Exploring the Limits of Fine-grained LLM-based Physics Inference via Premise Removal Interventions

Jordan Meadows[1*], **Tamsin James**[2*], **André Freitas**[1,3,4]
[1]University of Manchester, United Kingdom
[2]University of Birmingham, United Kingdom
[3]Idiap Research Institute, Switzerland
[4]National Biomarker Centre, CRUK-MI, United Kingdom

Abstract

Language models (LMs) can hallucinate when performing complex mathematical reasoning. Physics provides a rich domain for assessing their mathematical capabilities, where physical context requires that any symbolic manipulation satisfies complex semantics (*e.g.,* units, tensorial order). In this work, we systematically remove crucial context from prompts to force instances where model inference may be algebraically coherent, yet unphysical. We assess LM capabilities in this domain using a curated dataset encompassing multiple notations and Physics subdomains. Further, we improve zero-shot scores using synthetic in-context examples, and demonstrate non-linear degradation of derivation quality with perturbation strength via the progressive omission of supporting premises. We find that the models' mathematical reasoning is not physics-informed in this setting, where physical context is predominantly ignored in favour of reverse-engineering solutions.

1 Introduction

Language models demonstrate some level of mathematical ability (Lewkowycz et al., 2022; Liu et al., 2023; Azerbayev et al., 2023; Pan et al., 2024). This reasoning modality requires controlled symbolic behaviour involving the repeated application of mathematical operations (Valentino et al., 2023; Meadows et al., 2023a), and LMs struggle to deliver this reliably (Frieder et al., 2023; Liu and Yao, 2024). A particularly challenging mathematical domain is that of *Physics*, where equation derivations serve as a rich environment within which the mathematical inference capabilities of LMs may be thoroughly examined, yet in contrast to other forms of mathematical reasoning (Shakarian et al., 2023; Yuan et al., 2023; Wang and Lu, 2023), there are few examples of such efforts (Lewkowycz et al., 2022). While Mathematics proofs are closer to a logical argument over more abstract domain types, Physics derivations are centered around the integration of the abstraction of physical properties and laws into approximations and premises, where algebraic and calculus-related symbolic manipulations are performed to obtain novel equations, through a step-wise derivation, in close dialogue with empirical evidence.

An explanation for the lack of Physics-related approaches (Luo and Liu, 2018; Wu and Tegmark, 2019; Eivazi et al., 2022; Lewkowycz et al., 2022) is the fact that automating scientific discovery in mathematics has a long tradition in the context of automated theorem provers and proof assistants (Jiang et al., 2022; Lample et al., 2022), and this relies on the translation of mathematical proofs into logical forms (Szegedy, 2020; Wu et al., 2022). However, much of Physics is less compatible with logical formalisation (Kaliszyk et al., 2015; Davis, 2019; Meadows and Freitas, 2021; Yang et al., 2024; Davis, 2024). The flexibility of transformer-based models as soft reasoners (Clark et al., 2020) offers the opportunity to circumvent formalisation requirements, and develop models capable of detailed mathematical reasoning based on informally defined scientific knowledge. However, as with any form of mathematical reasoning, these models need to be built around models that support controlled, step-wise, symbolic inference.

One reason why current LMs fail in this regard, is because online resources used for training (such as Wikipedia and arXiv), and many mathematical datasets (Mishra et al., 2022; Hendrycks et al., 2021; Saxton et al., 2019), do not feature the required detail necessary for fine-grained reasoning. Physics derivations contain specific notations (*e.g.,* Dirac notation) that are relied upon to build symbolically complex text spans, and complex operations (*e.g.,* Laplace transform, Taylor expansion)

* Equal contributions.
Email addresses: jordan.meadows@postgrad.manchester.ac.uk; txj287@student.bham.ac.uk; andre.freitas@idiap.ch.

Findings of the Association for Computational Linguistics: EMNLP 2024, pages 6487–6502
November 12-16, 2024

form sophisticated dependencies between textual elements. Moreover, the derivations presented in papers and textbooks omit a significant number of steps, reinforcing difficulties in training models to perform more detailed calculations.

$$g(x) = \frac{\hbar}{2\pi i} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} e^{i(x-\chi)b} d\chi db$$

$$\int_{-\infty}^{\infty} e^{i(x-\chi)b} db = 2\pi\delta(x-\chi)$$

$$\frac{\hbar}{2\pi i} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} 2\pi\delta(x-\chi) d\chi db = \frac{\hbar}{i} \int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} \delta(x-\chi) d\chi$$ 1

$$\int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} 2\pi\delta(x-\chi) d\chi = 2\pi \frac{d\psi(x)}{dx}$$

$$\frac{\hbar}{i} \int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} \delta(x-\chi) d\chi = \frac{\hbar}{i} 2\pi \frac{d\psi(x)}{dx}$$ 2

$$g(x) = \frac{\hbar}{i} \left(\frac{d\psi(x)}{dx}\right)$$

Figure 1: An incorrect derivation generated by few-shot GPT-4 that scores high ROUGE (81), BLEU (71), and GLEU (71). Erroneous equations are denoted in red.

For instance, in Fig. 1, GPT-4 fails a derivation (from the present data) related to the Uncertainty Principle in Quantum Mechanics. The first error arises from attempting to substitute the RHS of the second equation into the RHS of the first, but GPT erroneously keeps the $\int_{-\infty}^{\infty} db$ in the LHS. The second error involves an incorrect evaluation of an integral over a Dirac delta function (by a factor of 2π), which is a critical result in Physics. Real-world research inherently involves equation manipulation which is out-of-distribution with respect to models' training data, either through a derivation's specific use of notation or its underlying reasoning. If state-of-the-art LMs fail at such basic manipulation, and if current evaluation metrics fail to account for such fine-grained errors (Welleck et al., 2022a; Meadows et al., 2023a), to what extent are leading methods appropriate for inference in mathematical domains (Davis, 2024)?

This paper aims to explore these considerations, contributing with:

(1.) A manually curated step-wise granular dataset comprising 1200 derivation steps over 218 fine-grained Physics derivations at approximately graduate-level difficulty. Examples span a diverse set of subdomains including electromagnetism, classical, quantum, and statistical mechanics. The derivations are aligned to reference those on Wikipedia, but have been manually augmented to include finer steps, and are mapped to prompts containing premises and goal equations.

(2.) Zero-shot and few-shot evaluation of GPT-4, GPT-3.5, and T5-related models on a *Derivation Generation* task using common text generation metrics, paired with a manual evaluation, to highlight model reasoning limitations related to Physics.

(3.) An exploration of models' out-of-distribution mathematical abilities centered on a controllable *Premise Removal* intervention.

Through these contributions, we measure and qualify the ability of current LMs in performing out-of-distribution, fine-grained, multi-step mathematical reasoning, and aim to provide empirical foundations for developing transformer-based reasoners suitable for assisting discovery in Physics and related fields.

2 Related work

Our focus is on the generation of step-wise detailed derivations of Physics equations with LMs (Brown et al., 2020; Ahmed and Devanbu, 2022; Song et al., 2022; Ge et al., 2023; Hu et al., 2023; Yang et al., 2023). While we presently consider *solely equations*, math generation exists in various forms, split between approaches that consider formal languages (First et al., 2023; Polu et al., 2022; Jiang et al., 2021; Polu and Sutskever, 2020) and those considering informal mathematical natural language (Ferreira and Freitas, 2020; Welleck et al., 2021; Ferreira et al., 2022; Valentino et al., 2022a; Meadows and Freitas, 2023). Informal reasoning approaches generally involve code generation as input to symbolic solvers (He-Yueya et al., 2023; Chen et al., 2022; Drori et al., 2022; Mandlecha et al., 2022; Hu and Yu, 2022; Chen et al., 2021), or directly generating math reasoning in natural language (Lewkowycz et al., 2022; Welleck et al., 2022a; Chowdhery et al., 2022; Lample and Charton, 2019). Numerous approaches exist for evaluating the mathematical and symbolic capabilities and robustness of models (Welleck et al., 2022b; Stolfo et al., 2022; Meadows et al., 2023b), in contexts such as solving math word problems (Roy et al., 2015; Liang et al., 2022; Yao et al., 2023). Various datasets exist containing mathematical reason-

