International Journal of Innovative Information Systems & Technology Research 14(1):11-22, Jan.-Mar., 2026

 www.seahipublications.org ISSN: 2467-8562

doi:10.5281/zenodo.18129624

Artificial Intelligence (AI) in Computational Science and Mathematics: Dynamics of the Application of Generative AI in Science and Mathematics Pedagogy

Omoghenemuko Greg Imoniyovwe[1], Edafeajiroke Michael Favour[2], Dedetolu Josephine[3], Ejakpovi Simeon Uvie[4]

[1]Department of Computer Science, College of Education, Warri, Delta State, Nigeria.
[2]Department of Computer Science, University of PortHarcourt, Rivers State, Nigeria.
[3]Department of Chemistry, College of Education, Warri, Delta State, Nigeria.
[4] Department of Mathematics, College of Education, Warri, Delta State, Nigeria.

E- mail:Omoghenemukog@gmail.com
Mobile Contact: +2348033839465

ABSTRACT
Computer and information and communication technology (ICT) are the major components of the digital age, while the Internet is the fulcrum of the information superhighway. The science of computing has given birth to a new born baby known as artificial intelligence (AI) in the 21st century. AI is the science of programming computer to simulate human behaviour or characteristics. Generative artificial intelligence (Gen AI) refers to the use of AI algorithms to create personalized learning experiences, simulate complex scientific phenomena, and facilitate interactive learning environments. The integration of Generative AI in science and mathematics has the potential to revolutionize the way students learn and interact with scientific and mathematical concepts. This article explores the applications of Generative AI in science and mathematics, including personalized learning, automated problem generation, and adaptive assessments. This article discusses how Generative AI can be used to create customized learning paths for students, provide real-time feedback, and enhance their understanding of mathematical differential calculus concepts. Additionally, we examine the potential benefits and challenges of implementing Generative AI in computational science and mathematics and provide examples of how it can be used to support student learning of differential equations. Generative AI has the potential to transform science and mathematics by providing students with personalized learning experiences that handles individual needs and abilities. With the application of Generative AI in the teaching and learning process, educators can create customized learning paths that adapt to each student's learning style, pace, and level of comprehension. This can help to improve student outcomes, increase engagement, and reduce frustration that is associated with learning complex scientific and mathematical concepts.
Keyword: Internet, artificial intelligence-AI, generative AI, science, mathematics, computational science, ICT, digital age.

INTRODUCTION

Teaching science and mathematics to learners (students) in a manner that students can grasp scientific and mathematical concepts has long been a challenge for many students since the exception of conceptualization of mathematics as a subject. With the advent of generative artificial intelligence (Gen AI), educators can now leverage powerful tools that assist science and mathematics teachers in planning instruction to support student learning and improve learners' outcomes. Generative AI can be used to generate scientific and mathematical problems, provide step-by-step solutions, and offer real-time feedback.

Generative artificial intelligence has become prevalent in discussions of educational technology, particularly in the context of science and mathematics. Gen AI models can engage in human-like conversation and generate answers to complex questions in real-time, with education reports accentuating their potential to make teachers' work more efficient and improve student learning (Walkington, 2025).

Generative artificial intelligence (Gen AI) has taken the world by storm since the release of ChatGPT by Open AI in November of 2022. This release marked an important milestone in the development of conversational artificial intelligence (AI) agents, driven by ChatGPT's ability to engage in human-like conversation and answer complex questions. Stakeholders immediately began imagining how these tools might be applied to education, and in November of 2024, Open AI released an online course to support K-12 teachers using ChatGPT in the classroom (Common Sense Education, 2024).

In mathematics education specifically, there has been much speculation of these models' ability to successfully solve mathematics problems (e.g., Garisto, 2022), given the history of computers' powerful and precise computational abilities being used to make mathematical breakthroughs, and the negative connotations mathematics work often has among K-12 US students. It has been over 2 years since ChatGPT's release, and research is rapidly emerging on its implications for mathematics education (Walkington, 2025).

The integration of generative artificial intelligence (Gen AI) in computational science and mathematics has the potential to transform the way students learn and interact with scientific and mathematical concepts.

