Global Education Ecology, 2025, 1(2), 16-27
https://doi.org/10.71204/a01mtr30

Research on the Design and Implementation of Early Childhood Programming Education under STEM Philosophy

Qimin Cao [1,*], Longtian Li [1]

[1] Huaiyin Normal University, Huai 'an 223300, China

*** Correspondence:**

Qimin Cao

1036884438@qq.com

Received: 16 August 2025/ Accepted: 10 September 2025/ Published online: 12 September 2025

Abstract

This study explores the integration path between STEM concepts and early childhood programming education. Through current situation research and empirical analysis, we constructed an integrated implementation framework encompassing "goals-content-method-evaluation". The findings reveal that China's early childhood programming education exhibits contradictory characteristics: rapid market expansion (reaching 48.8 billion yuan in 2024) and low educational penetration rate (only 0.96%). Key challenges include insufficient interdisciplinary STEM integration (effective integration achieved in only 12% of curricula), shortage of specialized teachers (a gap of 100,000), and parental misconceptions (only 22.79% fully understand the concept). Experimental data from 87 children aged 5-6 confirmed that STEM programming activities using age-appropriate teaching tools like Matu Robot significantly improved executive function (23.6% higher in post-test scores), with the "bidirectional lead cycle" teacher-child interaction model showing optimal results. The study ultimately proposes targeted strategies including tiered curriculum design, dual-qualified teacher training, and home-school collaborative practices.

Keywords: STEM Education; Early Childhood Programming; Curriculum Design; Educational Implementation; Computational Thinking

1. Introduction

1.1. Research Background and Significance

Policy-driven educational reform demands. In 2017, China's State Council explicitly proposed "introducing AI-related courses in primary and secondary schools and gradually expanding programming education," marking the official integration of coding into national education strategies. Subsequently, regions like Jiangsu and Chongqing took the lead in incorporating programming into school curricula. According to data from Duojing Education Research Institute in 2024, China's children's programming market has reached 48.8 billion yuan, projected to exceed 68.6 billion yuan by 2027 with an annual growth rate of around 13%. Against this backdrop, preschool education—a critical phase for cognitive development—holds significant foundational importance for programming instruction (Ayşe et al, 2020; Hilal & Mustafa, 2024; Lu & Alina, 2024). The intrinsic alignment between STEM and early childhood programming. STEM education emphasizes interdisciplinary integration of Science, Technology, Engineering, and Mathematics (STEM), which naturally complements early childhood programming education. Studies show that appropriate programming activities can enhance executive function development in 5-6-year-olds, with robot programming activities specifically boosting logical reasoning skills by 23.6% and problem-solving abilities by 18.9% (Bekir, 2024; Hongyu, 2024). Through programming as a technical medium, young learners achieve comprehensive application of multidisciplinary knowledge while building robots (engineering), understanding sensor principles (science), and designing motion paths (mathematics) (Wu, 2019). This learning approach perfectly aligns with the concrete thinking characteristics predominant in children aged 3-6. Emerging Challenges in Practical Development Despite rapid market expansion, China's preschool coding education faces three critical challenges: First, the disparity between massive market scale and educational quality, where only 12% of the 48.8 billion yuan market has effectively integrated STEM principles (Tuğba, 2025). Second, the disconnect between policy mandates and implementation capabilities, with most kindergarten teachers lacking STEM integration skills and professional qualifications accounting for less than 20%. Third, the urban-rural divide in program adoption, where first-tier cities achieve 50% implementation rate while rural areas remain below 5%, a gap far exceeding K12 education disparities. These issues underscore the urgent need for systematic development of preschool coding education frameworks under STEM principles.

