This is an Accepted Manuscript for *Calitatea Vieții*. Subject to change during the editing and production process.

# LLMs IN HIGHER EDUCATION: A COMPARATIVE EVALUATION OF MATHEMATICAL PROBLEM-SOLVING CAPABILITIES AND A BLOOM-ALIGNED HUMAN-AI COLLABORATION FRAMEWORK

**NATASA KOCESKA**
**SASO KOCESKI**
**MARIA POTES BARBAS**
**ANDREIA TALES VIEIRA**

*This paper presents a comparative evaluation of three freely accessible Large Language Models (LLMs) such as ChatGPT-4o, Gemini 2.5 Flash, and Claude 4.5 Sonnet, with respect to their capability to solve ordinary difference equations spanning linear homogeneous, non-homogeneous, and system-type formulations. A structured dataset of twenty problems was developed, and solutions were independently assessed by three domain experts using a standardised rubric; inter-rater reliability was confirmed through intraclass correlation coefficients exceeding 0.9. Quantitative evaluation employed precision, recall, F1-score, and single-factor ANOVA, which confirmed statistically significant inter-model differences at $p < 0.05$. All models achieved median performance scores above 7 out of 10, with ChatGPT recording the highest precision and F1-score. A systematic qualitative analysis, framed within Bloom's Taxonomy, reveals that LLMs demonstrate reliable autonomous competence at the foundational cognitive levels, Remember, Understand, and partially Apply and Analyse, while the Evaluate and Create levels remain exclusively within the domain of human cognitive contribution. This boundary, empirically identified in the present study, provides the theoretical basis for the Human-AI Collaboration framework for higher-education STEM instruction grounded in Bloom's Inverted Taxonomy, proposed herein: instruction should begin at the generative and evaluative apex of the cognitive hierarchy, where human contribution is indispensable, with LLMs serving as scaffolding infrastructure for the lower cognitive levels. The study further identifies a recurring failure mode common to all models − correct procedural narration coupled with*

**Address of the corresponding authors: Natasa KOCESKA**, Full professor at Faculty of Computer Science, GoceDelcev University, Stip, North Macedonia, e-mail: natasa.koceska@ugd.edu.mk (ORCID: https://orcid.org/0000-0002-3392-8871); **Saso KOCESKI**, Full professor at Faculty of Computer Science, GoceDelcev University, Stip, North Macedonia, e-mail: saso.koceski@ugd.edu.mk (ORCID: https://orcid.org/0000-0002-5513-1898); **Maria POTES BARBAS**, Full Professor, Instituto Politécnico de Santarém Escola Superior de Educação de Santarém, Portugal, e-mail: maria.barbas@sc.ipsantarem.pt (ORCID: https://orcid.org/0000-0001-8598-1650); **Andreia TELES VIEIRA**, Instituto Politécnico de Santarém Escola Superior de Educação de Santarém, Portugal, e-mail: andreia.vieira@my.istec.pt (ORCID: https://orcid.org/0000-0001-5528-9380).

CALITATEA VIEȚII, XXXVII, nr. 3, 2026, pp. , https://doi.org/10.46841/RCV.2026.03.04

*arithmetic execution errors and discusses its value as a deliberate instructional resource. The results have direct relevance for educators, curriculum designers, and researchers seeking to integrate LLMs responsibly into mathematical learning environments.*

***Keywords:*** *Large Language Models; Difference Equations; Bloom's Taxonomy; Human-AI Collaboration; Mathematical Problem-Solving; STEM Education.*

## INTRODUCTION

Difference equations, also commonly referred to as recurrence relations, constitute a foundational concept in discrete mathematics and are formally defined as equations that express the value of a sequence term as a function of one or more preceding terms (Kelley and Peterson 2001). Their generic form can be written as:

$$x(n+k)=f(n,x(n),x(n+1),\ldots,x(n+k-1)) \quad (1)$$

where x is the variable of interest, n is the discrete independent variable, k denotes the order of the equation, and f describes the functional relationship between terms. Despite their structural simplicity, difference equations underpin the mathematical modelling of discrete-time phenomena across a wide range of scientific and industrial domains, including economics and financial forecasting (Santiago *et al.* 2020; Kunsch *et al.* 2000; Leippold and Schärer 2017), population biology (Cushing 2019; Zhao, Kong and Wang 2020), control systems engineering (Belaghi and Sari 2021), algorithm analysis (Khelifa and Halim 2021), digital signal processing (Parveen Banu and Jothilakshmi 2021; Kavitha *et al.* 2024), pharmacokinetics (D'Cunha and Gorty 2022), social dynamics (Mickens 2015), and environmental science (Lu, Lin and Chorin 2016; DeAngelis and Yurek 2015).

From a pedagogical standpoint, difference equations present well-documented challenges for students in higher-education STEM programs. Proficiency in this domain requires the simultaneous command of discrete mathematics, linear algebra, and in more advanced cases elements of complex analysis. Students commonly encounter difficulty in recognising the structural type of an equation, setting up the appropriate solution procedure, and executing multi-step derivations with sufficient precision. These difficulties are compounded by the fact that errors introduced at intermediate steps propagate through the solution, making the identification and correction of mistakes particularly demanding. Intelligent tutoring systems capable of providing step-by-step guidance through such derivations could therefore constitute a valuable instructional resource, provided their reliability and limitations are empirically characterized.

The rapid development of generative AI and, in particular, of Large Language Models (LLMs) has introduced a new class of tools with potential applications across educational and research contexts. LLMs are deep learning models based on transformer architecture (Vaswani *et al.* 2017), trained on large-

scale text corpora through self-supervised language modelling objectives. Their ability to generate contextually coherent, human-like text across a wide range of domains has prompted growing interest in their capacity for mathematical reasoning. Prior research has demonstrated that LLMs can perform reliably on arithmetic and basic algebraic tasks (Xia and Yu 2023), and that transformer-based models can be trained to solve integration problems and ordinary differential equations (Koceski *et al.* 2025; Zlatanovska *et al.* 2026; Becker *et al.* 2023). However, studies examining more complex mathematical domains have consistently identified limitations in multi-step reasoning (Wang and Lü 2023) and formal proof generation (Polu and Sutskever 2020), even as chain-of-thought prompting has been shown to substantially improve performance on structured problem-solving tasks (Wei *et al.* 2022). Most relevantly to the present work, Seebut, Wongsason and Kim (2023) demonstrated that the combination of GPT and computational notebooks can support student exploration of numerical solutions to difference equations, concluding that such tools foster high levels of self-efficacy. Nonetheless, no study to date has systematically evaluated and compared the performance of multiple LLMs on a controlled set of analytically solved ordinary difference equations across the full range of structural types, assessed by independent domain experts, and interpreted within an explicit pedagogical framework.

The present study addresses this gap through a multi-dimensional comparative evaluation of three commercially available, free-tier LLMs ChatGPT-4o, Gemini 2.5 Flash, and Claude 4.5 Sonnet on a structured dataset of twenty difference equation problems. The evaluation combines quantitative metrics (expert ratings, precision, recall, F1-score, and ANOVA) with a qualitative analysis grounded in Bloom's Revised Taxonomy (Krathwohl 2002), enabling a principled mapping of model capabilities onto the cognitive demands of mathematical instruction. The study further extends its contribution beyond performance measurement by proposing a Human-AI Collaboration framework for STEM pedagogy grounded in Bloom's Inverted Taxonomy, which reframes the LLM performance profile not as a ceiling to be raised but as an architectural foundation for instructional design that begins at the generative and evaluative levels where human contribution is irreplaceable. The deliberate choice of exclusively free-tier models situates this framework within a broader argument for equitable access to AI-enhanced education.

## GENERATIVE AI AND LARGE LANGUAGEMODELS(LLMS)

A rapidly developing area of artificial intelligence called "generative AI" is trying to fulfill one of the human desires to build general artificial intelligence. This is advancing the previous state of the art AI algorithms. Namely, generative AI models can produce outputs in a way similar to human, in contrast to traditional

AI models, which are usually built to identify patterns and make predictions based on past data. This ability stems from advanced algorithms that allow these models to generate content by learning the underlying structure and distribution of the data they are trained on.

The creation of Large Language Models (LLMs) is one of the key developments in generative AI. Deep learning models, known as LLMs, are trained on large text corpora, which gives them the ability to comprehend and produce human-like outputs with a high level of coherence and fluency. The performance of the deep learning models stems from the ability to train them on large amounts of data and also on the number of layers and their architectures of artificial neural networks. These models make use of neural network architecture, specifically transformers. Transformers are better compared to other architectures because they totally avoid recursion, by processing sentences as a whole and by learning relationships between words thanks to multi-head attention mechanisms and positional embeddings.

During the training phase of LLMs, a model learns to predict the next word in a sentence based on the previous context using unsupervised learning on big datasets. In order to complete this task, which is called language modeling, the model must get a thorough grasp of grammar, semantics, and general knowledge. The biggest and most potent LLMs, have been trained on a variety of datasets that contain billions of words from books, papers, websites, and other textual sources. Because of their in-depth training, they are remarkably accurate and versatile when it comes to performing a variety of linguistic tasks, such as text completion, translation, summarization, and question answering.

In this study, three of them ChatGPT, Gemini and Claude were used.

ChatGPT is a cutting-edge language model created by OpenAI (OpenAI 2026). Its foundation is the Generative Pre-trained Transformer (GPT) architecture, which is excellent at producing text that makes sense in context. ChatGPT can carry out a variety of linguistic tasks, such as text completion, creative writing, and discussion, because it has been trained on a variety of internet texts. Because it can comprehend and produce responses that resemble those of a human, it can be helpful in applications like chatbots, virtual assistants, and automated customer service.

The company Anthropic developed the AI language model Claude (Anthropic 2026), with a focus on alignment and safety. Claude, which gets its name from the inventor of information theory, Claude Shannon, seeks to minimize the dangers connected with AI-generated material while offering dependable and contextually relevant solutions. To guarantee that its results are morally righteous and advantageous, it integrates alignment strategies and safety precautions. Claude is utilized in a number of situations where reliable and trustworthy AI replies are essential.

