Computers & Education 242 (2026) 105510

Contents lists available at ScienceDirect

Computers & Education

journal homepage: www.elsevier.com/locate/compedu

Teacher-led robot intervention in early primary school classrooms improves pupil and teacher outcomes

Amy A.E. Hughes [a,1], Sarah A. Gerson [a], Johanna E. van Schaik [b,*]

[a] Cardiff University, Cardiff University Centre for Human Developmental Science (CUCHDS), School of Psychology, Cardiff, Wales, UK
[b] Radboud University, Behavioural Science Institute, Nijmegen, the Netherlands

ARTICLE INFO

Keywords:
Primary education
Computational thinking
Programming
Robots
Teacher education

ABSTRACT

Programming is often taught through robots in early primary education to support young children's computational thinking (CT), but many teachers lack the confidence and training to use them effectively. This paper presents a school-based robot intervention for children aged 4–7 (n = 430) and their classroom teachers (n = 17), delivered under three conditions: Intervention (robot intervention only), Intervention+ (intervention plus teacher education), and Control (no intervention). The two intervention groups assessed whether teacher education, in addition to classroom robot experience, influenced pupils' prediction and debugging, transferable skills (programming transfer and picture sequencing), and teachers' beliefs (enjoyment, relevance, self-efficacy, anxiety). The intervention improved children's prediction and debugging scores significantly, but only Intervention+ significantly outperformed Control for both prediction and debugging. Performance on the programming transfer and picture sequencing tasks improved across all groups. Teachers in both intervention groups reported improved relevance beliefs, though only Intervention+ showed a significant difference from Control. Self-efficacy also improved significantly in Intervention+ only. These findings offer practical guidance for embedding programming with robots in primary education and underscore the importance of teacher education for significant impact.

1. Introduction

The increasing recognition of the need to develop children's digital literacy has seen its inclusion in primary school curricula worldwide (Balanskat & Engelhardt, 2015; Bers, 2020; Bocconi et al., 2022; Uzunboylu et al., 2017). One cornerstone of digital literacy, computational thinking (CT), is considered crucial for problem solving across domains (Hwb, 2024; Sysło & Kwiatkowska, 2015). Although not the only route to CT, programming education introduces CT concepts, providing an interactive and practical approach for teaching young learners CT skills. Within this context, educational robots have gained popularity as a tool for teaching programming to those under 7 years, offering a tangible, engaging tool for preliterate children (Wang et al., 2023). Despite its potential, the successful implementation of robots in classrooms is hindered by many early childhood teachers' belief that they are inadequately prepared to teach programming (Hughes, 2024). This highlights the need for well-controlled, teacher-led research to investigate how robots can be effectively used to benefit both pupils' learning and teachers' beliefs. Thus, we conducted a study in

* Corresponding author.
E-mail address: Johanna.vanSchaik@ru.nl (J.E. van Schaik).
[1] (Present Address) Sheffield Hallam University, School of Sport and Physical Activity.

https://doi.org/10.1016/j.compedu.2025.105510
Received 16 June 2025; Received in revised form 12 November 2025; Accepted 18 November 2025
Available online 20 November 2025

which teachers delivered a programming intervention in their early primary school classrooms (4-7-year-olds).

1.1. CT through programming in early primary education

Whilst definitions of CT vary, it is generally considered "the conceptual foundation required to solve problems effectively and efficiently" (Shute et al., 2017, p.10). For the youngest learners, the basis of CT often focuses on *sequencing* (Ching & Hsu, 2024), as is the case in the Welsh curriculum (Hwb, 2024), the context of the present study. Sequencing is an essential cognitive process for organising steps in a logical order to achieve a desired outcome (Brennan & Resnick, 2012) and is crucial for early mathematics and literacy learning (see Kazakoff & Bers, 2012 for review). Programming introduces sequencing to formulate and execute instructions (i.e. scripts; Hwb, 2024). In addition to writing scripts, pupils are asked to analyse and evaluate sequences by *predicting* script outcomes and *debugging* errors when the actual result does not match the intended one (Bers et al., 2019). Accordingly, sequencing, predicting, and debugging were selected as they constitute the first two progression steps in the new Curriculum for Wales, which outlines skills expected in primary education (e.g., "I can follow algorithms to determine their purpose and predict outcomes"; Hwb, 2024).

1.2. Educational robots for teaching programming in early primary education

Educational robots offer an effective and interactive way to teach programming to young children (Bers et al., 2019; Çetin & Demircan, 2020). Robots like KIBO (www.kinderlabrobotics.com) and Cubetto (www.PrimoToys.com) allow even preliterate children to write and analyse scripts, by providing a visible, tangible representation of their code and immediate feedback on their programming attempts (Bakala et al., 2021; Hughes, 2024).

Robots can be effective in teaching sequencing, prediction, and debugging (Ching & Hsu, 2024; Wang et al., 2023). However, when considering the implementation of robots in primary classrooms, existing research highlights several limitations of past research and remaining open questions. These include issues regarding the implementation of robots with young pupils, the methodological rigor of studies (e.g., integrating controls, intervention duration; Rapti & Sapounidis, 2024; Tselegkaridis & Sapounidis, 2022), and investigating the role of teachers (Sullivan & Bers, 2016).

1.2.1. Educational robots for learning sequencing

A systematic review from Ching and Hsu (2024) highlighted sequencing as one of the most studied CT skills in early primary education robotics literature. They noted that in studies by Bers and colleagues, robot activities were shown to facilitate the development of sequencing skills in young learners (e.g., Kazakoff & Bers, 2012, 2014). In a study by Kazakoff and Bers (2014), 34 children (4.5–6.5 years) completed three programming sessions with a robot over a few weeks (±1.5 h each). In these laboratory sessions, children worked one-on-one with a researcher and completed a range of programming challenges. Children's sequencing skills were assessed pre- and post-intervention using a picture sequencing task (from Baron-Cohen et al., 1986). Children as young as 4.5-years-old improved in sequencing.

Notably, this picture-based assessment was unrelated to the programming tasks and measured sequencing in a new context. Skill transfer across different contexts is central to CT ideas (Grover & Pea, 2018). These findings align with the broader trends identified in Ching and Hsu's (2024) review, which reported that improvements in sequencing skills were observed in most studies investigating robot interventions. However, it is unclear whether improvements are still found when interventions are delivered by teachers rather than experienced researchers.

1.2.2. Educational robots for learning prediction and debugging

The ability to predict the outcome of a script is a key aspect of programming. Past studies, such as Slangen et al. (2011), have shown that children as young as 10 can improve prediction skills through interactions with robots like Lego Mindstorms. However, this research used qualitative observation methods (which may introduce demand characteristics), with limited use of quantitative measures or control groups. There is also a notable lack of research on the effect of robots on prediction with children under seven years (Mangina et al., 2023), which is an important gap considering that younger children are now being introduced to programming as part of early education curricula (Bocconi et al., 2022; Hwb, 2024). Thus, our study recruited children aged 4–7 years and utilised quantitative measures to track prediction skills across the intervention.

Similarly, the role of robots in developing debugging skills has been explored (see Ching & Hsu, 2024 for review). Across six sessions, Bers et al. (2014) found that children as young as four developed debugging skills with a programmable robot (CHERP). Furthermore, Pugnali et al. (2017) found that children (4–7 years, $n = 28$) who completed a ~15-h programming curriculum using KIBO robots outperformed those using screen-based programming applications (like Scratch Jr) in debugging tasks. These studies illustrate the potential of educational robots in cultivating debugging skills among young children, although the small sample sizes highlight the need for further research with larger cohorts.

Furthermore, these studies (and others reviewed by Ching & Hsu, 2024) did not assess changes in debugging skills across time by administering pre- and post-intervention measures. Instead, previous research has observed the strategies young children employ while debugging (Angeli & Valanides, 2020) or has measured debugging abilities post-intervention only (Bers et al., 2014; Bers et al., 2019; Pugnali et al., 2017; Taylor & Baek, 2019). In the present study, we adopted a pre-post design with a control group (lacking in many robot intervention studies, see Rapti & Sapounidis, 2024) to investigate changes in debugging over time.

1.2.3. Programming transfer

A remaining question about educational robots is whether pupils can transfer programming skills from robots to screen-based programming. This is relevant given the widespread use of screen-based programming in later school years. As with robots, many screen-based programming languages and programming-like games exist, some of which can be used by preliterate pupils (e.g., Scratch Jr, www.scratchjr.org). While the current study does not directly compare different programming modalities, it examines whether pupils can spontaneously apply programming concepts learned with a tangible robot to a new, conceptually-similar but visually-novel, screen-based environment.

1.2.4. Not all educational robots are equal

Despite proposed benefits of robots, specific functionalities of these devices may impact their suitability for young children. For example, Bee-Bot (www.tts-group.co.uk) is a robot that requires children to remember instructions without providing a visual representation of the script, which can increase cognitive load and make debugging challenging (Bakala et al., 2021). KIBO, on the other hand, provides children with a clear display of their sequences as they manipulate and scan barcodes on physical wooden blocks. However, KIBO's blocks have been described as difficult to scan. González-González and colleagues' (2019) study using KIBO reported that children often did not wait for the notification to confirm that the code had been read before continuing with their script, leading to invisible mistakes.

