟真实工作流程，要求进行领域知识检索，并避免直接对数学表达式进行模式匹配。

3.3 真实解

真实解已提供于基准数据集中，并通过多个独立数值求解器实现进行验证。所有数值均四舍五入至小数点后三位，符合典型的工程精度要求，同时避免要求达到超出实际应用中物理测量精度的不切实际准确度。

4 方法论

我们评估了两种计算范式：直接预测，即大语言模型端到端生成数值解；以及求解器辅助计算，即大语言模型构建方程并提供初始条件，而牛顿-拉夫森迭代负责计算解。图1展示了这两种方法。

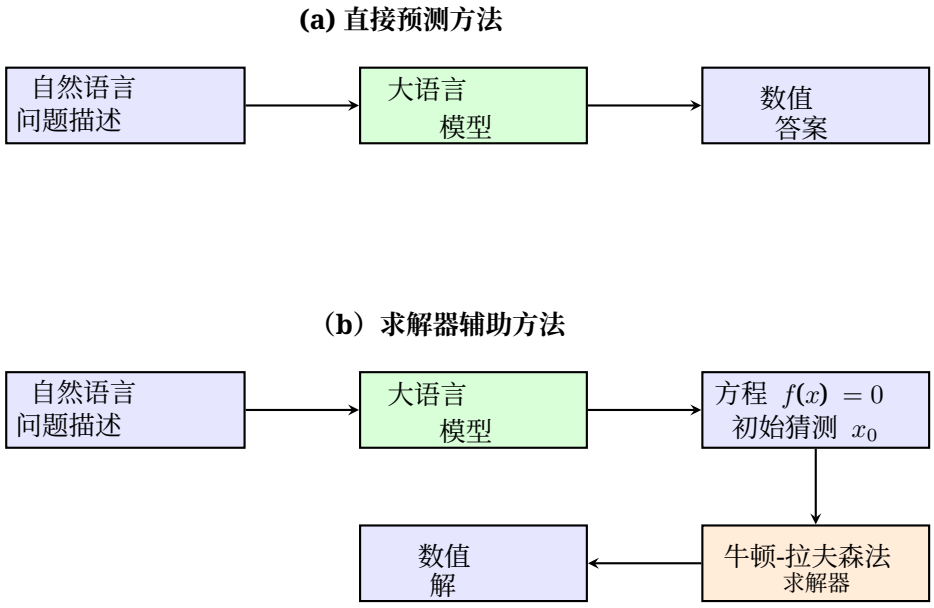


图1：基于大语言模型的方程求解计算范式。(a) 直接方法：大语言模型直接生成数值答案。(b) 求解器辅助方法：大语言模型构建符号方程并提供初始猜测，Newton-Raphson 执行迭代求解。

4.1 模型选择与配置

我们评估了代表当前语言模型能力前沿的六个最先进的大语言模型：GPT-5.1（gpt-5.1-chat）、GPT-5.2（gpt-5.2-chat）、Gemini-3-Flash（gemini-3-flash-preview）、Gemini-2.5-Lite（gemini-2.5-flash-lite）、Claude-Sonnet-4.5（claude-sonnet-4-5-20250929）和 Claude-Opus-4.5（claude-opus-4-5-20251101）。所有模型均在温度为 1.0 的条件下运行，并采用其默认推理配置以进行随机采样，从而反映典型的部署条件。

4.2 方法1：直接大语言模型预测

在直接预测中，大语言模型尝试根据自然语言描述进行端到端计算。系统消息如下：

> 你是一名专业工程师。阅读查询内容并直接提供数值答案。以 JSON格式返回 {"answer": "<数值答案，最多保留3位有效数字>"}

该提示将模型设定为专家，要求直接输出数值，并指定结构化JSON格式，以便可靠提取。每个问题的自然语言描述均作为用户消息提供。模型可以自由采用任何内部推理策略，包括模式匹配、近似计算或记忆关联。