In the field of generative AI, Gemini is another sophisticated AI model created by Google DeepMind (Google DeepMind 2026).It improves its problem-solving abilities by fusing reinforcement learning with natural language processing. Gemini is intended to provide complex language creation and understanding while also picking up new skills from feedback and interactions. Gemini's dual strategy enables them to perform very well in dynamic contexts and applications that are needed for sophisticated language understanding and adaptive learning.

## STATE OF THE ART

Since the early beginnings of Large Language Models (LLMs) scientists were excited and tried to challenge these models to solve various mathematical problems. This research has shown both promising results and significant challenges. Early studies focused primarily on basic arithmetic and simple algebraic problems, demonstrated surprising capabilities in these areas (Xia and Yu 2023). However, as researchers delved into more complex mathematical domains, the limitations of LLMs became increasingly apparent.

Parra *et al.* (2024) conducted a comprehensive study on the mathematical abilities of various LLMs, including GPT-3, using a diverse set of problems ranging from elementary school mathematics to graduate-level topics. The authors' findings revealed that while LLMs showed proficiency in certain areas, they struggled with more advanced concepts and often failed to provide consistent, reliable solutions to complex problems.

In the realm of symbolic mathematics, transformer-based models could be trained to solve integration problems and ordinary differential equations, as stated by Koceski *et al.* (2025) and Becker *et al.* (2023). Their work suggested that LLMs might have the potential to assist in formal mathematical reasoning, though the extent of this capability remained unclear.

Wang and Lü (2023) explored the ability of LLMs to perform multi-step reasoning in mathematics. They found that while models could sometimes generate step-by-step solutions, they often made errors in logical reasoning or failed to maintain consistency throughout lengthy problem-solving processes. This highlighted the challenge of endowing LLMs with robust mathematical reasoning capabilities.

The issue of mathematical accuracy in LLMs was further examined by Polu and Sutskever (2020), who developed a system for formal mathematical reasoning using GPT-3. While their approach showed promise in proving simple theorems, it also revealed the limitations of current LLMs in handling the rigorous logical structure required for advanced mathematical proofs.

Wei *et al.* (2022) introduced the concept of "chain-of-thought" prompting, which significantly improved the performance of LLMs on multi-step reasoning

tasks, including mathematical word problems. This technique demonstrated that the way problems are presented to LLMs can dramatically affect their problem-solving abilities.

Our study conducted on a very specific domain reveals interesting conclusions that can help in further development of LLMs as well as for scientific and teaching community for incorporating them in the process of research and education.

## EXPERIMENTAL METHOD

This chapter outlines the experimental method employed in our comparative study on the accuracy of different Large Language Models (LLMs) for solving difference equation problems. The method includes the selection of difference equation problems, criteria for evaluating accuracy, the process for presenting problems to each LLM, and the method for analyzing and comparing results. To ensure a comprehensive evaluation of the LLMs' capabilities, we selected a diverse set of difference equation problems that vary in difficulty and type. The selection process involved the following steps:

• *Identification of problem types* – Different types of difference equations have been selected. They can be classified as: linear homogenous difference equations from first and higher order, linear non-homogenous difference equations and systems of difference equations. Each category represents a distinct challenge for the LLMs, testing their ability to handle various mathematical structures and complexities.

• *Range of difficulty* − Within each type, we selected problems that span a range of difficulties. This includes simple problems that require basic algebraic manipulations, intermediate problems that involve more complex transformations or multiple steps, and advanced problems that require deep mathematical insights and sophisticated solution techniques.

• *Inclusion of edge cases* − To assess the robustness of the LLMs, we included edge cases such as problems of dependency of initial conditions, dealing with rounding off errors, non-periodicity, etc. These edge cases are designed to test the LLMs' ability to handle atypical and challenging scenarios.

To ensure consistency and fairness in our comparison, we developed a standardized process for presenting problems to each LLM:

• *Problem formulation:* Each difference equation problem is formulated in clear, unambiguous mathematical notation, accompanied by any necessary initial conditions.

• *Contextual information*: When relevant, we provide brief contextual information about the problem's origin or application to assess the model's ability to incorporate domain knowledge.

• *Instruction set:* A consistent set of instructions is provided with each problem, directing the model to "Solve the following difference equation, showing all steps clearly and explaining your reasoning."

• *Multiple attempts:* To account for potential variability in LLM outputs, we present each problem to each model multiple times (n=3) and consider the best performance for our analysis.

The analysis and comparison of results was performed independently by three selected domain experts and each of them followed a structured approach:

1. *Solution verification:* We verified the correctness of each solution provided by the LLMs by comparing it with the known correct solution. Discrepancies were noted and categorized based on the type and severity of the error.

2. *Sequential analysis:* The intermediate steps provided by the LLMs were analyzed to assess their logical coherence and mathematical accuracy. This involved evaluating whether the steps followed a valid problem-solving strategy and whether they aligned with standard mathematical practices.

3. *Quantitative assessment:* We calculated quantitative metrics such as the percentage of correctly solved problems and the average number of correct intermediate steps. These metrics provided an objective basis for comparing the performance of the LLMs.

4. *Qualitative assessment:* In addition to quantitative metrics, we performed a qualitative assessment of the LLMs' explanations and justifications. This included evaluating the clarity and comprehensibility of the solutions, as well as the models' ability to provide insightful and relevant commentary on the problems.

By following this comprehensive method, we aimed to provide an objective evaluation of the accuracy and effectiveness of different LLMs in solving difference equation problems.

## EXPERIMENTAL EVALUATION

For the purposes of this study the free versions of the LLMs, namely, ChatGPT-4o, Gemini 2.5 Flash, and Claude 4.5 Sonnet, were used. Despite the level of development of all previous LLMs, they are considered non-deterministic systems and therefore understanding the way to interact with them is vital for obtaining accurate answers. Consequently, all the prompts were crafted in a way to clearly state the context, and particular topics of interest, as well as to outline the desired response. So, as a prelude of the evaluation study, each of the LLMs was introduced to the subject by couple of general specific domain prompts. Namely, they were all asked the following prompt:

"Imagine that you are expert in mathematics with a strong experience in difference equations. Can you please give me a short definition of difference equations and tell me what they are used for?"

Different but accurate and elaborate answers were provided by all of them. The corresponding answers are presented in *Figures 1, 2,* and *3*.

*Figure 1*

**Answer provided on the first introductory question by ChatGPT**

Certainly!

A **difference equation** is a mathematical expression that relates the difference between successive values of a variable, typically representing a sequence of numbers. It is analogous to a differential equation, which relates rates of change, but difference equations are discrete rather than continuous.

## Definition:

A difference equation involves an equation of the form:

$$y_{n+1} = f(n, y_n, y_{n-1}, \ldots, y_{n-k})$$

where $y_n$ represents the sequence, $n$ is an integer representing the position in the sequence, and $k$ is the order of the difference equation.

## Uses:

Difference equations are widely used in various fields, including:

1. **Economics:** To model economic phenomena such as inflation rates, population growth, and stock prices.
2. **Engineering:** In digital signal processing, control systems, and telecommunications.
3. **Biology:** For modeling population dynamics, spread of diseases, and biological processes.
4. **Physics:** To study discrete systems and phenomena in areas like quantum mechanics and statistical mechanics.
5. **Finance:** For option pricing models and predicting future stock prices or financial indicators.
6. **Computer Science:** In algorithms, numerical analysis, and computer graphics.

They provide a powerful tool for modeling and analyzing systems that evolve in discrete steps, making them essential in both theoretical and applied mathematics.

*Figure 2*

**Answer provided on the first introductory question by Gemini**

Difference Equations: A Brief Overview

**A difference equation is a mathematical equation that relates the values of a function at different points in a discrete domain.** In simpler terms, it's an equation that describes how a quantity changes over time or across intervals, where time or the intervals are measured in discrete steps rather than continuously.

Applications of Difference Equations

Difference equations are incredibly versatile and find applications in a wide range of fields. Here are some key areas:

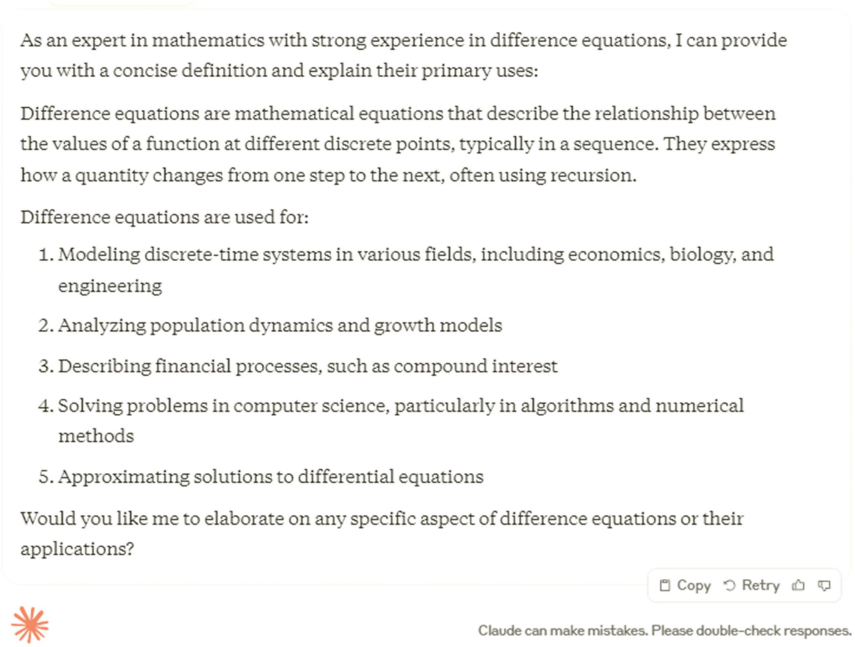
- **Economics:** Modeling economic systems, such as national income, inflation, and interest rates.
- **Biology:** Studying population dynamics, spread of diseases, and genetic inheritance.
- **Physics:** Analyzing discrete systems, such as crystal lattices and electrical circuits.
- **Engineering:** Designing control systems, signal processing, and numerical methods.
- **Computer Science:** Developing algorithms and analyzing data structures.

Essentially, whenever a system can be modeled as evolving in discrete steps, difference equations provide a powerful tool for understanding and predicting its behavior.

Would you like to explore a specific application or learn about solving difference equations?