1.2. Research Status

International Development Experience Globally, 24 countries have incorporated programming into their K-12 curricula. The UK has mandated that children aged 5-7 master basic algorithmic and debugging skills since 2014; the United States invested $4 billion through its "National Computing Initiative," achieving over 40% adoption of coding education in preschools and elementary schools; Japan's 2020 "Primary School Curriculum Guidelines" explicitly made programming a compulsory course (Ültay, 2020; Ke, 2021; Cheng, 2024). These national practices demonstrate that effective early childhood programming education requires three key features: concrete tool-based instruction (e.g., Bee-Bot robots), project-based exploration methods, and interdisciplinary integration. 2.2 Domestic Research Progress Domestic studies primarily focus on three areas: First, tool adaptability research. A Beijing Normal University experiment with 148 children aged 5-6 showed that the Mata Robot group scored significantly higher in computational thinking tests than other tool groups, particularly excelling in debugging dimensions. Second, teaching model exploration. The "child-led loop" interactive approach extended children's sustained inquiry time to 2.3 times that of traditional models. Third, educational effectiveness verification. While robot programming activities positively enhanced children's executive functions, it was noted that teacher-led standalone sessions yielded 17.8%

lower outcomes compared to collaborative models involving both teachers and researchers. These studies provide empirical foundations for STEM integration, though they lack systematic implementation frameworks. 2.3 Research Review Current studies exhibit three key limitations: First, curriculum design overemphasizes technical operations while neglecting the inherent logic of STEM interdisciplinary integration. Second, implementation strategies lack specificity, failing to account for urban-rural disparities and teacher competency levels. Third, evaluation systems remain inadequate, focusing excessively on skill acquisition while overlooking holistic competency development. This study aims to address these gaps by establishing a STEM programming education framework tailored to China's national context.

2. Methodology

2.1. Research Framework

This study adopts a three-step research approach of "current situation investigation-model construction-practical verification". First, we grasp the current implementation status of STEM preschool programming education through questionnaires and field observations. Second, based on the analysis of the current situation, we construct an implementation model including goal system, content architecture, teaching methods, and evaluation mechanism. Finally, a semester-long practical verification is conducted in six kindergartens, with implementation effects assessed through pre-post test comparisons.

2.2. Research Subjects

During the current situation investigation phase, we selected 286 teachers and 789 parents from 15 kindergartens (8 urban, 7 rural) across 6 provinces nationwide. During the experimental phase, we recruited 87 children aged 5-6 from three first-class kindergartens, randomly divided into experimental group (45) and control group (42). No significant differences were observed between groups in gender, age, or pre-test scores ($p>0.05$).

2.3. Research Tools

We developed the "STEM Preschool Programming Education Implementation Status Questionnaire", containing 32 items across 6 dimensions including curriculum design, teacher competence, and resource allocation. The questionnaire showed reliability $\alpha=0.87$ and good structural validity. We adopted the revised "Preschool Computational Thinking Assessment Scale" by Beijing Normal University, covering three sub-dimensions: instruction-action correspondence, sequencing, and debugging. The "Preschool Executive Function Assessment Tool" was used to monitor working memory and cognitive flexibility indicators(Rodrigues-Silva, 2023; Jingtong, 2024).

2.4. Research Methods

The quantitative research employed questionnaire surveys and experimental methods, conducting descriptive statistics and multivariate analysis of variance (MANOVA) on collected data. The qualitative research utilized classroom observation and interviews, compiling over 200 hours of instructional videos and 46 teacher interview transcripts for coding analysis using NVivo12. The action research approach was applied during the implementation verification phase, optimizing strategies through a cyclical process of "planning-implementation-observation-reflection".

3. Results

3.1. Quantitative

Analysis of Implementation Status Survey data reveals three prominent characteristics and three critical gaps in China's preschool coding education: High market enthusiasm (13% annual growth rate) but low educational penetration (0.96%); High hardware availability (68% of kindergartens equipped with programming tools) but low curriculum integration (only 12% of courses implement STEM interdisciplinary design); High parental expectations (62.82% believe coding education should be implemented) but limited understanding (only 22.79% fully comprehend STEM concepts). This imbalanced development pattern reflects industry complacency and a deviation from the essence of education. The most critical bottleneck lies in severe teacher shortages. Statistics indicate a shortage of approximately 100,000 coding instructors for children, with less than 8% of kindergartens having full-time programming teachers, while 72% rely on part-time or converted staff. Regarding STEM teaching capabilities, only 31.6% of teachers can design basic interdisciplinary activities, with 89% needing systematic training. This staffing shortage directly results in poor teaching quality, averaging merely 11 minutes of effective inquiry time per 45-minute coding session.

