# The Facilitative Effect of Early Programming Activities on Executive Function in Young Children

**Authors:** Xue Jing, Shou Yanna, Xue Jing

**Date:** 2024-01-11T00:00:00+00:00

## Abstract

In recent years, programming education has become increasingly prevalent in the field of early childhood education; however, few studies have investigated the effects of early programming activities on child development. Therefore, this study examines the role of early programming activities in the development of executive function in young children—a critical capacity for early academic achievement and school readiness. The experimental and control groups each consisted of 16 preschoolers, with equal gender distribution, all aged 5–6 years. Children in the experimental group participated in an 8-week early programming curriculum based on programmable toys (Bee-Bot) (one session per week, 30–40 minutes per session), and their mastery was assessed to evaluate the feasibility of early programming activities. Pre-test and post-test measures of executive function (working memory, regulation ability, inhibitory ability) were administered to children in both groups using cognitive tasks and teacher ratings. Results indicated that children in the experimental group were able to grasp most programming concepts and successfully complete the majority of programming activities; furthermore, after controlling for pre-test baseline levels, the experimental group significantly outperformed the control group on executive function cognitive tasks at post-test. Additionally, the experimental group demonstrated significantly higher performance on working memory-related behaviors compared to the control group, though no significant differences were observed in behaviors related to regulation ability and inhibitory ability. These findings demonstrate the feasibility of implementing early programming activities (based on programmable toys) in early childhood settings, while also indicating that such activities can enhance children's executive function.

## Full Text

# The Promotional Effect of Early Programming Activities on Preschoolers' Executive Function

**Xue Jing[1]*, Shou Yan-Na[2]**
([1] School of Psychological and Cognitive Sciences, East China Normal University, Shanghai 200062)
([2] College of Education, Shanghai Normal University, Shanghai 200234)

### Abstract

In recent years, programming education has increasingly emerged in early childhood education, yet few studies have examined its developmental effects on young children. This study investigated the impact of early programming activities on the development of executive function—a critical capacity for early academic skills and school readiness. The experimental and control groups each comprised 16 children (8 boys and 8 girls) aged 5-6 years. Children in the experimental group participated in an eight-week early programming curriculum using programmable toys (Bee-Bot), with sessions held once weekly for 30-40 minutes each, and their mastery was assessed to evaluate feasibility. Both groups completed pre- and post-tests of executive function (working memory, cognitive flexibility, inhibitory control) using cognitive tasks and teacher ratings. Results indicated that children in the experimental group could grasp most programming concepts and successfully complete the majority of activities. Moreover, after controlling for baseline pre-test performance, the experimental group significantly outperformed the control group on executive function cognitive tasks at post-test, and showed significantly fewer working memory-related behavioral problems. However, no significant group differences emerged for cognitive flexibility or inhibitory control behaviors. These findings demonstrate the feasibility of implementing early programming activities (based on programmable toys) in preschool settings and suggest that such activities can enhance children's executive function.

**Keywords:** early programming, executive function, preschool children

---

Twenty years ago, "learning computers" was a hot topic in family education. In 1997, *Preschool Education Research* reprinted an article from *Wenhui Daily* titled "Talking to Parents About Children Learning Computers," which recommended using computers as teaching tools to support children's learning in areas such as general knowledge, art, and music education—something that has become commonplace today. With advances in computer technology, children are now exposed to various electronic devices at much younger ages. New devices such as smartphones and tablets with touch screens have transformed how individuals interact with images, sounds, and videos, enabling even two-year-olds to easily manipulate touchscreens for independent exploration. A 2011 survey of mothers of children aged 2-5 found that most young children could operate smartphones and use a computer mouse to play games before they could tie their shoelaces [1]. Consequently, parents and educators worry about the potential negative effects of early and excessive exposure to electronic media. However, since the digital age makes it unavoidable for children to encounter electronic devices, scholars have begun urging parents and teachers to guide children away from consumptive activities like gaming toward creative activities mediated by electronic devices [2-4].