The more-recently developed Cubetto robot appears to overcome limitations identified from research with KIBO and Bee-bot robots. Inspired by Montessori principles, Primo Toys designed Cubetto to be hands-on, collaborative, and facilitate self-correction (i.e. debugging; Migliaresi, 2016). Like other educational robots, Cubetto's design avoids text, thus making it suitable for preliterate children. Cubetto includes an interface board on which the user lays out their visual script using tangible tokens (e.g., forward, right, and left turn functions) and a floor map. Children can navigate the robot by placing the desired tokens in the interface board and pressing the 'Go' button. Though, past studies have noted that the board's layout can lead to confusion because the orientation of tokens is different in the second row compared to the first and third rows (Clarke-Midura et al., 2023; Faber et al., 2019).

1.3. The role of teachers

Although demonstrating the effectiveness of robots for teaching programming and CT skills is a prerequisite, the ultimate challenge lies in translating these findings into everyday classroom practice. A recurring limitation in research on robot interventions is that most have been implemented by researchers rather than classroom teachers (e.g., Strawhacker & Bers, 2015; Sullivan & Bers, 2016). Although rare, there are exceptions where teachers have delivered programming curricula (e.g., Bers et al., 2019). Nevertheless, such studies remain limited and, in some cases (Sullivan & Bers, 2016), teachers were expected to learn and gain confidence in teaching programming content through observing a researcher delivering the content. However, findings from previous research (Girvan et al., 2016; Hughes, 2024) indicate that experiential learning is favoured by primary school teachers.

Furthermore, this researcher-led approach, while ensuring consistency, does not reflect the realities of everyday teaching, where teachers are the ones delivering lessons. The absence of teacher-led robot interventions leaves open important questions regarding whether the benefits of robots would be found if the classroom teacher (who is likely to have less knowledge of programming and robots, but more knowledge of their pupils and the broader curriculum) delivered the curriculum. Engaging teachers as active participants is important for both improving teacher confidence (Burke & Hutchins, 2007) and ensuring the long-term success of programming and robot interventions.

One of the main obstacles in engaging teachers, though, is that many early primary school teachers feel underprepared to teach programming (Hughes, 2024; Khanlari, 2016; Ohashi et al., 2018; Ray et al., 2020). Research suggests that teacher beliefs, particularly their self-efficacy, play a significant role in determining how effectively they implement new curricula (Lauermann and ten Hagen, 2021). For example, when teachers lack self-efficacy in their ability to teach programming, they may be less likely to engage pupils in meaningful learning, which can limit pupil outcomes (Ohashi et al., 2018). Therefore, teacher education programs that provide teachers with the tools and beliefs to implement robots are necessary to ensure that programming becomes an integral part of early primary education (Kim & Seo, 2018). In this study, teachers' beliefs refer to their enjoyment, perceived relevance (i.e., the extent to which they perceive the subject as meaningful, developmentally appropriate, and aligned with broader learning goals), self-efficacy, and anxiety about teaching programming-related content.

A major barrier to the effective delivery of programming education in early primary education is the lack of continuing education opportunities for teachers (Bers, 2010; Hughes, 2024), particularly those suited for their classroom's contexts (e.g., with younger pupils; Hughes, 2024). Providing teachers with education opportunities that include hands-on, experiential learning has the potential to improve their self-efficacy and, by extension, their teaching practices (Burke & Hutchins, 2007). The current study investigated whether teachers who received specific education showed greater improvements in their enjoyment, perceived relevance, self-efficacy and anxiety about teaching programming compared to those who did not.

1.4. Current study

As reviewed above, CT is frequently targeted within early primary education curriculums and is often developed through teaching programming. For young children, educational robots are promising tools to introduce sequencing, prediction, and debugging. However, to what extent such these skills are learned during school-based curricula and how easily pupils transfer these skills to unpractised programming contexts or unrelated instances of sequencing is unclear. Moreover, primary school teachers' uncertainty

and lack of experience with teaching programming education makes introducing these skills inside the classroom a challenge. Thus, this study aimed to investigate the impact of a 6-week teacher-led robot intervention on children's programming and CT skills and teachers' beliefs. To do so, we implemented three experimental conditions: Intervention (robot curriculum only), Intervention+ (robot curriculum and teacher education workshop) and Control (learning as normal). We assessed pupils' prediction and debugging, screen-based programming skills, and picture sequencing before and after the intervention. Teachers' beliefs (i.e. teacher enjoyment, relevance, self-efficacy, anxiety) were measured before and after Intervention + attended the teacher education workshop and again at the end of the intervention. This study aimed to answer the following research questions:

RQ1) Can a 6-week Cubetto curriculum improve children's prediction and debugging skills, as well as their performance on screen-based programming and picture sequencing tasks? Do outcomes differ for children if their teachers attend an education workshop?
RQ2) Can a 6-week robotics curriculum improve teachers' beliefs regarding programming and robots and how do changes vary as a result of whether teachers attended an education workshop?

2. Methods

2.1. Design

This study employed a pre-test post-test design with classrooms assigned to one of three conditions: Intervention (classroom intervention only), Intervention+ (classroom intervention plus teacher education) and Control (no intervention, learning as normal). Teachers in Intervention + attended a teacher education workshop to learn about the robot (Cubetto) and its accompanying lesson plans before they implemented the lessons in their classrooms. Not all teachers delivering the lessons received this additional input; teachers in Intervention only received an introductory guide and the lesson plans. Control (teachers and pupils) had no exposure to the robots or lessons until after post-intervention assessments. All children were pre-tested before the intervention conditions began and post-tested after the intervention conditions completed the program (see Fig. 1).

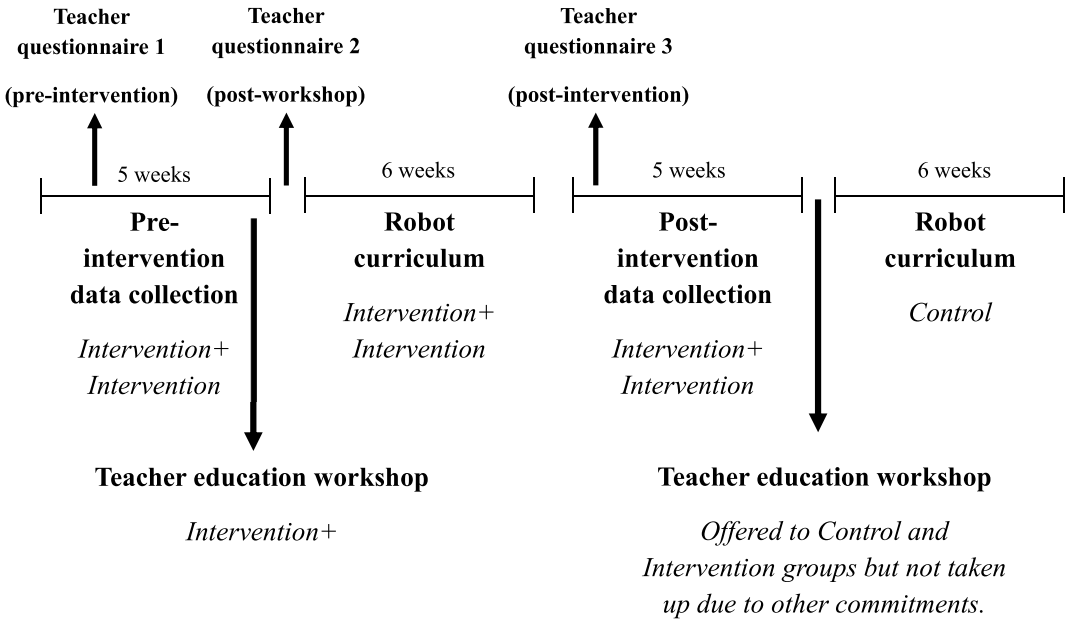

Fig. 1. Illustration of the project timeline.

2.2. Participants

Of 15 schools who volunteered, seven were chosen to participate based on physical resources and geographical restraints. Some of the early primary teachers (those teaching children 4–7 years) had previously completed an online survey investigating whether and, if so, to what extent teachers had experience with programming, and they had expressed interest in participating with their schools (see Hughes, 2024).

Each school (regardless of number of classrooms participating) was allocated to one of the three conditions, as it would not have been possible to prevent the spread of information between teachers if individual classrooms within a school were assigned to different conditions. Consequently, allocation was not completely randomised due to the size of the schools taking part. Two large schools (~180 pupils each) were each randomly assigned to the Intervention+ and Control conditions, and the five remaining (smaller) schools were assigned to the Intervention condition (~180 pupils in total).