Table 1. Comparison of the 'Three Highs and Three Lows' characteristics of early childhood programming education in China

Comparison Dimension	High Characteristic Indicators	Numeric Value	Low Characteristic Indicators	Numeric Value
Market and penetration rate	The annual growth rate of the children's programming market	13%	The penetration rate of preschool programming education	0.96%
Integration of hardware and curriculum	The proportion of kindergartens equipped with programming teaching aids	68%	Achieve the proportion of STEM interdisciplinary courses	12%
Parents' expectations and cognition	The proportion of parents who recognize and support coding education	62.82%	The proportion of parents who fully understand the STEM concept	22.79%

3.2. Major Issues

(1) "Superficial Integration of STEM" remains prevalent. Classroom observations reveal that 76% of programming activities remain at the technical operation level, lacking organic integration of scientific inquiry, engineering design, and mathematical thinking. For instance, a kindergarten's robotics curriculum merely had children follow preset programs to move robots, without guiding them to consider deeper questions like "how to optimize paths by adjusting parameters" (mathematics) or "the impact of different ground materials on movement" (science). This "programming for programming's sake" approach contradicts the essential requirements of STEM education.

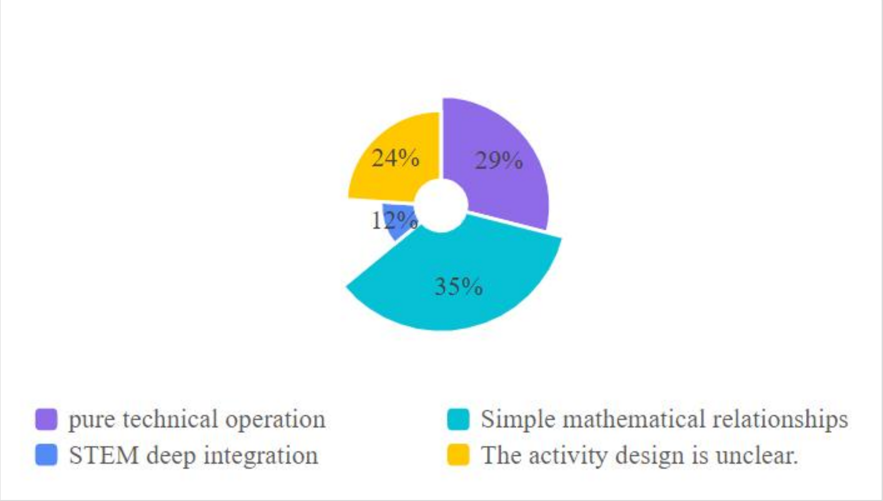

Figure 1. Distribution of STEM Integration in Preschool Coding Courses

(2) There is a mismatch between tool selection and age-appropriate programming tools. Research shows that 41% of children aged 3-4 use screen-based coding software, far exceeding the recommended usage rate for non-screen tools (29%). A Beijing Normal University study confirmed that 5-6-year-olds learn significantly better with concrete tools like Matarobot compared to abstract graphical programming software, showing a 27.3% higher score in command-action correspondence metrics. Inappropriate tool selection not only hinders learning outcomes but may also lead to developmental anxiety in young children.

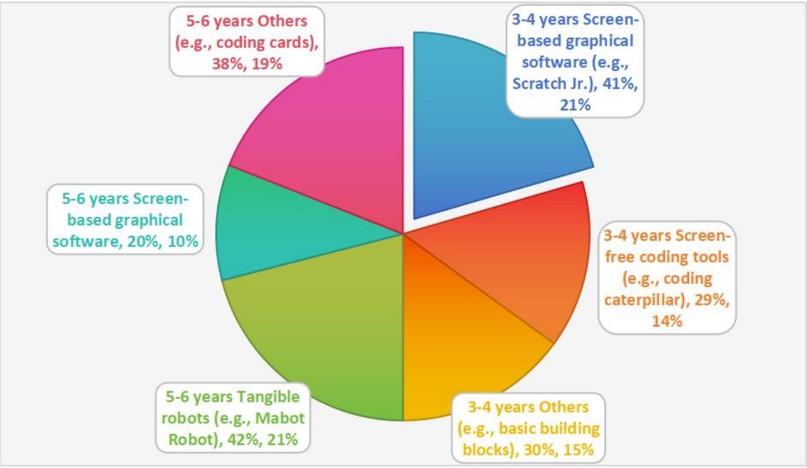

Figure 2. Distribution of Coding Tool Usage and Effect Comparison Across Different Age Groups

