Optimizing Triple Parallel Demonstration in AI and Machine Learning Education: An Agentic Approach to Integrated Scientific Reasoning

Charlotte Sennersten[1] & Kamilla Klonowska[2]

[1, 2] Computer Science Department, Kristianstad University, Sweden

Abstract

Advanced education in Computer Science (CS), Artificial Intelligence (AI), and Machine Learning (ML) demands not only theoretical mastery but also high-level cognitive abilities, such as scientific reasoning. Traditional pedagogy often teaches foundational mathematics, architectural design, and empirical testing sequentially, hindering the integrated, real-time applied understanding required for complex AI systems. This article proposes a conceptual Triple Parallel Demonstration (TPD), a novel pedagogical model that integrates three crucial streams 'Empirical Investigative Studies, Software Architecture/Data Variables, and Mathematical Exemplification' simultaneously within a real-time, agent-assisted tutoring environment. Drawing on principles from human-centric AI Agentic Design (emphasizing transparency, control, and facilitation) and inquiry-based instruction, the TPD framework utilizes a multi-agent architecture to manage and present these streams in parallel. This approach ensures that theoretical concepts are immediately validated against empirical evidence and instantiated within a transparent architectural context. By making causal relationships immediately discernible, the conceptual TPD model fosters empowered scientific reasoning skills, including hypothesis formulation and evidence evaluation, among Bachelor's and Master's level students. While currently the TPD model is a work-in-progress conceptual design with no human subjects involved, this framework addresses the recognized scarcity of custom-built technology for developing complex thinking skills within Education 4.0.

Keywords: *Triple Parallel Demonstration, Scientific Reasoning, AI Agents, Inquiry Learning, Online Tutors, Computer Science Education, Software Architecture*

1. Challenges in Computer Science Education: The Need for Integrated Reasoning

The landscape of Computer Science, particularly within the context of international curricula [1], has grown increasingly complex. This complexity is especially pronounced in specialized areas such as Artificial Intelligence (AI) and Machine Learning (ML). Within these fields, bachelor and master's-level students and teaching staff are required to move beyond compartmentalized knowledge structures. Success in these domains depends not only on technical proficiency, but also on the ability to understand and evaluate scientific information, formulate hypotheses, and solve problems through integrated inquiry. These are core components of scientific reasoning. Scientific reasoning can be defined as the ability to recognize and understand the scientific method, including the concepts, processes, and applications used in the pursuit of knowledge formation. Scientific literacy, as an outcome of this reasoning, equips individuals to make informed decisions and engage with issues relevant to the natural, physical, and social world. However, low levels of scientific reasoning, particularly as content complexity increases, remain a recognized educational challenge [2]. Addressing this challenge requires effective pedagogical interventions that offer students structured opportunities to analyze information, evaluate evidence, and develop their reasoning abilities.

The TPD model is intended to offer the *Research Methodology* (DA311B) and *Artificial Intelligence* (DA601A) course students an integrated, interactive environment for learning. Rather than isolating theory, blueprints, or data, this model will enable learners to engage directly with architectural concepts while simultaneously observing related mathematical feedback and empirical results, following a relational cause-and-effect design principle.

1.1 Inquiry-Based Approaches and Their Role in AI and ML Education

Pedagogical methods known for strengthening advanced scientific reasoning often employ investigative, inquiry-based activities. These activities serve as the foundation for a newly designed *Research Methodology* course (DA311B), in which students are expected to define research problems, formulate hypotheses, identify variables, implement models, collect and analyze data, and communicate their results. This process unfolds within a condensed 7.5-week timeframe, as supported by course literature by Wohlin et al. [3]. Furthermore, recent developments of a couple of AI courses (e.g. *Artificial Intelligence* (DA601A), *Introduction to Artificial Intelligence* (DA599A)) emphasize the importance of AI Agent Architectures, further underlining the need for integrated investigative approaches. Within AI and ML education, such investigations naturally span three highly interconnected domains: real-world data and phenomena (Empirical Approach), the system that processes this data (Software Architecture), and the algorithms that guide the system's behavior (Mathematical Use).