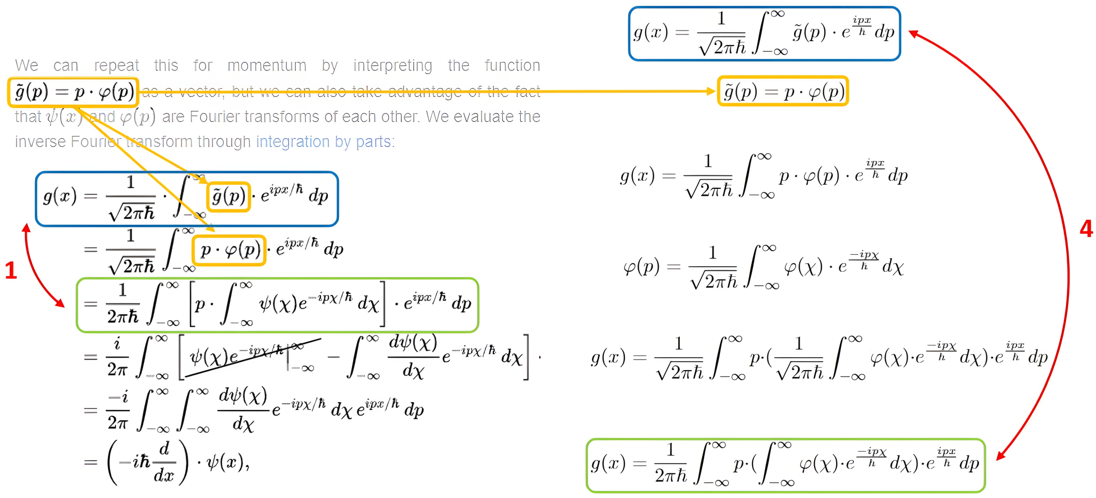

Figure 2: The difference between the Wikipedia proof (left) and our equational interpretation (right) of a reasoning chain related to the Uncertainty Principle in quantum mechanics. The (red) values represent the number of intermediate equations between equivalent equations in each representation, and highlights the detail gap.

ing (Ferreira and Freitas, 2020; Hendrycks et al., 2021; Welleck et al., 2021; Mishra et al., 2022), and to a lesser extent, Physics (Hendrycks et al., 2021; Lewkowycz et al., 2022; Meadows et al., 2022; Pan et al., 2024). We narrow our scope to isolate equational content from natural language descriptions, for the purpose of testing purely the equation manipulation capabilities of LMs in the Physics domain.

To summarise, we consider an equation Derivation Generation task, contribute a *fine-grained* Physics dataset spanning multiple subdomains, perturb input prompts with a Premise Removal intervention, and explore model performance and degradation due to perturbations. We aim for this data to improve the detailed step-wise equation derivation capabilities of LMs, and use it to analyse their ability to perform Physics reasoning with complex equational forms (Fig. 1 and 2). We later highlight the difference between coherent mathematical and physical reasoning (Fig. 4), and discuss underlying generation degradation laws and how LMs perform mathematics in this context.

3 Physics Dataset Construction

To elicit the level of mathematical detail and coherence from models as described in Section 1, we randomly select a number of derivations from Wikipedia spanning Electromagnetism, Quantum, Classical and Statistical Mechanics, and expand each example until the required granularity is obtained (approximately one operation per step). We rely on the support of two annotators with adequate Physics expertise (Master's level).

Fig. 2 gives an example of this rewriting and the departure from natural language. With respect to granularity differences, the single intermediate step (denoted by red) in the Wikipedia derivation on the left, is actually composed of multiple fine-grained steps naturally omitted for the sake of succinct communication online. Experienced physicists may indeed skip these steps within their own workings, but there is no guarantee that LMs can reliably perform them without error (Welleck et al., 2022a; Frieder et al., 2023; Meadows et al., 2023a; Liu and Yao, 2024; Quan et al., 2024). On the right-hand side, this single step is expanded into four finer-grained steps that *improve explainability*. The first and second equations in the expanded derivation are premises extracted from the text. The third equation is formed by substituting the second premise into the first, and the fourth equation is not explicitly written anywhere – it is the Fourier transform of $\varphi(\chi)$ described non-mathematically within the initial description now included for completeness. The remaining equations are obtained through substitution.

We algorithmically describe this expansion and annotation process for converting online derivations into examples rendered in Appendices B-D, in Alg. 1. This protocol was applied by each annotator within a double swap, review and refine setting. Initial derivations were split between each annotator, and after initial annotation, the datasets were swapped for review and changes were tracked. A second swap was performed for accepting the proposed changes.

The inclusion of the (implicit) Fourier transform and the (explicit) premise in Fig. 2, give an example of how a combined modality of natural language and equations is mapped into a single equational modality. The expansion of one step into four, describes the emphasis towards the explainable reasoning we wish to elicit from models.

High-quality (Villalobos et al., 2022) data of this kind is necessary to bridge an incompleteness gap between the communicative reasoning available online (used to train/evaluate models) and the reality of Physics calculations (Pan et al., 2024; Akrobotu et al., 2022; Meadows and Freitas, 2021; Mann et al., 2018; Hopfield, 1958). Establishing that models can robustly perform such reasoning is a prerequisite enabling their utility and application in theoretical discovery.

Algorithm 1 Derivation Annotation

1: `Define premises` from a given initial derivation.
2: `Derive intermediate equations` between initial equations.
3: `Re-write derivation` asserting one operation per step (approximately).
4: `Write derivation in LaTeX` as a sequence of equations.
5: `Annotate` "%PREM" within equation environments of premises.
6: `Re-organise LaTeX derivation` into self-contained 4-9 step sub-derivations.
7: `Re-annotate new sub-derivation premises` with "%PREM".
8: `Output multiple derivations` in LaTeX per initial derivation.

Data Analysis. We expand 60 derivations from Wikipedia examples following the discussed protocol, and extract 218 shorter examples resembling the right-hand side of Fig. 2. These are split into three categories: **Electromagnetism** (containing vector calculus), **Quantum Mechanics** (containing Dirac notation and commutation relations), and **Other Physics** (containing results from *classical* and *statistical mechanics*). Tab. 1 describes the number of examples within each subdomain, and example derivations are rendered in Appendices B (electromagnetism), C (quantum), and D (other).

Smaller models, such as BERT (Devlin et al., 2018) and T5 (Raffel et al., 2020), are limited to input sequences of up to 512 tokens. This corresponds to a LaTeX derivation comprising between 4-9 equations. The original expanded derivations are segmented into shorter examples to accommodate these limitations. The relevant length distributions are given in Fig. 3 alongside that of the synthetic dataset used to fine-tune MathT5 in related work (Meadows et al., 2023a), and to provide in-context examples for few-shot prompts in these experiments. Other similarities with the synthetic dataset include an overlap of 155 symbols and a similar step granularity.

Field of Physics	**# Derivations**
Electromagnetism	82
Quantum Mechanics	98
Other Physics	38
All fields	218

Table 1: Number of derivations in the dataset by field of Physics. The *Other* category corresponds to results from classical and statistical mechanics.

Otherwise, the Physics derivations contain significantly more symbol combinations, including limits of integration and entirely separate notation. In particular, this includes *Dirac notation* that is commonplace in Quantum Mechanics (such as $\int \langle x|\Psi\rangle^\dagger x'\delta(x-x') \langle x'|\Psi\rangle \, dx' = \langle x|\Psi\rangle^\dagger x \langle x|\Psi\rangle$, from C.3), and *vector calculus* involving the div, grad, and curl operators which are commonplace in Electromagnetism (such as $\nabla \cdot (\phi\nabla\phi) = (\nabla\phi)^2 + \phi\nabla^2\phi$, from B.2).

Premises are another crucial element of derivations. These are axiomatic equations deemed necessary for deriving a goal equation. The distribution of the number of premises per derivation is given in Fig. 4. A significant proportion of steps involves simply writing the premises and goal equation in the correct order with correct syntax (a non-trivial task for LMs (Chen et al., 2024; Meadows et al., 2023a)), or substituting expressions.

A final noteworthy property of the dataset is its compositionality towards longer derivations. Premises are separated from non-premises, and goal equations from some examples are premises in another, meaning that sequences may be chained together to form much longer examples of up to 20 steps. The dataset itself contains both prompts and target derivations. Output text takes the form of LaTeX equations conjoined by an "and" token, which provides a minimalistic template for defining equation boundaries. The dataset is available online[1], and includes lists of derivation-specific premises and references to original Wikipedia names alongside each example.

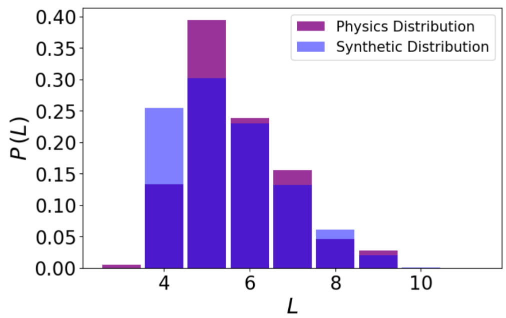

Figure 3: $P(L)$ is the probability that a given derivation contains L equations.

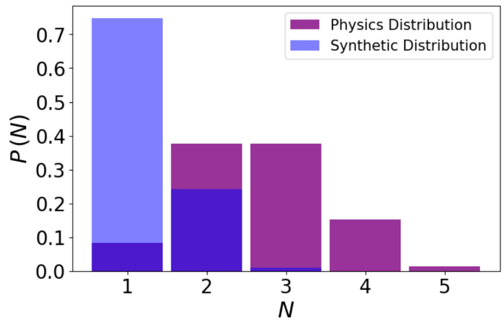

Figure 4: $P(N)$ is the probability that a given derivation contains N premise equations.