Two decades later, in July 2017, China's State Council released the *New Generation Artificial Intelligence Development Plan*, which particularly emphasized implementing AI education and recommended "setting up AI-related courses in primary and secondary schools, and gradually promoting programming education." Similarly, developed countries such as the United Kingdom and the United States have recently included early programming education in their basic education curricula and frameworks [5-6]. Early programming education activities resemble traditional construction activities in that both belong to the category of creative activities, differing only in their digital platform. Currently, most domestic and international research on programming learning focuses on primary and secondary school stages, with few studies examining programming learning in early childhood, particularly the developmental benefits of participating in early programming activities. Therefore, this study developed an early programming activity curriculum, implemented it using programmable toys as the medium, and employed an experimental design to examine the promotional effect of early programming activities on a crucial set of cognitive abilities in young children—executive function.

The English terms for programming are *Coding* or *Programming*, each emphasizing different aspects of the process [4]. First, programming is a construction based on code, which can be understood as abstract building blocks representing objects or actions. Similar to human language, code must follow certain syntax and rules that form a programming language system, such as the common Java language. Second, programming serves problem-solving by inputting code into a computer to control machines and achieve goals. Programs created through programming are common in daily life; for example, when a washing machine cleans clothes, it executes pre-inputted instructions from the manufacturer to complete a series of fixed procedures such as soaking, washing, and rinsing. Therefore, programming activities can essentially be viewed as construction occurring at an abstract level [7]. Unlike traditional construction activities where building occurs on concrete objects (such as blocks), children in programming activities manipulate abstract codes (such as computer instructions), which some have analogized as "electronic building blocks" [8]. In this sense, young children can engage in programming activities under certain conditions.

In the 1980s, the renowned cognitive and educational psychologist Seymour Papert applied Piaget's constructivist theory to computer learning and created a programming language for children called "LOGO" [9], whose design principles remain in use today. However, due to its text-input characteristics, this programming language was difficult for young preschool children to use. In the past decade, with the popularization of touchscreens, graphical programming languages for young children have emerged in endless streams, such as the ScratchJr application developed by MIT that runs on Android and iOS systems [10]. Some developers have even created programmable toys that depart from traditional computers and displays, such as the Bee-Bot developed by the American company Bee-Bot and the Code & Go Robot Mouse developed by Learning Resources. The characteristic of these programming tools is that they graphically and modularize code representing objects or actions, and operationally, early programming activities based on graphical programming software and programmable toys can be completely regarded as games similar to block construction.

The programming concepts and ideas underlying these activities can be glimpsed through programming activities based on programmable toys. The currently popular Bee-Bot in foreign kindergartens is shown in [Figure 1: see original paper]. Children input action instructions to control the bee by using seven buttons on its back representing different actions and functions (move forward [one step], move backward [one step], turn left [90 degrees], turn right [90 degrees], clear, execute, pause). Unlike traditional remote-controlled toys where children directly manipulate the toy for immediate movement via a remote, in problem-solving situations with programming toys, children must observe the specific path from the starting point to the destination, think ahead in code form, and input the instructions into the bee robot for execution (see a scenario in Figure 1, where circles represent destinations and the bottom of the grid shows the code that should be input in sequence). When encountering a complex problem, such as using a Bee-Bot with a pencil attachment to draw a square through movement, children must think through the route to draw the square and program the action instructions in code form: forward, turn right, forward, turn right, forward, turn right, forward, turn right. Similarly, children can discover the looping pattern in this code—repeating forward-turn right four times—thereby enhancing their understanding of squares.

Early childhood programming education can use programmable toys and their activities to teach developmentally appropriate early programming concepts. For example, the UK's basic education curriculum for information science requires 5-7-year-olds to understand concepts such as sequences and algorithms, decomposition and composition of processes, repeating patterns and loops, and debugging [5]. Recent research shows that with the help of programmable toys, children as young as four can successfully program simple robots [10-13]. Slightly older children aged 5-6 can also use complex action instructions in advanced programming platforms (such as ScratchJr) to control virtual characters and create interactive story scripts similar to animated film plots [10]. Through early programming activities, children can also master knowledge and phenomena related to early STEM (science, technology, engineering, and mathematics) education, such as cause-and-effect relationships, counting, and estimation [13,14].