1.2 Limitations of Sequential Instruction and the Case for Parallel Integration

Despite the foundational importance of these domains, current educational structures often address them sequentially and in isolation. Typically, instruction begins with abstract mathematical theory, proceeds with architectural implementation, and concludes with empirical testing. While Research Methodology or Thesis courses aim to foster an understanding of the interaction among these elements within the scientific process, this sequential approach does not accurately reflect the real-world complexity faced by software, AI, and ML engineers. In professional contexts, these individuals must simultaneously consider design, mathematical reasoning, and empirical evidence. Therefore, there is a clear need for an educational model that enables the concurrent demonstration of all three streams. Such a model would be especially valuable within a controlled online environment but would also benefit physical and hybrid teaching formats.

1.3 Problem formulation

Sequential teaching often hinders integrated learning. To help students grasp how concepts connect, a tutoring platform such as an AI agentic system and/or three projection screens is needed to enhance problem-solving and make computational and analytical processes transparent. The system should cover empirical investigation, model mathematical prediction, and reveal software architecture(s) like variables and control flow. Without this parallel approach, deep conceptual learning is challenging and fall back on sequential demonstration. While not yet implemented, this article presents a work-in-progress pedagogical design aimed at hands-on teaching and student experience.

1.4 Research Question

The following research question serves as an initial framework to define and refine the problem space over time: How can an online real-time AI tutoring system be effectively optimized using a Triple Parallel Demonstration (TPD) framework 'which integrates empirical research, software architecture featuring data variables, and mathematical examples' to enhance scientific reasoning among graduate students in Computer Science, Artificial Intelligence, and Machine Learning?

Although the potential benefits of parallel demonstration or multi-agent transparency in scientific reasoning are compelling, the absence of empirical evidence at present is formally noted as a hypothesis for future research.

2. A Brief Review of the Pedagogical Challenge

Internationally, we share a common challenge: as educators, we must find effective ways to demonstrate and model knowledge on current educational platforms. This includes developing our own approaches, utilizing public AI language models, and creating new strategies to keep up with global recommendations, guidelines and advancements [4].

A review of international foundational Computer Science curricula [1] and creating an initial search of the above-mentioned keywords (Triple Parallel Demonstration, Scientific Reasoning, AI Agents, Inquiry Learning, Online Tutors, Computer Science Education, Software Architecture) gave 16,300 results. Narrowing this keyword search string down to publications from 2025 yielded 3,540 hits. Restricting the selection to articles published from 2026 gave 101 publications (January 2026), but only three were truly pertinent: two general reviews and a single dissertation. All three provided only minimal direct information on the subject, which is outlined below.

A longitudinal review by Prayuda et al. [5] examined 350 articles published between 2000 and 2025, focusing on how artificial intelligence (AI) was increasingly being incorporated into education. Their study highlighted ongoing challenges regarding pedagogical effectiveness, global equity, and critical digital literacy, and provided guidance for future interdisciplinary research and inclusive education policy development. Frumin et al. focused on mapping generative AI research in higher education examined over 4,000 Scopus-indexed publications, ranging from 2022 to 2024, to outline trends in generative AI research within higher education [6]. McLaughlin's dissertation [7] provides a qualitative assessment of the challenges faced by instructional designers in online higher education due to the rapid development of generative artificial intelligence (AI) tools. These professionals must continually adapt their instructional methodologies and critically evaluate strategies for integrating AI technologies with human-centered learning practices. This situation closely parallels the experiences of teaching staff.

Further exploration using Notebook LM and the designated keywords highlighted two significant studies. The first was a systematic literature review examining how artificial intelligence could be integrated to support inquiry-based science teaching and learning [8]. The second study presented an initial transition from traditional "push-style" lectures to a student-driven "pull-style" approach to active learning [9]. In this framework, students participated in personalized, inquiry-based activities while educators concentrated on high-level curriculum planning and advanced Q&A sessions. The researchers also introduced a specialized analytical framework to evaluate anonymized chat transcripts, assessing educational engagement through metrics such as topic coverage, topic depth, and turn-level elaboration.

Due to the absence or scarcity of prior studies addressing the TPD path for these keywords, this research designated 'model formation' as a primary keyword. Consequently, two pertinent articles published before 2026 were identified. Perry's 2015 study [10] sought to establish a robust foundation for software engineering, providing comprehensive empirical support similar to that found in the natural and behavioral sciences. He pointed out that software engineering was still fairly new, with its core theories often being implicit rather than clearly outlined. Additionally, research and practices in the discipline were usually validated through anecdotes instead of systematic evaluation. A 2010 collaborative article from the University of Rome, IT University of Copenhagen, and the University of British Columbia called for more systematic empirical research on software architecture tools and methods. It aimed to shift focus from anecdotal evidence to rigorous studies and discussed challenges and findings from various empirical approaches, including experiments, reviews, and surveys [11]. These two latter articles highlight the need for systematic knowledge in software engineering, particularly when using AI and data-driven methods, and stress the value of organizing information to improve understanding for students and educators.