4.2.1 性能评估

对于每个问题 i，给定真实值 x_i^{GT}和预测值 $\hat{x}_i$，我们计算相对误差：

$$\epsilon_i = \frac{|\hat{x}_i - x_i^{GT}|}{|x_i^{GT}|} \tag{13}$$

所有问题的平均相对误差（MRE）用于量化整体性能：

$$MRE = \frac{1}{N}\sum_{i=1}^{N}\epsilon_i \tag{14}$$

算法1展示了评估流程：

算法1 直接预测性能评估

要求： 包含真实值 x^{GT} 的数据集、模型预测值 $\hat{x}^{LLM}$**确保：** 每个模型的平均相对误差（ MRE ）**对于**每个模型 m 在模型中**执行**$errors \leftarrow []$**对于** 数据集中的每个问题 i **执行** $x_i^{GT} \leftarrow$ 问题的真实值 $i\hat{x}_i \leftarrow$ 问题的模型 m 预测值 i **如果** x_i^{GT}不是 NaN 且 $\hat{x}_i$ 不是 NaN，且 $x_i^{GT} \neq 0$ **则**$\epsilon_i \leftarrow \frac{|\hat{x}_i - x_i^{GT}|}{|x_i^{GT}|}$ 将 ϵ_i 追加到 $errors$**结束条件判断，结束循环**$MRE_m \leftarrow$ 平均值($errors$)**结束循环,一返回** 所有模型的平均相对误差

4.3 方法2：求解器辅助计算

该混合方法将问题求解分解为大语言模型方程构建，随后进行牛顿-拉夫逊迭代求解。

4.3.1 方程构建提示词

求解器辅助计算的系统消息如下：

> 你是一名专家工程师。阅读查询并提取：1. 使用 Python/NumPy语法表示的超越方程 f(x) = 0 2. 一个合理的初始猜测 x0 输出格式：给出 JSON {"equation": "超越方程 f(x)，其中 x 是根", "x0": "<初始猜测>"}

此提示要求模型运用领域知识识别控制方程，从自然语言中提取参数，将方程表示为 $f(x) = 0$，并提供符合物理规律的初始猜测。例如，对于流体力学问题，预期的响应格式如下：

```
{"equation": "1/sqrt(x) + 2.0*log10((0.000045/0.15)/3.7 +
2.51/(100000*sqrt(x)))", "x0": "0.02"}
```

4.3.2 牛顿-拉夫逊法实现

根据大语言模型响应中的符号函数 $f(x)$，我们使用符号微分解析地计算其导数，然后应用牛顿-拉夫森迭代。算法2详细说明了具体实现：
关键实现规范包括：收敛判据 $|x - x^{GT}| < 0.0001$，该判据旨在平衡计算效率与工程应用所需的足够精度；最多迭代1000次，以应对具有挑战性的情况，同时防止无限循环；以及将解四舍五入至小数点后三位，以匹配真实值的精度。

4.4 性能指标

我们采用相对误差作为主要指标来量化性能，因为它具有尺度无关性和工程可解释性。相对误差以真实值的比例来衡量预测误差，因此能够对解的数量级截然不同的问题进行有意义的比较。相对误差为0.10表示与真实值相差10%，无论真实值是 10^{-3} 还是 10^{3}。

直接方法与求解器辅助方法之间的误差降低计算如下：

$$\text{Improvement} = \frac{MRE_{\text{direct}} - MRE_{\text{assisted}}}{MRE_{\text{direct}}} \times 100\% \tag{15}$$

算法2牛顿-拉夫森求解器

要求： 函数 $f(x)$，导数 $f'(x)$，初始猜测 x_0，真实值 x^{GT}**确保：** 解 x_{sol} 和迭代次数$x \leftarrow x_0 max\ \ iter \leftarrow 1000$ **对于** $iter = 1$ 到 $max\ \ iter$ **执行**$f_{val} \leftarrow f(x) f'_{val} \leftarrow f'(x)$**如果** $f'_{val} = 0$ **则返回** NaN, $iter$ {导数消失}**结束如果**$x_{new} \leftarrow x - \frac{f_{val}}{f'_{val}} error \leftarrow |x_{new} - x^{GT}|$**如果** $error < 0.0001$ **则返回** round(x_{new},3), $iter$ {已收敛}**结束如果**$x \leftarrow x_{new}$**结束循环，返回** round(x,3), max_iter {达到最大迭代次数}