4 Derivation Generation and Generalisation Capabilities

The experimental analysis occurs in two parts: evaluation on an in-distribution test set in the setting of a Derivation Generation task, followed by an out-of-distribution evaluation facilitated by the progressive removal of premises from in-distribution prompts. Perturbed prompts are out-of-distribution with respect to either a model's training data or in-context examples.

4.1 The Derivation Generation task

Given a goal equation G and premises $\mathcal{P}$ arranged within some prompt template $t(\mathcal{P}, G)$, a given model $\mathcal{M}$ must generate a sequence of equations $\hat{\mathcal{D}}$ which represents a reasonable derivation of G. A derivation is generated through $\mathcal{M} : t(\mathcal{P}, G) \mapsto \hat{\mathcal{D}}$, which is then compared to an idealised ground truth $\mathcal{D}^*$. Some idealised metric M^* scores the generated derivation through $M^* : (\mathcal{D}^*, \hat{\mathcal{D}}) \mapsto \mathcal{S}$. Assuming a suitable prompt t, we generally aim to optimise the following problem to find the best model according to the given metric, through:

$$\mathcal{M}^* = \underset{\mathcal{M}}{\text{argmax};} \, M^*\big(\mathcal{D}^*, \mathcal{M} : t(\mathcal{P}, G) \mapsto \hat{\mathcal{D}}\big).$$

However, we do not have access to ideal derivations $\mathcal{D}^*$ corresponding to templates $t(\mathcal{P}, G)$, as many derivations may reasonably derive G, yet may differ from $\mathcal{D}^*$. We also do not have access to ideal metric M^* suitable for accurately scoring individual $\hat{\mathcal{D}}$. Instead, we manually produce coherent ground truths $\tilde{\mathcal{D}}^*$ and *assume* that the quality of model derivations is reflected monotonically (on average) with scores obtained from the text generation metrics. We then conventionally determine $\mathcal{M}^*$ through

$$\mathcal{M}^* = \underset{\mathcal{M}}{\text{argmax};} \, \tfrac{1}{N} \textstyle\sum_{i=1}^{N} M\big(\tilde{\mathcal{D}}^*{}_i, \mathcal{M} : t(\mathcal{P}_i, G_i) \mapsto \hat{\mathcal{D}}_i\big),$$

where N is the number of ground truth derivations. In this work, we consider M as a reference-based generation metric (*e.g.,* ROUGE) used to evaluate derivations at scale, but we contrast this with a reference-free human evaluation of derivations. An example prompt $t(\mathcal{P}, G)$ is given below.

$$\textit{Given} \quad q(a) = e^a$$

$$\textit{and} \quad G(a) = -e^a + \tfrac{d}{da} q(a),$$

$$\textit{then obtain} \quad e^{G(a)} = 1$$

Prompting LMs. The specific details for zero-shot and few-shot prompts are described in Appendix A, but we outline a brief summary description. For zero-shot prompts, a simple task description is prefixed to the above template which emphasises the equational focus of the output template. For few-shot prompts, a total of 5 in-context examples are selected from the synthetic training set (used to train MathT5) following Meadows et al. (2023a). We note an inherent similarity with chain-of-thought prompting (Wei et al., 2023) due to the nature of the task, but prompts deviate from this due to the exclusion of natural language.

4.2 Controlled Premise Removal

The goal of premise removal is to systematically remove crucial mathematical and physical context

[1] https://github.com/jmeadows17/transformers-for-physics

from prompts, such that a given model must either recall this missing context *and* use it appropriately, or derive the goal equation via an alternative route. If a model understands the underlying reasoning, then the removal of premises should not incur derivation errors. If errors do occur as a function of the number of premises removed, we can study how certain reasoning capabilities degrade as the strength of the perturbation increases.

The prompt template accommodates the removal of premises without introducing uncontrolled perturbations to other mathematical terms or natural language. This relatively pure intervention on the input space (Stolfo et al., 2022; Pearl, 2009) introduces a secondary *premise selection* problem (Alama et al., 2014; Wang et al., 2017; Ferreira and Freitas, 2020; Valentino et al., 2022b; Meadows and Freitas, 2023) in tandem with the main task. If successful Derivation Generation involves the application of mathematical operations to equations during inference, such as algebraic manipulation and calculus, then Premise Removal introduces the requirement that models must either generate a missing premise, or find an alternative derivation route. We can increase the severity of the distribution shift (Fig. 4) by progressively removing premises.

More formally, if the prompt $t(\mathcal{P}, G)$ is a template containing (ordered) premises $p_i \in \mathcal{P}$, and the goal equation G, through string concatenation $(+)$ we can define this perturbation as

$$t(\mathcal{P}, G, S; \alpha, \beta, \gamma) = \alpha + p_1 + \sum_{i=2}^{|\mathcal{P}|-S-1} (\beta + p_i) + \gamma + G \tag{1}$$

where α, β, γ are natural language sequences held constant with respect to any p_i or G. Importantly, *S directly controls the perturbation strength.* S premises are removed from the prompt (reverse chronologically) such that if $S = 0$ we recover the original in-distribution prompt. In this work we consider $S \in \{0, 1, 2\}$, and are hence restricted to derivations containing $|\mathcal{P}| \geq 3$ (about half of the data, Fig. 4), as at least one premise is required by the prompt template.

To briefly demonstrate, the example prompt in Section 4.1 corresponds to $(|\mathcal{P}|, S) = (2, 0)$, whereas $(|\mathcal{P}|, S) = (2, 1)$ corresponds to the prompt below (*i.e.,* $\alpha + p_1 + \gamma + G$). Both would be excluded from this analysis as $|\mathcal{P}| = 2$.

Given $\quad q(a) = e^a$

then obtain $\quad e^{G(a)} = 1$

5 Evaluation

We describe details of models and metrics in Appendix A. Here we report key results from the Derivation Generation experiments, beginning with an evaluation of unperturbed prompts (Section 5.1) followed by an exploration of performance degradation due to Premise Removal perturbations (Section 5.2).

5.1 Derivation Generation

	ROUGE	**BLEU**	**GLEU**
T5-base	13.6	1.8	7.2
T5-large	9.5	1.0	5.0
FLAN-T5-base	9.2	2.7	6.4
FLAN-T5-large	7.0	0.6	3.5
MathT5-base	70.6	58.6	60.7
MathT5-large	69.6	60.2	62.1
GPT-3.5 *(ZS)*	56.8	48.3	51.3
GPT-3.5	77.7	67.3	70.3
GPT-4 *(ZS)*	77.1	60.3	66.4
GPT-4	**84.3**	**78.0**	**79.1**

Table 2: Evaluation results for derivation generation with the Physics dataset. *(ZS)* refers to zero-shot performance. Otherwise, all T5 results are zero-shot, and GPT results are few-shot.

Synthetic in-context examples improve inference. The few-shot approach used to prompt the GPT models does not use Physics derivations as in-context examples, and therefore *does not learn any biases present in the Physics data*. Instead, synthetic derivations are included in prompts that demonstrate the required granularity between steps, while helping to force model output into the required template. Few-shot learning over general mathematical text significantly improves performance over zero-shot prompting, and notably, few-shot GPT-3.5 outperforms zero-shot GPT-4 across all metrics in Tab. 2.

Agreement between models on the relative difficulty of Physics domains. The dataset is divided into the Electromagnetism, Quantum Mechanics, and Other subdomains (Tab. 1). According to the metrics, model derivations obtain highest scores on Electromagnetism and lowest on Other (Tab. 3). Notably, the quantum derivations are generally the most difficult and feature domain specific (braket) notation and relatively complex equational forms, so it is unexpected that models generate more co-

	Field	ROUGE	BLEU	GLEU
MathT5-base	*EM*	75.9	64.8	66.3
	QM	68.4	55.7	58.2
	Other	63.2	50.6	52.9
MathT5-large	*EM*	75.1	67.9	69.1
	QM	67.9	57.3	59.2
	Other	60.0	48.6	52.3
GPT-3.5	*EM*	82.7	71.1	75.3
	QM	76.7	66.7	69.1
	Other	67.6	59.1	61.1
GPT-4	*EM*	86.4	81.0	81.8
	QM	83.6	77.8	78.9
	Other	81.0	70.9	73.0

Table 3: Derivation generation results by field of Physics. *EM* is Electromagnetism, *QM* is Quantum Mechanics, and *Other* contains derivations from Classical and Statistical Mechanics.

herent reasoning than in other subdomains, as the metrics suggest. A manual evaluation of GPT-4 derivations confirms these scores are misleading (Tab. 4).