A preliminary search on the topic of pedagogy yielded a couple of relevant articles, which are referenced here. Creely and Carabott [12], within the context of a Generative AI track, found that teacher positionality is evolving from serving as an authoritative knowledge source to embracing the role of facilitator. Relationality redefines AI as a co-creative presence within the learning environment, while functionality introduces new approaches to assessment, feedback, and creative output. These foundational principles guide the formulation of the '*Integrated AI-Oriented Pedagogy Model*' aimed at transforming classroom practice and teacher education. The article's focus is on the pedagogical shifts emerging from GenAI integration, using three ontological dimensions 'positionality, relationality and functionality' to understand how roles and practices are changing. Their conceptual framework draws on phenomenology, particularly post-intentional phenomenology, to highlight the lived experience of these changes, and on posthumanism to attend to the agency and co-presence of AI systems in education. Suaidah et al. [13] aimed to evaluate learning activities within science education that foster scientific reasoning skills and to identify optimal instructional strategies. The authors conducted a literature review of 20 articles indexed in the Scopus database between 2017 and 2022. Their analysis revealed that investigative activities are the most commonly employed intervention to enhance scientific reasoning skills in science learning contexts.

The review study conducted by Patiño el al. [14] highlights complex thinking and how this is a desired competency in 21st-century university students, so therefore technology-based teaching and learning strategies must be carefully considered when training students in complex reasoning skills. The results of Patiño et al.'s review indicated that custom-built technological development for complex thinking development software incorporating emerging technologies is scarce at present. Further research is needed to document the interventions that train students interactively in complex thinking skills using Education 4.0 technologies.

UNESCO's AI competency framework for students [4], along with the associated framework matrix, serves as a comprehensive guide for defining learning outcomes across four key domains: 'Human-centered mindset,' 'Ethics for AI,' 'AI techniques and applications,' and 'AI system design.' The matrix incorporates three progressive mastery levels -'Understand,' 'Apply,' and 'Create'- which provide structured guidance for skill development. For the AI system design track, the knowledge objectives cited below are aligned with the conceptual TPD model.

"Architecture design: Students are expected to be able to cultivate basic methodological knowledge and technical skills to configure a scalable, maintainable and reusable architecture for an AI system covering layers of data, algorithms, models and application interfaces. Students are expected to develop the interdisciplinary skills necessary to leverage datasets, programming tools and computational resources to construct a prototype AI system. This includes the expectation that they apply deepened human-centered values and ethical principles in their configuration, construction and optimization" (p. 25/80) [4].

3. CS@HKR

The Computer Science Department at Kristianstad University (CS@HKR) has many years of experience in applying different educational models. One of them is constructive alignment, which ensures that course design, teaching activities, and assessment are explicitly connected to syllabus learning outcomes. Another example was the creation of project courses throughout the three-year bachelor's program, designed to combine programming skills with databases, software engineering, system design, and sustainable development, thereby reflecting real-world system complexity. In parallel, an *Applied Mathematics* (MA142A) course was introduced to support formal reasoning and analytical competence in technical contexts.

Despite these efforts, many of these initiatives remained structurally isolated; mathematical reasoning, programming practice, and architectural thinking were often taught sequentially rather than interactively. A constructive example of attempted integration is the *Introduction to Computer Science* (DA100D) course for first-year students, where mathematical logic is directly connected to algorithm construction and low-level programming in assembler. In this setting, mathematics becomes operational rather than abstract, forming the foundation for logical thinking required in software creation. However, even such integrations have tended to remain confined within individual courses rather than forming a sustained, parallel structure across the program.

This accumulated experience underscores the need for an approach such as the TPD model, in which empirical reasoning, software architecture, and mathematical formalization are presented in parallel rather than sequentially. Such integration reflects the interconnected nature of real-world problem-solving and aligns with contemporary educational demands.