[Figure 1: see original paper] Bee-Bot and a problem-solving scenario

Although early STEM education researchers believe that participating in programming education activities can help children learn digital-age knowledge, communicate with computer systems, or prepare for future programming-related work and cultivate interest in relevant fields [3], advocates of early programming education argue that more importantly, the ideas and concepts in programming activities can provide children with powerful thinking tools [9]. When children control computers/machines to achieve specific goals through programming, they must understand their own problem-solving thinking patterns, thus promoting metacognition as they think about their thinking while giving instructions to the computer. Therefore, in programming activities, children not only "learn by doing" but more importantly achieve better learning outcomes by reflecting on what they do. Moreover, if children can think about and solve problems through computer language, they can also use this "powerful idea" to solve practical problems [9]. Currently, most research on programming education at home and abroad focuses on primary school and above, with few studies examining the implementation and effects of programming education in preschool [16]. Studies from primary school stages show that programming education promotes children's cognitive and academic development, particularly in STEM-related abilities [17-20]. Recent research on early childhood programming education has begun to focus on its promotional effects on child development. Ages 5-7 represent a critical transition period in cognitive development, during which children gradually develop more abstract and logical thinking patterns, and programming experiences can help establish connections between concrete and symbolic representations, stimulating their thinking processes to support this transition [21]. Additionally, some recent studies have found that early programming learning can improve children's problem-solving abilities [22,23], attention, and self-control [24], thereby supporting early development and learning.

In programming activities, since children need to pre-plan problem-solving strategies and clearly list the steps of solutions, they likely employ a set of higher-order cognitive abilities—executive function. Executive function comprises a set of higher-order cognitive abilities involved in goal-directed activities that play an important role in children's early academic abilities and school readiness [25,26]. The currently widely accepted model includes three basic components: working memory, inhibitory control, and attention shifting [27]. At the psychological and behavioral level, working memory refers to the ability to temporarily store and manipulate information, inhibitory control refers to suppressing inappropriate impulsive behaviors and responses, and attention shifting refers to the ability to switch between different mental sets. More advanced executive functions also include planning—the ability to pre-plan problem solutions [28]. Analyzing from the perspective of required executive functions, programming activities are goal-directed activities where children need planning abilities to think through specific methods and steps for problem-solving. During plan execution, children must maintain goals, rules, and steps in working memory while suppressing dominant impulsive responses and shifting between different rules and perspectives. Therefore, early programming activities may promote the development of children's executive function, yet no studies have used experimental methods to examine this research question.

Thus, the purpose of this study is to develop an early programming activity curriculum and implement it with senior kindergarten children, using an experimental design to examine both the feasibility of early programming education and its effect on children's executive function development.

## Method

### Participants

Participants were 32 senior kindergarten children (mean age = 54.78 months, SD = 3.30 months) from a kindergarten in Shanghai, all without physical, psychological, or developmental disabilities. They were divided into experimental and control groups using age- and gender-matching, with 16 children in each group (8 boys and 8 girls).

### Materials

**1) Children's Executive Function Performance** The Head-Toes-Knees-Shoulders task was used to assess executive function. In this task, children played a game touching their head, feet, shoulders, or knees [29]. Initially, children followed the experimenter's instructions (e.g., "touch your head/feet/shoulders/knees"), then were required to do the opposite: when asked to touch their head, they touched their feet; when asked to touch their shoulders, they touched their knees (and vice versa). The experimenter gave 20 instructions (5 each for head, feet, shoulders, knees). Each correct response scored 2 points, incorrect responses scored 0, and self-corrections after errors scored 1 point. The maximum score was 40. This task requires children to remember four rules (working memory), inhibit direct response impulses (inhibitory control), and shift between different rules (attention shifting), making it a comprehensive measure of executive function.

**2) Children's Executive Function Behaviors** The Childhood Executive Functioning Inventory, developed by Thorell and Nyberg [30] and previously translated into Chinese, was used. Teachers rated each child's executive function-related behaviors in kindergarten life over the past week. The scale contains 24 items using a 5-point rating (1 = completely not true to 5 = completely true). The original scale comprised four subscales: working memory, planning ability, regulation ability, and inhibitory ability. Since all items were reverse-scored, higher dimension scores indicated more behavioral problems in the corresponding executive function component. The scale has demonstrated good reliability and validity in previous research [31].