Physics	# Derivations	Accuracy
Electromagnetism	76	88
Quantum Mechanics	74	69
Other Physics	29	83
All	179	79

Table 4: Manual evaluation of 179 derivations generated by few-shot GPT-4.

To supplement previous analysis, we manually evaluate the coherence of the top scoring model's derivations in each subdomain (Tab. 4), and describe instances where reasoning failures *violate laws of Physics* without necessarily including mathematical errors (Fig. 5). Notably, the ***misuse of minus signs*** frequently contributes to incoherent reasoning, and ***equations can be recalled incorrectly even if they are given in the prompt***.

Use of Mathematics which violates Physics. The coherence of a derivation does not depend on the correct application of Mathematics alone. Fundamental physical assumptions guide which mathematical steps are allowed, and in many cases, such as those in Fig. 5, GPT-4 fails to interpret this. For example, the **Electromagnetism** excerpt is only valid if the electric field $\mathbf{E}$ and charge density ρ are constant throughout space. In this case, all integrals indeed cancel to give Gauss' law. As this law is true when quantities are spatially dependent, this reasoning is flawed. In the **Quantum** excerpt, both equations are true simultaneously only if $\hat{x}$ and $\hat{p}$ commute. Assuming each are respectively the quantum mechanical operators representing position and momentum, GPT-4 violates Heisenberg's uncertainty principle. The **Classical** excerpt is from a derivation that attempts to obtain Snell's law. GPT-4 asserts that light cruises across optical boundaries at 1 m/s without refraction.

5.2 Premise Removal

Following from Eq. 1, we progressively remove premises from prompts and report corresponding scores in Tab. 5.

The non-linear degradation in derivation quality reported by text generation metrics is supported by manual evaluation. Across all metrics, the average performance degradation due to premise removal is non-linear with respect to the number of premises removed, and the score decrease from $(S = 0) \rightarrow (S = 1)$ is on average less than that of $(S = 1) \rightarrow (S = 2)$. However, this alone does not tell us the extent that removing premises *leads to mathematical errors*, as it may be the case that models select alternative derivation paths which inherently lead to lower scores, despite the correct use of mathematics (Meadows et al., 2023a).

To better assess the effect of premise removal, we manually evaluate 300 derivations for mathematical coherence. As with Tab. 4, our evaluation is *reference-free* (Deutsch et al., 2022; Ke et al., 2022; Zhao et al., 2020) with respect to the reasoning itself, although premises and goal equations are necessarily compared. Notably, we allow one or two missing steps given a coherent path can be traced through the derivation, and we are lenient if a derivation's underlying physics is not appropriately considered. The focus of the evaluation is the *correct use of mathematics*, but sometimes the physical context requires specific mathematical behaviour (e.g., Fig. 5).

To give helpful examples, in the supplementary material[2], the key concept in C.89 is to equate kinetic energy with work and rearrange for velocity, but rather than introduce kinetic energy as a premise, GPT *reverse-engineers* the derivation from the goal equation, which we mark as correct. The derivations A.54 and B.54 would be marked as correct if they did not skip the premise

[2] https://github.com/jmeadows17/transformers-for-physics/blob/main/Supplementary_Model_Derivations.pdf

Electromagnetism	**Quantum**	**Classical**
$\frac{\iiint_V \nabla \cdot \mathbf{E} dV}{\iiint_V dV} = \frac{\iiint_V \frac{\rho}{\varepsilon_0} dV}{\iiint_V dV}$	$a^\dagger a = \frac{m\omega}{2\hbar}(\hat{x}^2 - \frac{i}{m\omega}\hat{x}\hat{p} + \frac{i}{m\omega}\hat{p}\hat{x} + \hat{p}^2)$	$c \sin\theta_1 = n_1 \sin\theta_1 = c \sin\theta_2 = n_2 \sin\theta_2$
$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0}$	$a^\dagger a = \frac{m\omega}{2\hbar}(\hat{x}^2 + \frac{i}{m\omega}\hat{x}\hat{p} - \frac{i}{m\omega}\hat{p}\hat{x} + \hat{p}^2)$	$n_1 \sin\theta_1 = n_2 \sin\theta_2$

Figure 5: Excerpts from few-shot GPT-4 derivations that violate well-documented Physics.

	ROUGE			**BLEU**			**GLEU**		
	$S=0$	$S=1$	$S=2$	$S=0$	$S=1$	$S=2$	$S=0$	$S=1$	$S=2$
MathT5-base	69.5	64.1	56.4	58.5	55.1	49.0	59.9	56.7	51.3
MathT5-large	69.1	64.4	55.7	60.6	56.4	46.8	62.1	58.1	49.9
GPT-3.5	77.7	75.8	60.2	68.7	65.6	44.4	70.5	68.4	51.5
GPT-4	**85.6**	**81.0**	**70.5**	**79.7**	**73.8**	**58.9**	**80.6**	**75.7**	**63.5**

Table 5: Results from the Premise Removal perturbation analysis.

	$S=0$	$S=1$	$S=2$
GPT-4	83	73	57

Table 6: GPT accuracy as S premises are removed from the prompt.

$z = \langle f|g\rangle$, but the reasoning as it stands would allow $z \geq \langle f|g\rangle$, which is false.

Overall, the manual evaluation gives a *liberal estimate* of model accuracy on the dataset in Tab. 6. We would expect Physics professionals to give lower scores, particularly for $S = 2$. The non-linear degradation in accuracy due to the gradual removal of premises aligns closely with scores obtained from the automatic metrics (notably BLEU).

Substitution errors. A fundamental component of mathematical reasoning is the substitution of equivalent terms. A significant proportion of errors involve substitution.

Language models derive equations by reverse-engineering. Working backwards from the result, this approach is characterised by a lack of understanding of the problem. Due to the prevalence of this reasoning in the outputs, this is perhaps the core mechanism behind how language models derive given equations. An example is given below:

$\sigma_p^2 = \int_{-\infty}^{\infty} p^2|\phi(p)|^2 dp$ (initial premise)

$\langle g| = \int_{-\infty}^{\infty} p^2|\phi(p)|^2 dp$

$|g\rangle = 1$

$\sigma_p^2 = \langle g|g\rangle$ (goal equation)

GPT-4 has clearly recognised that it must define terms for $\langle g|$ and $|g\rangle$, but a fundamental lack of understanding of Dirac notation (from quantum mechanics) has resulted in defining them as scalar quantities instead of vectors.

6 Conclusion

We explore the mathematical ability of LMs in the context of a derivation generation task using a novel dataset of equation derivations and prompt interventions, with the intent of emulating granular mathematical workings and exploring how models perform detailed calculations involving a variety of Physics notations.

We apply premise removal interventions on prompts to reveal a non-linear relationship between the perturbation strength and average derivation quality, as reported by manual and automatic scores. We find that models (particularly GPT-4) derive goal equations from premises through reverse-engineering intermediate steps without appropriate consideration of basic underlying Physics. This becomes increasingly apparent as we progressively remove premises from few-shot prompts.

Furthermore, many algebraic errors arise from attempts at *substitution*. While it is challenging to build synthetic datasets that reflect all aspects of mathematical reasoning (Toshniwal et al., 2024), it is more manageable to focus on the application of individual operations (such as substitution), and use symbolic engines to apply them in the context of vast vocabularies of symbols and notations representative of target subdomains, which could potentially bolster models' out-of-distribution mathematical abilities. Our use of synthetic in-context examples (involving substitution) improves models' evaluation scores.

Although exploration of the synergy between language models and symbolic engines is underway (Yang et al., 2024; Davis, 2024), Physics derivations rely on physical behaviour which is implicitly assumed (*e.g.,* commutative properties of quantum operators), yet is ignored by LMs. Such assumptions govern the reasoning paths allowed in derivations, and are hence independent of the ability to perform fine-grained algebraic manipulation. This is effectively a premise/operation selection problem, and integrating symbolic solvers does not inherently improve this limitation (Liu and Yao, 2024). We find contemporary language models severely lacking in this regard, and suggest research efforts should be directed towards LM-based querying of appropriate knowledge bases during inference alongside solver integration (Trinh et al., 2024).

Acknowledgements

This work was partially funded by the Swiss National Science Foundation (SNSF) project NeuMath (200021_204617).

7 Limitations

We have not extensively optimised prompts for model performance. However, our few-shot approach mirrors related work (Azerbayev et al., 2023; Meadows et al., 2023a). Our focus on fine-grained mathematical transformations between equations excludes natural language (*e.g.,* Fig. 2) as a design choice. Text generation metrics (*e.g.,* ROUGE) fail to account for mathematical errors in individual derivations, however we contrast this with a human evaluation of 300 derivations in the case of GPT, which supports our conclusions. Due to the small number of human annotators responsible for curating the dataset and manually evaluating model outputs, both elements of this research may vary with the number of annotators or their experience in Physics. Our focus in this regard is a self-consistent set of derivations and assessment criteria agreed upon by the annotators involved.