In line with UNESCO's framework, recommended pedagogical strategies and writing guidelines incorporate scenario-based practices that intentionally leverage AI to strengthen students' writing proficiency and reasoning skills. This approach is currently implemented in the Research Methodology and AI courses, in collaboration with the university library to support structured referencing handling.

The TPD model builds on this foundation by supporting students in understanding relationships between concepts while simultaneously developing their own problem-solving strategies. Although still conceptual, the framework is informed by extensive experience in guiding students as they formulate research questions, design and implement solutions, and interpret results. Throughout this process, whether supported by AI tools or not, it is essential that the reasoning remains transparent, logically coherent, and clearly articulated in the written work.

4. The Triple Parallel Demonstration Model

AI and ML systems are complex, with many interconnected factors. Teaching Computer Science students, especially at the Master's level, requires clarifying how mathematical models, design decisions, and empirical results are causally linked.

Scientific reasoning including deduction, induction, abduction and distinguishing causation from correlation is fundamental to the development of intelligent systems. Within this context, machine learning predominantly focuses on inductive reasoning.

The Triple Parallel Demonstration (TPD) framework (figure 1) requires simultaneous engagement with three synchronized learning streams, guided by a human-focused AI agent in real time. This setup uses inquiry-based methods and ensures AI reliability. The framework aims to help students understand how meta- and object-data are connected, use TPD tools effectively, and eventually design their own agent frameworks for scientific applications with empirical rigor.

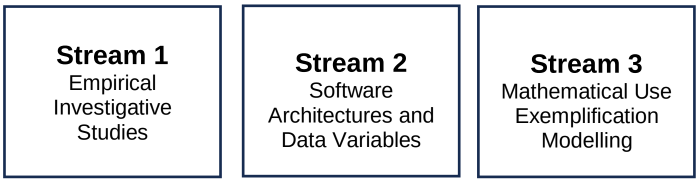

Fig. 1. The basis of a TPD framework.

4.1 TPD Stream 1: Empirical Investigative Studies (Scientific Reasoning in Parallel)

The first triplet stream emphasizes empirical investigation, a key intervention for developing scientific reasoning. In AI/ML, students analyze real-world problems or datasets through scientific inquiry -a process common to Computer Science and other disciplines during thesis writing.

Stream 1 has two main components: first, students are expected to understand the empirical process itself (referred to as the student comprehension model). Later, this same stream can be incorporated into an agent system or enhanced with AI LLMs (known as the student execution model). The second component runs in real time, requiring students to:

1) Formulate Hypotheses: Define a scientific problem and create testable hypotheses about the behavior of the AI/ML model or characteristics of the data.

2) Data Use, Exploration, and Collection: Access data, model inputs, and simulation outcomes via the online tool -much like gathering data to test hypotheses and finding evidence to back up claims.

3) Real-Time Data Analysis: Receive data analyses (such as graphs, charts, or bar diagrams) simultaneously with empirical activities. Students must identify empirical evidence and construct logical arguments based on these findings.

This process is essential for developing skills in correlational reasoning, probabilistic reasoning, and hypothesis-deductive reasoning, all of which are key for evaluating AI/ML models.

Machine learning presents a challenge because it relies on data-driven pattern discovery, which may not fit traditional hypothesis testing methods. This highlights the need to reconsider how we approach scientific reasoning in this context.

4.2 TPD Stream 2: Software Architectures and Data Variables (Agentic Transparency)

The second parallel stream addresses the technical framework, including software architecture and data variable flow. For ‘The student comprehension model’, it examines how the architecture supports or hinders the aims of TPD, followed by a related student execution model. While AI LLMs can assist, their knowledge often becomes fragmented and lacks consistent logic.

An effective online tutoring system must follow Agentic Design Principles:

Transparency: Users should know when AI is used and understand its workflow. The agent's internal processes, including data variables and model parameters, need to be visible and adjustable in real time.

Control: Students should be able to set preferences, edit prompts, and remove data. They must have the ability to modify architectural elements like activation functions or layer counts and instantly see their impact on results.

Facilitation: The agent connects students with relevant knowledge and context without replacing the learning process, providing real-time guidance for each learning stage.

This approach links architectural decisions directly to mathematical results and empirical performance, helping students better judge evidence and make scientific claims.