Socioeconomic status of participating schools was estimated using the Welsh Index of Multiple Deprivation (WIMD) based on school postcodes. The WIMD ranks small areas in Wales from 1 (most deprived) to 10 (least deprived decile). The Intervention+ school was in a highly deprived area (decile 1). The Intervention condition included five schools spanning a range of deprivation levels (deciles 9, 7, 4, 2, and 2). The Control school was in a least deprived area (decile 10).

Seventeen teachers (15 female, mean age = 39.3; see Table 1) and ~550 pupils completed the robot curriculum taught by their teachers during regular class time. Data was collected from 430 of these children whose parents provided written opt-in consent (50.93 % female, mean age = 5.9, range = 4.41 to 7.65). Data from children with formal diagnoses of developmental delay or needing one-to-one teaching support were excluded. Participant distribution was: Intervention+ = 149, Intervention = 137, Control = 144.

2.3. Intervention materials

2.3.1. Robot curriculum

Teachers in Intervention+ and Intervention received Cubetto robot kits (1 robot per 6 children, as advised by Primo Toys; see Fig. 2).

2.3.1.1. Lesson plans. Teachers completed six structured weekly lessons (created by Primo Toys and an early primary teacher), focusing on skills like sequencing, prediction, and debugging (see OSF). These lessons were designed to reflect a constructivist, Montessori-inspired pedagogy, using play-based activities. The plans framed programming as a cross-curricular tool, supporting teachers to embed programming across subjects rather than teach it in isolation. Lesson one introduced Cubetto and basic sequencing. Lesson two involved programming Cubetto to collect shapes on the map. Lesson three focused on predicting script outcomes using a snakes and ladders game. Lesson four advanced prediction skills through peer work and reviewing each other's scripts (see Fig. 2). Lessons five and six introduced debugging, with pupils correcting scripts to navigate mazes and obstacle courses.

2.3.1.2. Introductory guide. All teachers in Intervention+ and Intervention received Primo Toys' Teaching Guide (see OSF) covering Cubetto's functions and instructional strategies for classroom integration.

2.3.2. Teacher education workshop

Teachers in Intervention+ attended a 3-h robot workshop led by two researchers before commencing the 6-week robot curriculum. The workshop, informed by a teacher focus group and survey (Hughes, 2024), included three components: a presentation on programming concepts, hands-on Cubetto practice, and discussions on incorporating programming into existing curricula. Teachers explored classroom setups such as whole-class demonstrations and small-group activities and reviewed example lesson plans. The workshop promoted a constructivist, interdisciplinary approach through hands-on exploration, collaborative planning, and strategies for integrating programming across subjects (see OSF).

2.3.3. Implementation fidelity

Teachers completed feedback forms after each lesson, noting the time spent using Cubetto, modifications to lesson plans, and reflections on positive and negative aspects. Control teachers were also asked to complete weekly feedback forms reporting any programming or CT activities planned within their curriculum. They were not encouraged to teach these topics before post-intervention data collection, but they were not discouraged from carrying out their planned lessons.

All six feedback forms were completed for the six classrooms participating in Intervention+. In Intervention, all six forms were completed for three of the five classrooms. For the remaining two classrooms, teachers completed five of the forms, with the final form missing despite confirmation that the last session was delivered. No teachers implemented any of the additional lesson plans beyond the six compulsory lessons. Lesson duration ranged from 1 to 2 h. Control group feedback forms had a very low response rate, with only two teachers returning forms across the intervention period; neither reported delivering CT-related activities.

2.4. Measures

2.4.1. Pupil assessments

Two assessment types were used: whole-class-administered paper-and-pencil programming tasks and individually administered

Table 1
Details of intervention participants, grouped by condition.

	Intervention + n (% female)			Intervention n (% female)			Control n (% female)		
	Reception	Year 1	Year 2	Reception	Year 1	Year 2	Reception	Year 1	Year 2
Teachers	1 (100)	2 (100)	1 (100)	3 (100)	3 (100)	3 (33.33)	2 (100)	2 (100)	2 (100)
Pupils	47 (53.19)	49 (42.86)	52 (53.84)	52 (53.84)	39 (66.67)	46 (45.65)	40 (47.5)	52 (50)	52 (48.08)

Note: Two teachers taught a mixed classes of Reception/Year 1 children and Year 1/Year 2 children, however the total number of participating teachers was 17.

Fig. 2. Prediction Activity
Note: Pupil completing a prediction activity during lesson four of the robot curriculum. Cubetto robot is on the map. The X represents the child's prediction.

transfer tasks given to a random subsample of children within each condition. Hands-on activities with Cubetto were not used to avoid disadvantaging the Control group and to ensure usability at pre-test. Measures were piloted with similarly aged children (4–7 years) to ensure they were developmentally appropriate. Per class, all assessments were completed on the same day.

2.4.1.1. Whole-class assessment. All pupils completed three paper-pencil assessments; however, data was only collected from those with consent. All tasks were administered in the same session in the following order: visual perspective taking[2], prediction, and debugging. Not all pupils completed all tasks, leading to more missing data for debugging than prediction. Before the prediction task, a brief PowerPoint introduced a cartoon Cubetto robot and its movement tokens, similar in appearance to the real tokens. It was explained that the robot needed a sequence of instructions to move across a grid (analogous to the floor map), with each coloured token representing a different command. Animations of each token's function were shown twice, with a verbal check to ensure understanding. Subsequently, children received specific instructions per task.

2.4.1.1.1. Prediction. This task assessed whether children could predict the robot's final position on a grid based on a script (see Fig. 3a). Two demonstration trials were conducted via the slideshow: one with only "forward" tokens and another with a "turn" token. Pupils worked in individual booklets on 10 trials of increasing difficulty, based on token number, type (only forward vs. turn), and robot orientation (facing toward participants was more challenging than facing the same direction). To answer each trial, children drew an "X" on the grid to indicate where they predicted the robot's final position would be. Each correct trial earned 1 point.

2.4.1.1.2. Debugging. This task was based on previous research (see Strawhacker & Bers, 2019, Solve It tasks). To introduce the task, an animation was shown where a robot failed to reach a target due to an incorrect script. The incorrect token was removed, creating a gap, and the class selected the correct token to complete the script. The corrected script was then run, showing the robot reaching the target. Two such demonstration trials were conducted: one with a missing forward token and one with a missing turn token. Participants then completed ten trials of increasing difficulty in their booklets by themselves (see Fig. 3b). Although 10

[2] One additional whole-class assessment was collected but not reported for brevity. A Visual-Perspective Taking task was used to assess children's ability to visualise a display from another observer's perspective. Details of this task are available on OSF.

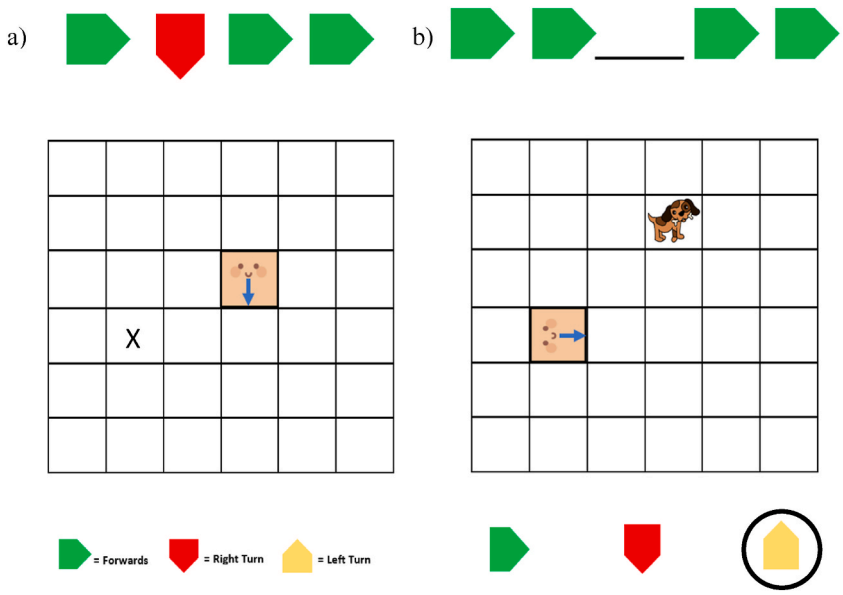

Fig. 3. *Note:* Fig. 3a) Prediction task. The script is shown at the top of the page. Blue arrow on the robot represents its orientation. Reminders of token functions are at the bottom of the page. The "X" on the grid shows the correct answer for this trial. Fig. 3b) Debugging task. The script is shown at the top of the page. Children chose a token to fill the gap by circling it. The circled token is the correct answer for this trial ("left turn" token). (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

debugging trials were completed by children, 8 trials were included in the final analyses due to a misprint in the assessment booklet at trial 9, which was therefore excluded. In case the misprint in trial 9 mislead pupils, trial 10 was also removed from the final analyses. For every trial, an animation was shown via the slideshow and pupils subsequently needed to circle the missing token in their booklet. Each correct trial earned 1 point.