Computational Science

Computational science refers to the science domain where computer science, mathematics, and domain-specific sciences like physics, chemistry, biology, economics, etc, convergence to analyse and solve complex problems (Omoghenemuko, 2025).

Artificial Intelligence (AI) and Generative Artificial Intelligence (Gen AI)

There is no classic distinction between artificial intelligence (AI) and generative artificial intelligence (Gen AI). There fact is that Gen AI is a subset of AI. However, the definitions of the two terms will give a clearer understanding to readers from all works of life.

Artificial intelligence (AI) refers to the ability of machines to perform tasks that typically require human intelligence that involves learning, problem-solving, decision-making, and perception. Omoghenemuko (2025) defines artificial intelligence (AI) as the science and technology of programming computers to simulate or mimic human intelligence through the use of computer systems that can perform tasks that typically require human intelligence, such as: Learning; problem-solving; reasoning; perception; language understanding and translation, etc.

Generative artificial intelligence (Gen AI) is a type of artificial intelligence that creates new contents, such as texts, images, videos, music, based on patterns it learns from existing data and algorithms. Generative AI refers to the use of AI algorithms to create personalized learning experiences, simulate complex scientific phenomena, and facilitate interactive learning environments. It works by using neural networks to analyse large datasets and then generate original output in response to a prompt or input (Omoghenemuko, 2025).

Application of Generative AI in Mathematics

Some ways Gen AI can be applied in mathematics include:

1. **Automated Problem Generation**

Generative AI can be used to generate mathematical problems that cater to individual students' needs and abilities. For example, the following equation can be generated using Generative AI:

$\frac{d}{dx}(3x^2 + 2x - 5) = 6x + 2$

Generative AI can generate similar equations with varying levels of difficulty, allowing students to practice and reinforce their understanding of mathematical concepts.

2. **Personalized Learning**

Generative AI can create customized learning paths for students, adapting to their learning style, pace, and level of understanding. By analyzing student data and performance, Generative AI can identify areas where students need additional support and provide targeted interventions. Suppose a student has a generated-AI customized problem on *integration by substitution*, as follow:

Evaluate $\int(2x + 1)\, e^{(x^2+x)}\, dx$

The mathematical algorithm adapted by the generative AI might help a student solve the problem by:

1. Providing the student with a hint: Look for a substitution to simplify the integral.
2. Offering the student, a step: Let $u = x^2 + x$, then $du = (2x + 1)\, dx$.
3. Guiding the students to rewrite the original integral: $\int e^u\, du$
4. Ask the student for the answer: $e^{(x^2+x)} + C$

The generative AI adapts to a given problem based on the input supplied, then it offers explanations or next steps as required.

Generative AI supports personalized mathematics learning by promoting:

- **Adaptive and customized content.** Gen AI can create tailored practice problems and exercises based on a student's specific learning pace and proficiency, ensuring they are challenged appropriately without being overwhelmed.
- **Instant, step-by-step feedback.** Gen AI tools can offer immediate feedback on problem-solving attempts and provide detailed, step-by-step solutions to complex problems, helping students learn from mistakes in real-time.
- **Concept visualization.** For abstract concepts, Gen AI can generate interactive graphs, 3D models, and other visual aids to help students grasp difficult topics more effectively.
- **Personalized interest-based problems.** Gen AI can create mathematics word problems that are more relatable to a student's background and interests, which helps to increase engagement and motivation.
- **Intelligent tutoring.** Some AI systems act as an intelligent tutor, offering a persistent learning companion that provides guidance and explanations, especially for students who may not have consistent access to human tutors.

3. **Adaptive Assessment**

Generative AI can be used to create adaptive assessments that adjust to each student's level of understanding. By generating questions that are tailored to individual students' needs, educators can gain a more accurate understanding of student knowledge and understanding.

Specialized Gen AI Tools for Mathematics

Many think that general-purpose large language models (LLMs) such as ChatGPT, Claude, can be used effectively for mathematics instruction. But it is a fallacy of hasty generalisation. ChatGPT, Claude, have significant limitations in mathematical calculation reliability and accuracy.