*Figure 3*

**Answer provided on the first introductory question by Claude**




Interpreted through Bloom's Revised Taxonomy, results indicate that all three LLMs reliably operate at the first cognitive level of it i.e. Remember − which requires the ability to recall, define, and label factual knowledge. This is consistent with the training methodology of transformer-based LLMs, which are optimised on large-scale text corpora encompassing extensive mathematical literature and therefore possess robust declarative domain knowledge. The next cognitive level of Bloom's Revised Taxonomy is Understand – which requires the ability to summarise, classify, and explain concepts. To verify LLM competence at this level, all three models were prompted with 20 difference equations and asked to classify each in terms of linearity, homogeneity, and degree. All three models achieved maximum marks on this task, confirming reliable and consistent competence at the Understand level. Representative answers for a single equation are presented in *Figures 4, 5,* and *6*.

*Figure 4*

**Answer provided by ChatGPT when asked to characterize the difference equation a(n+2)+a(n+1)+2a(n)=sin(2n)**

Let's analyze the given difference equation:

$$a(n+2) + a(n+1) + 2a(n) = \sin(2n)$$

### Degree of the Difference Equation

The degree of a difference equation is determined by the highest order of the difference (the difference between the indices of $a(n)$) present in the equation.

In this case, the equation involves $a(n+2)$, $a(n+1)$, and $a(n)$. The highest order difference is 2 (from $n$ to $n+2$). Therefore, the degree of the difference equation is 2.

### Homogeneity

A difference equation is homogeneous if it can be written in the form where all terms involve the dependent variable (in this case, $a(n)$) and its shifts, without any additional constant or function of $n$.

The given equation:

$$a(n+2) + a(n+1) + 2a(n) = \sin(2n)$$

has a non-zero term $\sin(2n)$ on the right side, which does not involve the dependent variable $a(n)$ or its shifts. Therefore, the equation is non-homogeneous.

### Linearity

A difference equation is linear if the dependent variable and its shifts appear to the first power and there are no products or nonlinear functions (such as squares or other powers, exponentials, or logarithms) of the dependent variable or its shifts.

In the given equation:

$$a(n+2) + a(n+1) + 2a(n) = \sin(2n)$$

each term $a(n+2)$, $a(n+1)$, and $a(n)$ appears to the first power, and there are no products or nonlinear functions of these terms. The right-hand side $\sin(2n)$ is a function of $n$ and does not affect the linearity of the equation. Therefore, the equation is linear.

### Summary

- Degree: 2
- Homogeneity: The equation is non-homogeneous.
- Linearity: The equation is linear.

Thus, the difference equation $a(n+2) + a(n+1) + 2a(n) = \sin(2n)$ is a second-degree, non-homogeneous, linear difference equation.

*Figure 5*

**Answer provided by Gemini when asked to characterize the difference equation a(n+2)+a(n+1)+2a(n)=sin(2n)**

**Analyzing the Difference Equation: a(n+2) + a(n+1) + 2a(n) = sin(2n)**

**Homogeneity**

- **Non-homogeneous:** The presence of the term `sin(2n)` on the right-hand side makes the equation non-homogeneous. A homogeneous equation would have only terms involving the dependent variable or its shifts.

**Linearity**

- **Linear:** The dependent variable and its shifts appear only to the first power and are not multiplied together or involved in any nonlinear operations. Therefore, the equation is linear.

**Degree**

- **Second order:** The largest difference in the subscripts of the dependent variable is between `n+2` and `n`, which is 2. Therefore, the equation is of second order.

**In conclusion,** the given difference equation is a **second-order, linear, non-homogeneous difference equation.**

The presence of the non-homogeneous term `sin(2n)` complicates the solution process compared to a homogeneous equation. Techniques like the method of undetermined coefficients or variation of parameters are often used to solve such equations.

Would you like to explore methods for solving this type of difference equation?

*Figure 6*

**Answer provided by Claude when asked to characterize the difference equation a(n+2)+a(n+1)+2a(n)=sin(2n)**

Certainly, I'll analyze the given difference equation:

a(n+2) + a(n+1) + 2a(n) = sin(2n)

Let's break it down in terms of homogeneity, linearity, and degree:

1. Homogeneity: This equation is non-homogeneous. A homogeneous difference equation has zero on the right-hand side. Here, we have sin(2n) on the right-hand side, which is not zero for all n.
2. Linearity: This equation is linear. It satisfies both additivity and homogeneity properties with respect to a(n):

- The terms involving a(n) are all of first degree (no squares or higher powers).
- The coefficients of a(n+2), a(n+1), and a(n) are constants (1, 1, and 2 respectively).
- The right-hand side, sin(2n), does not involve a(n).

3. Degree: The degree of a difference equation is determined by the highest order difference that appears in the equation. In this case, the highest order term is a(n+2), which represents a second-order difference. Therefore, this is a second-degree or second-order difference equation.

To summarize, the equation a(n+2) + a(n+1) + 2a(n) = sin(2n) is:

- Non-homogeneous
- Linear
- Second-order (degree 2)

Would you like me to explain any of these characteristics further or discuss methods for solving this type of equation?

Copy Retry

Claude can make mistakes. Please double-check responses.

To probe competence at the Apply and Analyse levels of Bloom's Revised Taxonomy − which require, respectively, the execution of procedures to solve known problem types and the decomposition of complex problems into constituent parts through identification of structural relationships, the following prompt was administered in sequence to all three models:

*"You've mentioned that population dynamics can be modeled using difference equations. Can this problem be modeled with a difference equation:*

*A young pair of rabbits (one of each sex) is placed on an island.*

*A pair of rabbits does not breed until they are 2 months old.*

*After they are 2 months old, each pair of rabbits produces another pair each month.*

*And the pairs leave the island after reproducing twice.*

*If so, can you define the recurrence relation for the number of pairs of rabbits on the island after n months? Can you calculate the number of pairs of rabbits after 12 months?"*

This task simultaneously engaged three cognitive levels of Bloom's Revised Taxonomy. At the Understand level, models were required to recognise that the population dynamics described belong to the class of phenomena modelable by difference equations. At the Analyse level, models were required to decompose a multi-conditional natural language narrative into discrete formal mathematical constraints and identify the logical dependencies between them − the defining cognitive act of this level. At the Apply level, models were required to iterate the recurrence relation computationally to obtain a numerical result. All three LLMs successfully completed all three operations: they derived the correct recurrence relation, spontaneously identified its structural equivalence with the Fibonacci sequence, a cross-domain relational inference that constitutes the most demanding Analyse-level operation observed in this study, and correctly calculated the number of rabbit pairs after 12 months. This task therefore provides the strongest evidence in the study that current free-tier LLMs can operate reliably at the Analyse level of Bloom's Revised Taxonomy when problems are appropriately contextualised and prompted.

After this initial conversation each of the 20 difference equations were sequentially submitted to each of the LLMs as a separate prompt. Following the defined method, in order to account for potential variability in LLM outputs, each problem was presented to each model three times and the best performance was considered as a final answer for further analysis. The set of problems is given in *Table no. 1*.

*Table no. 1*

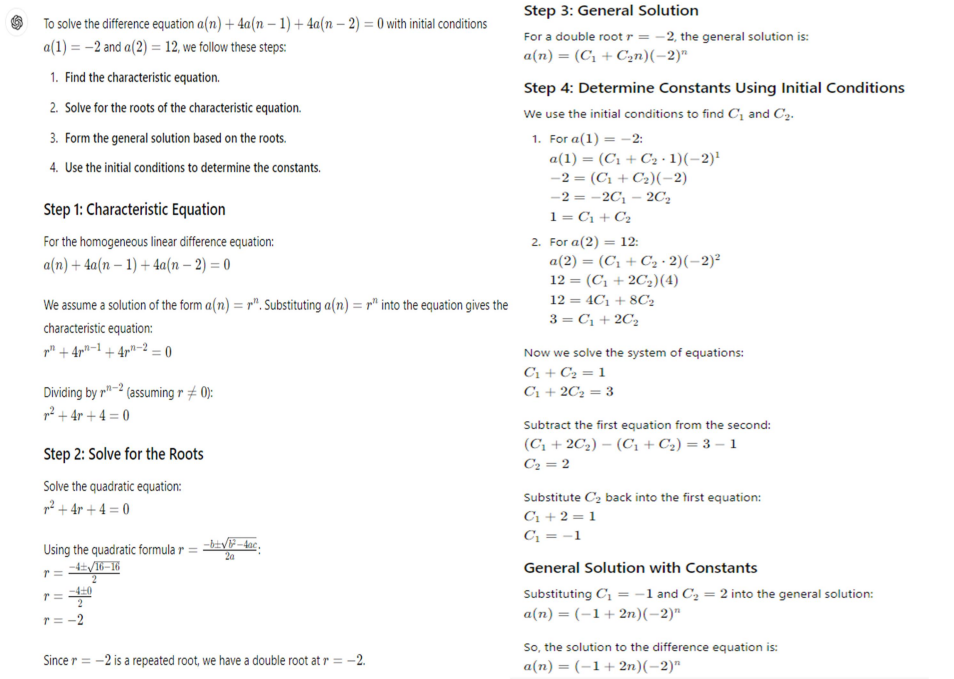
**Set of difference equation problems that were posed to all LLMs for solution**

| ID | Equation |
|---|---|
| 1 | $a_n = 4a_{n-1} - 4a_{n-2}, a_0 = 3, a_1 = 8$ |
| 2 | $a_n = 2a_{n-1} + 8a_{n-2}$ |
| 3 | $a_n = 3a_{n-1} + 4a_{n-2}, a_0 = 1, a_1 = 2\ for\ all\ n \geq 2$ |
| 4 | $a_n + 3a_{n-2} = 0, a_1 = 1, a_2 = 3, for\ all\ n \geq 3$ |
| 5 | $a_n + 4a_{n-1} + 4a_{n-2} = 0, a_1 = -2, a_2 = 12, for\ n \geq 3$ |
| 6 | $6a_n - 5a_{n-1} + a_{n-2} = 2$ |
| 7 | $6a_n - 5a_{n-1} + a_{n-2} = n, a_1 = 1, a_2 = 2, for\ n \geq 3$ |
| 8 | $6a_n - 5a_{n-1} + a_{n-2} = 2^n$ |
| 9 | $6a_n - 5a_{n-1} + a_{n-2} = (1/2)^n$ |
| 10 | $a_n + a_{n-1} - 6a_{n-2} = n, a_0 = 0, a_1 = 2, for\ n \geq 2$ |
| 11 | $a_{n+3} - 2a_{n+2} - 5a_{n+1} + 6a_n = 0$ |
| 12 | $a_{n+4} - 2a_{n+2} + a_n = 0$ |
| 13 | $a_{n+3} - 2a_{n+1} + 4a_n = 0$ |
| 14 | $a_{n+2} - 3a_{n+1} + 2a_n = 5^n$ |
| 15 | $a_{n+2} - 3a_{n+1} - 4a_n + 12 = 0$ |
| 16 | $a_{n+2} - 4a_n = 2^n$ |
| 17 | $a_{n+1} - 2a_n = \cos 2n$ |
| 18 | $a_{n+2} + a_{n+1} + 2a_n = \sin 2n$ |
| 19 | $u_{n+1} - 3u_n - 2v_n = -n$<br>$v_{n+1} - u_n - 2v_n = n$<br>$u_0 = 1, v_0 = 0$ |
| 20 | $u_{n+1} - 7u_n - 10v_n = 0$<br>$v_{n+1} - u_n - 4v_n = 0$<br>$u_0 = 3, v_0 = 2$ |