2.4.1.1.3. Individual assessments. Due to time constraints, a random subsample of participants (Intervention+ = 104, Intervention = 86, Control = 118; total $n = 308$) completed four tasks[3] in a 20-min, one-to-one session with a researcher. Within each classroom, teachers selected consenting pupils to participate using a class list. These tasks were administered in the following order: verbal beliefs questionnaire, sequencing task, executive functioning, and programming transfer. Pupils were free to end the test-session of their own volition, leading to different numbers of missing data sets per task.

2.4.1.1.4. Programming transfer. Lightbot Jr (www.lightbot.com), a programming app for children aged 4-to-8, was chosen as it also involves grid-based movement, but is unique from the Cubetto tasks children completed. Pupils played on a tablet, guiding a robot to reach a goal (i.e., lighting up all blue tiles) by creating scripts (forward, turn, lightbulb). They were given 5 min to play and could complete up to nine levels, with on-screen instructions read aloud by the researcher. Children received no direct help, only prompts like *"what do you think?"* if they asked for assistance.

However, including too many tokens could in some cases be ignored in this app (whereas a Cubetto would then end up in a wrong location), leading to level progression despite incorrect scripts. Therefore, screen recordings were coded for programming efficiency: 2 points were awarded for levels completed with the minimum number of tokens used, and 1 point for levels completed with extra tokens. Scores per trial were totalled for a total score.

2.4.1.1.5. Sequencing. A picture sequencing task, adapted from previous studies (Baron-Cohen et al., 1986; Kazakoff et al., 2013), assessed sequencing skills. The task featured six sequences of activities with increasing difficulty: two 4-card (easy, e.g., making a sandwich), two 5-card (medium), and two 6-card (hard). Participants were instructed to place the picture cards in the correct order, from left-to-right. There was no time limit for this task. Each correct sequence scored 1 point, for a total range of 0–6.

2.4.2. Teacher beliefs questionnaire

Teachers completed a questionnaire at three points: pre-intervention, post-workshop (after Intervention + completed the workshop; before implementing the robot curriculum), and post-intervention (after the robot curriculum). Two validated attitude measures were combined: the Dimensions of Attitudes toward Science scale (van Aalderen-Smeets & Walma van der Molen, 2013) and the Preschool Teacher Attitudes and Beliefs toward Science questionnaire (Maier et al., 2013). This combination enabled a comprehensive

[3] Two additional individual pupil measures were collected but are not reported in this paper for brevity. Pupils completed an iPad-based assessment of executive functioning (the Minnesota Executive Function Scale; Carlson & Schaefer, 2012) and a verbal beliefs questionnaire (an adaptation of a measure used by Master et al., 2017). These are not the focus of this manuscript but more information about them is available on OSF.

evaluation of teachers' beliefs (enjoyment, relevance, self-efficacy and anxiety) and teaching practices, which served as a manipulation check of increased programming content during the intervention. Duplicate questions and questions too specifically related to physical sciences were removed. For example, *"I demonstrate experimental procedures (e.g., comparing objects to see if they will sink or float) in my classroom"* was removed.

The final questionnaire contained 52 items (see OSF), compartmentalised into eight subscales, rated on a five-point Likert scale ("strongly disagree" to "strongly agree"). Subscales included the enjoyment of teaching programming (5 items, $\alpha = .91$), relevance of teaching programming (16 items, $\alpha = .88$), anxiety about teaching programming (4 items, $\alpha = .86$), self-efficacy in teaching programming (8 items, $\alpha = .85$), difficulty of teaching programming (3 items, $\alpha = .60$), gender-stereotypical beliefs (4 items, $\alpha = .71$), contextual factors influencing programming education (5 items, $\alpha = -0.20$), and current teaching practices related to programming (7 items, $\alpha = 0.74$).

We analysed five of these eight subscales within this manuscript: teacher enjoyment, relevance, anxiety, self-efficacy, and teaching practices. The remaining subscales (difficulties teaching programming, gender beliefs and perceptions of external barriers) are not explored in the main text for several reasons. The difficulty of teaching programming subscale required teachers to think about how "most teachers" felt about teaching programming rather than their own personal beliefs. We had no expectations that the intervention would alter their ratings of others' beliefs. This study did not manipulate external factors or barriers teachers may face when trying to teach programming and robot education, nor was it an intervention targeting teachers' gender stereotype beliefs. Analysis of these other subscales can be found on OSF.

2.4.2.1. Questionnaire responses. The questionnaire had the highest response rate post-intervention (88 % of teachers responded), with lower participation at pre-intervention and post-workshop (53 % of teachers responded). The Intervention+ and Intervention groups had consistent response rates across all time points (≥67 %), while the Control group had the most missing data, especially at post-workshop (100 % missing). Additionally, two teachers from the Control group did not complete the beliefs questionnaire at any of the three timepoints.

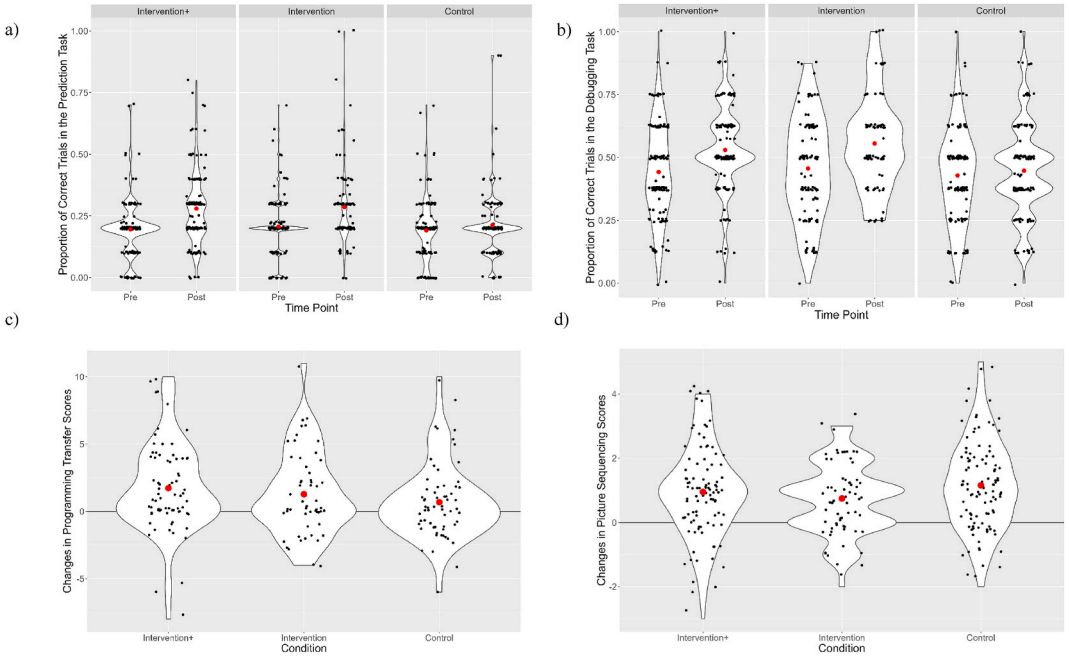

Fig. 4. Pupil Performance on Assessments
Note: Fig. 4a) Prediction; Fig. 4b) Debugging; Fig. 4c) Programming Transfer; Fig. 4d) Sequencing. Results are grouped by condition. Group means displayed in red. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

2.5. Data analysis

Multilevel modelling was used to account for hierarchical data: pupils nested within classrooms/teachers, classrooms within schools and schools within an experimental condition (Hox et al., 2017; Koepsell et al., 1992). The analysis used R Statistical Software (R Core Team, 2021) with the lme4 package (Bates et al., 2015), applying Maximum Likelihood Estimation and Full Information Maximum Likelihood for missing data (Grund et al., 2018).

When analysing pupil data, all base models included fixed effects of condition (Intervention+, Intervention, Control). Additionally, for binomial assessment measures, interaction effects of condition and assessment time-point (i.e., pre-intervention; post-intervention) were also included. For non-binomial measures, interaction terms could not be modelled because difference scores (pre to post change) were analysed when individual trial data were unavailable. For teacher data, all models included interactions of condition and time-point (i.e., pre-intervention, post-workshop, and post-intervention).

Initially, both teacher and pupil were considered random intercepts to account for variability at the classroom and pupil levels. However, singularity errors were encountered due to the complexity of the random effects structure (Barr, 2013). To resolve this, only teacher was retained as a random intercept, as it provided a better fit for the data based on Akaike Information Criterion values (Akaike, 1974). This decision was supported by literature indicating that teacher effects significantly influence pupil achievement (Gustafsson, 2003). Random slopes were excluded from models analysing pupil outcomes as data was collected at two time points. However, time was included as a random slope in the models analysing teacher beliefs as data was collected at three time points.

3. Results

3.1. Pupils

3.1.1. Prediction

Prediction task data was collected from 409 pupils (at pre- and/or post-testing, 21 pupils did not provide data at both timepoints). Data are visualised in Fig. 4a. The analysis revealed a significant interaction effect of condition and time on children's prediction performance (see Table 2). Children in Intervention+ showed significantly larger improvements in scores compared to those in Control. The differences between Intervention and Control were not significant. No significant differences were found between Intervention+ and Intervention.