### Procedure

**1) Pre-test** Trained undergraduates administered the Head-Toes-Knees-Shoulders task individually to children in both groups, while classroom teachers completed the executive function inventory for each child.

**2) Implementation of Programming Activities** Children in the experimental group participated in a six-week programming curriculum developed with reference to international frameworks for early childhood programming education, using the programmable toy Bee-Bot as the medium. The curriculum was designed and revised through discussion between the authors and kindergarten teachers. The framework and content are shown in . Activities were conducted in groups of eight (requiring two sessions for the experimental group), led by one of the authors and an assistant.

Framework and Content of Early Programming Activities

**I. Exploring Programmability** - 1.1 Exploring programmable toys: Children discussed the functions of buttons on everyday appliances and transferred this knowledge to programmable toys (power switches, buttons) - 1.2 Input and output: "I am a robot" activity—using instruction symbols to control a teacher acting as a "robot" to understand input and output concepts - 1.3 Using programming modules: Through trial and guidance, children explored the functions of each button (as shown in 1-1 and 1-2 in Figure 1)

**II. Programming Concepts** - 2.1 Sequence and algorithm: Explaining the meaning of sequence and algorithm through activities like "having a meal" (algorithm refers to specific steps to solve problems) - 2.2 Decomposition and composition: In the "Bee Visits Friends" activity, due to complex paths (including impassable obstacles X), children were guided to decompose the path into several parts according to turns (as shown in 1-3 and 1-4 in Figure 1) - 2.3 Debugging activity: "Bee Gets Lost"—pre-inputting a set of incorrect instructions into the bee and requiring children to use instruction cards to identify errors and modify them so the bee could successfully reach the destination

**III. Complex Programming** - 3.1 Perspective shifting: "Bee Goes Home"—the bee's perspective differs from the child's perspective (as shown in 1-5 in Figure 1). Children must input instructions from the bee's perspective, requiring multiple perspective shifts as the bee makes several turns - 3.2 Forbidden button: "Bee That Only Moves Backward"—the forward button on the bee is sealed with a colored label (as shown in 1-6 in Figure 1), and children can only use backward and turn buttons to control the bee to reach the destination

The programming activities consisted of eight sessions, with sample programming problems shown in [Figure 2: see original paper]. To assess children's understanding of programming concepts and the effectiveness of activity participation, each child's performance in five areas—initial exploration, sequence and algorithm, composition and decomposition, debugging, and complex programming—was evaluated using a 3-point scale (2 = proficient mastery, 1 = basic mastery, 0 = cannot master).

**3) Post-test** Following the programming activities, both groups were re-administered the Head-Toes-Knees-Shoulders task, and classroom teachers again completed the executive function inventory for each child.

[Figure 2: see original paper] Sample problems from programming activities (all from child's perspective; circles represent destinations, X represents impassable squares; Part I difficulty: 1-1 and 1-2; Part II difficulty: 1-3 and 1-4; Part III difficulty: 1-5 and 1-6)

## Results

### Children's Performance in Programming Activities

Examination of experimental group children's performance revealed that all children could proficiently use buttons for control. Regarding programming concepts covered in the curriculum (sequence and algorithm, composition and decomposition), most children could understand and master them, with the remaining children achieving basic mastery. However, only 25% of children could successfully complete debugging activities, 69% could basically complete them, and 6% could not master them. For complex programming activities, 12% and 44% of children could proficiently or basically complete them, respectively, while 44% could not complete them. Overall, children in the experimental group mastered basic activities well but performed relatively poorly on complex activities with higher cognitive demands.

Experimental Group Children's Learning of Programming Concepts

| Activity | Proficient Mastery | Basic Mastery | Cannot Master |
|---|---|---|---|
| Initial Exploration | 100% (16) | 0% (0) | 0% (0) |
| Sequence and Algorithm | 81% (13) | 19% (3) | 0% (0) |
| Composition and Decomposition | 62% (10) | 38% (6) | 0% (0) |
| Debugging | 25% (4) | 69% (11) | 6% (1) |
| Complex Programming | 12% (2) | 44% (7) | 44% (7) |

Descriptive statistics for pre-test and post-test variables for both groups are shown in . Preliminary paired t-tests indicated that both experimental and control groups showed significant improvement from pre-test to post-test (experimental group t-values ranged from 3.96-7.36; control group t-values ranged from 4.95-7.08; all $p < 0.001$), suggesting that developmental maturation over time may account for the experimental group's progress. To examine whether the experimental group's improvement was significantly greater than the control group's, one-way analysis of covariance (ANCOVA) was conducted, controlling for pre-test performance to examine group effects on each dependent variable.