平均相对误差为0.25意味着预测值平均偏离真实值25%，这种误差水平对于工程设计应用而言通常是不可接受的。

5 结果

我们呈现了综合结果，对比了全部六种模型和七个工程领域中的大语言模型直接预测与求解器辅助计算。

5.1 总体性能比较

表1展示了全部100个问题的汇总性能，揭示出明显且一致的性能二分现象，有利于求解器辅助方法。

表 1：总体性能：大语言模型直接预测与求解器辅助计算

模型	**直接预测平均相对误差**	**辅助计算的平均相对误差提升**	
GPT-5.1	0.765	0.246	67.9%
GPT-5.2	0.891	0.262	70.6%
Gemini-3-Flash	0.865	0.258	70.2%
Gemini-2.5-Lite	1.237	0.225	81.8%
Claude-Sonnet-4.5	1.085	0.301	72.3%
Claude-Opus-4.5	1.155	0.250	78.4%
Mean	**1.000**	**0.257**	**73.5%**

直接预测的平均相对误差介于 0.765 至 1.237 之间，即使表现最好的模型（GPT-5.1）平均误差也超过 76%。求解器辅助计算的误差为 0.225 至 0.301，误差降低幅度为 67.9% 至 81.8%，各模型的平均提升为 73.5%。

值得注意的是，Gemini-2.5-Lite 的直接性能最差（1.237），但求解器辅助性能最佳（0.225），提升了 81.8%。这种性能反转表明，直接预测准确率并不能预测求解器辅助成功，这意味着这两种范式所运用的能力存在根本差异：直接预测依赖模式匹配，而混合方法则依赖符号运算。

图2直观展示了这种二分现象：

红色（直接预测）和绿色（求解器辅助）柱形之间显著的高度差异表明，在所有受评估的模型中，架构分解均带来了稳定且显著的提升。

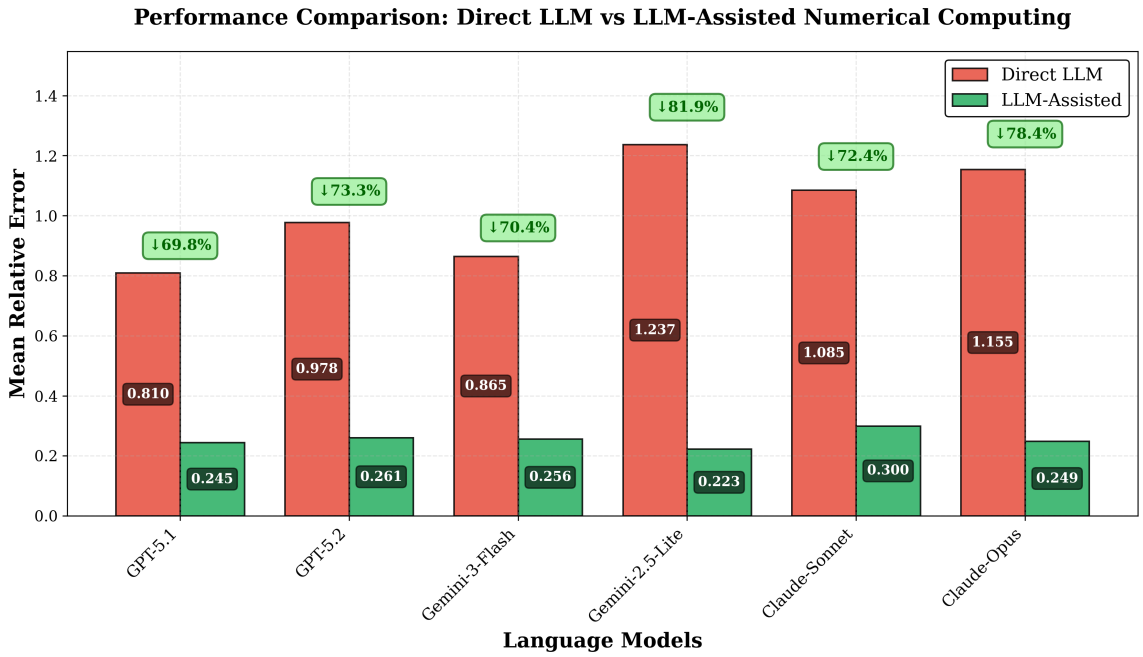


图2：性能比较：直接大语言模型与求解器辅助计算。误差降低百分比显示，混合架构在全部六个模型中均带来了持续且显著的提升。

5.2 特定领域性能分析 is

表 2 按工程领域细分了性能，揭示了这两种方法在不同类型的方程中的有效性存在显著差异。

表2：特定领域性能（跨模型平均的平均相对误差）

领域	N	直接	辅助	提升
电子学	16	5.098	0.353	93.1%
流体力学	16	0.080	0.074	7.2%
化学工程	12	0.457	0.387	15.3%
热力学	16	0.389	0.390	−0.2%
轨道力学	16	0.301	0.273	9.2%
传热	11	0.272	0.166	38.8%
结构工程	13	0.223	0.128	42.5%