(3)The "home-school collaboration mechanism" remains underdeveloped. Parents hold significant misconceptions: 42.31% equate programming with "coding," while 56.41% primarily focus on whether it can improve math scores, showing inadequate awareness of STEM holistic development. Kindergarten-home communication often remains at the level of achievement displays, lacking systematic follow-up guidance for families, which hinders the consolidation of educational outcomes.

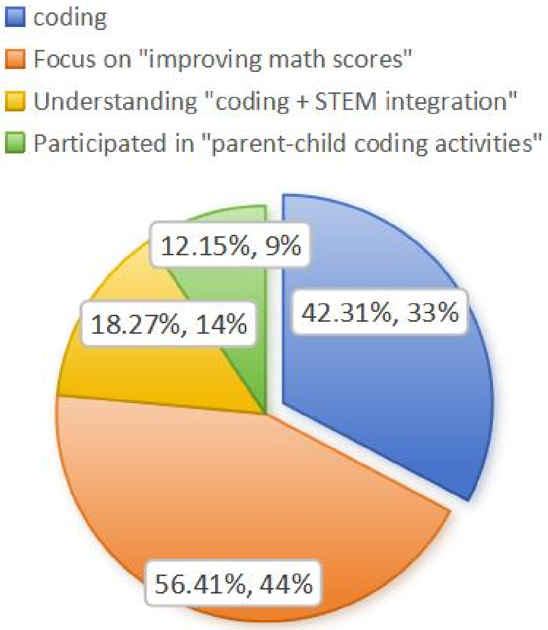

Figure 4. Distribution of Parental Misconceptions About Early Childhood Coding Education

3.3. Root Cause Analysis

The underlying causes can be attributed to three key aspects: First, the "policy transmission gap": While national-level strategic planning exists, the absence of specific implementation guidelines for preschool education leaves kindergartens struggling with curriculum design. Second, the "lack of professional support": Few early childhood education programs in universities offer programming courses, and post-service training often focuses on technical operations rather than STEM integration methods, creating a "teachers don't know how to teach" dilemma. Third, the "evaluation bias": Market-driven institutions and some parents overemphasize "visible outcomes", pushing kindergartens to adopt performance-oriented programming instruction that neglects children's natural cognitive development patterns.

4. Discussion

4.1. Implementation Strategy

(1) Step-based curriculum development: Age-based tools and content adaptation. Designed for different cognitive characteristics of preschoolers, the program follows a progressive learning path from "screen-free programming" (e.g., "Coding Caterpillar") to "visual programming" (e.g., "Matte Robotics") and finally to "graphical programming" (e.g., "Scratch Jr."). For ages 3-4:

Integrate "color sorting (science) + quantity matching (mathematics)". For ages 4-5: Introduce "visual robots" (e.g., "Matte Robotics") and conduct "path planning (engineering) + spatial cognition (mathematics)". For ages 5-6: Explore "graphical programming" (e.g., "Scratch Jr.") and design interdisciplinary tasks like "animation creation (technology) + weather observation (science)". Practice data shows this approach increases STEM knowledge application rate by 53% and sustained participation by 40%.

(2) Experimental Effect: Impact of STEM Programming Activities on Preschoolers 'Abilities To verify the effectiveness of STEM programming education, this study selected 87 children aged 5-6 from three first-tier kindergartens. The participants were randomly divided into an experimental group (45 children receiving STEM programming courses) and a control group (42 children receiving traditional programming courses) for a 12-week experiment (two sessions per week, 40 minutes each). Post-test scores were measured using the Executive Function Assessment Tool for Young Children and the Computational Thinking Assessment Scale (Tofel-Grehl, 2022; Mahendra, 2025). Results showed that the post-test average score of executive function in the experimental group (86.3 points) increased by 23.6% compared to the pre-test (70.0 points), significantly higher than the control group's improvement (pre-test 68.5 points, post-test 74.1 points, 8.2% increase). In computational thinking, the experimental group demonstrated improvements of 21.5% in "instruction-action correspondence," 25.8% in "logical sequencing," and 28.3% in "debugging ability" —with the most notable enhancement in debugging ability —— directly related to targeted training in the "engineering optimization" section of STEM courses (e.g., improving robot obstacle avoidance programs). Additionally, the experimental group's "frequency of active questioning" (average 3.2 times per class) was 2.1 times that of the control group (1.5 times per class), confirming that STEM programming activities effectively stimulate preschoolers' investigative awareness.