- Large language model (LLM), is *a machine learning model- a* type of artificial intelligence algorithm that uses deep learning techniques and massively large data sets to understand, summarize, generate and predict new content. The term *generative AI* also is closely connected with LLMs, which are, in fact, a type of generative AI that has been specifically architected to help generate text-based content.

- Over millennia, humans developed spoken languages to communicate. Language is at the core of all forms of human and technological communications; it provides the words, semantics and grammar needed to convey ideas and concepts. In the AI world, a language model serves a similar purpose, providing a basis to communicate and generate new concepts.

- The first AI language models trace their roots to the earliest days of AI. The Eliza language model debuted in 1966 at MIT and is one of the earliest examples of an AI language model. All language models are first trained on a set of data, then make use of various techniques to infer relationships before ultimately generating new content based on the trained data. Language models are commonly used in natural language processing (NLP) applications where a user inputs a query in natural language to generate a result that can comprehend and generate human language.

Gen AI mathematics-support softwares are specialized mathematical tools that receive additional fine-tuning for mathematical concepts, with the principle that ensures tools' support (not circumvent) productive struggle.

Specialized Gen AI mathematical tools that can be optimally used for mathematical pedagogy include: Khanmigo, Mathway, Photomath, Gauthmath, Snorkl, Mathos AI, Wolfram Alpha, Desmos, and Julius AI. These tools are better suited for computational tasks than general-purpose LLMs.

Strategies for Using GenAI to Support Deeper Mathematics Learning

Student use cases, outlines three specific ways students can leverage specialized AI tools to enhance their learning while maintaining productive cognitive engagement. These are:

1. **AI tutoring**: Uses specialized tools to help students think through problems, offering private support without judgment.
2. **Error analysis and feedback**: Identifies specific areas where understanding breaks down, providing safe spaces for students to ask questions.
3. **Procedural skills practice**: Creates personalized practice problems aligned with curriculum, allowing students to customize content to their interests

Teacher use cases presents six strategic applications for educators to incorporate Gen AI into their instructional practices while preserving the essential mathematical thinking processes, by;

- **Creating lessons and materials**: Gen AI helps develop lesson plans, instructional content, and targeted interventions aligned to learning objectives.
- **Generating authentic practice problems**: Gen AI creates contextual problems incorporating real-world scenarios and student interests.
- **Providing differentiation**: Gen AI develops modified assignments with multiple entry points for diverse learners.

- **Breaking down tasks and concepts**: Gen AI unpacks challenging content by identifying prerequisite skills and common errors.
- **Analyzing student data**: Gen AI identifies trends and action items while protecting student privacy.
- **Creating visual representations**: Gen AI generates graphs, charts, and interactive tools to illustrate mathematical concepts

It is imperative to note that, each use case includes detailed descriptions, instructional benefits, and caution points.

Algorithm

An algorithm is a set of finite logical step by step instructions that are designed to accomplish a specific task or to solve a specific problem. Algorithms usually take one or more inputs, execute them systematically through a series of logically well-defined steps, and provide one or more outputs. Algorithms are typically associated with computer science and are an essential element of computer programming. Algorithms can be used to accomplish a variety of computational tasks, such as performing calculations, they are also used to create relationships such as social media link or finding information in databases, such as surfing for information on the internet data warehouse or repository. An algorithm is a finite chronological solution steps to a well-defined problem (Omoghenemuko, 2019).

Algorithms may be designed and used outside computer science as well. They can be executed manually by humans or executed automatically by various digital devices. for instance, one can decide to perform long division manually on paper or use a calculator to do the same operation. Users do not need to understand the inner workings and configuration of algorithms in order to use them. In fact, many algorithms used by organisations are closely guarded secrets, blocking users from seeing exactly how they work and what happens within (Omoghenemuko, 2019).

There are different types of algorithms. They include: Search algorithm; sort algorithm; encryption algorithm; greedy algorithm; recursive algorithm; backtracking algorithm; divide-and-conquer algorithm; brute force algorithm; dynamic programming algorithm; hash algorithm; randomized algorithm; etc.