Results showed that after controlling for pre-test performance on the Head-Toes-Knees-Shoulders task, there was a significant group effect on post-test performance, $F(1, 29) = 6.33$, $p < 0.05$, $^2 = 0.18$. Similarly, after controlling for pre-test working memory, there was a significant group effect on post-test working memory behavioral problems, $F(1, 29) = 11.49$, $p < 0.05$, $^2 = 0.28$. However, after controlling for pre-test regulation ability, no significant group effect emerged for post-test regulation problems, $F(1, 29) = 0.53$, $p > 0.05$, $^2 = 0.02$. After controlling for pre-test inhibitory ability, no significant group effect emerged for post-test inhibitory problems, $F(1, 29) = 0.45$, $p > 0.05$, $^2 = 0.02$.

Descriptive Statistics for Experimental and Control Groups (Mean ± SD)

| Measure | Experimental Group | Control Group |
|---|---|---|
| | Pre-test | Post-test |
| Head-Toes-Knees-Shoulders Task | 33.19 ± 2.93 | 36.19 ± 2.48 |
| Working Memory Behaviors | 2.18 ± 0.67 | 1.71 ± 0.55 |
| Regulation Behaviors | 2.24 ± 0.43 | 1.85 ± 0.41 |
| Inhibitory Behaviors | 3.24 ± 0.97 | 2.70 ± 0.99 |

## Discussion

### Feasibility of Early Programming Education for Young Children

This study designed an early programming education activity curriculum based on international frameworks appropriate for young children and empirically examined its impact on executive function development. Consistent with previous research [10-12], this study demonstrates the feasibility of early programming education in preschool settings. First, since most senior kindergarten children already have sufficient experience operating electronic toys and constructing with objects, they could well understand and remember that button symbols represent specific actions and grasp input and output concepts. Children could also transform a series of concrete actions (moving forward, backward, turning) into abstract button symbols (or with the aid of paper, pen, or cards), input them into the programmable toy, and finally execute and output the concrete actions. Second, through analogical teaching methods (e.g., comparing to the process of eating at a fast-food restaurant: ordering, paying, receiving food), children could better understand preliminary programming concepts like algorithms (problem-solving procedures), sequence, decomposition, and composition. These programming concepts share similar features with everyday problem-solving, making them relatively easy for children to understand. However, children showed difficulty with debugging activities—that is, reflecting on and identifying problematic steps from incorrect solutions. This is consistent with cognitive characteristics of preschool children shown in previous research, which found that young children have difficulty improving upon currently inappropriate solutions and tend to envision new solutions or start over [32]. Finally, children performed slightly worse on activities with high executive function demands. In these activities, children often confused their own perspective with the bee's perspective when thinking of solutions, sometimes controlling the bee's movement from their own perspective rather than the bee's.

Overall, in basic programming activities, children easily understood and mastered the meaning and functions of each button. In simple problem-solving situations, they could mentally plan the route from start to destination, transform it into button symbols for input, and finally check the output effect (whether the destination was reached). However, they showed operational difficulties in more challenging activities with higher executive function demands. Since this study only examined initial learning, children might perform better with longer-term practice.

### Effects of Programming Activities on Children's Executive Function

By comparing post-test results between experimental and control groups (controlling for pre-test), this study found that participating in programming activities significantly improved children's performance on executive function cognitive tests and significantly reduced the frequency of working memory-related behavioral problems, but showed no effects on inhibitory or regulation abilities. Similar to studies in primary schools and international exploratory research, this study found that programming activities can improve children's comprehensive cognitive abilities, though effects on specific cognitive abilities differ [16]. As analyzed previously, programming activities require children to pre-plan specific problem-solving methods, demanding working memory and planning abilities to mentally simulate and evaluate solution steps. During input, children must maintain inputted instructions and code in working memory, constantly keeping track of what has and has not been input. Therefore, programming activities place substantial demands on planning and working memory abilities, and participation can train these two cognitive components (in the rating scale, the planning ability subscale correlated too highly with the working memory subscale and was combined into the working memory dimension). The Head-Toes-Knees-Shoulders task, as a comprehensive executive function measure, has been shown to correlate highly with working memory [33], as children must remember rules to complete it successfully. Thus, participating in programming activities improved children's performance on this task and significantly reduced working memory-related behavioral problems.