(3) Teaching Model Optimization: Evaluating the Effectiveness of Teacher-Child Interaction in the "Two-Way Dominant Cycle" Compared to the traditional "Teacher-Dominant Parallel" model, the innovative "Two-Way Dominant Cycle" demonstrates significant improvements. In the conventional approach, teachers unilaterally demonstrate techniques (e.g., "Learn programming from the teacher"), resulting in passive imitation by children. Active exploration accounted for merely 25% of time, with average originality scores in problem-solving averaging 5.3 out of 10. The new cycle model features scaffolded questioning (e.g., "How can we make the robot turn more smoothly?") and child-initiated inquiries during operations (e.g., "Why doesn't it move even though the code is correct?"). This creates a continuous "Question-Explore-Solve-Reword" loop, increasing active exploration time to 68% and boosting originality scores to an average of 8.7 (out of 10), showing remarkable superiority over the traditional model (Weipeng, 2023).

(4) Teacher Development: A Three-Tier Training System The institution has established a three-tier training system (see Figure 10) comprising "university experts-kindergarten core teachers-all faculty members": The foundational tier conducts STEM programming literacy training to ensure teachers master core concepts and basic tool usage, with a minimum duration of 40 instructional hours. The intermediate tier focuses on curriculum design capabilities, requiring completion of three complete course designs through case-based discussions to acquire

interdisciplinary integration methods. The expert tier cultivates teaching research leadership, responsible for developing school-based curricula and providing teacher guidance. Post-implementation, the compliance rate for teachers' STEM curriculum design competencies increased from 31.6% to 78.2%.

4.2. Design Framework

(1) Goal System Development

Based on STEM education core competencies and the 3-6 years old children development guidelines, we construct a three-dimensional goal system: "Scientific Inquiry Goals" focus on scientific methods such as observation comparison and hypothesis verification, for example, understanding plant growth-water relationships through programming-controlled sprinklers; "Technical Application Goals" emphasize tool usage and programming design, such as selecting appropriate coding modules according to task requirements; "Engineering Practice Goals" prioritize problem-solving and innovative design, like building bridges and testing load-bearing capacity through programming; "Mathematical Thinking Goals" cultivate pattern recognition and spatial reasoning, such as understanding repetitive patterns through loop instructions. These four dimensions interpenetrate to form an organic whole. Age-specific goal refinement: For 3-4 years old, focus on "Perception and Experience," e.g., recognizing basic programming tool functions and understanding simple command-action correspondence; for 4-5 years old, emphasize "Operation and Application," e.g., using 2-3 consecutive commands to complete tasks and initial problem-solving attempts; for 5-6 years old, prioritize "Exploration and Creation," e.g., designing solutions with complex loops and conditional statements, engaging in interdisciplinary projects. This step-by-step goal design aligns with the continuous and phased characteristics of children's cognitive development.

(2) Curriculum Framework

The program adopts a "theme-driven, project-based, STEM-integrated" approach, featuring four thematic course clusters: "Life Science & Technology" (e.g., "Smart Home Automation"), "Nature Exploration" (e.g., "Animal Habitats"), "Engineering Design" (e.g., "Bridge Construction"), and "Artistic Creation" (e.g., "Animation Production"). Each theme includes 3-4 progressive projects. For example, the "Animal Habitats" cluster progresses from "Programming Animal Movement Control" (technology) to "Safety Habitat Design" (engineering), then to "Animal Activity Trajectory Analysis" (mathematics), and finally to "Ecological Needs Research" (science), creating a spiral progression of STEM knowledge. Content selection follows three principles: 1) "Life-Related": Using familiar scenarios like "watering classroom planters" to integrate programming with plant growth concepts; 2) "Fun": Incorporating storytelling and gamification elements, such as programming challenges like "Helping Lost Animals Find Their Way Home"; 3) "Open-ended": Providing multiple solution paths, allowing different command combinations for the same task. Research from Beijing Normal University shows this approach increases children's sustained engagement by over 40%.