Overall ethical impact. This work explores a systematic way to elicit the mathematical/symbolic inference properties of Transformer-based models in a mathematical language processing task. As such, it contributes in the direction of a critique of the reasoning capabilities and the biases of these models, particularly in the Physics domain.

References

Toufique Ahmed and Premkumar Devanbu. 2022. Few-shot training llms for project-specific code-summarization. In *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*, pages 1–5.

Prosper D Akrobotu, Tamsin E James, Christian FA Negre, and Susan M Mniszewski. 2022. A qubo formulation for top-τ eigencentrality nodes. *Plos one*, 17(7):e0271292.

Jesse Alama, Tom Heskes, Daniel Kühlwein, Evgeni Tsivtsivadze, and Josef Urban. 2014. Premise selection for mathematics by corpus analysis and kernel methods. *Journal of Automated Reasoning*, 52(2):191–213.

Zhangir Azerbayev, Hailey Schoelkopf, Keiran Paster, Marco Dos Santos, Stephen McAleer, Albert Q Jiang, Jia Deng, Stella Biderman, and Sean Welleck. 2023. Llemma: An open language model for mathematics. *arXiv preprint arXiv:2310.10631*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. 2022. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *arXiv preprint arXiv:2211.12588*.

Xinyun Chen, Ryan A Chi, Xuezhi Wang, and Denny Zhou. 2024. Premise order matters in reasoning with large language models. *arXiv preprint arXiv:2402.08939*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al.

2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Peter Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *arXiv preprint arXiv:2002.05867*.

Ernest Davis. 2019. The use of deep learning for symbolic integration: A review of (lample and charton, 2019). *arXiv preprint arXiv:1912.05752*.

Ernest Davis. 2024. Mathematics, word problems, common sense, and artificial intelligence. *Bulletin of the American Mathematical Society*.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2022. On the limitations of reference-free evaluations of generated text. *arXiv preprint arXiv:2210.12563*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Iddo Drori, Sarah Zhang, Reece Shuttleworth, Leonard Tang, Albert Lu, Elizabeth Ke, Kevin Liu, Linda Chen, Sunny Tran, Newman Cheng, et al. 2022. A neural network solves, explains, and generates university math problems by program synthesis and few-shot learning at human level. *Proceedings of the National Academy of Sciences*, 119(32):e2123433119.

Hamidreza Eivazi, Mojtaba Tahani, Philipp Schlatter, and Ricardo Vinuesa. 2022. Physics-informed neural networks for solving reynolds-averaged navier–stokes equations. *Physics of Fluids*, 34(7).

Deborah Ferreira and André Freitas. 2020. Natural language premise selection: Finding supporting statements for mathematical text. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2175–2182.

Deborah Ferreira, Mokanarangan Thayaparan, Marco Valentino, Julia Rozanova, and Andre Freitas. 2022. To be or not to be an integer? encoding variables for mathematical text. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 938–948.

Emily First, Markus N Rabe, Talia Ringer, and Yuriy Brun. 2023. Baldur: Whole-proof generation and repair with large language models. *arXiv preprint arXiv:2303.04910*.

Simon Frieder, Luca Pinchetti, Ryan-Rhys Griffiths, Tommaso Salvatori, Thomas Lukasiewicz, Philipp Christian Petersen, Alexis Chevalier, and Julius Berner. 2023. Mathematical capabilities of chatgpt. *arXiv preprint arXiv:2301.13867*.

Yingqiang Ge, Wenyue Hua, Jianchao Ji, Juntao Tan, Shuyuan Xu, and Yongfeng Zhang. 2023. Openagi: When llm meets domain experts. *arXiv preprint arXiv:2304.04370*.

Joy He-Yueya, Gabriel Poesia, Rose E. Wang, and Noah D. Goodman. 2023. Solving math word problems by combining language models with symbolic solvers.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

JJ Hopfield. 1958. Theory of the contribution of excitons to the complex dielectric constant of crystals. *Physical Review*, 112(5):1555.

Yangyang Hu and Yang Yu. 2022. Enhancing neural mathematical reasoning by abductive combination with symbolic library. *arXiv preprint arXiv:2203.14487*.

Zhiqiang Hu, Yihuai Lan, Lei Wang, Wanyu Xu, Ee-Peng Lim, Roy Ka-Wei Lee, Lidong Bing, and Soujanya Poria. 2023. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*.

Albert Qiaochu Jiang, Wenda Li, Jesse Michael Han, and Yuhuai Wu Lisa. 2021. Language models of isabelle proofs. In *6th Conference on Artificial Intelligence and Theorem Proving*.

Albert Qiaochu Jiang, Wenda Li, Szymon Tworkowski, Konrad Czechowski, Tomasz Odrzygóźdź, Piotr Miłoś, Yuhuai Wu, and Mateja Jamnik. 2022. Thor: Wielding hammers to integrate language models and automated theorem provers. *Advances in Neural Information Processing Systems*, 35:8360–8373.

Cezary Kaliszyk, Josef Urban, Umair Siddique, Sanaz Khan-Afshar, Cvetan Dunchev, and Sofiene Tahar. 2015. Formalizing physics: automation, presentation and foundation issues. In *International Conference on Intelligent Computer Mathematics*, pages 288–295. Springer.

Pei Ke, Hao Zhou, Yankai Lin, Peng Li, Jie Zhou, Xiaoyan Zhu, and Minlie Huang. 2022. Ctrleval: An unsupervised reference-free metric for evaluating controlled text generation.

Guillaume Lample and François Charton. 2019. Deep learning for symbolic mathematics. *arXiv preprint arXiv:1912.01412*.

Guillaume Lample, Timothee Lacroix, Marie-Anne Lachaux, Aurelien Rodriguez, Amaury Hayat, Thibaut Lavril, Gabriel Ebner, and Xavier Martinet. 2022. Hypertree proof search for neural theorem proving. *Advances in Neural Information Processing Systems*, 35:26337–26349.

Aitor Lewkowycz, Anders Andreassen, David Dohan, Ethan Dyer, Henryk Michalewski, Vinay Ramasesh, Ambrose Slone, Cem Anil, Imanol Schlag, Theo Gutman-Solo, et al. 2022. Solving quantitative reasoning problems with language models. *arXiv preprint arXiv:2206.14858*.

Zhenwen Liang, Jipeng Zhang, Lei Wang, Wei Qin, Yunshi Lan, Jie Shao, and Xiangliang Zhang. 2022. Mwp-bert: Numeracy-augmented pre-training for math word problem solving. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 997–1009.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. 2023. Evaluating the logical reasoning ability of chatgpt and gpt-4. *arXiv preprint arXiv:2304.03439*.

Haoxiong Liu and Andrew Chi-Chih Yao. 2024. Augmenting math word problems via iterative question composing. *arXiv preprint arXiv:2401.09003*.

MinZhong Luo and Li Liu. 2018. Automatic derivation of formulas using reforcement learning. *arXiv preprint arXiv:1808.04946*.

Pratik Mandlecha, Snehith Kumar Chatakonda, Neeraj Kollepara, and Pawan Kumar. 2022. Hybrid tokenization and datasets for solving mathematics and science problems using transformers. In *Proceedings of the 2022 SIAM International Conference on Data Mining (SDM)*, pages 289–297. SIAM.

Charlie-Ray Mann, Thomas J Sturges, Guillaume Weick, William L Barnes, and Eros Mariani. 2018. Manipulating type-i and type-ii dirac polaritons in cavity-embedded honeycomb metasurfaces. *Nature communications*, 9(1):1–11.

Jordan Meadows and André Freitas. 2021. Similarity-based equational inference in physics. *Physical Review Research*, 3(4):L042010.

Jordan Meadows and André Freitas. 2023. Introduction to mathematical language processing: Informal proofs, word problems, and supporting tasks. *Transactions of the Association for Computational Linguistics*, 11:1162–1184.

Jordan Meadows, Marco Valentino, and Andre Freitas. 2023a. Generating mathematical derivations with large language models.

Jordan Meadows, Marco Valentino, Damien Teney, and Andre Freitas. 2023b. A symbolic framework for systematic evaluation of mathematical reasoning with transformers.

Jordan Meadows, Zili Zhou, and Andre Freitas. 2022. Physnlu: A language resource for evaluating natural language understanding and explanation coherence in physics. *arXiv preprint arXiv:2201.04275*.

Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, et al. 2022. Lila: A unified benchmark for mathematical reasoning. *arXiv preprint arXiv:2210.17517*.

Andrew Mutton, Mark Dras, Stephen Wan, and Robert Dale. 2007. Gleu: Automatic evaluation of sentence-level fluency. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 344–351.