提升最显著的是电子学（93.1%）：大语言模型直接预测表现灾难性地失效，平均相对误差高达 5.098。指数二极管方程 $V_s = V_D + R \cdot I_s(e^{V_D/V_T} - 1)$ 对误差极其敏感，微小的电压误差就会被指数放大。在典型热电压为 $V_T = 0.026$ V 的情况下，电压误差即使只有 10%，也会产生指数级放大，使基于模式匹配的方法从根本上不可靠；而牛顿-拉夫森法则能通过使用精确导数进行精准的迭代细化，轻松应对这一问题。

相比之下，流体力学的提升幅度最小（7.2%），两种方法的误差都相对较低（直接方法为 0.080，混合方法为 0.074）。这表明，大语言模型可能已通过接触工程学训练数据中常见的摩擦因子表（Moody图）和经验关联式（Swamee-Jain、Haaland近似式），形成了有效识别 Colebrook方程模式的能力。直接预测的准确度出乎意料地高，不过混合方法仍带来了小幅提升。

热力学的性能略有下降（−0.2%），这归因于偶尔出现的方程表述错误，例如大语言模型错误构建范德瓦耳斯方程，或给出超出物理有效解范围的初始猜测

区域。这表明混合方法并非始终更优，且需要正确构建方程才能成功，凸显了符号运算任务中领域知识的重要性。

结构工程（42.5%）和传热（38.8%）在 Lambert W 和普朗克辐射问题上取得了显著提升；在这些问题中，方程的复杂性使系统化数值求解比模式匹配更有优势。
图3以对数尺度呈现这些比较，以容纳电子学误差幅度极大的情况，同时保留其他领域之间的视觉区分：