Pairwise comparisons using the *emmeans* package showed that children in Intervention+ (odds ratio = 0.61, z = −5.24, $p < 0.001$) and Intervention (odds ratio = 0.66, $z = -3.86$, $p < 0.001$) were significantly more likely to perform better at post-intervention than at pre. Odds of improving did not increase for children in Control (odds ratio = 0.85, $z = -1.56$, $p = .12$).

3.1.2. Debugging

Debugging task data was collected from 397 children (at pre- and/or post-testing, 33 pupils did not provide data at both time-points). Fig. 4b illustrates the proportion of correct trials on the debugging task, grouped by condition and time point. The analysis revealed a significant interaction effect of condition and time on children's debugging performance (see Table 2). Children in Intervention+ showed significantly larger improvements in their scores compared to those in Control. There were no significant differences in improvements between Intervention and Control, nor between Intervention+ and Intervention.

Pairwise comparisons indicated that children in Intervention+ (odds ratio = 0.69, z = −4.16, $p < 0.0001$) and Intervention (odds

Table 2
Pupil model results.

	Model summaries					
	β	*SE*	*t*	95 % CI	*p*	
Prediction						
Intervention + vs Control	0.33	0.14	2.41	[0.06, 0.61]	.02	*
Intervention vs Control	0.25	0.15	0.71	[-0.04, 0.54]	.09	.
Intervention + vs Intervention	0.56	0.14	0.58	[-0.19, 0.04]	.56	
Debugging						
Intervention + vs Control	0.30	0.13	2.35	[0.05, 0.55]	.02	*
Intervention vs Control	0.22	0.15	1.49	[-0.07, 0.50]	.14	
Intervention + vs Intervention	0.09	0.14	0.59	[-0.37, −0.20]	.56	
Programming Transfer						
Intervention + vs Control	0.95	0.92	1.03	[-0.79, 2.68]	.33	
Intervention vs Control	0.68	0.86	0.79	[-0.93, 2.29]	.44	
Intervention + vs Intervention	−0.27	0.91	−0.29	[-1.98, 1.45]	.77	
Sequencing						
Intervention + vs Control	−0.18	0.22	−0.80	[-0.60, 0.25]	.44	
Intervention vs Control	−0.41	0.23	−1.78	[-0.85, 0.02]	.09	.
Intervention + vs Intervention	0.23	0.24	0.97	[-0.22, 0.69]	.35	

Note: . p < 0.1, *p < .05. Intervention + vs Control, illustrates whether there is an interaction between Conditions (Intervention+ and Control) and Time (pre- and post-intervention).

ratio = 0.75, $z = -2.54$, $p = .01$) were significantly more likely to perform better at post-intervention. Odds of improving did not increase for children in Control (odds ratio = 0.93, $z = -0.78$, $p = .44$).

3.1.3. Programming transfer

190 children were included in this analysis, having completed the programming transfer assessments pre- and post-intervention (missing $n = 118$; data missing at one timepoint $n = 56$, loss of screen recordings $n = 62$). At each time point, total scores could range from 0 to 18. Difference scores were calculated by subtracting post-intervention scores from pre-intervention scores and are illustrated per condition in Fig. 4c.

The model did not show a significant main effect of condition on children's improvement in programming (see Table 2). Neither Intervention+ nor Intervention significantly differed from Control. Similarly, scores in Intervention+ did not differ significantly from Intervention. Instead, t-tests showed that improvements in programming scores differed significantly from 0 across all conditions (all $p < .001$).

3.1.4. Sequencing

267 children were included in this analysis, having completed the picture sequencing assessments pre-and post-intervention (missing $n = 41$). At each time point, total scores could range from 0 to 6. Fig. 4d illustrates changes in sequencing scores per condition.

The model did not show a significant main effect of condition on children's improvement in sequencing (see Table 2). Neither Intervention+ nor Intervention significantly differed from Control. Similarly, scores in Intervention+ did not differ significantly from Intervention. Instead, t-tests showed that improvements in sequencing scores differed significantly from 0 across all conditions (all $p < .001$).

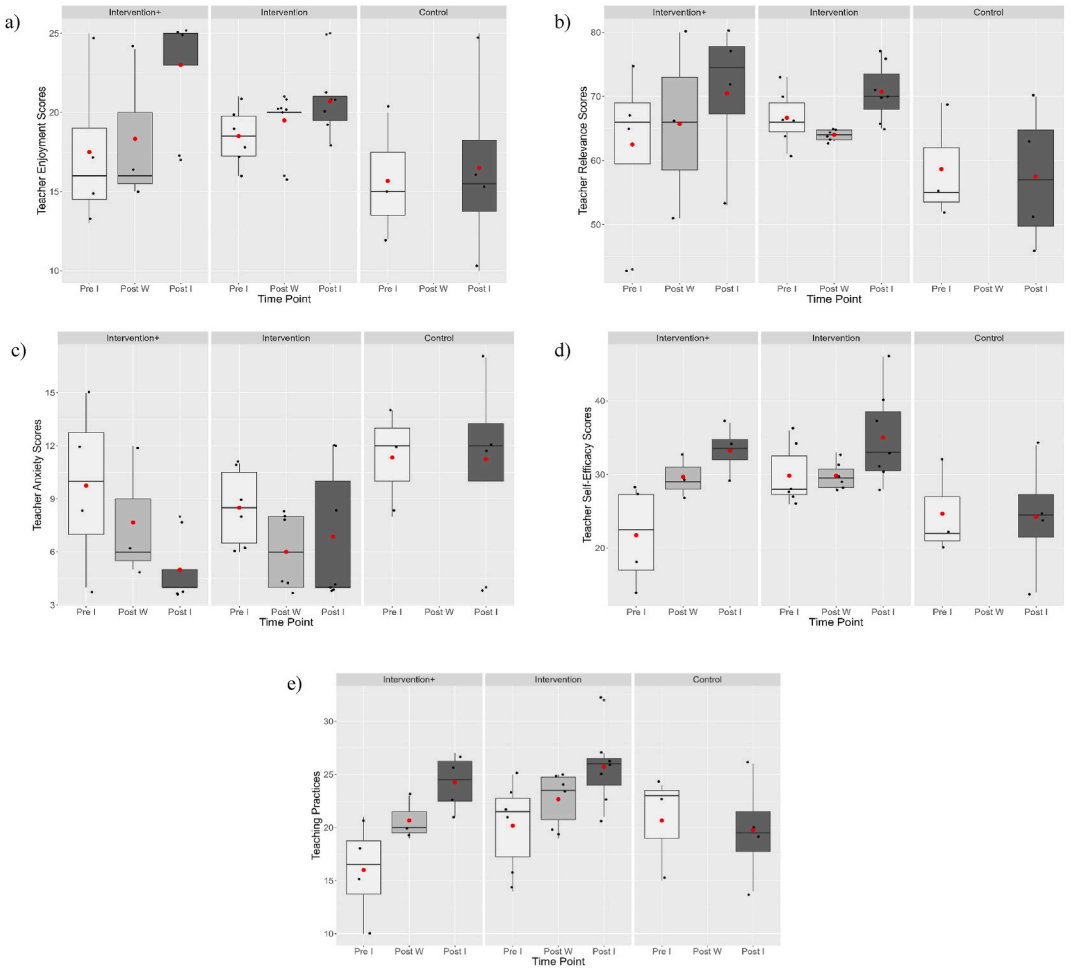

Fig. 5. Scores from Teacher Beliefs Questionnaires

Note: Fig. 5a) Enjoyment; Fig. 5b) Relevance; Fig. 5c) Self-Efficacy; Fig. 5d) anxiety; Fig. 5e) Teaching Practices. Results are grouped by condition with three time points. Pre-intervention (Pre-I), post-workshop (Post-W) and post-intervention (Pre-I). Group means displayed in red. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

Table 3
Teacher model results.