Differential Calculus and How Gen AI Can Use Algorithms to Solve Differential Calculus Problems

Differential calculus is a branch of mathematics that deals with rates of change and slopes of curves. Gen AI can be used to solve differential calculus problems in in various fields, such as: (i) Physics- for modeling motion, optimization of physical systems; (ii) Engineering- for design optimization, control systems; (iii) Economics- for modeling economic systems, optimization of resources using various algorithms.

Some ways Gen AI can be applied in differential calculus include:

1. **Optimization**: Gen AI can use optimization algorithms like gradient descent to find maxima or minima of functions.
2. **Symbolic differentiation**: Gen AI can use algorithms like symbolic manipulation to find derivatives of functions.
3. **Numerical differentiation**: Gen AI can use numerical methods like finite differences to approximate derivatives.

Generative AI Algorithms Used in Differential Calculus

Some Gen AI algorithms used in differential calculus include:

1. *Gradient descent*: This is an optimization algorithm that uses derivatives to minimize or maximize functions.
2. *Automatic differentiation*: This is a technique that computes derivatives of functions using computational graphs.
3. *Symbolic differentiation*
4. *Numerical differentiation*

An algorithm for solving a first-order differential equation

1. Start

2. Write the differential equation in a standard form (e.g., dy/dx = f(x,y) or standard linear form)

3. Is order = 1?
   - If Yes then Step 4.
   - If No then Step 10.
4. Is it linear (dy/dx + P(x)y = Q(x))?
   - If Yes then Step 5.
   - If No then Step 6.
5. Use integrating factor: μ(x)=exp(∫P dx). Solve for y, then go to 18 (Stop)

6. Is it separable (dy/dx = g(x)h(y))?
   - If Yes then Step 7.
   - If No then Step 8.
7. Separate variables and integrate: ∫dy/h(y)=∫g(x)dx. Solve for y, then Step 18

8. Is it exact or homogeneous (M(x,y)+N(x,y) y' = 0 and ∂M/∂y = ∂N/∂x)?
   - If Yes then Step 9.
   - If No then Step 11.
9. Find potential function Φ(x,y): dΦ = M dx + N dy. Solve Φ=C. Then Step 18

10. Is it linear with constant coefficients (n-th order)?
    - If Yes then Step 12.
    - If No then Step 13.
11. Can substitution (e.g., y=vx, Bernoulli, homogeneous) transform it?

    - If Yes then Step 6 or 4 (depending on transformed type) — show arrow back to the appropriate decision.
    - If No then Step 14.
12. Solve homogeneous solution (characteristic polynomial). Find particular solution (method of coefficients or variation of parameters). Combine: y = y_c + y_ p., then Step 18

13. Is Laplace transform appropriate (initial value problem, piece-wise)?
    - If Yes then Step 15.
    - If No then Step 14.
14. Is numerical method needed (no closed form or initial-value)?
    - If Yes then Step 16.
    - If No then Step 17.
15. Apply Laplace transforms, solve algebraic equation for Y(s), inverse transform to get y(t).