Just as illustration, provided answers from all LLMs on problem No.5 of the defined set of problems are presented on *Figures 7–9.*

*Figure 7*

**Answer provided by ChatGPT when prompted to solve the problem No. 5**



To solve the difference equation $a(n) + 4a(n-1) + 4a(n-2) = 0$ with initial conditions $a(1) = -2$ and $a(2) = 12$, we follow these steps:

1. Find the characteristic equation.
2. Solve for the roots of the characteristic equation.
3. Form the general solution based on the roots.
4. Use the initial conditions to determine the constants.

### Step 1: Characteristic Equation

For the homogeneous linear difference equation:
$a(n) + 4a(n-1) + 4a(n-2) = 0$

We assume a solution of the form $a(n) = r^n$. Substituting $a(n) = r^n$ into the equation gives the characteristic equation:
$r^n + 4r^{n-1} + 4r^{n-2} = 0$

Dividing by $r^{n-2}$ (assuming $r \neq 0$):
$r^2 + 4r + 4 = 0$

### Step 2: Solve for the Roots

Solve the quadratic equation:
$r^2 + 4r + 4 = 0$

Using the quadratic formula $r = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$:
$r = \frac{-4 \pm \sqrt{16 - 16}}{2}$
$r = \frac{-4 \pm 0}{2}$
$r = -2$

Since $r = -2$ is a repeated root, we have a double root at $r = -2$.

### Step 3: General Solution

For a double root $r = -2$, the general solution is:
$a(n) = (C_1 + C_2 n)(-2)^n$

### Step 4: Determine Constants Using Initial Conditions

We use the initial conditions to find $C_1$ and $C_2$.

1. For $a(1) = -2$:
   $a(1) = (C_1 + C_2 \cdot 1)(-2)^1$
   $-2 = (C_1 + C_2)(-2)$
   $-2 = -2C_1 - 2C_2$
   $1 = C_1 + C_2$
2. For $a(2) = 12$:
   $a(2) = (C_1 + C_2 \cdot 2)(-2)^2$
   $12 = (C_1 + 2C_2)(4)$
   $12 = 4C_1 + 8C_2$
   $3 = C_1 + 2C_2$

Now we solve the system of equations:
$C_1 + C_2 = 1$
$C_1 + 2C_2 = 3$

Subtract the first equation from the second:
$(C_1 + 2C_2) - (C_1 + C_2) = 3 - 1$
$C_2 = 2$

Substitute $C_2$ back into the first equation:
$C_1 + 2 = 1$
$C_1 = -1$

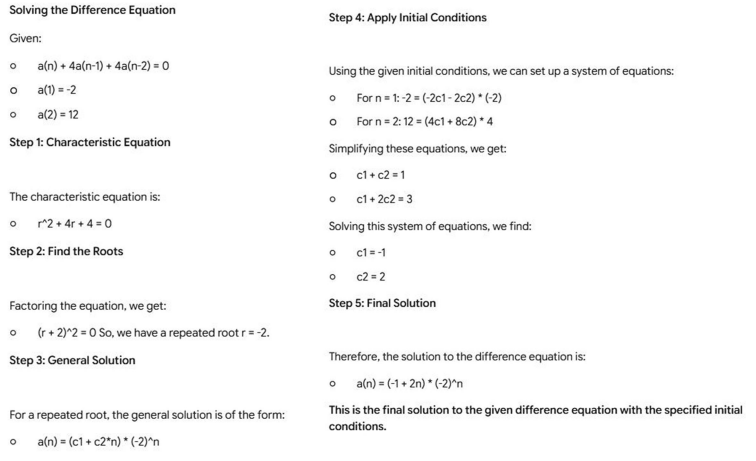
### General Solution with Constants

Substituting $C_1 = -1$ and $C_2 = 2$ into the general solution:
$a(n) = (-1 + 2n)(-2)^n$

So, the solution to the difference equation is:
$a(n) = (-1 + 2n)(-2)^n$

*Figure 8*

**Answer provided by Gemini when prompted to solve the problem No. 5**



**Solving the Difference Equation**

Given:

- a(n) + 4a(n-1) + 4a(n-2) = 0
- a(1) = -2
- a(2) = 12

**Step 1: Characteristic Equation**

The characteristic equation is:

- r^2 + 4r + 4 = 0

**Step 2: Find the Roots**

Factoring the equation, we get:

- (r + 2)^2 = 0 So, we have a repeated root r = -2.

**Step 3: General Solution**

For a repeated root, the general solution is of the form:

- a(n) = (c1 + c2*n) * (-2)^n

**Step 4: Apply Initial Conditions**

Using the given initial conditions, we can set up a system of equations:

- For n = 1: -2 = (-2c1 - 2c2) * (-2)
- For n = 2: 12 = (4c1 + 8c2) * 4

Simplifying these equations, we get:

- c1 + c2 = 1
- c1 + 2c2 = 3

Solving this system of equations, we find:

- c1 = -1
- c2 = 2

**Step 5: Final Solution**

Therefore, the solution to the difference equation is:

- a(n) = (-1 + 2n) * (-2)^n

**This is the final solution to the given difference equation with the specified initial conditions.**

*Figure 9*

**Answer provided by Claude when prompted to solve the problem No. 5**

Given equation: a(n) + 4a(n-1) + 4a(n-2) = 0, with initial conditions a(1) = -2 and a(2) = 12

Step 1: Determine the characteristic equation
The characteristic equation is:
r^2 + 4r + 4 = 0

Step 2: Solve the characteristic equation
This is a quadratic equation. We can solve it by factoring:
(r + 2)^2 = 0
r = -2 (double root)

Step 3: Write the general solution
Since we have a double root, the general solution takes the form:
a(n) = (A + Bn)(-2)^n, where A and B are constants to be determined

Step 4: Use initial conditions to find A and B
For n = 1: -2 = (A + B)(-2)
For n = 2: 12 = (A + 2B)(4)

From the first equation: A + B = 1
From the second equation: A + 2B = 3

Solving these simultaneously:
B = 2
A = -1

Step 5: Write the final solution
a(n) = (-1 + 2n)(-2)^n

We can verify this solution using the initial conditions:
For n = 1: a(1) = (-1 + 2(1))(-2)^1 = -2
For n = 2: a(2) = (-1 + 2(2))(-2)^2 = 12

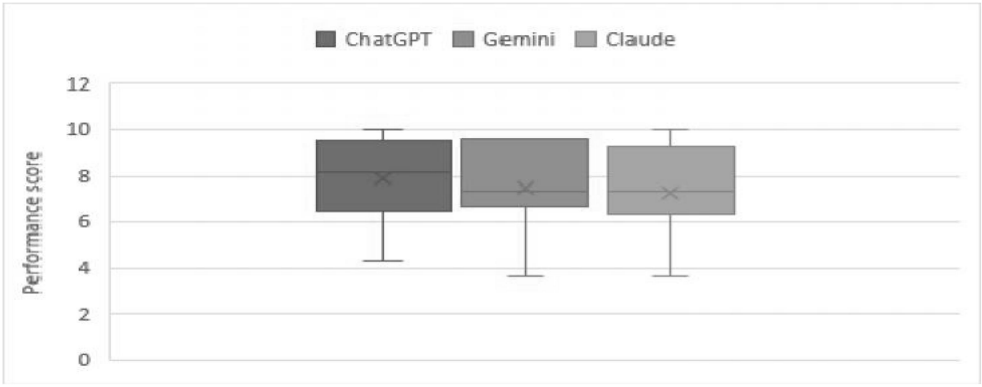
Therefore, the solution to the difference equation a(n) + 4a(n-1) + 4a(n-2) = 0, with a(1) = -2 and a(2) = 12, is:

a(n) = (-1 + 2n)(-2)^n

Following the defined evaluation method, each of the selected domain experts carefully evaluated the selected answer given by each of the LLMs with a mark from 1 to 10. The overall result of the evaluation was obtained as a sum of the average marks for each problem in the given set of problems. The results are presented on *Figure 10*.