	Model summaries					
	β	*SE*	*t*	95 % CI	*p*	
Enjoyment						
Intervention + vs Control	2.29	1.24	1.84	[-0.01, 4.60]	0.09	.
Intervention vs Control	0.44	1.14	0.38	[-1.64, 2.59]	0.71	
Intervention + vs Intervention	−2.29	1.06	−1.74	[-3.79, 0.16]	0.11	
Relevance						
Intervention + vs Control	5.35	1.80	2.97	[0.86, 8.75]	0.008	**
Intervention vs Control	3.44	1.66	2.07	[0.25, 6.60]	0.051	.
Intervention + vs Intervention	1.91	1.53	1.25	[-0.99, 4.81]	0.23	
Self Efficacy						
Intervention + vs Control	5.25	2.12	0.89	[1.30, 9.98]	0.03	*
Intervention vs Control	1.74	1.94	0.89	[-1.84, 6.38]	0.39	
Intervention + vs Intervention	3.52	1.83	1.93	[-0.42, 6.98]	0.08	.
Anxiety						
Intervention + vs Control	−0.18	0.22	−0.80	[-0.60, 0.25]	0.44	
Intervention vs Control	−0.41	0.23	−1.78	[-0.85, 0.02]	0.09	.
Intervention + vs Intervention	0.23	0.24	0.97	[-0.22, 0.69]	0.35	
Teaching Practices						
Intervention + vs Control	4.51	1.38	3.28	[1.97, 7.07]	0.01	*
Intervention vs Control	3.10	1.27	2.45	[0.77, 5.46]	0.03	*
Intervention + vs Intervention	1.41	1.17	1.21	[-0.77, 3.57]	0.25	

Note: . $p < 0.1$, *$p < .05$, **$p < 0.01$. Intervention + vs Control, illustrates whether there is an interaction between Conditions (Intervention+ and Control) and Time (pre-intervention, post-workshop and post-intervention).

3.2. Teachers

3.2.1. Enjoyment

Fig. 5a illustrates teacher enjoyment scores. The model did not show a significant interaction effect of condition and time on teachers' enjoyment scores (see Table 3). Neither Intervention+ nor Intervention significantly differed from Control in reported change of enjoyment across all time points. Similarly, scores in Intervention+ did not differ significantly from Intervention.

3.2.2. Relevance

Fig. 5b illustrates teacher relevance scores. The model showed a significant interaction effect of condition and time on teachers' relevance scores (see Table 3). Intervention+ had significantly larger improvements in relevance scores than Control. Although scores in Intervention descriptively improved over time, these changes were only marginally different from Control. Improvement scores in Intervention+ did not differ significantly from Intervention.

The *pairs()* function in the *emmeans* package (Lenth, 2021) was used for post-hoc follow-ups. Results showed that relevance scores significantly improved overall (i.e., between pre-intervention and post-intervention) for teachers in Intervention+ ($B = -8.00$, $SE = 1.90$, $t = -4.22$, $p = .001$). However, relevance scores showed no significant increases from before to after the workshop ($p = .08$), or from post-workshop to post-intervention ($p = .33$). For Intervention teachers, relevance scores significantly improved between post-workshop and post-intervention time points ($B = -6.82$, $SE = 1.55$, $t = -4.41$, $p = .001$), and overall (pre-intervention to post-intervention; $B = -4.15$, $SE = 1.55$, $t = -2.69$, $p = .04$). Scores did not significantly improve overall in Control ($p = .42$).

3.2.3. Self-efficacy

Fig. 5c illustrates teacher self-efficacy scores. The model showed a significant interaction effect of condition and time on teachers' self-efficacy scores (see Table 3). Intervention+ showed significantly larger improvements in self-efficacy scores than Control. Although scores in Intervention descriptively improved over time, these changes were not significantly different from Control. Similarly, improvement in Intervention+ did not differ significantly from Intervention.

Post hoc tests were used to further explore the significant improvements seen in Intervention+. Results showed that self-efficacy scores significantly improved overall (i.e., between pre-intervention and post-intervention) for teachers in Intervention+ ($B = -11.50$, $SE = 2.70$, $t = -4.27$, $p = .001$). Self-efficacy scores showed significant increases between pre-intervention and post-workshop ($B = -9.00$, $SE = 3.00$, $t = -3.00$, $p = .02$), but no significant differences were found between post-workshop and post-intervention ($p = .69$). Additionally, no significant improvements were found between any time points for Intervention (p's $> .16$) and Control ($p = .87$).

3.2.4. Anxiety

Fig. 5d illustrates teacher anxiety scores. Despite positive descriptive trends, the model did not show a significant interaction effect of condition and time on teachers' anxiety scores (see Table 3). Neither Intervention + nor Intervention significantly differed from Control. Similarly, scores in Intervention+ did not differ significantly from Intervention.

3.2.5. Teaching practices

Fig. 5e illustrates reported teaching practices. This analysis served as a manipulation check (whether the robot curriculum was implemented). There was a significant interaction effect of condition and time on teachers' practice scores. Both Intervention+ and Intervention had significantly higher increases in practice scores compared to Control. Scores in Intervention + did not differ significantly from the Intervention.

Post-hoc analysis showed teaching practices improved overall (i.e., more programming content was taught by the end of the intervention than before) for teachers in Intervention+ ($B = -8.25$, $SE = 1.73$, $t = -4.77$, $p < 0.001$) and Intervention ($B = -5.42$, $SE = 1.40$, $t = -3.87$, $p = .003$). Delivery of programming content did not significantly increase between pre-intervention and post-workshop (all p's > 0.1). Teaching scores did not significantly increase overall in Control ($p = .92$).

4. Discussion

This study assessed a 6-week robot curriculum across three conditions (inclusive of a control group and a group receiving an additional teacher education workshop) using a pre-test post-test design. We compared changes in pupils' prediction and debugging, their performance on transfer tasks (programming and sequencing in new contexts), and teachers' beliefs between conditions.

4.1. Programming skills: prediction and debugging

We found the robot curriculum significantly improved pupils' performance on paper-and-pencil, Cubetto-related prediction and debugging measures. However, only Intervention+ differed significantly from Control in improvement. Although children in Intervention showed pre–post gains, their improvements did not differ significantly from Control.

These findings extend previous research that has explored the influence of robots on prediction and debugging skills in children. Regarding prediction skills, Slangen et al. (2011) concluded, on the basis of qualitative observations, that learning with robots could improve 10- to 11-year-olds' prediction skills. Turning to debugging, Ching and Hsu (2024) found that previous research had used observation methods to explore debugging strategies used by young children (Angeli & Valanides, 2020) or had assessed debugging skills with quantitative measures at post-intervention only (Bers et al., 2014; Bers et al., 2019; Pugnali et al., 2017; Taylor & Baek, 2019). The present study extends these findings using an intervention with controls and a whole-class-administered paper-and-pencil assessment, which have the benefit of being easier to administer and scalable. Interestingly, this also indicates that pupils were capable of the near-transfer (Perkins & Salomon, 1992) of these skills to visually- and conceptually-similar but two-dimensional measures.

Moreover, these findings suggest a central role of the teacher in pupils' learning outcomes. Though performance varied across all conditions, only children in Intervention+ showed significant improvement relative to children in Control. This indicates that researcher-led lessons, as discussed in the introduction, may indeed not transfer to the classroom where teachers are not educated on programming. Instead, the teachers of Intervention+, due to their expertise or beliefs, likely foster learning in specific ways that support learners. Future fine-grained investigations of teaching practices and pupils' individual differences could elucidate which aspects of instruction are effective for which learners and how these aspects might be supported through teacher education sessions like the one attended by the teachers in this study.

4.2. Transfer of programming and CT

Though descriptively, Intervention+ improved more on the programming transfer task than Intervention and Control, and Intervention more than Control, neither condition significantly differed from Control. Instead, all conditions improved on the task, which was identical at pre-test and post-test. Thus, it is unclear whether the curriculum facilitated the transfer of programming knowledge from a tangible robot to a new screen-based robot above and beyond testing and maturation effects. This aligns with existing literature suggesting young children often struggle to transfer skills between different modalities (Moser et al., 2015; Perkins & Salomon, 1992). In their conceptual paper, Kallia and Cutts (2023) even suggested that transferring programming skills from physical to digital environments should not be a focus for children under 8 years due to its difficulty. Taken together, whilst the visually- and conceptually-similar paper-and-pencil prediction and debugging tasks show some cross-modality transfer, the Lightbot Jr findings reiterate that researchers and educators need to be aware of the demands of transfer, and that it cannot be assumed that all pupils will be able to spontaneously transfer skills acquired by programming robots to visually- and/or conceptually-different screen-based programming environments later in their school career.

Similarly, it seems that the sequencing skills pupils needed to use to program the robot did not transfer to the picture sequencing task. Instead, all conditions improved on average in picture-sequencing. Thus, improvements in sequencing may arise naturally from other classroom activities and are not specific to programming (as suggested by the improvements across all conditions). These results diverge from findings from Kazakoff and Bers (2014), who found that approximately 4.5 h of one-to-one robot programming improved children's sequencing skills on a similar assessment. However, it is unclear whether this one-on-one instruction provided additional instructional support for generalising sequencing skills, which may not be feasible in classroom settings. Additionally, no comparisons were made to a control group. In the current study, sequencing was treated in the context of programming, but the curriculum did not elaborately support pupils to generalise this skill to unrelated contexts. It could thus be the case that transferring CT skills out of programming contexts does not occur spontaneously for all pupils and may require extra instructional support. This raises questions about whether improvements observed in previous studies truly reflect transferrable learning or are limited to task-specific contexts.

Many of the studies reviewed by Ching and Hsu (2024) measured sequencing only within programming tasks rather than in

separate transfer assessments. This suggests that previous studies' reported improvements in sequencing may reflect task-specific learning rather than transfer. If children improve sequencing within programming tasks but struggle to apply this skill in unrelated contexts, this reinforces the need for explicit instructional strategies to support transfer. Future research could explore whether structured small-group instruction or targeted interventions offer comparable benefits whilst remaining scalable.