4.3 TPD Stream 3: Mathematical Use Exemplification (Integrated Modeling)

The third stream operates concurrently to illustrate the mathematical principles underpinning the system's behaviour. In artificial intelligence and machine learning, mathematics offers the essential framework for logical reasoning and informed decision-making. Mathematics serves as the fundamental basis of AI; thus, mastering its application and design is a critical aspect of this stream. It is important for students to understand the interactions among the various streams. From both implementation and end-user perspectives, these interactions should be presented as clear cause-and-effect relationships, demonstrating the interconnectivity needed for comprehensive understanding.

In the future implementation of the TDP, when a student engages in an investigative activity (Stream 1) that involves modifying a data variable within the architecture (Stream 2), the corresponding mathematical implications should be promptly visualized and clearly explained in a third concurrent view. The following example illustrates the functionality of this approach:

When a student changes input scaling (Empirical), the system should show the transformation functions (Mathematical) and verify the updated data variable range (Architecture).

The system illustrates the mathematical formalization of specific reasoning types, such as proportional and correlational reasoning, within its code architecture (Architecture). These formulations subsequently inform the empirical conclusions (Empirical) derived from the results.

The triple nature ensures that students do not just passively receive mathematical concepts but actively see how these concepts are "reconstructed through experimental activities" and used to evaluate evidence. The agent's role here is to simplify flows and dynamically generate cues to direct the user's attention to the specific mathematical concept being demonstrated at the right moment.

5. The AI Agent Architecture Supporting Scientific Reasoning

To support students and staff, the TPD model will be implemented as an AI multiagent system, extending beyond existing projection solutions. The architecture includes three sub-agents: one for Empirical Investigative Studies, another for Software Architecture/Data Variables, and a third for Mathematical Exemplification, each aligned with the core streams previously outlined.

The TPD model's strength is its ability to show real-time relational changes across three streams, allowing students to observe how these changes interact. For example, see illustration figure 2.