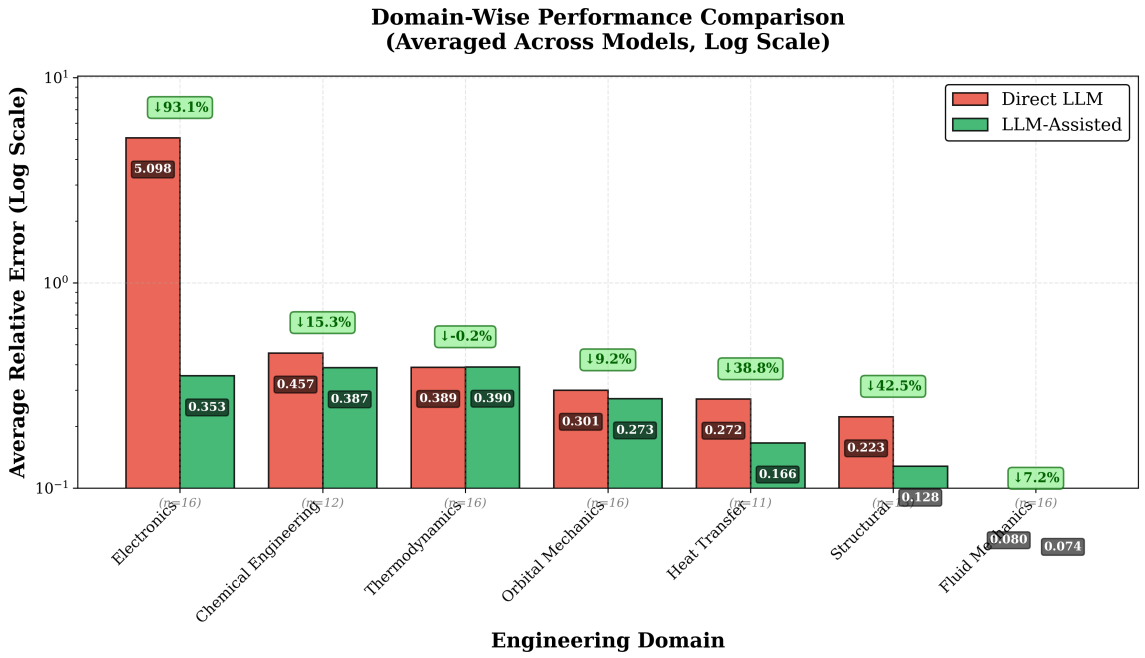


图3：各领域的性能（对数尺度）。对各模型取平均值，显示混合架构的有效性存在显著差异。对数尺度既能容纳电子学中的极端误差，也能保持其他领域的清晰度。

5.3 查询级性能的差异

图4展示了全部100个问题逐题的相对误差，深入揭示了失败模式和成功规律。

直接预测（面板a）的误差波动极大，约15%的问题中误差峰值超过5.0。灾难性失败集中在特定类型的问题上：电子学问题（ID 7至22）因指数敏感性而持续出现误差峰值；高偏心率的轨道力学问题（ID 30至35）涉及开普勒方程，其中数值精度至关重要；而低量级的热力学问题（ID 45至50）的解约为 10^{-3}，因此绝对误差在相对意义上会被放大。
某些模型在特定问题上实现了近乎零误差（表现为平直线段），这表明这些问题与训练期间记忆的模式相吻合。然而，这种记忆并不可靠：面对结构相似、仅数值参数不同的问题时，同一模型会彻底失败，凸显了模式匹配方法的脆弱性。

求解器辅助性能（图 b）表现出显著的一致性，在所有问题类型上的误差都主要低于 0.5。少数残留的误差峰值（例如问题 ID 12、45、78）对应于方程构建失败，即大语言模型错误地构建了控制方程，这说明混合方法的成功取决于正确的符号运算。