    then Step 18

16. Choose numerical method (Euler, Runge–Kutta, etc.). Implement and compute approximate solution, then Step 18

17. Consider qualitative or special analytic methods (phase plane, series solution, perturbation), then Step 18

18. Stop- the present general solution and apply initial/boundary conditions to get specific solution.

Figure 1

A group of flowcharting symbols and their functions

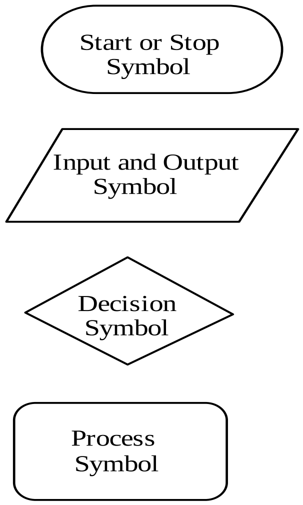
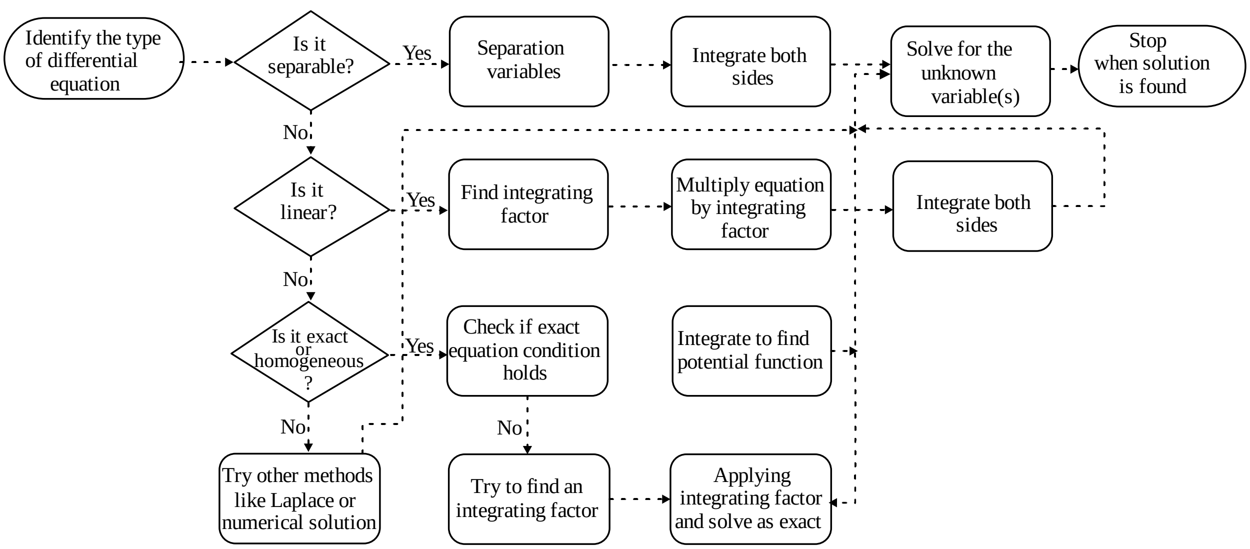


Figure 2

A flowchart showing the algorithmic structure for solving a first-order differential equation



Generative AI Algorithm for Implementing Gradient Descent in Python Computer Programming Language

A Gen AI algorithm for implementing gradient descent in Python computer programming language, is:

1. Start
2. Initialize parameters (weights, biases)
3. Compute the loss function (error) for the current parameters
4. Compute the gradient of the loss function with respect to each parameter
5. Update the parameters by subtracting a fraction of the gradient (learning rate)
6. Repeat steps 2-4 until convergence and solution
7. Stop computation