Haining Pan, Nayantara Mudur, Will Taranto, Maria Tikhanovskaya, Subhashini Venugopalan, Yasaman Bahri, Michael P Brenner, and Eun-Ah Kim. 2024. Quantum many-body physics calculations with large language models. *arXiv preprint arXiv:2403.03154*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Judea Pearl. 2009. Causal inference in statistics: An overview. *Statistics surveys*, 3:96–146.

Stanislas Polu, Jesse Michael Han, Kunhao Zheng, Mantas Baksys, Igor Babuschkin, and Ilya Sutskever. 2022. Formal mathematics statement curriculum learning. *arXiv preprint arXiv:2202.01344*.

Stanislas Polu and Ilya Sutskever. 2020. Generative language modeling for automated theorem proving. *arXiv preprint arXiv:2009.03393*.

Xin Quan, Marco Valentino, Louise A. Dennis, and André Freitas. 2024. Verification and refinement of natural language explanations through llm-symbolic theorem proving.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

Subhro Roy, Tim Vieira, and Dan Roth. 2015. Reasoning about quantities in natural language. *Transactions of the Association for Computational Linguistics*, 3:1–13.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. 2019. Analysing mathematical reasoning abilities of neural models. *arXiv preprint arXiv:1904.01557*.

Paulo Shakarian, Abhinav Koyyalamudi, Noel Ngu, and Lakshmivihari Mareedu. 2023. An independent evaluation of chatgpt on mathematical word problems (mwp). *arXiv preprint arXiv:2302.13814*.

Chan Hee Song, Jiaman Wu, Clayton Washington, Brian M Sadler, Wei-Lun Chao, and Yu Su. 2022. Llm-planner: Few-shot grounded planning for embodied agents with large language models. *arXiv preprint arXiv:2212.04088*.

Alessandro Stolfo, Zhijing Jin, Kumar Shridhar, Bernhard Schölkopf, and Mrinmaya Sachan. 2022. A causal framework to quantify the robustness of mathematical reasoning with language models. *arXiv preprint arXiv:2210.12023*.

Christian Szegedy. 2020. A promising path towards autoformalization and general artificial intelligence. In *International Conference on Intelligent Computer Mathematics*, pages 3–20. Springer.

Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. 2024. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *arXiv preprint arXiv:2402.10176*.

Trieu H Trinh, Yuhuai Wu, Quoc V Le, He He, and Thang Luong. 2024. Solving olympiad geometry without human demonstrations. *Nature*, 625(7995):476–482.

Marco Valentino, Deborah Ferreira, Mokanarangan Thayaparan, André Freitas, and Dmitry Ustalov. 2022a. Textgraphs 2022 shared task on natural language premise selection. In *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, pages 105–113.

Marco Valentino, Deborah Ferreira, Mokanarangan Thayaparan, André Freitas, and Dmitry Ustalov. 2022b. TextGraphs 2022 shared task on natural language premise selection. In *Proceedings of TextGraphs-16: Graph-based Methods for Natural Language Processing*, pages 105–113, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Marco Valentino, Jordan Meadows, Lan Zhang, and André Freitas. 2023. Multi-operational mathematical derivations in latent space. *arXiv preprint arXiv:2311.01230*.

Pablo Villalobos, Jaime Sevilla, Lennart Heim, Tamay Besiroglu, Marius Hobbhahn, and Anson Ho. 2022. Will we run out of data? an analysis of the limits of scaling datasets in machine learning.

Mingzhe Wang, Yihe Tang, Jian Wang, and Jia Deng. 2017. Premise selection for theorem proving by deep graph embedding. *arXiv preprint arXiv:1709.09994*.

Tianduo Wang and Wei Lu. 2023. Learning multi-step reasoning by solving arithmetic tasks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1229–1238.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

Sean Welleck, Jiacheng Liu, Ronan Le Bras, Hannaneh Hajishirzi, Yejin Choi, and Kyunghyun Cho. 2021. Naturalproofs: Mathematical theorem proving in natural language. *arXiv preprint arXiv:2104.01112*.

Sean Welleck, Jiacheng Liu, Ximing Lu, Hannaneh Hajishirzi, and Yejin Choi. 2022a. Naturalprover: Grounded mathematical proof generation with language models. *arXiv preprint arXiv:2205.12910*.

Sean Welleck, Peter West, Jize Cao, and Yejin Choi. 2022b. Symbolic brittleness in sequence models: on systematic generalization in symbolic mathematics. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8629–8637.

Tailin Wu and Max Tegmark. 2019. Toward an artificial intelligence physicist for unsupervised learning. *Physical Review E*, 100(3):033311.

Yuhuai Wu, Albert Q. Jiang, Wenda Li, Markus N. Rabe, Charles Staats, Mateja Jamnik, and Christian Szegedy. 2022. Autoformalization with large language models.

Jingfeng Yang, Hongye Jin, Ruixiang Tang, Xiaotian Han, Qizhang Feng, Haoming Jiang, Bing Yin, and Xia Hu. 2023. Harnessing the power of llms in practice: A survey on chatgpt and beyond. *arXiv preprint arXiv:2304.13712*.

Kaiyu Yang, Aidan Swope, Alex Gu, Rahul Chalamala, Peiyang Song, Shixing Yu, Saad Godil, Ryan J Prenger, and Animashree Anandkumar. 2024. Leandojo: Theorem proving with retrieval-augmented language models. *Advances in Neural Information Processing Systems*, 36.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, and Songfang Huang. 2023. How well do large language models perform in arithmetic tasks? *arXiv preprint arXiv:2304.02015*.

Wei Zhao, Goran Glavaš, Maxime Peyrard, Yang Gao, Robert West, and Steffen Eger. 2020. On the limitations of cross-lingual encoders as exposed by reference-free machine translation evaluation. *arXiv preprint arXiv:2005.01196*.

A Further details on models, metrics, and prompting

Models. T5 (Raffel et al., 2020) is an encoder-decoder transformer where all pre-training objectives are formulated as text generation (and therefore do not require different loss functions). FLAN-T5 (Chung et al., 2022) is T5 fine-tuned on instructions, and outperforms T5 in a variety of tasks. The GPT models (Brown et al., 2020) are decoder-only transformer-based models trained on large-scale natural (and mathematical) language corpora. We evaluate 8 models on derivation generation: the base and large variants of T5 and FLAN-T5, GPT-3.5, GPT-4, MathT5-base, and MathT5-large (Meadows et al., 2023a). MathT5-large is a version of FLAN-T5-large fine-tuned for 25 epochs on 15K (LaTeX) synthetic mathematical derivations (containing 4 - 10 equations), that were generated using a symbolic engine. It outperforms the few-shot performance of GPT-4 and GPT-3.5 on derivation generation in ROUGE, BLEU, BLEURT, and GLEU scores, and shows some generalisation capabilities. It was trained on 155 Physics-related symbols, but struggles with out-of-vocabulary symbols. MathT5-base is the equivalent, but uses T5-base as the initialised model before fine-tuning. Instantiated few-shot prompts are fed to the instruction-based models (in this case GPT) models through the OpenAI API[3], with temperature set to 0 to minimise non-deterministic effects.

Computation and hyperparameters. Hyperparameters used in T5 models are the default hyperparameters of MathT5 defined in the MathT5.py script on the Hugging Face website. There was no training involved in the experiments of this paper, and models were evaluated on a single GTX 1070 for up to a week in total.

Zero-shot prompts. We prefix the following sentence to the prompt template discussed in the main paper:

"Derive the final equation using the premise equations from the following prompt (denoted by "Prompt:"). Give only the equations involved in the derivation. Do not include any text other than equations each separated by "and". Prompt: ".

Few-shot prompts. For each initial prompt, such as the example template in the main paper, a set of 5 example templates (and their derivations) are randomly selected from the set of *synthetic derivations*. We select in-context examples by filtering only those containing *more than one premise*, and with no given intermediate equations, to better match the Physics prompts. The examples are then fit into the few-shot prompt below:

The following examples consist of a prompt (denoted by Prompt:) and a mathematical derivation (denoted by Derivation:). Each derivation contains LaTeX equations separated by "and".

The training prompts are appended after this description, then continues:

Now given the following prompt, generate the derivation. Ensure equations are split by the word "and".

The evaluation prompt is inserted here, prefixed by *"Prompt:"*.

Metrics. ROUGE, BLEU, and GLEU are all metrics used to evaluate the quality of text generated by machine translation or other natural language processing tasks, but they differ in their approaches and specific applications.

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) (Lin, 2004) focuses on the overlap of n-grams, word sequences, and word pairings between the generated text and reference texts, emphasising *recall*. It is widely used in summarisation evaluation and other tasks where capturing the essence of the reference material is critical. BLEU (Bilingual Evaluation Understudy) (Papineni et al., 2002) measures the *precision* of n-gram matches between the output and reference texts, adjusted by a brevity penalty to discourage overly short translations. BLEU is predominantly used in machine translation to assess the closeness of the translation to human-produced texts.