*Figure 10*

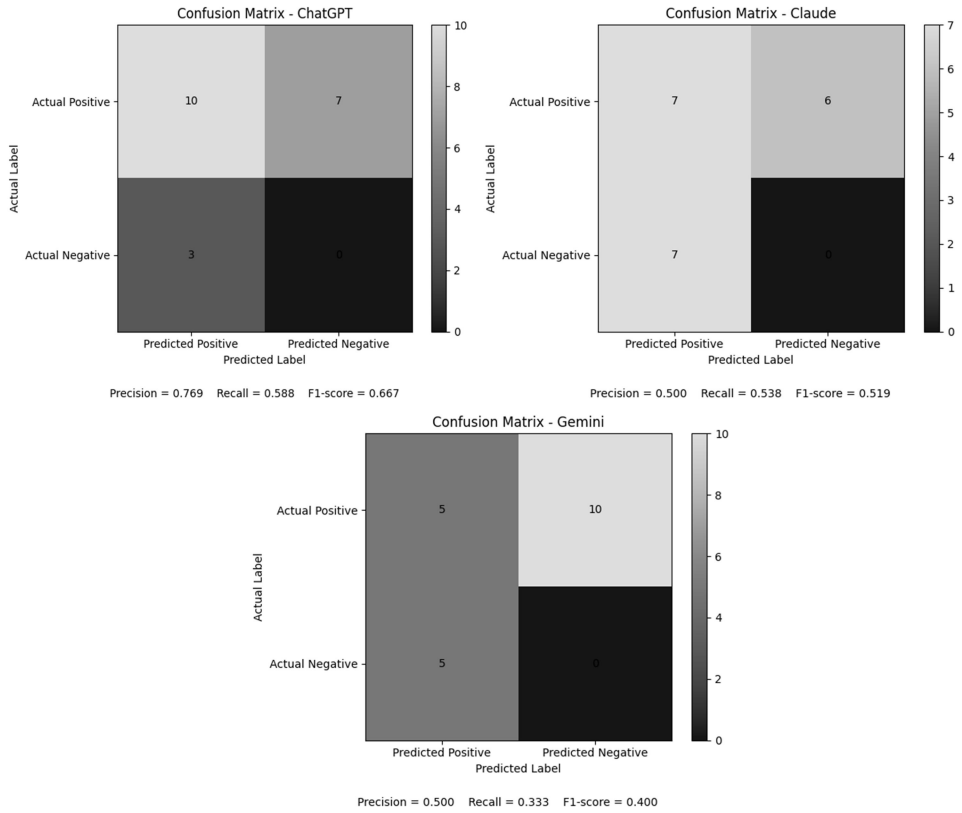
**Overall performance score for evaluated LLM models.**



Moreover, Precision, Recall and F1-score were calculated for each of the models for the given set of problems. The results are presented on the *Figure 11*.

*Figure 11*

**Confusion matrixes and calculated values for Precision, Recall and F1-score for evaluated LLM models.**

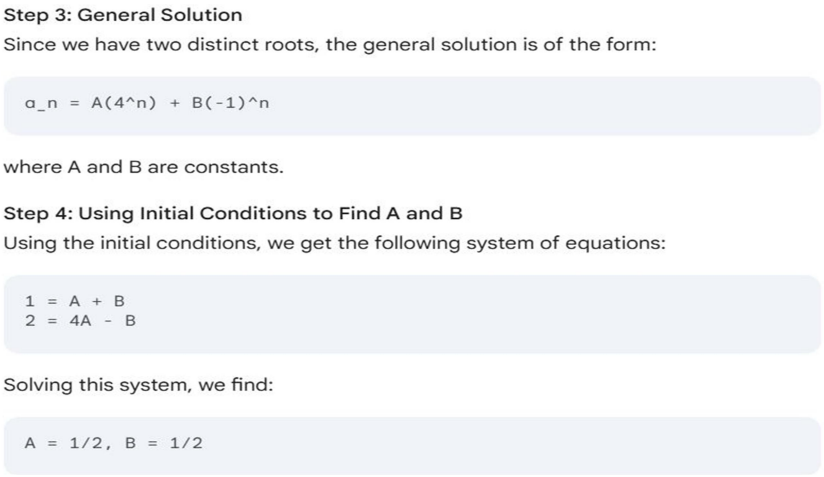



The results show that median scores of all LLMs are above 7 out of 10. Among all LLMs ChatGPT stands out as the most accurate model in terms of precision and has the highest F1 score, suggesting a relatively balanced approach between accuracy and coverage of solvable problems. On the other hand, Claude's and Gemini's lower precision and F1 scores suggest these models struggle with accuracy and may solve certain problems accurately. Qualitative assessment of the results showed that in general all LLMs are capable to comprehend the problem, offering a logical algorithm to solve it and elaborate the internal steps. In this process ChatGPT is by far superior to its competitors. However, when it comes to formalization of the solution, in many cases, almost all LLMs were struggling with

the basic arithmetic operations. This is illustrated in *Figure 12*, which shows Gemini's response when asked to solve difference equation number 3. Gemini accurately described the entire methodology for solving the equation, beginning with the determination of the general solution. After successfully solving the characteristic equation, it proceeded to find the coefficients using the initial conditions. However, it was unable to solve the resulting simple system of two linear equations.

*Figure 12*

**LLMs are struggling with relatively simple tasks.**
**Gemini fails to solve simple system of two linear equations.**




To evaluate the statistical significance of differences in accuracy among three LLMs solving difference equations, we have created two hypotheses: Null Hypothesis ($H_0$): The accuracy of all three LLM models is statistically the same and an Alternative Hypothesis ($H_1$): At least one model has a significantly different accuracy compared to the others. We performed an ANOVA Single Factor analysis. The results obtained showed that the null hypothesis is not valid at $p < 0.05$ which proves the alternate hypothesis i.e. that at least one of the evaluated models has a significantly different accuracy compared to the others.

By analogy we have created two hypotheses related to the quality of the mathematical explanations given by the models. In this context the Null Hypothesis was formulated as follows: $H_0$: "There is no difference in the quality of mathematical explanations between the models." And the $H_1$: "One model provides a statistically superior mathematical explanation compared to the others." Results obtained by the ANOVA Single Factor analysis showed that the null hypothesis is not valid at $p < 0.05$ which proves the alternate hypothesis i.e. that at least one of

the evaluated models provides statistically superior mathematical explanation compared to the others.

The results from ANOVA Single Factor analysis in both cases are presented in *Figures 13 and 14*.

*Figure 13*

**ANOVA Single Factor analysis showed that at least one of the evaluated models has a significantly different accuracy compared to the others.**

Anova: Single Factor

SUMMARY

| *Groups* | *Count* | *Sum* | *Average* | *Variance* |
|---|---|---|---|---|
| ChatGPT | 20 | 15.7 | 0.785 | 0.038016 |
| Claude | 20 | 12.68 | 0.634 | 0.104615 |
| Gemini | 20 | 10.68 | 0.534 | 0.122299 |

ANOVA

| *Source of Variation* | *SS* | *df* | *MS* | *F* | *P-value* | *F crit* |
|---|---|---|---|---|---|---|
| Between Groups | 0.63868 | 2 | 0.31934 | 3.616132 | 0.033225 | 3.158843 |
| Within Groups | 5.03366 | 57 | 0.08831 | | | |
| | | | | | | |
| Total | 5.67234 | 59 | | | | |

*Figure 14*

**ANOVA Single Factor analysis showed that at least one of the evaluated models provides statistically superior mathematical explanation compared to the others.**

Anova: Single Factor

SUMMARY

| *Groups* | *Count* | *Sum* | *Average* | *Variance* |
|---|---|---|---|---|
| ChatGPT | 20 | 16.35 | 0.8175 | 0.052314 |
| Claude | 20 | 12.02 | 0.601 | 0.112778 |
| Gemini | 20 | 9.67 | 0.4835 | 0.111592 |

ANOVA

| *Source of Variation* | *SS* | *df* | *MS* | *F* | *P-value* | *F crit* |
|---|---|---|---|---|---|---|
| Between Groups | 1.14823 | 2 | 0.574115 | 6.224937 | 0.003588 | 3.158843 |
| Within Groups | 5.25701 | 57 | 0.092228 | | | |
| | | | | | | |
| Total | 6.40524 | 59 | | | | |

Moreover, the analysis showed that all LLMs performed best with the linear homogeneous difference equations and exhibit difficulties with non-homogeneous and especially with systems of difference equations.

Therefore, a general conclusion is that LLMs hold significant potential to serve as valuable resources for supporting the learning of difference equations, offering personalized instruction and adaptive learning experiences. However, when it comes to the solution of difference equations, caution should be taken, as human expertise is still essential for understanding the underlying mathematical concepts and interpreting the results generated by LLMs.

In order to measure the level of agreement among the three independent domain experts who have rated selected problems, inter-rater reliability was assessed using the two-way random, single score intraclass correlation (ICC). It is a numerical value that varies between 0 meaning total disagreement and 1 perfect agreement among the raters. It is useful metrics as it assesses the reliability of ratings by comparing the variability of different ratings of the same subject to the total variation across all ratings and all subjects. Intraclass correlation (ICC) was calculated separately for every LLM based on ANOVA using MS Excel software. The results for ChatGPT are presented on *Figure 15*. Considering the fact that all ICC are above 0.9, according to the research presented by Koo and Li (2016), we may conclude that there is excellent agreement among experts.

*Figure 15*

**Calculation of ICC based on ANOVA for ChatGPT**

ANOVA: Two-Factor Without Replication

| *SUMMARY* | *Count* | *Sum* | *Average* | *Variance* |
|---|---|---|---|---|
| 1 | 3 | 30 | 10 | 0 |
| 2 | 3 | 30 | 10 | 0 |
| 3 | 3 | 30 | 10 | 0 |
| 4 | 3 | 29 | 9.666667 | 0.333333 |
| 5 | 3 | 30 | 10 | 0 |
| 6 | 3 | 28 | 9.333333 | 0.333333 |
| 7 | 3 | 26 | 8.666667 | 0.333333 |
| 8 | 3 | 25 | 8.333333 | 0.333333 |
| 9 | 3 | 25 | 8.333333 | 0.333333 |
| 10 | 3 | 25 | 8.333333 | 0.333333 |
| 11 | 3 | 24 | 8 | 1 |
| 12 | 3 | 23 | 7.666667 | 0.333333 |
| 13 | 3 | 22 | 7.333333 | 0.333333 |
| 14 | 3 | 21 | 7 | 1 |
| 15 | 3 | 19 | 6.333333 | 0.333333 |
| 16 | 3 | 24 | 8 | 0 |
| 17 | 3 | 19 | 6.333333 | 0.333333 |
| 18 | 3 | 18 | 6 | 0 |
| 19 | 3 | 13 | 4.333333 | 0.333333 |
| 20 | 3 | 15 | 5 | 0 |
| | | | | |
| Expert 1 | 20 | 157 | 7.85 | 2.344737 |
| Expert 2 | 20 | 161 | 8.05 | 3.734211 |
| Expert 3 | 20 | 158 | 7.9 | 3.147368 |

ANOVA

| *Source of Variation* | *SS* | *df* | *MS* | *F* | *P-value* | *F crit* |
|---|---|---|---|---|---|---|
| Rows | 164.4 | 19 | 8.652632 | 30.16514 | 2.83E-17 | 1.867332 |
| Columns | 0.433333 | 2 | 0.216667 | 0.755352 | 0.476769 | 3.244818 |
| Error | 10.9 | 38 | 0.286842 | | | |
| | | | | | | |
| Total | 175.7333 | 59 | | | | |

**ICC** **0.907767**

## QUALITATIVE ANALYSIS AND PEDAGOGICAL IMPLICATIONS

The quantitative results reported in the preceding section − median performance scores above 7/10, ANOVA-confirmed inter-model differences ($p < 0.05$), and ICC values exceeding 0.9, establish a statistically rigorous baseline for comparing the three evaluated Large Language Models. However, numbers alone are insufficient to guide instructional practice. A complementary qualitative lens is necessary to translate raw performance data into actionable pedagogical insights, particularly within the context of higher-education STEM curricula. The standard bottom-up progression provides a principled framework for characterising both what LLMs can contribute reliably and where human cognitive engagement remains indispensable across the full spectrum of mathematical learning.