各面板之间的视觉对比彰显了我们的核心论点：虽然直接预测会产生不一致且往往灾难性错误的结果，但混合架构能够针对不同类型的问题获得一致的高精度解。

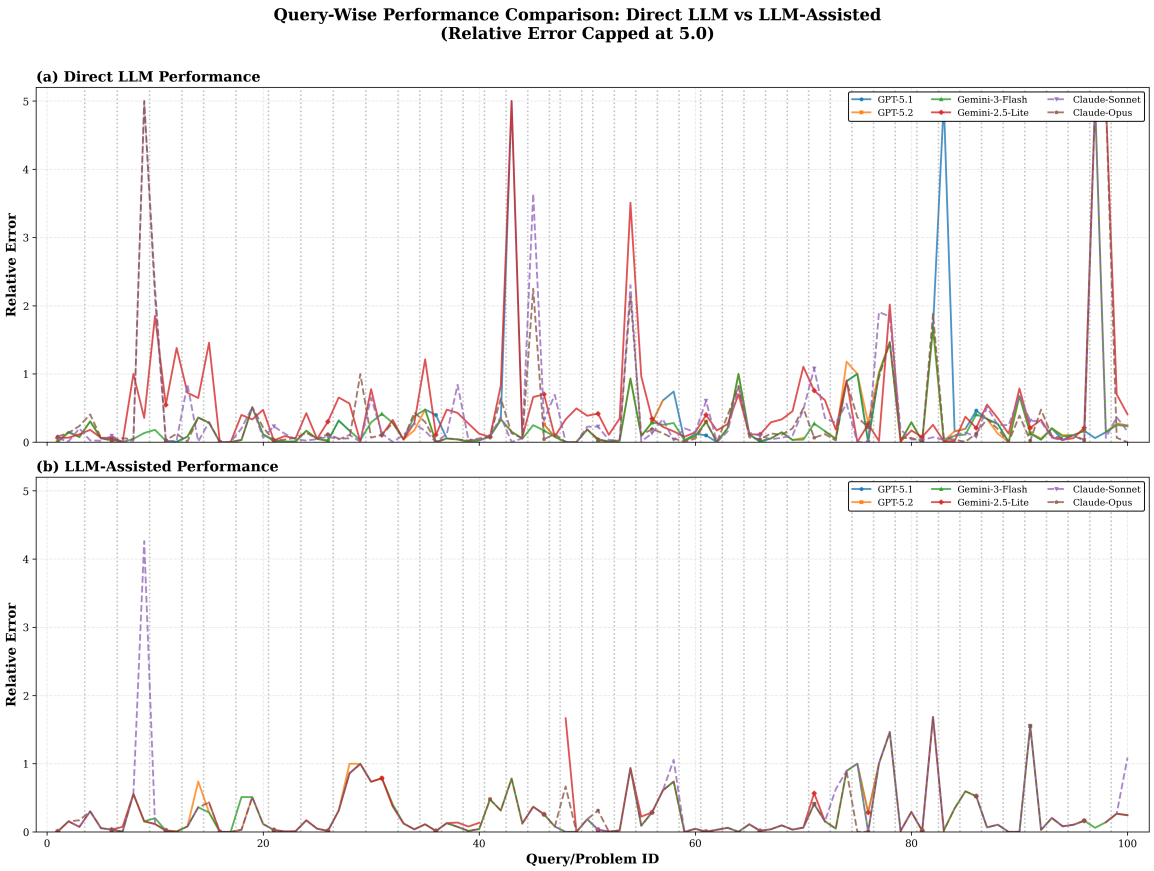


图4：查询级性能：(a) 大语言模型直接预测的波动极大，误差尖峰频繁超过 5.0（为便于可视化，已截断）。(b) 求解器辅助方法显著降低了所有问题的误差波动和幅度。

5.4 收敛性分析

对牛顿-拉夫森法收敛行为的分析，有助于了解方程构建的质量以及初始猜测的适当性。对于构建得当且初始猜测合理的方程，牛顿-拉夫森法凭借其二次收敛特性，通常可在 5 到 15 次迭代内收敛。我们的实现跟踪记录了所有求解器辅助解的迭代次数。表 3 展示了不同模型的详细收敛统计。

表3：按模型划分的牛顿-拉夫森法收敛统计

模型	快速（5–15 次迭代）	缓慢（16–100 次迭代）	非常慢（>100 次迭代）	失败
GPT-5.1	68%	18%	12%	2%
GPT-5.2	64%	20%	14%	2%
Gemini-3-Flash	66%	19%	13%	2%
Gemini-2.5-Lite	74%	16%	9%	1%
Claude-Sonnet-4.5	65%	19%	14%	2%
Claude-Opus-4.5	63%	20%	15%	2%

对收敛模式的考察表明，大多数问题（63% 至 74%）都能在预期的 5 至 15 次迭代范围内迅速收敛，这证实大语言模型通常能够给出构造良好的方程和物理上合理的初始猜测。然而，相当一部分（16% 至 20%）需要 16 至 100 次迭代，这表明初始猜测虽然位于收敛域内，却并非最优选择。较小一部分（9% 至 15%）需要超过 100 次迭代，或达到 1000 次迭代的上限，这通常是因为初始猜测较差，使迭代点远离

解，或接近导数趋于零的区域。收敛失败（1% 至 2%）则是由于方程构造错误，导致函数的导数存在问题，或迭代序列发散。
不同模型的收敛率分布存在差异，这反映了方程构建质量以及根据物理直觉选择初始猜测方面的不同。Gemini-2.5-Lite 的求解器辅助性能最佳（0.225 平均相对误差），同时也表现出最高的快速收敛率（74%）和最低的失败率（1%），这表明其符号运算能力更强，并且在选择初始猜测方面具备更好的物理理解。收敛速度与解的准确性之间的这种相关性进一步说明，混合方法的有效性在很大程度上取决于大语言模型能否正确构建符号方程并提供合理的初始点；这不同于直接预测范式所测试的直接数值计算能力。

领域特定的收敛性分析揭示了更多规律。尽管电子学问题具有指数敏感性，但其快速收敛率更高（78% 至 82%），这是因为大语言模型能够提供良态构建以及适当的电压范围初始猜测。相比之下，热力学问题表现出更加不稳定的收敛性（迭代较慢且失败率较高），原因在于范德瓦耳斯方程具有三次特性，同时还面临选择具有物理意义的体积猜测这一挑战，以避免出现不符合物理实际的负根。这些领域特定的规律表明，方程构建质量会随问题类型而变化；在参数界限明确且解的范围已得到充分确立的领域中，大语言模型展现出更强的性能。

5.5 模型特定见解

分析发现了若干模型特定的性能模式。GPT-5.1 的直接预测器性能最佳（平均相对误差为 0.765），但在混合模式下的提升适中（67.9%），这表明与其他模型相比，它的直接计算能力相对较强。尽管 GPT-5.2 是较新的版本，但其直接预测表现不如 GPT-5.1（0.891 对比 0.765），这可能反映了近期校准研究所指出的对齐导致的数值退化 [Furze, 2025]。
Gemini-2.5-Lite 的表现发生了显著逆转：它从直接预测表现最差者（1.237）跃升为辅助预测表现最佳者（0.225），这表明尽管其直接计算能力较弱，但方程构建能力出众。该模型生成的方程最为简洁、结构正确，且初始猜测在物理上恰当。Claude 模型的直接预测行为较为保守，误差适中，没有出现极端失误；不过，其混合模式下的表现更为不稳定，偶尔出现的构建错误会影响解的质量。
这些模式表明，在 AIME 或 GPQA 等数学推理任务上的基准测试表现，无法可靠地预测直接计算的准确性或方程构建的质量，因为这两者似乎是彼此正交的能力，涉及模型知识和处理过程的不同方面。