GLEU (Generalized Language Understanding Evaluation) (Mutton et al., 2007) is similar to BLEU in its use of n-gram overlap but was specifically designed for evaluating grammatical error corrections. GLEU includes modifications to accommodate the nuances of grammar correction by considering both the presence of corrected n-grams and penalising uncorrected errors, without the need for tuning across different numbers of reference texts. While ROUGE emphasises capturing the essence of the text through recall, BLEU focuses on precision, and GLEU targets the specific domain of grammatical correctness. The latter is used in related work (Welleck et al., 2022a).

B Electromagnetism

B.1 Gauss' law: Derivation from Coulomb's law

$$\mathbf{E}(\mathbf{r}) = \frac{1}{4\pi\varepsilon_0}\int \frac{\rho(\mathbf{s})(\mathbf{r}-\mathbf{s})}{|\mathbf{r}-\mathbf{s}|^3} d^3\mathbf{s} \tag{2}$$

$$\nabla\cdot\mathbf{E}(\mathbf{r}) = \nabla\cdot\frac{1}{4\pi\varepsilon_0}\int \frac{\rho(\mathbf{s})(\mathbf{r}-\mathbf{s})}{|\mathbf{r}-\mathbf{s}|^3} d^3\mathbf{s} \tag{3}$$

$$\nabla\cdot\mathbf{E}(\mathbf{r}) = \frac{1}{4\pi\varepsilon_0}\int \rho(\mathbf{s})\nabla\cdot\frac{(\mathbf{r}-\mathbf{s})}{|\mathbf{r}-\mathbf{s}|^3} d^3\mathbf{s} \tag{4}$$

$$\nabla\cdot\frac{(\mathbf{r}-\mathbf{s})}{|\mathbf{r}-\mathbf{s}|^3} = 4\pi\delta(\mathbf{r}-\mathbf{s}) \tag{5}$$

[3] https://platform.openai.com/overview

$$\nabla \cdot \mathbf{E}(\mathbf{r}) = \frac{1}{4\pi\varepsilon_0} \int \rho(\mathbf{s}) 4\pi\delta(\mathbf{r}-\mathbf{s}) d^3\mathbf{s} \quad (6)$$

B.2 Uniqueness theorem for Poisson's equation 2

$$\nabla \cdot (\phi\nabla\phi) = (\nabla\phi)^2 + \phi\nabla^2\phi \quad (7)$$

$$\nabla^2\phi = 0 \quad (8)$$

$$\nabla \cdot (\phi\nabla\phi) = (\nabla\phi)^2 \quad (9)$$

$$\int_V \nabla \cdot (\phi\nabla\phi) dV = \int_V (\nabla\phi)^2 dV \quad (10)$$

$$\int_V \nabla \cdot (\phi\nabla\phi) dV = \int_S \phi\nabla\phi \cdot d\mathbf{S} \quad (11)$$

$$\int_S \phi\nabla\phi \cdot d\mathbf{S} = \int_V (\nabla\phi)^2 dV \quad (12)$$

B.3 Lorentz force: Derivation of Lorentz force from classical Lagrangian (LHS)

$$L = \frac{m}{2}(\dot{x}^2+\dot{y}^2+\dot{z}^2)+q(\dot{x}A_x+\dot{y}A_y+\dot{z}A_z)-q\phi \quad (13)$$

$$\frac{\partial L}{\partial \dot{x}} = \frac{\partial}{\partial \dot{x}}(\frac{m}{2}(\dot{x}^2+\dot{y}^2+\dot{z}^2)+q(\dot{x}A_x+\dot{y}A_y+\dot{z}A_z)-q\phi) \quad (14)$$

$$\frac{\partial L}{\partial \dot{x}} = \frac{m}{2}\frac{\partial}{\partial \dot{x}}(\dot{x}^2+\dot{y}^2+\dot{z}^2) + q\frac{\partial}{\partial \dot{x}}(\dot{x}A_x+\dot{y}A_y+\dot{z}A_z) - q\frac{\partial}{\partial \dot{x}}\phi \quad (15)$$

$$\frac{\partial}{\partial \dot{x}}\phi = 0 \quad (16)$$

$$\frac{\partial L}{\partial \dot{x}} = \frac{m}{2}\frac{\partial}{\partial \dot{x}}(\dot{x}^2+\dot{y}^2+\dot{z}^2) + q\frac{\partial}{\partial \dot{x}}(\dot{x}A_x+\dot{y}A_y+\dot{z}A_z) \quad (17)$$

B.4 Ampere's circuital law: Proof of equivalence 2

$$\nabla \times \mathbf{H} = \mathbf{J}_f + \frac{\partial \mathbf{D}}{\partial t} \quad (18)$$

$$\nabla \times \frac{1}{\mu_0}\mathbf{B} = \nabla \times \mathbf{H} + \mathbf{J}_M \quad (19)$$

$$\nabla \times \frac{1}{\mu_0}\mathbf{B} = \mathbf{J}_f + \frac{\partial \mathbf{D}}{\partial t} + \mathbf{J}_M \quad (20)$$

$$\mathbf{D} = \varepsilon_0\mathbf{E} + \mathbf{P} \quad (21)$$

$$\frac{\partial \mathbf{D}}{\partial t} = \frac{\partial}{\partial t}(\varepsilon_0\mathbf{E} + \mathbf{P}) \quad (22)$$

$$\frac{\partial \mathbf{D}}{\partial t} = \frac{\partial}{\partial t}\varepsilon_0\mathbf{E} + \frac{\partial}{\partial t}\mathbf{P} \quad (23)$$

$$\nabla \times \frac{1}{\mu_0}\mathbf{B} = \mathbf{J}_f + \frac{\partial}{\partial t}\varepsilon_0\mathbf{E} + \frac{\partial}{\partial t}\mathbf{P} + \mathbf{J}_M \quad (24)$$

C Quantum Mechanics

C.1 Uncertainty principle: Kennard inequality proof part 2.7

$$g(x) = \frac{1}{2\pi\hbar}\int_{-\infty}^{\infty} p\cdot(\frac{\hbar}{ip}\int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} e^{\frac{-ip\chi}{\hbar}} d\chi)\cdot e^{\frac{ipx}{\hbar}} dp \quad (25)$$

$$g(x) = \frac{1}{2\pi\hbar}\int_{-\infty}^{\infty} \frac{\hbar}{i}\int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} e^{\frac{-ip\chi}{\hbar}} d\chi \cdot e^{\frac{ipx}{\hbar}} dp \quad (26)$$

$$g(x) = \frac{1}{2\pi i}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} e^{\frac{-ip\chi}{\hbar}} d\chi \cdot e^{\frac{ipx}{\hbar}} dp \quad (27)$$

$$e^{\frac{-ip\chi}{\hbar}} \cdot e^{\frac{ipx}{\hbar}} = e^{\frac{i}{\hbar}(x-\chi)p} \quad (28)$$

$$g(x) = \frac{1}{2\pi i}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{d\psi(\chi)}{d\chi} e^{\frac{i}{\hbar}(x-\chi)p} d\chi dp \quad (29)$$

C.2 Uncertainty principle: Kennard inequality proof part 3.3

$$\sigma_p^2 = \int_{-\infty}^{\infty} p^2 |\varphi(p)|^2 dp \tag{30}$$

$$|\tilde{g}(p)|^2 = p^2 |\varphi(p)|^2 \tag{31}$$

$$\sigma_p^2 = \int_{-\infty}^{\infty} |\tilde{g}(p)|^2 dp \tag{32}$$

$$\int_{-\infty}^{\infty} |\tilde{g}(p)|^2 dp = \int_{-\infty}^{\infty} |g(x)|^2 dx \tag{33}$$

$$\sigma_p^2 = \int_{-\infty}^{\infty} |g(x)|^2 dx \tag{34}$$

$$\langle g|g \rangle = \int_{-\infty}^{\infty} |g(x)|^2 dx \tag{35}$$

$$\sigma_p^2 = \langle g|g \rangle \tag{36}$$

C.3 Expectation value: integral expression 3

$$\left\langle \hat{X} \right\rangle_\Psi = \int \int \langle x|\Psi \rangle^\dagger x' \delta(x - x') \left\langle x' \middle| \Psi \right\rangle dx dx' \tag{37}$$

$$\int \langle x|\Psi \rangle^\dagger x' \delta(x - x') \left\langle x' \middle| \Psi \right\rangle dx' = \langle x|\Psi \rangle^\dagger x \langle x|\Psi \rangle \tag{38}$$

$$\left\langle \hat{X} \right\rangle_\Psi = \int \langle x|\Psi \rangle^\dagger x \langle x|\Psi \rangle dx \tag{39}$$

$$\langle x|\Psi \rangle = \Psi(x) \tag{40}$$

$$\left\langle \hat{X} \right\rangle_\Psi = \int \Psi^\dagger(x) x \Psi(x) dx \tag{41}$$