Recent studies (e.g., Wei *et al.* 2022; Cobbe *et al.* 2021) suggest that high accuracy often masks inconsistencies in systematic reasoning, reinforcing the needs for qualitative verification. The present findings confirm this perspective by revealing that correctness at the final response level does not necessarily imply reliability at intermediate reasoning steps, a distinction that is pedagogically critical when LLMs are integrated into instructional workflows.

### Cognitive-Level Mapping via Bloom's Revised Taxonomy

The experimental protocol was structured in a manner that mirrors the hierarchical progression of Bloom's Revised Taxonomy (Anderson and Krathwohl 2001) enabling a direct mapping between observed LLM behaviours and the six cognitive levels of the taxonomy. This mapping is synthesised in *Table no. 2*, drawn directly from the empirical observations described in the Experimental Evaluation section.

At the Remember level, all three LLMs accurately recalled definitions, described the formal properties of difference equations, and situated them within their applied domains when prompted with introductory questions. At the Understand level, all models correctly classified all twenty equations by linearity, homogeneity, and degree, achieving maximum marks. Performance at these two foundational levels was consistent and reliable across all models, confirming that current free-tier LLMs possess robust declarative and conceptual mathematical knowledge.

This finding aligns with prior literature characterising LLMs as highly effective semantic retrieval systems trained on large-scale corpora (Kasneci *et al.* 2023), though it should be noted that such performance reflects probabilistic pattern matching rather than grounded conceptual understanding.

At the Apply level, a more nuanced picture emerges. Models executed correct solution procedures on standard problem types and as demonstrated by the rabbit population dynamics task, successfully connected procedural knowledge to real-

world scenarios. However, a systematic failure mode was identified across all models, correct procedural reasoning was frequently accompanied by arithmetic execution errors within those same derivations. This divergence between declarative procedural knowledge and computational precision is most clearly illustrated by Gemini's response to problem number 3, in which the model accurately narrated the entire solution methodology yet failed to solve the resulting simple system of two linear equations. This constitutes a failure of arithmetic execution at the Apply level rather than an absence of the higher-order reasoning that produced the correct strategy.

At the Analyse level, all three models demonstrated genuine competence in structural decomposition and relational reasoning. This level therefore marks the empirical ceiling of consistent LLM competence within the present experimental protocol. However, it remains an open question whether this analytical competence reflects genuine abstraction or learned pattern generalisation from training data, as suggested by studies on emergent abilities in large-scale models. This distinction is critical for assessing the transferability of LLM-supported reasoning to novel problem domains.

At the Evaluate level, performance was absent from all models. No model demonstrated the capacity to critically appraise the quality of its own outputs, identify its own computational errors, or exercise calibrated judgement over its derivations. This evaluative function was performed exclusively by the three human domain experts, whose authoritative judgement was validated by ICC values exceeding 0.9. While LLMs can simulate evaluative language when prompted, the absence of self-verification mechanisms and metacognitive awareness limits their ability to perform authentic evaluation, a limitation also noted in recent analyses of alignment and reasoning reliability (Bubeck *et al.* 2023).

At the Create level, performance was similarly absent: no model spontaneously generated novel problem variants, devised alternative solution strategies, or produced original mathematical formulations unprompted. Creative synthesis remained exclusively a human contribution throughout the study. This observation is consistent with broader findings that LLM creativity is largely recombinatorial rather than generative in a genuinely novel sense, reinforcing the necessity of human-led ideation in advanced STEM contexts (Kasneci *et al.* 2023, Bubeck *et al.* 2023).

*Table no. 2* presents the complete taxonomy mapping with empirical evidence and pedagogical implications for each level.

*Table no. 2*

**Mapping of LLM capabilities to Bloom's Revised Taxonomy levels with pedagogical implications.**

| Bloom's Level | Cognitive Demand | LLM Evidence in This Study | Pedagogical Implication |
|---|---|---|---|
| 1. Remember | Recall, Define, Label | All three LLMs accurately defined difference equations and recalled domain knowledge when prompted with introductory questions. | LLMs are reliable retrieval aids; suitable for pre-lecture priming or vocabulary review. |
| 2. Understand | Summarise, Classify, Explain | All LLMs correctly classified 20 equations by linearity, homogeneity, and degree, achieving maximum marks. | LLMs scaffold conceptual comprehension reliably; classification tasks may be delegated to LLMs for independent student use. |
| 3. Apply | Execute, Implement, Solve | Models executed correct solution strategies on standard problem types and connected difference equations to real-world scenarios; however, systematic arithmetic execution failures occurred across models, most severely on systems of equations. | LLMs serve as dynamic worked-example generators but require student verification of all computational steps; comparison tasks build procedural discipline. |
| 4. Analyse | Differentiate, Decompose, Identify Relationships | LLMs decomposed multi-step problems and identified logical sub-procedures; this is the highest cognitive level reliably demonstrated in this study. | Instructors should assign decomposition and error-identification tasks; the rabbit-Fibonacci case provides a concrete classroom illustration of cross-domain analytical reasoning. |
| 5. Evaluate | Critique, Judge, Appraise | Not demonstrated by LLMs. Authoritative evaluation was performed exclusively by the three human domain experts, validated by ICC values exceeding 0.9. No model appraised the quality of its own outputs or identified its own errors. | Students should apply structured rubrics to LLM-generated solutions, replicating the expert-rater methodology of this study; evaluative judgement is developed as a student competency through practice. |
| 6. Create | Design, Construct, Formulate | Not demonstrated. No model generated novel problem variants, alternative solution methods, or original mathematical formulations unprompted. Creative synthesis remained exclusively a human contribution. | Instructors should design capstone tasks requiring students to author original difference equation problems and critically compare LLM-generated solutions against their own model answers. |

The data in *Table no. 2* reveals a consistent ceiling effect: LLMs demonstrate strong and reliable performance at the lower three cognitive levels (Remember, Understand, Apply), with genuine Analyse-level competence, but are entirely absent at the Evaluate and Create levels. This pattern delineates a principled boundary between cognitive tasks that LLMs can scaffold reliably and those that must remain the primary responsibility of the human learner and instructor. It emerges as a key design principle for structuring Human–AI interaction in educational settings, rather than merely a limitation of current technology.

## Qualitative Error Patterns and Their Pedagogical Significance

Qualitative inspection of individual model outputs revealed two recurring error classes with distinct pedagogical significance:

• *Procedural accuracy with arithmetic breakdown.* As illustrated by Gemini's failure to solve a simple 2×2 linear system despite correctly narrating the broader solution strategy, LLMs can describe what should be done without providing an exact solution. This maps onto the gap between Bloom's Apply and Analyse levels: the model possesses declarative procedural knowledge and analytical decomposition capability but lacks consistent computational precision. For instructors, this pattern is instructionally valuable it creates a template for teaching error-detection as a higher-order skill. Assigning students the task of identifying and correcting LLM arithmetic errors is an activity that transforms model imperfection into a pedagogical resource.

• *Differential performance across problem types.* All LLMs performed best on linear homogeneous difference equations and worst on systems of difference equations. This gradient directly reflects the structural complexity of the problem taxonomy and is consistent with the expectation that more compositional, multi-step reasoning challenges current-generation LLMs. Instructors can exploit this gradient by sequencing LLM-assisted tasks from homogeneous to system-type problems, calibrating the level of human oversight required at each stage.

These qualitative findings underscore that traditional summative assessments which privilege correct final numerical answers, are poorly suited to an educational environment in which LLMs can generate structurally plausible but computationally unreliable solutions. A pedagogically responsible response to this challenge involves redesigning assessments around Bloom's upper cognitive levels, which current LLMs cannot contribute autonomously:

• *Evaluate-level tasks:* Students critically assess the quality of an LLM-generated solution using a domain-expert rubric, mirroring the expert-rater methodology employed in this study.

• *Analyse-level tasks:* Students identify the precise location and nature of an arithmetic or logical error in an LLM output and produce a corrected derivation with justification.

• *Create-level tasks:* Students design a novel difference equation problem that tests a specific concept, then compare LLM-generated solutions against their own model answer.

Such task designs simultaneously leverage the practical utility of LLMs and preserve the integrity of higher-order mathematical reasoning as an irreducibly human competency.

## PEDAGOGICAL INNOVATION: HUMAN-AI COLLABORATION IN STEM

The empirical and qualitative evidence presented in this paper converges on a single structural insight with direct consequences for STEM curriculum design: LLMs demonstrate reliable autonomous competence up to the Analyse level of Bloom's Revised Taxonomy, while the Evaluate and Create levels remain exclusively within the domain of human cognitive contribution. Rather than treating this asymmetry as a limitation to be overcome, the present section proposes that it constitutes the precise architectural foundation for a principled Human-AI Collaboration framework − one grounded in Bloom's Inverted Taxonomy and designed to exploit the complementary cognitive profiles of LLMs and human learners systematically and deliberately. This interpretation aligns with emerging perspectives in AI in education, which conceptualise LLMs not as replacements for human cognition but as cognitive partners that redistribute intellectual labour across human–machine systems (Holmes and Tuomi 2022).