However, this study found that programming activities did not improve children's regulation or inhibitory abilities, despite these being required in complex programming activities. A possible reason is that children's performance on complex programming activities was significantly worse than on the first two parts. For instance, due to higher cognitive demands, nearly half of the children could not successfully complete advanced programming tasks, which may have affected the potential for improving inhibitory and regulation abilities. As a preliminary exploratory study, the programming curriculum was not specifically designed to enhance children's inhibitory and shifting abilities; future research could integrate different components of executive function to train specific aspects.

## Conclusion and Educational Recommendations

### Conclusion

This study demonstrates the feasibility of implementing early programming activities based on programmable toys and indicates that participation in early programming activities can enhance children's executive function.

### Educational Recommendations

Given the widespread availability of electronic media, kindergartens are encouraged to attempt implementing early programming activities. Programming education can be conducted through multiple approaches: First, programmable toys can be used. The tangible nature of programmable toys not only increases children's interest in programming activities but also helps them establish connections between abstract code and concrete actions, while avoiding increased "screen time" since they don't require electronic screens. Second, tablet software similar to programmable toys can be employed. Such software (e.g., ScratchJr) offers richer interfaces and interactive functions compared to programmable toys, allowing children to learn more advanced programming concepts (such as pattern recognition and loops, conditions and consequences) and engage in creative activities like drawing and story creation while writing scripts. Finally, early programming activities can be integrated with early STEM education and drama education. For example, programming activities may involve counting, estimation, and visuospatial abilities, allowing integration with certain components of early mathematics education.

## References


[1] AVG Technologies. digital skills study. Retrieved www.beijing-kids.com/files/avg-digital-skills-study-full-briefing.pdf

[2] Cui, Y. Y. (2017). Early childhood programming: To learn or not to learn? *People's Daily Overseas Edition*, September 11, 2017, p. 008

[3] Geist, E. (2014). Using tablet computers with toddlers and young preschoolers. *Young Children*, 69,

[4] Geist, E. (2016). Robots, programming and coding, Oh my! *Childhood Education*, 92, 298-304.

[5] Government of the United Kingdom. (2013). *National curriculum in England: Computing programmes of study.* Retrieved from https://www.gov.uk/government/uploads/system/uploads/attachment_{data}/file/239033/PRIMARY_{{{n
{Computing}.pdf

[6] K–12 Computer Science Framework. (2016). Retrieved from http://www.k12cs.org.