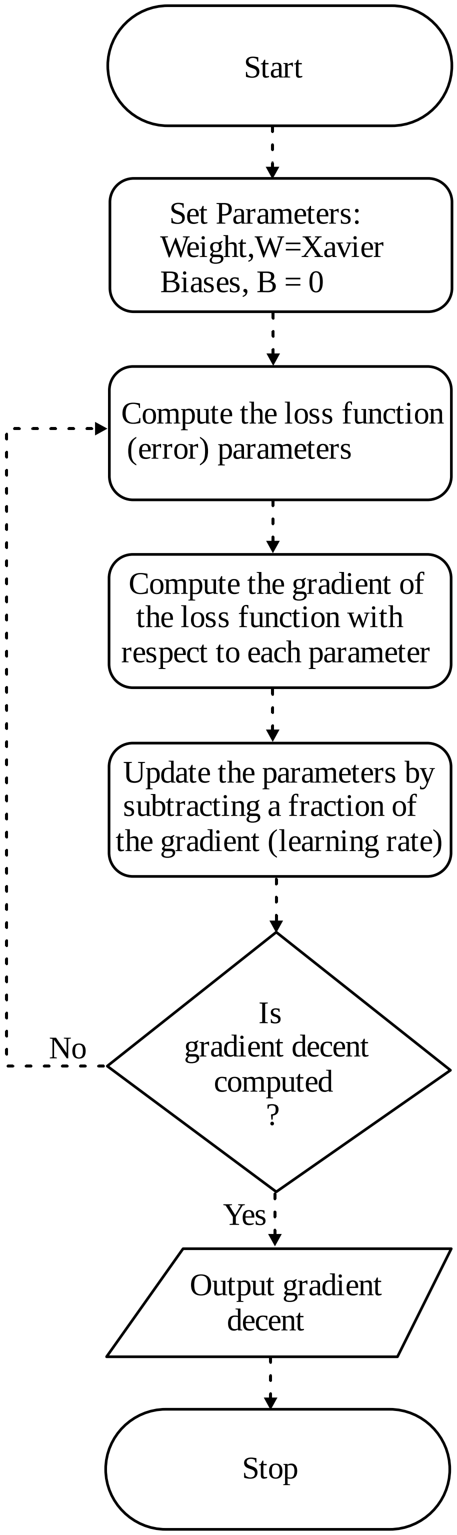


Figure 3
Flowchart for implementing gradient decent

This algorithm can be implemented using various programming languages and libraries, such as Python with NumPy, TensorFlow, PyTorch, etc.

An example using PyTorch is as follows:

```
import numpy as np

# Define the loss function (mean squared error)
def loss (y_pred, y_true):
   return np. mean ((y_pred - y_true) ** 2)

# Define the gradient descent update rule
def gradient_descent (weights, learning_rate, gradient):
   return weights - learning_rate * gradient

# Example usage
weights = np. array ([1.0, 2.0])
learning_rate = 0.01
gradient = np. array ([0.5, 0.3])
new_weights = gradient_descent (weights, learning_rate, gradient)
print(new_weights)
```

This code Implements gradient descent, a fundamental optimisation algorithm in machine learning. Breaking it down:

- *Loss function:* The code defines a mean squared error (MSE) loss function, commonly used for regression problems.

- *Gradient descent update rule:* The gradient_descent function updates the model weights based on the gradient of the loss function and the learning rate.

- *Example usage:* The code demonstrates how to use the gradient_descent function to update weights.

Some potential improvements include:
Add bias term- consider adding a bias term to the weights.
Use a more complex model- apply gradient to a more complex mode, like linear regression or neural networks.
Visualisation the optimization process- plot the loss function and weight updates to visualise the optimisation process.

Generative AI Algorithm for Implementing Automatic Differentiation in Python Computer Programming Language

1. Represent the function as a computational graph
2. Apply the chain rule to compute derivatives of each operation in the graph
3. Compute the derivative of the output with respect to each input using backpropagation
4. Return the computed derivatives

Library functions like TensorFlow or PyTorch provide automatic differentiation capabilities. An example using PyTorch is as follows:

```
import torch
# Define a simple function
def my_function(x):
   return x ** 2

# Create a tensor with requires_grad=True
x = torch. tensor (2.0, requires_grad=True)

# Compute the output
output = my_function(x)
# Compute the gradient using backward ( )
output. backward ( )

# Print the gradient
print (x.grad)
```

Generative AI Algorithm for Implementing Symbolic Differentiation in Python Computer Programming Language

1. Parse the mathematical expression into an abstract syntax tree (AST)
2. Apply differentiation rules to each node in the AST (e.g., power rule, product rule)
3. Simplify the resulting derivative expression
4. Return the derivative expression

Library functions like SymPy provide symbolic differentiation capabilities. An example using sympy is as follows:

```
import sympy as sp

# Define a symbolic variable
x = sp.symbols ('x')

# Define a function
f = x ** 2

# Compute the derivative
f_prime = sp.diff (f, x)

print(f_prime)
```

Generative AI Algorithm for Implementing Numerical Differentiation in Python Computer Programming Language

1. Choose a small value for the step size (h)
2. Evaluate the function at x and x+h
3. Compute the difference quotient (f(x+h) - f(x))/h
4. Return the approximate derivative value.

```
import numpy as np

# Define a function
def my_function(x):
   return x ** 2
```

```
# Define the numerical differentiation function
def numerical_derivative (f, x, h=1e-6):
   return (f (x + h) – f (x) ) / h
# Compute the derivative at x=2
x = 2.0
derivative = numerical_derivative (my_function, x)
print (derivative)
```