Taken together, whereas CT skills are assumed to be important for everyone as they can theoretically be applied in a wide range of contexts (Wing, 2006), this study adds to mounting evidence that the generalisation of CT skills to other contexts may not be so straight forward (Scherer et al., 2019). Skills related to the Cubetto programming curriculum (i.e., debugging and prediction tasks) showed improvement, however, picture sequencing and Lightbot Jr programming tasks, which were less-directly connected to the Cubetto curriculum, showed general maturation effects across conditions. Future research should investigate how instructional support can help children transfer newly acquired programming skills to other domains. Furthermore, even if direct transfer does not occur, early exposure to programming may still offer valuable benefits, such as enhancing motivation and building confidence (Master et al., 2017), which could contribute to long-term learning.

4.3. Teachers' beliefs

With respect to teachers' beliefs, teachers in Intervention and Intervention+ improved their relevance beliefs across the intervention. However, only Intervention+ differed significantly from Control. Additionally, this study found that self-efficacy beliefs only significantly improved in Intervention+. This study did not find significant differences in improvements in enjoyment or reductions in anxiety scores across the groups, although positive descriptive trends were observed in the Intervention+ group. Prior research by Kim et al. (2015) and Bers et al. (2013) has similarly highlighted the importance of structured teacher education programmes in fostering positive beliefs in teaching programming with robots. In this study, Intervention showed positive trends in their relevance and self-efficacy beliefs across time, illustrating the value of hands-on teaching experiences. Furthermore, the current findings indicate that even a relatively short, 3-h teacher education workshop can significantly impact teachers' positive beliefs when paired with hands-on curriculum delivery. Thus, these findings emphasise the value of concise, targeted professional development sessions in educational settings and indicate that with targeted and hands-on early primary teacher education, a little time-investment can go a long way.

However, potential limitations of this study are the missing questionnaire responses and unequal distribution of teachers across conditions, with 5 teachers in Intervention+, 9 in Intervention, and 6 in Control. Although the small and uneven sample sizes in Intervention+ and Control may limit the generalisability of the findings, the significant positive outcomes observed in Intervention+ across teachers and pupils suggests that the brief teacher education workshop had a meaningful effect despite the unequal sample sizes.

4.4. Implications

4.4.1. (Inter)national context

This study provides actionable insights for embedding programming with robots into primary education, highlighting the importance of teacher education in achieving sustainable impact. Conducted in Wales, where such curriculum changes are currently underway (Hwb, 2024) and where there are vast differences in socio-economic prosperity (i.e. WIMD scores), this study addresses challenges faced globally as many countries adopt programming and CT in primary education (Balanskat & Engelhardt, 2015; Bers, 2020; Bocconi et al., 2022; Uzunboylu et al., 2017). Core principles such as hands-on learning and collaborative problem-solving resonate internationally, with the European Commission (Bocconi et al., 2022) highlighting the growing popularity of educational robots. The methods used in this study offer strategies for integrating robots (like Cubetto and possibly other robots that offer program visualisation) into early education.

4.4.2. Addressing shared barriers

Three key barriers identified by the European Commission (Bocconi et al., 2022) were addressed in this study: (1) Lack of Trained Teachers, (2) Competing Curriculum Priorities and (3) Assessment of CT/Programming Skills. This study delivered a concise teacher workshop and found it, alongside a classroom intervention, significantly improved relevance and self-efficacy beliefs, showing the potential impact of minimal but targeted professional development. To assist with challenges related to competing priorities, this study integrated robots into cross-curricular activities, thus maximising teachers' time and increasing exposure, making CT education more manageable. Finally, these quantitative, paper-based assessments offer practical, scalable tools to evaluate programming skills effectively.

4.4.3. Recommendations for teacher education

This study highlights recommendations for teacher education in the areas of CT, programming and robotics. Firstly, tailoring teacher education programs to include content developmentally appropriate for early primary school children can be beneficial for both teachers and pupils. In this study, the use of tailored activities for younger learners aimed to ensure accessibility and engagement during the intervention. Additionally, teacher education workshops should provide hands-on learning opportunities with robots to give teachers space to think about what would work best for them in practice (Hughes, 2024), and how they could tailor sessions to meet the needs of their pupils specifically. Adopting this approach in this study was important as teaching is not a "one size fits all" exercise. Instead, teachers deliver lessons in different ways depending on the abilities of their pupils and the types of resources and technologies available to them. This personalised approach aligns with the flexible, learner-centred New Curriculum for Wales, which encourages schools to design curricula reflecting their pupils' needs and interests (Hwb, 2024). More broadly, the findings of this study

suggest that with appropriate support, teachers can confidently integrate robots and programming into early primary education to benefit pupil's learning.

CRediT authorship contribution statement

Amy A.E. Hughes: Writing – review & editing, Writing – original draft, Visualization, Software, Resources, Project administration, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Sarah A. Gerson:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Johanna E. van Schaik:** Writing – review & editing, Supervision, Resources, Methodology, Funding acquisition, Conceptualization.

Ethical approval statement

Ethical approval was granted by the Cardiff University School of Psychology Ethics Committee (EC.21.September 11, 6434). Online opt-in consent was obtained from headteachers, teachers and parents/caregivers. Children provided verbal consent at the start of each testing session.

Funding statement

This research was funded by the Economic and Social Research Council, granted to Sarah A. Gerson and Johanna E. van Schaik, with in-kind contributions from Primo Toys/Moravia Education and Techniquest (reference 2267832).

Declaration of competing interest

The Cubetto robots used in this study were provided by Primo Toys/Moravia Education. Primo Toys/Moravia Education were not involved in any discussions regarding the study design nor played any role in the collection or analysis of the data; interpretation of the results; or writing of the manuscript.

Acknowledgements

We are grateful to the schools, teachers, and children who participated in this research. We also thank the BSc and MSc students – Lloyd, Alexandra, Chara, Georgie, Zoe, Matt, Ellie, and Jamie – for their support with data processing. Special thanks to Dr Dominic Guitard and Dr Kelsey Frewin for their statistical advice, and to Vicky Simmons for her invaluable help in collecting the large volume of data. We would also like to extend our gratitude to the reviewers for providing fruitful feedback. We appreciate the support of Primo Toys/Moravia Consulting for providing essential resources. This project was funded by the Economic and Social Research Council through the Doctoral Training Partnership.

Data availability

All data, analysis scripts and results, and example materials, are available on Open Science Framework: https://osf.io/xkgrq/overview?view_only=80d9a49536d2493da220a0099d0d23d7

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control, 19*(6), 716–723.

Angeli, C., & Valanides, N. (2020). Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in Human Behavior, 105*, Article 105954.

Bakala, E., Gerosa, A., Hourcade, J. P., & Tejera, G. (2021). Preschool children, robots, and computational thinking: A systematic review. *International Journal of Child-Computer Interaction, 29*, Article 100337.

Balanskat, A., & Engelhardt, K. (2015). Computing our future. Computer programming and coding. *Priorities, school curricula and initiatives across Europe*. Brussels: European Schoolnet.

Baron-Cohen, S., Leslie, A. M., & Frith, U. (1986). Mechanical, behavioural and intentional understanding of picture stories in autistic children. *British Journal of Developmental Psychology, 4*(2), 113–125.

Barr, D. J. (2013). Random effects structure for testing interactions in linear mixed-effects models. *Frontiers in Psychology, 4*, Article 54057.

Bates, D., Maechler, M., Bolker, B., Walker, S., Christensen, R. H. B., Singmann, H., … Bolker, M. B. (2015). Package 'lme4'. *Convergence, 12*(1), 2.

Bers, M. U. (2010). The TangibleK robotics program: Applied computational thinking for young children. *Early Childhood Research & Practice, 12*(2), Article n2.

Bers, M. U. (2020). *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Routledge.

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education, 72*, 145–157.

Bers, M. U., González-González, C., & Armas–Torres, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education, 138*, 130–145.

Bers, M., Seddighin, S., & Sullivan, A. (2013). Ready for robotics: Bringing together the T and E of STEM in early childhood teacher education. *Journal of Technology and Teacher Education, 21*(3), 355–377.