[7] Soloway, E. (1986). Learning to program = learning to construct mechanisms and explanations. *Communications of the ACM*, 29(9), 850-858.
[8] Liu, X. (2018). Tracing the origin: Exploring the essence of programming education. *China Science and Technology Education*, (1), 10-13.
[9] Papert, S. (1993). *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books: New York, NY, USA. ISBN 0465046746.
[10] Portelance, D. J., Strawhacker, A. L., & Bers, M. U. (2016). Constructing the ScratchJr programming language in the early childhood classroom. *International Journal of Technology and Design Education*, 26,
[11] Sullivan, A., Elkin, M., & Bers, M. U. (2015, June). KIBO robot demo: Engaging young children in programming and engineering. In *Proceedings of the 14th international conference on interaction design and children* (pp. 418-421). ACM.
[12] Sullivan, A., Kazakoff, E. R., & Bers, M. U. (2013). The wheels on the Bot go round and round: Robotics curriculum in pre-kindergarten. *Journal of Information Technology Education: Innovations in Practice*, 12, 203-219.
[13] Sullivan, A., Ber, U. M., & Mihm, C. (2017). *Imagining, playing, and coding with KIBO: Using robotics to foster computational thinking in young children*. Siu-cheung KONG The Education University of Hong Kong: Hong Kong, China. ISBN 9789887703440.
[14] Peta Wyeth (2008). How Young Children Learn to Program with Sensor, Action, and Logic Blocks. *Journal of the Learning Sciences*, 17(4), 517-550.
[15] Portelance, D. J., Strawhacker, A. L., & Bers, M. U. (2016). Constructing the ScratchJr programming language in the early childhood classroom. *International Journal of Technology and Design Education*, 26,
[16] Jung, S. E., & Won, E. S. (2018). Systematic Review of Research Trends in Robotics Education for Young Children. *Sustainability*, 10(4), 905.
[17] Barker, B. S., & Ansorge, J. (2007). Robotics as means to increase achievement scores in an informal learning environment. *Journal of Research on Technology in Education*, 39(3), 229-243.
[18] La Paglia, F., Rizzo, R., & La Barbera, D. (2011). Use of robotics kits for the enhancement of metacognitive skills of mathematics: A possible approach. *Stud Health Technol Inform*, 167, 26-30.
[19] Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers and Education*, 72, 145-157.
[20] Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2013). The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood. *Early Childhood Education Journal*, 41(4), 245-255.
[21] Clements, D. H., & McMillen, S. (1996). Rethinking "concrete" manipulatives. *Teaching Children Mathematics*, 2(5), 270-279.
[22] Hussain, S., Lindh, J., & Shukur, G. (2006). The effect of LEGO training on pupils' school performance in mathematics, problem solving ability and attitude: Swedish data. *Educational Technology and Society*, 9, 182-194.
[23] Lindh, J., & Holgersson, T. (2007). Does LEGO training stimulate pupils' ability to solve logical problems? *Computers & Education*, 49(4), 1097-1111.

[24] Di Lieto, M. C., Inguaggiato, E., Castro, E., Cecchi, F., Cioni, G., Dell'Omo, M., … & Dario, P. (2017). Educational robotics intervention on executive functions in preschool children: A pilot study. *Computers in Human Behavior*, 71, 16-23.
[25] Zhou, X., Zhao, Z. G., Li, J., Kang, D., Tian, L. L., Cheng, Y. C., … & Wang, G. H. (2013). The influence of cognitive factors on early mathematics learning difficulties in children. *Preschool Education Research*, (11), 3-13.
[26] Best, J. R., Miller, P. H., & Naglieri, J. A. (2011). Relations between executive function and academic achievement from ages 5 to 17 in a large, representative national sample. *Learning and Individual Differences*, 21(4), 327-336.
[27] Miyake, A., Friedman, N. P., Emerson, M. J., Witzki, A. H., Howerter, A., & Wager, T. D. (2000). The unity and diversity of executive functions and their contributions to complex "frontal lobe" tasks: A latent variable analysis. *Cognitive Psychology*, 41(1), 49-100.
[28] Anderson, P. (2002). Assessment and development of executive function (EF) during childhood. *Child Neuropsychology*, 8(2), 71-82.
[29] Ponitz, C. C., McClelland, M. M., Matthews, J. S., & Morrison, F. J. (2009). A structured observation of behavioral self-regulation and its contribution to kindergarten outcomes. *Developmental Psychology*, 45(3), 605-.
[30] Thorell, L. B., & Nyberg, L. (2008). The Childhood Executive Functioning Inventory (CHEXI): A new rating instrument for parents and teachers. *Developmental Neuropsychology*, 33(4), 536-552.
[31] Wei, W., Xie, Q. B., Zhu, J. J., He, W., & Li, Y. (2018). Reliability and validity analysis of the Childhood Executive Functioning Inventory in Chinese preschool children. *Chinese Journal of Clinical Psychology*, 26, 26-29.
[32] Epstein, A. S. (2003). How Planning and Reflection Develop Young Children's Thinking Skills. *Young Children*, 58(5), 28-36.
[33] McClelland, M. M., Cameron, C. E., Duncan, R., Bowles, R. P., Acock, A. C., Miao, A., & Pratt, M. E. (2014). Predictors of early growth in academic achievement: The head-toes-knees-shoulders task. *Frontiers in Psychology*, 5:599.

*Note: Figure translations are in progress. See original paper for figures.*