### Theoretical Foundation: Bloom's Inverted Taxonomy as a Collaboration Architecture

Where Bloom's Revised Taxonomy prescribes a bottom-up instructional sequence (building from foundational recall toward higher-order creation), Bloom's Inverted Taxonomy argues that learning is most effectively initiated at the complex, generative end of the cognitive hierarchy, with lower-order knowledge acquired organically as a byproduct of engagement with authentic, higher-order tasks. In the context of AI-enhanced STEM instruction, this inversion acquires a new and specific empirical grounding from the present study. The LLM performance profile documented here, defines with precision the boundary at which human cognitive contribution becomes indispensable. That boundary falls between the Analyse and Evaluate levels, and it is exactly at that boundary that the inverted framework locates the primary entry point of instruction.

This yields a three-zone collaboration architecture. The human-primary zone encompasses the Create and Evaluate levels, where instruction begins and where the primary cognitive investment of both students and instructors is concentrated. LLMs have no autonomous competence in this zone; they supply artefacts −

solutions, classifications, derivations, that human learners evaluate, critique, and transcend. The collaborative zone encompasses the Analyse level, where the most productive human-AI interaction occurs: the LLM contributes structural decomposition and relational reasoning while the student supplies computational verification, error correction, and critical appraisal of the analytical conclusions reached. The LLM-scaffolded zone encompasses the Apply, Understand, and Remember levels, where the AI system operates with sufficient reliability to reduce the demand on instructor time and student prior knowledge, allowing both to be redirected toward the higher zones. Instruction does not begin here; it arrives here as consolidation after higher-order engagement has established the functional need for foundational content. This three-zone model can also be interpreted through the lens of the Vygotsky's Zone of Proximal Development (Shabani, Khatib and Ebadi 2010), with LLMs partially assuming the scaffolding role traditionally held by instructors, although without adaptive awareness or diagnostic capability.

Critically, the inverted taxonomy provides the rationale for this resequencing: because LLMs now reliably supply the scaffolding that traditionally occupied the opening sessions of a mathematics module, the justification for beginning instruction at those levels is substantially weakened. Students can be positioned at the Create and Evaluate levels from the first session, drawing on LLM-mediated retrieval and classification support as required, rather than spending instructional time on content that AI systems now handle reliably and on demand. However, this resequencing introduces new instructional risks, particularly the potential for uncritical acceptance of AI-generated outputs, which requires the explicit integration of verification and reflection as core learning activities.

## Practical Implementation: A Bloom-Inverted LLM Pedagogy for Difference Equations

The following five-phase instructional sequence operationalises the collaboration architecture for a higher-education module on difference equations, proceeding from the apex of the inverted taxonomy downward.

*Phase 1* − Generative Entry Point (Create). The module opens without prior formal instruction. Students are presented with a real-world scenario drawn from population biology, economics, or signal processing and are asked to formulate a difference equation that models it, specifying its structural type, order, and initial conditions. They then submit the formulated problem to two LLMs, document the AI responses, and produce a reflective commentary comparing the LLM formulations against their own. No definitional or procedural instruction is provided at this stage; the task surfaces existing intuitions, generates productive cognitive conflict, and establishes the LLM as an object of critical scrutiny from the first session rather than as an authoritative source.

*Phase 2* − Evaluative Dialogue (Evaluate). Students apply a structured expert-rater rubric, modelled directly on the evaluation instrument used in this study, to a set of LLM-generated solutions for assigned difference equation problems. Working in small groups, they compare ratings, resolve inter-rater disagreements through structured discussion, and produce a consolidated evaluative report. This phase replicates the ICC methodology of the present research at the classroom scale, positioning evaluative domain judgement as a student-developed competency acquired through practice rather than as a criterion transmitted by the instructor.

*Phase 3* − Critical Analysis (Analyse). Students receive LLM solutions containing arithmetic and logical errors of the type documented in this study, specifically, solutions in which the procedural reasoning architecture is correct but computational execution fails at a sub-step. They are required to locate each error, classify it by type and severity, produce a fully corrected derivation with explicit justification, and reflect on what the error reveals about the boundary between LLM analytical competence and computational reliability. This phase transforms the central empirical finding of the present study into a deliberate instructional mechanism, developing analytical precision and verification discipline through direct engagement with AI failure.

*Phase 4* − Procedural Engagement (Apply). Students submit the structured problem set to the LLM individually, document the complete solution process generated, and systematically compare LLM step sequences against worked examples from the course text, flagging every point of divergence and assessing whether it represents an error, an alternative valid approach, or a presentational difference. The LLM functions here as a dynamic worked-example generator, but the comparison task ensures that student engagement remains active, critical, and analytically grounded in the competencies developed in the preceding phases.

*Phase 5* − Foundation Consolidation (Remember/Understand). The module closes with explicit consolidation of the definitional and classificatory knowledge that underpins all preceding phases. Students interact with the LLM using the introductory prompts from this study, annotate the responses for accuracy and completeness against the course reference materials, and produce a structured conceptual map of difference equation types, their formal properties, and their solution methods. By this point, students have encountered and functionally used this foundational content extensively in higher-order contexts; this phase formalises and consolidates what the preceding phases have made operationally necessary, completing the downward traversal of the inverted taxonomy.

Across all five phases, a critical design principle is the deliberate exposure of students to LLM errors as a pedagogical resource rather than a failure condition. This aligns with recent educational research suggesting that confronting and correcting erroneous AI outputs can strengthen metacognitive awareness and deepen conceptual understanding.

### From Instructional Sequence to Transferable Professional Competency

The Human-AI Collaboration framework proposed here pursues an objective that extends beyond effective difference-equation instruction. By initiating instruction at the Create and Evaluate levels and structuring the entire learning sequence around principled critical engagement with LLM outputs, the framework cultivates transferable competencies directly relevant to contemporary STEM professional practice: the capacity to critically appraise AI-generated technical content, to identify with precision the boundary between reliable and unreliable AI performance, to verify computational outputs that AI systems produce with structural confidence but arithmetic inconsistency, and to contribute original mathematical reasoning that current AI systems cannot replicate. The expert raters whose ICC agreement exceeded 0.9 represent the professional ideal toward which this pedagogy is oriented − domain specialists capable of exercising authoritative, calibrated, and independent judgement over AI-generated mathematical content. The five-phase inverted sequence described above is a scalable programme for developing that capacity in undergraduate STEM students, and its theoretical coherence derives from the alignment between the empirical findings of this study and the pedagogical logic of Bloom's Inverted Taxonomy. In this broader context, the findings point toward a shift in the epistemological foundations of STEM education, from an emphasis on solution production to a focus on solution validation and critical interpretation, reflecting the evolving role of AI in professional scientific practice.

## CONCLUSION

This study evaluated three freely accessible Large Language Models on a structured set of 20 difference equation problems spanning linear homogeneous, non-homogeneous, and system-type equations. The results yield a picture that is simultaneously encouraging and cautionary, with direct implications for both the research community and practitioners engaged in AI-enhanced STEM instruction.

All three models achieved median performance scores above 7 out of 10, attesting to the considerable mathematical reasoning capability of current-generation generative AI. ChatGPT emerged as the most accurate model, recording the highest precision and F1-score. ANOVA analysis confirmed that inter-model performance differences are statistically significant at $p < 0.05$, while intraclass correlation coefficients exceeding 0.9 validate the reliability of the expert evaluation framework.

These aggregate metrics must, however, be read alongside the qualitative error patterns that constitute the study's most instructionally significant finding. Across all models, a systematic divergence was observed between procedural

competence and computational execution: LLMs consistently identified correct solution strategies yet frequently failed at elementary arithmetic operations within those same derivations. The practical risk in an educational context is therefore not that students receive unhelpful responses, but that they may receive confident, structurally plausible, yet numerically erroneous ones.

Mapped through Bloom's Revised Taxonomy, the results define a coherent and actionable cognitive boundary: LLMs demonstrate reliable competence at the Remember, Understand, and Apply levels, while the Evaluate and Create levels remain exclusively within the domain of human cognitive contribution. Reframed through Bloom's Inverted Taxonomy, this boundary becomes the architectural foundation of the Human-AI Collaboration framework proposed in this paper: instruction should begin at the levels where human contribution is irreplaceable, with LLMs providing scaffolding infrastructure for the lower cognitive tiers, rather than ascending toward higher-order engagement only after foundational content has been transmitted.

The present study is not without limitations. The dataset of twenty problems, while representative, constrains generalisability, and the evaluation captures each model at a single point in time despite ongoing version updates. Future research should expand the problem dataset, track model performance longitudinally across version releases, extend evaluation to additional or domain-specialised LLMs, and empirically test student learning outcomes under the Bloom-Inverted Human-AI instructional designs proposed here. The critical insight emerging from this work is not that LLMs can solve difference equations, but that their specific pattern of strengths and failures, precisely mapped through Bloom's Revised Taxonomy and reinterpreted through its inverted counterpart, provides instructors with a principled basis for designing collaborative learning environments that are more effective, more equitable, and more aligned with the higher-order competencies that STEM education is ultimately tasked with developing. Extending this evaluation framework to other STEM disciplines would reveal whether the observed cognitive boundary between procedural reasoning and computational execution is domain-general or subject-specific. Furthermore, broadening the scope to include a wider array of mathematical topics, from calculus and linear algebra to statistics and discrete mathematics, would enable the construction of a more comprehensive competency map for LLMs, thereby informing discipline-tailored pedagogical strategies for human-AI collaboration across the entire STEM curriculum.

## *Declaration of AI use*

The authors declare that they used ChatGPT to assist with language revision, grammar checking, and the improvement of language clarity.

## REFERENCES

Anderson, Lorin W., and David R. Krathwohl. 2001. “A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom’s Taxonomy of Educational Objectives: Complete Edition.” Addison Wesley Longman, Inc.

Anthropic. 2026. “Claude 4.5 Sonnet” https://claude.ai

Belaghi, Mahmoud, and Murat Sarı. 2021. “Behaviour of the First-Order q-Difference Equations.” An International Journal of Optimization and Control Theories and Applications (IJOCTA) 11 (1): 68–74. https://doi.org/10.11121/ijocta.01.2021.00908

Becker, Sören, Michal Klein, Alexander Neitz, Giambattista Parascandolo, and Niki Kilbertus. 2023. “Predicting Ordinary Differential Equations with Transformers.” In International conference on machine learning, pp. 1978-2002.PMLR, 2023.arXiv. https://doi.org/10.48550/arxiv.2307.12617

Bubeck, Sébastien, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, EceKamar, Peter Lee, *et al.* 2023. “Sparks of Artificial General Intelligence: Early Experiments with GPT-4.” arXiv. https://doi.org/10.48550/arXiv.2303.12712.