Bocconi, S., Inamorato dos Santos, A., Chioccariello, A., Cachia, R., Kampylis, P., Giannoutsou, N., Dagienė, V., Punie, Y., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V., & Stupurienė, G. (2022). In A. Inamorato dos Santos, R. Cachia, N. Giannoutsou, & Y. Punie (Eds.), *Reviewing computational thinking in compulsory education : State of play and practices from computing education*. Publications Office of the European Union. https://data.europa.eu/doi/10.2760/126955.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American educational research association, 1* p. 25). Vancouver, Canada.
Burke, L. A., & Hutchins, H. M. (2007). Training transfer: An integrative literature review. *Human Resource Development Review, 6*(3), 263–296.
Çetin, M., & Demircan, H.Ö. (2020). Empowering technology and engineering for STEM education through programming robots: A systematic literature review. *Early Child Development and Care, 190*(9), 1323–1335.
Ching, Y. H., & Hsu, Y. C. (2024). Educational robotics for developing computational thinking in young learners: A systematic review. *TechTrends, 68*(3), 423–434.
Clarke-Midura, J., Lee, V. R., Shumway, J. F., Silvis, D., Kozlowski, J. S., & Peterson, R. (2023). Designing formative assessments of early childhood computational thinking. *Early Childhood Research Quarterly, 65*, 68–80.
Faber, H. H., Koning, J. I., Wierdsma, M. D., Steenbeek, H. W., & Barendsen, E. (2019). Observing abstraction in young children solving algorithmic tasks. In *International conference on informatics in schools: Situation, evolution, and perspectives* (pp. 95–106). Cham: Springer International Publishing.
Girvan, C., Conneely, C., & Tangney, B. (2016). Extending experiential learning in teacher professional development. *Teaching and Teacher Education, 58*, 129–139.
González-González, C. S., Herrera-González, E., Moreno-Ruiz, L., Reyes-Alonso, N., Hernández-Morales, S., Guzmán-Franco, M. D., & Infante-Moro, A. (2019). Computational thinking and Down syndrome: An exploratory study using the KIBO robot. *Informatics, 6*(2), 25.
Grover, S., & Pea, R. (2018). Computational thinking: A competency whose time has come. *Computer Science Education: Perspectives on teaching and learning in school, 19*(1), 19–38.
Grund, S., Lüdtke, O., & Robitzsch, A. (2018). Multiple imputation of missing data for multilevel models: Simulations and recommendations. *Organizational Research Methods, 21*(1), 111–149.
Gustafsson, J. E. (2003). What do we know about effects of school resources on educational results? *Swedish Economic Policy Review, 10*(2), 77-77.
Hox, J., Moerbeek, M., & Van de Schoot, R. (2017). *Multilevel analysis: Techniques and applications*. Routledge.
Hughes, A. (2024). *Integrating educational robotics in primary school classrooms: Exploring teacher and pupil perspectives (doctoral dissertation*. Cardiff University.
Hwb. (2024). Descriptions of learning. *Science and Technology*. Retrieved from https://hwb.gov.wales/curriculum-for-wales/science-and-technology/descriptions-of-learning/.
Kallia, M., & Cutts, Q. (2023). Conceptual development in early-years computing education: A grounded cognition and action based conceptual framework. *Computer Science Education, 33*(4), 485–511.
Kazakoff, E., & Bers, M. (2012). Programming in a robotics context in the kindergarten classroom: The impact on sequencing skills. *Journal of Educational Multimedia and Hypermedia, 21*(4), 371–391.
Kazakoff, E. R., & Bers, M. U. (2014). Put your robot in, put your robot out: Sequencing through programming robots in early childhood. *Journal of Educational Computing Research, 50*(4), 553–573.
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*, 245–255.
Khanlari, A. (2016). Teachers' perceptions of the benefits and the challenges of integrating educational robots into primary/elementary curricula. *European Journal of Engineering Education, 41*(3), 320–330.
Kim, C., Kim, D., Yuan, J., Hill, R. B., Doshi, P., & Thai, C. N. (2015). Robotics to promote elementary education pre-service teachers' STEM engagement, learning, and teaching. *Computers & Education, 91*, 14–31.
Kim, K. R., & Seo, E. H. (2018). The relationship between teacher efficacy and students' academic achievement: A meta-analysis. *Social Behavior and Personality: An International Journal, 46*(4), 529–540.
Koepsell, T. D., Wagner, E. H., Cheadle, A. C., Patrick, D. L., Martin, D. C., Diehr, P. H., … Dey, L. J. (1992). Selected methodological issues in evaluating community-based health promotion and disease prevention programs. *Annual Review of Public Health, 13*(1), 31–57.
Lauermann, F., & ten Hagen, I. (2021). Do teachers' perceived teaching competence and self-efficacy affect students' academic outcomes? A closer look at student-reported classroom processes and outcomes. *Educational Psychologist, 56*(4), 265–282.
Lenth, R. (2021). Emmeans: Estimated marginal means, aka least-squares means. R package version 1.6. 0.
Maier, M. F., Greenfield, D. B., & Bulotsky-Shearer, R. J. (2013). Development and validation of a preschool teachers' attitudes and beliefs toward science teaching questionnaire. *Early Childhood Research Quarterly, 28*(2), 366–378.
Mangina, E., Psyrra, G., Screpanti, L., & Scaradozzi, D. (2023). *Robotics in the context of primary and pre-school education: A scoping review*. IEEE Transactions on Learning Technologies.
Master, A., Cheryan, S., Moscatelli, A., & Meltzoff, A. N. (2017). Programming experience promotes higher STEM motivation among first-grade girls. *Journal of experimental child psychology, 160*, 92–106.
Migliaresi, G. (2016). "A very special Montessori moment…" Cubetto passes the Montessori test!. September 6 https://www.primotoys.com/cubetto-featured-montessori-international/.
Moser, A., Zimmermann, L., Dickerson, K., Grenell, A., Barr, R., & Gerhardstein, P. (2015). They can interact, but can they learn? Toddlers' transfer learning from touchscreens and television. *Journal of Experimental Child Psychology, 137*, 137–155.
Ohashi, Y., Kumeno, F., Yamachi, H., & Tsujimura, Y. (2018). Readiness of Japanese elementary school teachers to begin computer-programming education. In *2018 IEEE international conference on teaching, assessment, and learning for engineering (TALE)* (pp. 807–810). IEEE.
Perkins, D. N., & Salomon, G. (1992). Transfer of learning. *International encyclopedia of education, 2*(2), 6452–6457.
Pugnali, A., Sullivan, A., & Bers, M. U. (2017). The impact of user interface on young children's computational thinking. *Journal of Information Technology Education: Innovations in Practice, 16*, 171.
Rapti, S., & Sapounidis, T. (2024). "Critical thinking, communication, collaboration, creativity in kindergarten with educational robotics": A scoping review (2012–2023). *Computers & Education, 210*, Article 104968.
Ray, B. B., Rogers, R. R., & Hocutt, M. M. (2020). Perceptions of non-STEM discipline teachers on coding as a teaching and learning tool: What are the possibilities? *Journal of Digital Learning in Teacher Education, 36*(1), 19–31.
Scherer, R., Siddiq, F., & Sánchez Viveros, B. (2019). The cognitive benefits of learning computer programming: A meta-analysis of transfer effects. *Journal of Educational Psychology, 111*(5), 764–792. https://doi.org/10.1037/edu0000314
Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*, 142–158.
Slangen, L., van Keulen, H., & Gravemeijer, K. (2011). What pupils can learn from working with robotic direct manipulation environments. *International Journal of Technology and Design Education, 21*, 449–469.
Strawhacker, A., & Bers, M. U. (2015). "I want my robot to look for food": Comparing Kindergartner's programming comprehension using tangible, graphic, and hybrid user interfaces. *International Journal of Technology and Design Education, 25*(3), 293–319.
Strawhacker, A., & Bers, M. U. (2019). What they learn when they learn coding: Investigating cognitive domains and computer programming knowledge in young children. *Educational Technology Research & Development, 67*, 541–575.
Sullivan, A., & Bers, M. U. (2016). Robotics in the early childhood classroom: Learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. *International Journal of Technology and Design Education, 26*, 3–20.
Sysło, M. M., & Kwiatkowska, A. B. (2015). Introducing a new computer science curriculum for all school levels in Poland. In *Informatics in schools. Curricula, competences, and competitions: 8th international conference on informatics in schools: Situation, evolution, and perspectives, ISSEP 2015, Ljubljana, Slovenia, September 28-October 1, 2015, proceedings 8* (pp. 141–154). Springer International Publishing.
Taylor, K., & Baek, Y. (2019). Grouping matters in computational robotic activities. *Computers in Human Behavior, 93*, 99–105.
Tselegkaridis, S., & Sapounidis, T. (2022). A systematic literature review on STEM research in early childhood. *STEM, robotics, mobile apps in early childhood and primary education: Technology to promote teaching and learning* (pp. 117–134).
Uzunboylu, H., Kınık, E., & Kanbul, S. (2017). An analysis of countries which have integrated coding into their curricula and the content analysis of academic studies on coding training in Turkey. *TEM Journal, 6*(4), 783.

van Aalderen-Smeets, S., & Walma van der Molen, J. (2013). Measuring primary teachers' attitudes toward teaching science: Development of the dimensions of attitude toward science (DAS) instrument. *International Journal of Science Education, 35*(4), 577–600.

Wang, K., Sang, G. Y., Huang, L. Z., Li, S. H., & Guo, J. W. (2023). The effectiveness of educational robots in improving learning outcomes: A meta-analysis. *Sustainability, 15*(5), 4637.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.

R Core Team (2021). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.