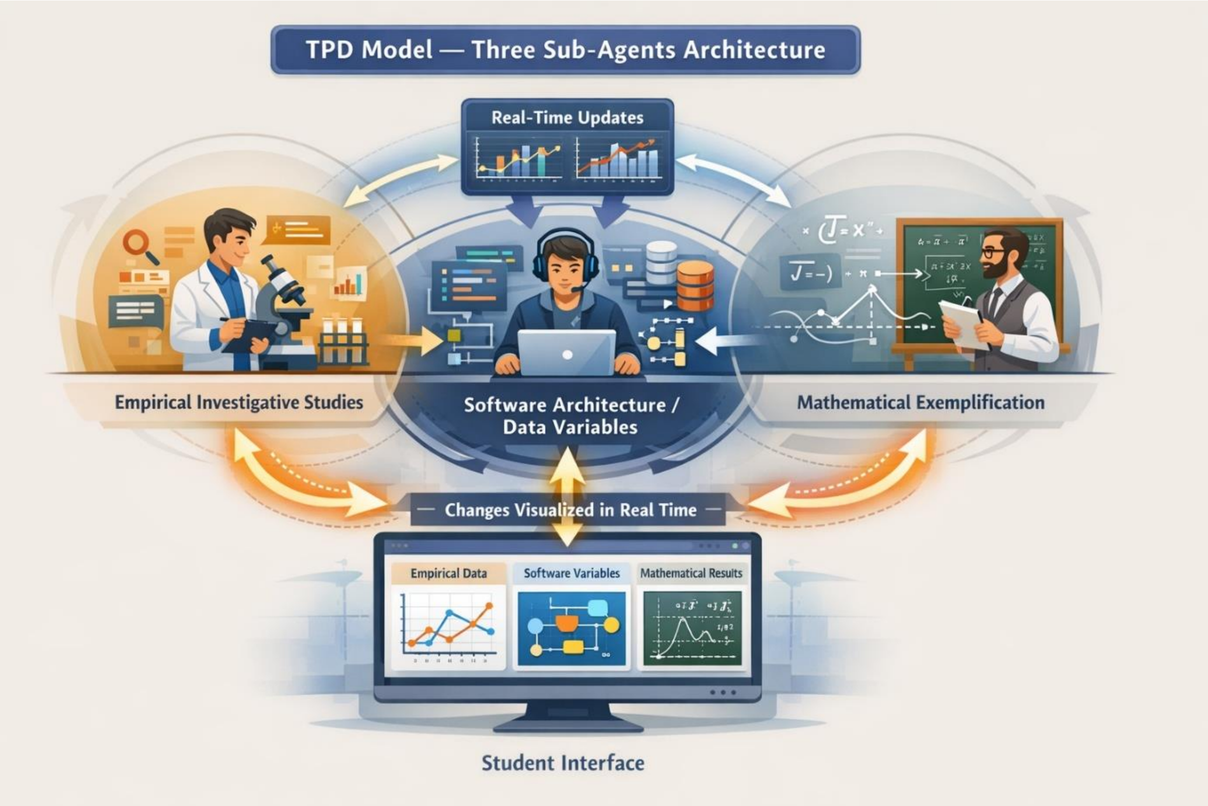

Fig. 2. A visual representation of the real-time TPD multi-agent architecture, generated using ChatGPT 5.2 (prompt describing the visualization requirements for the AI agent architecture, accessed on 09/01/26).

Although these streams can use pre-existing Large Language Models (LLMs) sequentially, this approach often fails to provide students or staff with a clear view of the scientific reasoning abilities and the underlying relationships in reasoning all at once. The flowchart in figure 3 shows how to implement the TPD scientific reasoning framework in parallel and supports its online relational knowledge system.

This flowchart presents a high-level overview of the TPD framework's implementation. The prompt window visualization is central, illustrating not only text responses but also how these answers connect and demonstrate the relationship among the three streams to both students and staff.

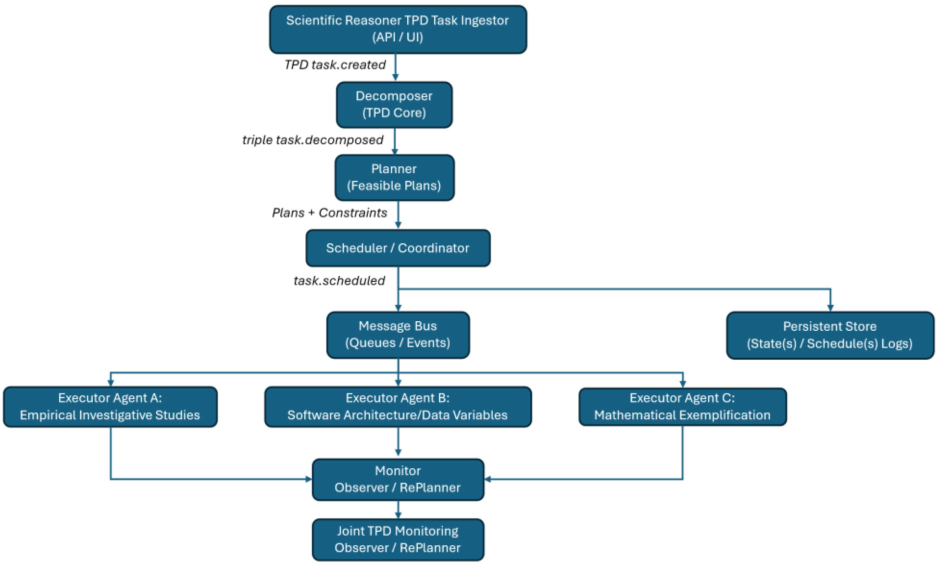

Fig. 3. A flowchart showing the TPD AI agent overview outline.

This scientific reasoning environment has one main goal, *to show relationships* between the multi-streams studied.

6. Conclusion and Discussion

Integrated, high-level thinking, characterized as scientific reasoning, is increasingly necessary within advanced technical disciplines, prompting the need for pedagogical approaches that transcend conventional sequential instruction. The Triple Parallel Demonstration (TPD) framework, underpinned by a specialized, transparent, and controllable AI agentic tutoring system, presents a methodology designed to enhance instructional delivery for Bachelor's and Master's students in Computer Science.

The TPD model enforces the concurrent implementation of Empirical Investigative Studies, Software Architecture Exposition, and Mathematical Exemplification. This approach will hopefully help and engage students in essential activities that cultivate scientific reasoning, including hypothesis formulation, data analysis, and evidence evaluation. Integrating principles of AI agentic design such as transparency and user control within the online learning environment supports knowledge construction, rendering causal relationships among theory, systems, and real-world phenomena immediately discernible rather than gradually deduced. Future research should prioritize the development of a prototype TPD system and the execution of randomized controlled studies to empirically assess improvements in integrated scientific reasoning compared to traditional sequential methodologies.