Cobbe, Karl, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert *et al.* 2021.“Training Verifiers to Solve Math Word Problems.” arXiv. https://doi.org/10.48550/arXiv.2110.14168

Cushing, J. M. 2019. “Difference Equations as Models of Evolutionary Population Dynamics.” Journal of Biological Dynamics 13 (1): 103–127. https://doi.org/10.1080/17513758.2019.1574034

D’Cunha, Shilpa, and V. R. Lakshmi Gorty. 2022. “Drug Level Modelling with Difference and Differential Equations.” Journal of Physics: Conference Series 2287 (1): 012011. https://doi.org/10.1088/1742-6596/2287/1/012011

DeAngelis, Donald L., and SimeonYurek. 2015. “Equation-Free Modeling Unravels the Behavior of Complex Ecological Systems.” Proceedings of the National Academy of Sciences 112 (13): 3856–3857. https://doi.org/10.1073/pnas.1503154112

Google DeepMind. 2026. “Gemini 2.5 Flash.” https://gemini.google.com/app

Holmes, Wayne, and Ilkka Tuomi. 2022. “State of the Art and Practice in AI in Education.” European Journal of Education 57 (4): 542–570.https://doi.org/10.1111/ejed.12533

Kasneci, Enkelejda, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, *et al.* 2023. “ChatGPT for Good? On Opportunities and Challenges of Large Language Models for Education.”Learning and Individual Differences 103: 102274. https://doi.org/10.1016/j.lindif.2023.102274

Kavitha, D. Dorathy Prema, L. Francis Raj, Sandeep Kautish, Abdulaziz S. Almazyad, Karam M. Sallam, and Ali Wagdy Mohamed. 2024. “Fuzzy Difference Equations in Diagnoses of Glaucoma from Retinal Images Using Deep Learning.” Computer Modeling in Engineering and Sciences 139 (1): 801–816. https://doi.org/10.32604/cmes.2023.030902

Kelley, Walter G., and Allan C. Peterson. 2001. “Difference Equations: An Introduction with Applications.” Academic Press.

Khelifa, Amira, and Yacine Halim. 2021. “General Solutions to Systems of Difference Equations and Some of Their Representations.” Journal of Applied Mathematics and Computing 67 (1–2): 439–453. https://doi.org/10.1007/s12190-020-01476-8

Koceski, Saso, Natasa Koceska, Limonka K. Lazarova, Marija Miteva, and Biljana Zlatanovska. 2025. “Exploring the Performance of ChatGPT for Numerical Solution of Ordinary Differential Equations.” Journal of Technology and Science Education 15 (1): 18–34. https://doi.org/10.3926/jotse.2709

Koo, T. K., and M. Y. Li. 2016. “A Guideline of Selecting and Reporting Intraclass Correlation Coefficients for Reliability Research.” Journal of Chiropractic Medicine 15 (2): 155–163. https://doi.org/10.1016/j.jcm.2016.02.012

Krathwohl, David R. 2002. “A Revision of Bloom’s Taxonomy: An Overview.” Theory into Practice 41 (4): 212–218. https://doi.org/10.1207/s15430421tip4104_2

Kunsch, Pierre L., Alain Michel Theys, A. Chevalier, and J.-P. Iacopetta. 2000. “A System Dynamics Model of Stock Price Movements.” In Decision Making: Recent Developments and Worldwide Applications, 157–171. Springer US. https://doi.org/10.1007/978-1-4757-4919-9_11

Leippold, Markus, and Steven Schärer. 2017. “Discrete-Time Option Pricing with Stochastic Liquidity.” Journal of Banking and Finance 75: 1–16. https://doi.org/10.1016/j.jbankfin.2016.11.014

Lu, Fei, Kevin Lin, and Alexandre Chorin. 2016. “Comparison of Continuous and Discrete-Time Data-Based Modeling for Hypoelliptic Systems.” Communications in Applied Mathematics and Computational Science 11 (2): 187–216. https://doi.org/10.2140/camcos.2016.11.187.

Mickens, Ronald E. 2015. “Difference Equations: Theory, Applications and Advanced Topics.” https://doi.org/10.1201/b18186

OpenAI. 2026. “ChatGPT-4o.” https://chat.openai.com

Parra, Verónica, Patricia Sureda, Ana Corica, Silvia Schiaffino, and Daniela Godoy. 2024. “Can Generative AI Solve Geometry Problems? Strengths and Weaknesses of LLMs for Geometric Reasoning in Spanish.” International Journal of Interactive Multimedia and Artificial Intelligence 8 (5): 65–74. https://doi.org/10.9781/ijimai.2024.02.009.

Parveen Banu, M., and R. Jothilakshmi. 2021. “Forward and Backward Difference Equations in Digital Signal Processing.” Journal of Physics: Conference Series 1964 (2): 022022. https://doi.org/10.1088/1742-6596/1964/2/022022.

Polu, Stanislas, and Ilya Sutskever. 2020. “Generative Language Modeling for Automated Theorem Proving.” arXiv. https://doi.org/10.48550/arxiv.2009.03393

Shabani, Karim, Mohamad Khatib, and Saman Ebadi. 2010. “Vygotsky’s Zone of Proximal Development: Instructional Implications and Teachers’ Professional Development.” English Language Teaching 3 (4): 237–248.https://doi.org/10.5539/elt.v3n4p237.

Santiago, Renato, Matheus Koengkan, José Alberto Fuinhas, and António Cardoso Marques. 2020. “The Relationship between Public Capital Stock, Private Capital Stock and Economic Growth in the Latin American and Caribbean Countries.” International Review of Economics 67 (3): 293–317. https://doi.org/10.1007/s12232-019-00340-x

Seebut, Supot, Patcharee Wongsason, and Dojin Kim. 2023. “Combining GPT and Colab as Learning Tools for Students to Explore the Numerical Solutions of Difference Equations.” Eurasia Journal of Mathematics, Science and Technology Education 20 (1). https://doi.org/10.29333/ejmste/13905

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. “Attention Is All You Need.” In Advances in Neural Information Processing Systems.arXiv. https://doi.org/10.48550/arXiv.1706.03762.

Wang, Tianduo, and Wei Lü. 2023. “Learning Multi-Step Reasoning by Solving Arithmetic Tasks.” In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics, vol. 2, 1229–1238.Short Papers. https://doi.org/10.18653/v1/2023.acl-short.106

Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models.” Advances in Neural Information Processing Systems 35: 24824–24837. https://arxiv.org/abs/2201.11903

Xia, Jing, and Xinguo Yu. 2023. “Abstractive Analysis of Traditional and GPT-Based Methods for Solving Algebra Problems.” In Proceedings of the 7th International Conference on Education and Multimedia Technology, 101–105. https://doi.org/10.1145/3625704.3625768

Zlatanovska, Biljana, Marija Miteva, Natasa Koceska, Saso Koceski, Limonka Koceva Lazarova, and Mirjana Kocaleva. 2026. “Obtaining an Analytical Solution of First-Order Differential Equations with ChatGPT.” Mathematics and Informatics 69, no. 5. https://doi.org/10.53656/math2026-4-7-oas

Zhao, Jing, Xiangming Kong, and Chunxiao Wang. 2020. "The Boundedness of the Solution of a Class of Biological Population Models." Journal of Physics: Conference Series 1670 (1): 012023. IOP Publishing. https://doi.org/10.1088/1742-6596/1670/1/012023

*Această lucrare prezintă o evaluare comparativă a trei Modele de Limbaj de Mari Dimensiuni (LLM) accesibile gratuit – ChatGPT-4o, Gemini 2.5 Flash și Claude 4.5 Sonnet – privind capacitatea acestora de a rezolva ecuații cu diferențe finite obișnuite, acoperind formulări liniare omogene, neomogene și sub formă de sisteme. A fost dezvoltat un set de date structurat, format din douăzeci de probleme, iar soluțiile au fost evaluate independent de trei experți în domeniu, pe baza unei rubrici standardizate; fidelitatea inter evaluatori a fost confirmată prin coeficienți de corelație intraclasă de peste 0,9. Evaluarea cantitativă a utilizat precizia, fidelitatea (recall), scorul F1 și analiza variațională unifactorială (ANOVA), care a confirmat diferențe semnificative din punct de vedere statistic între modele, la un prag $p<0,05$. Toate modelele au obținut scoruri mediane de performanță de peste 7 din 10, iar ChatGPT a înregistrat cea mai mare precizie și cel mai mare scor F1. O analiză calitativă sistematică, încadrată în Taxonomia lui Bloom, relevă faptul că LLM-urile demonstrează o competență autonomă de încredere la nivelurile cognitive fundamentale – A-și aminti, A înțelege și, parțial, A aplica și A analiza –, în timp ce nivelurile A evalua și A crea rămân exclusiv în domeniul contribuției cognitive umane. Această graniță, identificată empiric în cadrul acestui studiu, oferă baza teoretică pentru cadrul de Colaborare Om-IA în predarea disciplinelor STEM din învățământul superior, fundamentat pe Taxonomia Inversată a lui Bloom, propus în această lucrare: instruirea ar trebui să înceapă de la apexul generativ și evaluativ al ierarhiei cognitive, unde contribuția umană este indispensabilă, iar LLM-urile servesc drept infrastructură de sprijin (scaffolding) pentru nivelurile cognitive inferioare. De asemenea, studiul identifică o tipologie recurentă de eroare comună tuturor modelelor – nararea procedurală corectă dublată de erori de execuție aritmetică – și discută valoarea acesteia ca resursă pedagogică deliberată. Rezultatele au o relevanță directă pentru cadrele didactice, designerii de curriculum și cercetătorii care urmăresc integrarea responsabilă a LLM-urilor în mediile de învățare a matematicii.*

***Cuvinte-cheie**: Modele de Limbaj de Mari Dimensiuni; Ecuații cu diferențe finite; Taxonomia lui Bloom; Colaborare Om-IA; Rezolvarea de probleme matematice; Educație STEM.*

*Received: 16.05.2026* *Accepted: 10.09.2026*