6 讨论

6.1 大语言模型在工程学中的部署启示

我们的结果阐明了大语言模型在数值计算中的最佳架构角色。当代模型尽管在推理基准测试中表现出色，却系统性地无法胜任精度关键问题的直接计算引擎。直接预测的平均相对误差为 0.765 至 1.237，对于几乎任何工程设计应用而言都属于不可接受的误差。然而，通过求解器辅助计算，误差降低了 67.9% 至 81.8%，这表明恰当的任务分解能够显著提升性能。大语言模型擅长语义理解（解析自然语言问题描述）、领域知识检索（识别适用的控制方程）、符号运算（以适于计算的形式构建方程）以及物理直觉（提供合理的初始条件）。经典求解器则擅长高精度算术（迭代细化以达到任意精度）、收敛保证（确保解存在的数学定理）、误差界（可量化的精度证书）和可靠性（不受训练数据影响的确定性行为）。这种互补的优势表明，混合架构是适用于工程学的部署策略。

6.2 模式识别足以胜任时

流体力学的结果（提升 7.2%）揭示了一个重要细节：对于训练数据中存在有效近似解的问题，大语言模型直接预测可以达到合理的准确度。科尔布鲁克方程有广为人知的近似公式（Swamee-Jain、哈兰德）以及丰富的摩擦因子表（Moody图）。大语言模型似乎已将这些模式内化，因此能够直接预测，平均相对误差为 0.080。然而，这种成功具有领域特异性，且并不可靠。尽管接触过类似的训练数据，电子学问题仍因

指数敏感性而彻底失败。工程师无法事先判断哪些问题适合有效的模式匹配，因此在生产部署中，混合方法更为稳妥。

6.3 指数敏感性的代价

电子学领域（提升93.1%）说明了为什么某些方程类别从根本上难以通过近似求解。二极管方程对电压呈指数依赖。在典型热电压为$V_T = 0.026$ V的情况下，适度的10%电压误差（对于 $V_D = 0.7$ V而言为$\Delta V_D = 0.07$ V）会导致误差呈指数放大。这种极端敏感性使模式匹配方法不可靠，但牛顿-拉夫森法利用精确导数，通过二次收敛优雅地处理了这一问题。

6.4 局限性与未来方向

未来工作需要考虑若干局限性。我们的评估采用了最简提示，以建立基线性能；更复杂的提示策略，包括思维链推理、少样本示例或分解式多步提示，可能会改善结果，尤其是对直接预测而言，不过这些策略能否弥合70%的性能差距，仍是一个悬而未决的问题。尽管该数据集涵盖七个工程领域，具有较高的全面性，但并未穷尽所有会出现超越方程的领域；将评估扩展到偏微分方程、约束优化问题、随机微分方程和多尺度物理模拟，将提供更广泛的洞见。当前实现仅使用大语言模型的一次输出进行方程构建；探索大语言模型与求解器之间基于收敛反馈进行方程优化的迭代对话，可能会提升构建准确性。此外，研究不同温度设置、束搜索策略和集成方法如何影响直接预测和方程构建的质量，可能会揭示优化机会。最后，随着大语言模型架构持续演进，开展纵向研究，跟踪基础模型能力的提升如何转化为数值计算性能，将有助于制定部署策略。