$$\Psi^\dagger(x)\Psi(x) = |\Psi(x)|^2 \tag{42}$$

$$\left\langle \hat{X} \right\rangle_\Psi = \int x |\Psi(x)|^2 dx \tag{43}$$

C.4 Hellmann–Feynman theorem 2

$$\frac{dE_\lambda}{d\lambda} = \frac{d}{d\lambda} \langle \Psi_\lambda | \hat{H}_\lambda | \Psi_\lambda \rangle \tag{44}$$

$$\frac{d}{d\lambda} \langle \Psi_\lambda | \hat{H}_\lambda | \Psi_\lambda \rangle = \left\langle \frac{d\Psi_\lambda}{d\lambda} \right| \hat{H}_\lambda | \Psi_\lambda \rangle + \langle \Psi_\lambda | \frac{d\hat{H}_\lambda}{d\lambda} | \Psi_\lambda \rangle + \langle \Psi_\lambda | \hat{H}_\lambda \left| \frac{d\Psi_\lambda}{d\lambda} \right\rangle \tag{45}$$

$$\frac{dE_\lambda}{d\lambda} = \left\langle \frac{d\Psi_\lambda}{d\lambda} \right| \hat{H}_\lambda | \Psi_\lambda \rangle + \langle \Psi_\lambda | \frac{d\hat{H}_\lambda}{d\lambda} | \Psi_\lambda \rangle + \langle \Psi_\lambda | \hat{H}_\lambda \left| \frac{d\Psi_\lambda}{d\lambda} \right\rangle \tag{46}$$

$$\langle \Psi_\lambda | \hat{H}_\lambda = \langle \Psi_\lambda | E_\lambda \tag{47}$$

$$\hat{H}_\lambda | \Psi_\lambda \rangle = E_\lambda | \Psi_\lambda \rangle \tag{48}$$

$$\frac{dE_\lambda}{d\lambda} = E_\lambda \left\langle \frac{d\Psi_\lambda}{d\lambda} \middle| \Psi_\lambda \right\rangle + \langle \Psi_\lambda | \frac{d\hat{H}_\lambda}{d\lambda} | \Psi_\lambda \rangle + E_\lambda \left\langle \Psi_\lambda \middle| \frac{d\Psi_\lambda}{d\lambda} \right\rangle \tag{49}$$

D Other

D.1 Euler-Lagrange equation: Derivation 2

$$\frac{dJ_\varepsilon}{d\varepsilon} = \int_a^b \frac{dL_\varepsilon}{d\varepsilon} dx \tag{50}$$

$$\frac{dL_\varepsilon}{d\varepsilon} = \frac{\partial L_\varepsilon}{\partial g_\varepsilon} \eta(x) + \frac{\partial L_\varepsilon}{\partial g'_\varepsilon} \eta'(x) \tag{51}$$

$$\frac{dJ_\varepsilon}{d\varepsilon} = \int_a^b (\frac{\partial L_\varepsilon}{\partial g_\varepsilon} \eta(x) + \frac{\partial L_\varepsilon}{\partial g'_\varepsilon} \eta'(x)) dx \tag{52}$$

$$\frac{dJ_\varepsilon}{d\varepsilon} = \int_a^b \frac{\partial L_\varepsilon}{\partial g_\varepsilon} \eta(x) dx + \int_a^b \frac{\partial L_\varepsilon}{\partial g'_\varepsilon} \eta'(x) dx \tag{53}$$

$$\left. \frac{dJ_\varepsilon}{d\varepsilon} \right|_{\varepsilon=0} = \left. (\int_a^b \frac{\partial L_\varepsilon}{\partial g_\varepsilon} \eta(x) dx + \int_a^b \frac{\partial L_\varepsilon}{\partial g'_\varepsilon} \eta'(x) dx) \right|_{\varepsilon=0} \tag{54}$$

$$\left. \frac{dJ_\varepsilon}{d\varepsilon} \right|_{\varepsilon=0} = \left. (\int_a^b \frac{\partial L_\varepsilon}{\partial g_\varepsilon} \eta(x) dx) \right|_{\varepsilon=0} + \left. (\int_a^b \frac{\partial L_\varepsilon}{\partial g'_\varepsilon} \eta'(x) dx) \right|_{\varepsilon=0} \tag{55}$$

D.2 Snell's law: from Fermat's principle

$$T = \frac{(x^2+a^2)^{\frac{1}{2}}}{v_1} + \frac{(b^2+(l-x)^2)^{\frac{1}{2}}}{v_2} \tag{56}$$

$$T = \frac{(x^2+a^2)^{\frac{1}{2}}}{v_1} + \frac{(b^2+l^2-2lx+x^2)^{\frac{1}{2}}}{v_2} \tag{57}$$

$$\frac{dT}{dx} = \frac{d}{dx}(\frac{(x^2+a^2)^{\frac{1}{2}}}{v_1} + \frac{(b^2+l^2-2lx+x^2)^{\frac{1}{2}}}{v_2}) \tag{58}$$

$$\frac{dT}{dx} = \frac{d}{dx}(\frac{(x^2+a^2)^{\frac{1}{2}}}{v_1}) + \frac{d}{dx}(\frac{(b^2+l^2-2lx+x^2)^{\frac{1}{2}}}{v_2}) \tag{59}$$

$$\frac{d}{dx}(\frac{(x^2+a^2)^{\frac{1}{2}}}{v_1}) = \frac{x}{v_1(x^2+a^2)^{\frac{1}{2}}} \tag{60}$$

$$\frac{d}{dx}(\frac{(b^2+l^2-2lx+x^2)^{\frac{1}{2}}}{v_2}) = \frac{x-l}{v_2((x-l)^2+b^2)^{\frac{1}{2}}} \tag{61}$$

$$\frac{dT}{dx} = \frac{x}{v_1(x^2+a^2)^{\frac{1}{2}}} + \frac{x-l}{v_2((x-l)^2+b^2)^{\frac{1}{2}}} \tag{62}$$

D.3 Maxwell-Boltzmann: energy distribution 2

$$|\mathbf{p}|^2 d|\mathbf{p}| = m(2mE)^{\frac{1}{2}} dE \tag{63}$$

$$d^3\mathbf{p} = 4\pi|\mathbf{p}|^2 d|\mathbf{p}| \tag{64}$$

$$d^3\mathbf{p} = 4\pi m(2mE)^{\frac{1}{2}} dE \tag{65}$$

$$f_{\mathbf{p}}(\mathbf{p}) = (2\pi mkT)^{-\frac{3}{2}} e^{-\frac{\mathbf{p}^2}{2mkT}} \tag{66}$$

$$f_{\mathbf{p}}(\mathbf{p}) d^3\mathbf{p} = (2\pi mkT)^{-\frac{3}{2}} e^{-\frac{\mathbf{p}^2}{2mkT}} d^3\mathbf{p} \tag{67}$$

$$f_{\mathbf{p}}(\mathbf{p}) d^3\mathbf{p} = (2\pi mkT)^{-\frac{3}{2}} e^{-\frac{\mathbf{p}^2}{2mkT}} 4\pi m(2mE)^{\frac{1}{2}} dE \tag{68}$$

$$f_E(E) dE = f_{\mathbf{p}}(\mathbf{p}) d^3\mathbf{p} \tag{69}$$

$$f_E(E) dE = (2\pi mkT)^{-\frac{3}{2}} e^{-\frac{\mathbf{p}^2}{2mkT}} 4\pi m(2mE)^{\frac{1}{2}} dE \tag{70}$$

$$f_E(E) = 2(\frac{E}{\pi})^{\frac{1}{2}} (\frac{1}{kT})^{\frac{3}{2}} e^{-(\frac{E}{kT})} \tag{71}$$

D.4 Wave equation: stress pulse in a bar

$$\frac{\partial^2}{\partial t^2} u(x+h,t) = \frac{KL^2}{Mh^2}\big(u(x+2h,t) - 2u(x+h,t) + u(x,t)\big) \tag{72}$$

$$\lim_{h\to 0} \frac{\partial^2 u(x+h,t)}{\partial t^2} = \lim_{h\to 0} \frac{KL^2}{Mh^2}\big(u(x+2h,t) - 2u(x+h,t) + u(x,t)\big) \tag{73}$$

$$\lim_{h\to 0} \frac{u(x+2h,t) - 2u(x+h,t) + u(x,t)}{h^2} = \frac{\partial^2 u(x,t)}{\partial x^2} \tag{74}$$

$$\lim_{h\to 0} \frac{\partial^2 u(x+h,t)}{\partial t^2} = \frac{KL^2}{M} \frac{\partial^2 u(x,t)}{\partial x^2} \tag{75}$$

$$\lim_{h\to 0} \frac{\partial^2 u(x+h,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial t^2} \tag{76}$$

$$\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{KL^2}{M} \frac{\partial^2 u(x,t)}{\partial x^2} \tag{77}$$