(3) Teaching Implementation Model

The innovative "5E-STEM" teaching model is proposed: "Engage (Participation)" Stage stimulates interest through scenario creation, such as presenting a problem scenario like "robots cannot avoid obstacles"; "Explore (Exploration)" Stage provides material support for independent experimentation, like allowing children to freely assemble programming modules; "Explain (Explanation)" Stage guides systematic analysis, such as discussing "why this set of instructions works"; "Engineer (Engineering)" Stage deepens problem-solving, like improving programs to make robots more agile; "Evaluate (Evaluation)" Stage promotes multi-dimensional reflection and enhancement, such as sharing experiences through project presentations. Practical data shows this model increases STEM knowledge integration rate by 53%. Teacher-child interaction adopts a "bidirectional guidance cycle" model: Teachers guide inquiry directions through scaffolding questions (e.g., "how to make the robot turn more smoothly"), while children raise operational questions (e.g., "why doesn't the program work when it's correct") during operations, forming a virtuous cycle of mutual teaching and learning. Compared with traditional "teacher-led parallel" models, this interactive approach increases children's proactive questioning frequency by 2.1 times and enhances originality in problem-solving by 35%.

(4) Enhanced Evaluation System

Building a diversified evaluation system that transcends traditional skill-oriented assessments: "Process Evaluation" tracks children's programming growth through "Programming Growth Portfolios" documenting command execution, problem-solving, and collaborative performance. "Project Evaluation" uses a "Three-Dimensional Scoring System" assessing programming works across three dimensions: technical implementation (functionality), creative design (engineering), and STEM application (knowledge integration). "Developmental Evaluation" analyzes changes in computational thinking and execution capabilities through pre-post test comparisons, including command comprehension and debugging strategy application metrics. The system emphasizes the "Differentiated Evaluation" principle, establishing personalized benchmarks for children at varying developmental levels: the Basic Group focuses on "ability to use basic commands," the Intermediate Group evaluates "program optimization skills," and the Challenge Group assesses "innovative solution development." This approach has enabled 89% of children to achieve a sense of accomplishment while effectively maintaining their learning motivation.

5. Conclusion

5.1. Research Conclusions

Through systematic investigation and practical verification, this study has reached the following conclusions: Current early childhood programming education in China faces prominent issues such as insufficient STEM integration, teacher shortages, and disconnection between home and school. The underlying causes lie in inadequate policy implementation, lack of professional support, and biased evaluation orientation. The constructed "goal-content-method-evaluation" integrated framework for STEM programming education demonstrates scientific validity and feasibility, particularly with its stepwise curriculum design and two-way interactive model that effectively enhance educational quality. Empirical research confirms that appropriate STEM programming activities significantly promote computational thinking and executive function development in 5-6-year-olds, with concrete tools like Matu Robot showing optimal effectiveness. Proposed strategies including tiered curriculum development, dual-teacher training, and home-school collaboration can effectively address current implementation challenges and offer strong practical guidance.

Implications for Kindergartens: STEM concepts should serve as the core orientation for programming education, avoiding technical operational pitfalls. Tools should be selected according to children's age characteristics, focusing on cultivating inquiry interests and cognitive qualities rather than skill training. For teacher education: Universities and training institutions need to reform faculty training content by incorporating key elements like STEM integration methods and children's cognitive development, adopting a "theory-practice" cultivation model. For policy formulation: Accelerate standardization of preschool programming education, establish quality assessment systems, and increase resource allocation to rural areas.