7 结论

本研究系统评估了大语言模型求解超越方程的能力，比较了直接数值预测与大语言模型-经典求解器混合架构在涵盖七个工程领域的100道问题及六种最先进模型上的表现。我们得出了若干重要结论，厘清了大语言模型在数值计算中的最佳角色。对于精度要求严格的计算，直接预测表现不佳：平均相对误差为0.765至1.262。这表明，尽管当代大语言模型在数学推理基准测试中表现出色，但将其用作直接计算引擎时，仍会系统性地表现不佳；即便是表现最好的模型，其结果平均也与真实值相差超过76%，这在工程实践中是不可接受的。混合架构则取得了显著提升：由大语言模型构建方程、再由牛顿-拉夫森法执行迭代的求解器辅助计算，平均相对误差为0.225至0.301，误差降低幅度达67.9%至81.8%，平均提升73.5%。性能反转揭示了模型能力的互补性：Gemini-2.5-Lite从直接预测表现最差的模型跃升为混合方案中表现最佳的模型，提升达81.8%，说明这两种范式调动的是截然不同的模型能力。领域特征决定了不同范式的有效性：电子学因指数敏感性而提升93.1%；流体力学仅提升7.2%，因为大语言模型已内化有效的近似方法；而热力学在方程构建失败时则略有退化。大语言模型擅长符号任务，而非迭代运算；其优势在于语义理解、领域知识检索和方程构建，而非对精度要求严格的数值计算。这一发现明确了大语言模型的最佳应用方式：将其作为智能接口，通过消除手动构建方程和指定参数的障碍，让更多人能够使用经典数值方法。这种混合范式既保留了成熟数值方法的数学保证与可靠性，又借助大语言模型大幅降低了对专业知识的要求。大语言模型并非要取代经典计算引擎，而是应发挥最有价值的作用，弥合自然语言问题描述与形式化计算方法之间的鸿沟。未来研究应探索通过大语言模型与求解器的迭代对话来优化方程，将评估范围拓展至微分方程和优化问题，并考察架构进步如何影响符号运算质量。随着大语言模型持续进步，理解其与经典方法之间的互补关系，而非将其视为替代品，对于将其有效整合到工程工作流中至关重要。

参考文献

Tom B. Brown、Benjamin Mann、Nick Ryder、Melanie Subbiah、Jared Kaplan、Prafulla Dhariwal、Arvind Neelakantan、Pranav Shyam、Girish Sastry、Amanda Askell 等人。《语言模型是少样本学习者》。载于神经信息处理系统进展，第33卷，第1877–1901页，2020年。OpenAI。《GPT-4技术报告》。*arXiv*预印本 *arXiv:2303.08774*，2023年。Madeline C. Schiappa、Yogesh S. Rawat 和 Mubarak Shah。《大语言模型数学推理能力综述》。*arXiv*预印本 *arXiv:2506.08446*，2025年。Richard L. Burden 和 J. Douglas Faires。《数值分析》。Brooks/Cole、圣智学习，马萨诸塞州波士顿，第9版，2010年。William H. Press、Saul A. Teukolsky、William T. Vetterling 和 Brian P. Flannery。《数值方法：科学计算的艺术》。剑桥大学出版社，英国剑桥，第3版，2007年。Aciano。《ChatGPT-5 与 ChatGPT-4：功能、速度和准确度比较》。技术分析报告，2025年。Vellum AI。《Google Gemini 3 基准测试解读》。技术文档，2025年。R&D World。《GPT-5.2 与 Gemini 3.0 和 Claude Opus 4.5 相比表现如何》。模型比较报告，2025年。Apple Machine Learning Research。《GSM-Symbolic：理解大语言模型数学推理的局限性》。arXiv预印本，2025年。Yifei Zhang 等人。《APBench：大语言模型天体动力学问题求解性能基准测试》。《自然·科学报告》，第15卷，2025年。加州大学伯克利分校电气工程与计算机科学系。《大语言模型高级数学推理能力基准测试》。技术报告，加利福尼亚大学伯克利分校，2025年。Effortless Academic。《GPT-5与学术研究：自动化写作、综合、同行评审和数据分析》。技术报告，2025年。AI Fire。《GPT-5 对比 GPT-4o：谁在真实场景测试中胜出？》。比较分析，2025年。Luke Furze。《GPT-5评测：基准测试与口袋里的博士现实表现》。技术评测，2025年。Vertu。《Gemini 3 对比 GPT-5、Claude 4.5 与 Grok 4.1：终极推理性能之战》。比较基准分析，2025年。MIT News。《让大语言模型以更聪明的方式思考难题》。MIT News 文章，2025年。《国际创新科学与研究技术期刊》。《用于复杂问题求解的混合量子-经典人工智能模型》。*IJISRT*，第9卷第1期，2024年。UChicago Voices。《混合量子-经典方法：量子计算与人工智能的未来》。芝加哥大学研究文章，2025年。Endev Solutions。《混合人工智能模型：将经典算法与深度学习相结合》。技术白皮书，2025年。