Benefits of Application of Generative AI in Science and Mathematics

Some of the benefits of application of generative AI in science and mathematics include:

1. **It promotes enhanced engagement**: Generated AI can make science learning more interactive and engaging, increasing student motivation and interest.
2. **It promotes simulation-based learning**: It is imperative to state that AI-powered simulations can model complex scientific phenomena, allowing students to explore and interact with virtual experiments in a safe and controlled environment.
3. **It provides virtual laboratories**: AI-powered virtual labs can provide students with hands-on experience with scientific equipment and experiments, reducing costs and increasing accessibility
4. **It improves students' understanding**: AI-powered simulations and personalized learning experiences can help students develop a deeper understanding of complex scientific concepts.
5. **It increases accessibility**: Generated AI can provide equal access to high-quality science for students with disabilities, remote or underserved communities.
6. **It helps in efficient assessment**: AI-powered assessment tools can help teachers evaluate students' performance more efficiently and accurately.

Challenges of Application of Generative AI in Science and Mathematics

Some of the challenges to the application of generative AI in science and mathematics, are the following.

1. **Accuracy and reliability**: The issue of ensuring the accuracy and reliability of generative AI outputs and addressing issues of bias and fairness. Gen AI models can sometimes produce incorrect or nonsensical results, and a significant challenge is the lack of true reasoning consistency. For instance, if Gen AI used the wrong formular to solve a problem, it definitely returns a wrong answer. Human oversight is critical for verifying the correctness of AI-generated proofs or solutions.
2. **Impact on learning**: There are concerns about how relying on AI might affect students' mental calculation skills. The goal is to use AI to support "productive struggle," a healthy challenge in learning, rather than making the learning experience too easy or counterproductive.
3. **Implementation and assessment**: Educators will definitely face challenges in integrating AI effectively into the curriculum and updating assessment methods to account for its use. Teachers need training on how to use these tools to optimize learning experiences.
4. **Issue of Bias and Fairness**: AI-powered systems can perpetuate existing biases if not designed with fairness and equity in mind.
5. **Data quality**: The effectiveness of generative AI in science and mathematics depends on the quality of the data used to train the AI models.
6. **Ethical concerns**: Potential issues include the risk of AI being used to create fraudulent proofs, which could undermine trust in scientific and mathematical research, and the need for responsible deployment to ensure fairness and privacy.

CONCLUSION

Generative AI in science and mathematics has the potential to transform the way students learn and interact with scientific and mathematical concepts. By leveraging Generative AI, educators can create personalized learning experiences, automate problem generation, and provide adaptive assessments. As the use of Generative AI in science and mathematics continues to evolve, it is essential to address the challenges and ensure that these tools are used effectively to support science and mathematics students' learning.

REFERENCES

Cardona, M. A., Rodríguez, R. J., & Ishmael, K. (2023). Artificial intelligence and the future of teaching and learning: Insights and recommendations. Office of Educational Technology, U.S. Department of Education. https://policycommons.net/artifacts/3854312/ai-report/4660267/

Garisto, D. (2022). AI language models are struggling to "get" math: Should this be telling us something? *IEEE Spectrum.* https://spectrum.ieee.org/large-language-models-math

Omoghenemuko, G. I. (2025). Computational science and mathematics of social media networks. A lecture note. Unpublished

Omoghenemuko, G. I. (2024). Information and communication technology in education: A global perspective *(*1st ed*.).* COEWA Publishers, ISBN 978-33794-6.

Sean, M. K. (2024). What are large language models (LLMs). https://www.techtarget.com/whatis/definition/large-language-model-LLM.

Walkington, C. (2025). The implications of generative artificial intelligence for mathematics education. https://doi.org/10.1111/ssm.18356.

Walkington, C., Milton, S., Pando, M., Lipsmeyer, L., Sager, M., & Beauchamp, T. (2024). Proceedings of the 15th international congress on mathematical education (ICME-15). In *Adolescents using generative AI to engage in mathematical problem-posing*. Springer. https://link.sprnger.com/book/10.1007/978-1-4757-4238-1