7. Future Work

Since all Computer Science students follow the same process in their Research Methodology course, a TPD platform would be highly useful for both educators and students. It would help students to quickly grasp connections and understand how these relationships result in cause-and-effect relationships across the three streams. This potential TPD implementation can also be carried out by Computer Science students in their later Thesis work.

Acknowledgement

The authors developed the concepts presented in this article following encouragement received at the international conference 'New Perspectives in Science Education 2026,' and wish to express their gratitude to the reviewer(s) for their valuable feedback.

REFERENCES

[1] Kumar, A. N., Raj R. K., et al., "Computer Science Curricula 2023", Association for Computing Machinery, New York, NY, United States, 2024, ISBN: 979-8-4007-1033-9, DOI: https://doi.org/10.1145/3664191

[2] Blumentahl, R., "THINKING ISSUES, An AI Useless Case: A Need for Curricular Cultural Computing Literacy", ACM Inroads, Volume 16, Number 4 (2025), Pages 10-14, DOI: 10.1145/3773089

[3] Wohlin, C., Runeson, P., Höst, M., Ohlssson, M., "Experimentation in Software Engineering", 2nd edition, Springer Berlin Heidelberg, 2024, DOI: 10.1007/978-3-662-69306-3

[4] Miao, F., Shiohira, K., and Lao, N., "AI competency framework for students", Paris: UNESCO, 2024, https://doi.org/10.54675/JKJB9835

[5] Prayuda, M. S., Ginting, F. Y. A., and Tamba, L., "Tracing two decades of artificial intelligence in education: A bibliometric analysis of trends, themes, and future directions (2000–2025)", European Journal of Educational Research, 2026, 15(1), 285-304. https://doi.org/10.12973/eu-jer.15.1.28.

[6] Frumin, I., Vorochkov, A., Kiryushina, M., Platonova, D. and Terentiev, E., "Mapping the Generative AI Research in Higher Education: 2022–2024 Insights", Higher Education Quarterly 80, no. 1: e70075, 2026. DOI: https://doi.org/10.24059/olj.v28i3.4501

[7] McLaughlin, A., "Instructional Designers' Perceptions on Generative AI Integration in Higher Education", Diss., Walden University, 2026. DOI: https://doi.org/10.24059/olj.v28i3.4501

[8] Ramli, N. A., & Mahmud, S. N. D., "Integration of Artificial Intelligence to Support Inquiry-Based Science Teaching and Learning: A Systematic Literature Review", International Journal of

Academic Research in Progressive Education and Development, 14(2), 2025, 893–911, DOI: https://doi.org/10.24059/olj.v28i3.4501

[9] Simmhan, Y., & Kulkarni, V., "Towards AI Agents for Course Instruction in Higher Education: Early Experiences from the Field", arXiv preprint arXiv:2510.20255, 2025.

[10] Perry, D. E., "A framework for exploring unifying theories of empirical software engineering, Science of Computer Programming, Volume 101, 2015, p. 99-123, ISSN 0167-6423, https://doi.org/10.1016/j.scico.2014.11.011

[11] Falessi, D., Babar, M.A., Cantone, G., and Kruchten, P., "Applying empirical software engineering to software architecture: challenges and lessons learned", Empirical Software Engineering, 15(3), 2010, p. 250-276, DOI:10.1007/s10664-009-9121-0

[12] Creely, E., and Carabott, K., "Teaching and learning with AI: an Integrated AI-Oriented Pedagogical Model", The Australian Educational Researcher: A Publication of the Australian Association for Research in Education, 2025, p.1–22. DOI:10.1007/s10664-009-9121-0

[13] Suaidah, H. L., Susantini, E., and Hariyono, E., "Determining Learning Activities to Promote Scientific Reasoning in Science Learning: A Literature Review", IJORER: International Journal of Recent Educational Research, 4(3), 2023, p. 271-285. DOI:10.1007/s10664-009-9121-0

[14] Patiño, A., Ramírez-Montoya, M. S., and Ibarra-Vazquez, G., "Trends and Research Outcomes of Technology-Based Interventions for Complex Thinking Development in Higher Education: A Review of Scientific Publications.", Contemporary Educational Technology, 15 (4): 447, 2023. DOI:10.30935/cedtech/13416