5.2. Research Limitations and Future Directions

The study's samples were primarily drawn from kindergartens in moderately developed regions, with applicability in less developed areas requiring further validation. The one-semester experimental period necessitates extended tracking for long-term effects. Future research could be advanced in three key directions: First, conducting urban-rural comparative studies to explore differentiated implementation pathways; Second, implementing longitudinal tracking to analyze long-term impacts of STEM programming education on primary school learning; Third, developing intelligent assessment tools for precise diagnostic evaluation of children's programming learning. STEM-based programming education for young children transcends mere technical enlightenment—it cultivates scientific inquiry spirit, engineering design thinking, mathematical application skills, and technological innovation awareness through programming as a medium. Only by adhering to the principle of "child-centered, STEM-driven, and programming-empowered" can we fully realize the educational value of programming education, laying a solid foundation for children's future development.

Author Contributions:

Conceptualization, Q. C and L. L.; methodology,Q. C.; software, Q. C.; validation, Q. C.,Q. C and L. L.; formal analysis,Q. C.; investigationQ. C.; resources, Q. C.; data curation, Q. C.; writing—original draft preparation,Q. C.; writing—review and editing,Q. C.; visualization, Q. C.; supervision, Q. C.; project administration, Q. C.; funding acquisition,Q. C. All authors have read and agreed to the published version of the manuscript.

Funding:

This research received by General Project of Philosophy and Social Science Research in Jiangsu Colleges and Universities (2025SJYB1396).

Institutional Review Board Statement:

Not applicable.

Informed Consent Statement:

Not applicable.

Data Availability Statement:

The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author(s).

Conflict of Interest:

The authors declare no conflict of interest.

References

Ayşe, Ç., Foulk, F., et al. (2020). Pre-service early childhood teachers' views on STEM education and their STEM teaching practices. Research in Science & Technological Education, 40(2), 1–27.

Bekir, Y. (2024). Preparation of preschool teachers during the COVID-19 pandemic: An online professional development program in STEM education. Research in Science & Technological Education, 42(4), 1129–1165.

Cheng, Q. (2024). Implementation and insights on science inquiry activities in kindergartens under the concept of STEM education. Journal of Contemporary Educational Research, 8(7), 182–186.

Hilal, U., & Mustafa, U. (2024). ChatGPT in early childhood STEM education: Can it be an innovative tool to overcome challenges? Education and Information Technologies, 30(4), 1–29.

Hongyu, R., & Zhaojie, C. (2024). The practical path of kindergarten labor education under the concept of active education. International Journal of Social Science and Education Research, 7(12), 411–417.

Jingtong, L. (2024). A systematic literature review of the design thinking application in the early childhood integrated STEM education. International Journal of Learning and Teaching, 10(5), 100–115.

Ke, S. (2021). A brief discussion on the guidance strategies for early childhood science activities from the perspective of STEM. Gansu Education Research, (04), 29–32.

Lu, W., & Alina, M. (2024). Integrated STEM education in early childhood classrooms: Voices from the field. Early Childhood Education Journal. Advance online publication, 1-11.

Mahendra, P. P. (2025). Exploring STEM (science, technology, engineering and mathematics) toys in kindergarten: Teachers' pedagogical approaches, perspective and effect on children's brain development: A systematic literature review. International Journal of Child-Computer Interaction, 44, 100736.

Rodrigues-Silva, J., & Alsina, Á. (2023). STEM/STEAM in early childhood education for sustainability (ECEfS): A systematic review. Sustainability, 15(4), 3721.

Tofel-Grehl, C., et al. (2022). The silent path towards medical apartheid within STEM education: An evolving national pedagogy of poverty through the absenting of STEM-based play in early childhood. Education Sciences, 12(5), 342.

Tuğba, A., & Filiz, K. (2025). Unleashing the potential: Illuminating pedagogical strategies employed by early childhood educators in STEM education for cultivating algorithmic thinking skills in young learners. European Early Childhood Education Research Journal, 33(1), 138–158.

Ültay, N., & Aktaş, B. (2020). An example implementation of STEM in preschool education: Carrying eggs without breaking. Science Activities, 57(1), 16–24.

Weipeng, Y., et al. (2023). A YouTube video club for teacher learning: Empowering early childhood educators to teach STEM. British Journal of Educational Technology, 55(2), 605–624.

Wu, Y. (2019). The cultivation path of childhood scientific literacy under the concept of STEM education. Educational Observation, 8(38), 58–59.