Programming Education to Preschoolers: Reflections and Observations from a Field Study

Alaaeddin Swidan and Felienne Hermans
Delft University of Technology
{Alaaeddin.swidan, F.f.j.hermans}@tudelft.nl

Abstract

In recent years, there has been a rise in methods and tools dedicated to programming education for children of primary school age. In this paper, we present our experience of providing five programming sessions to a group of eleven children between four and six years. Our sessions followed problem-solving and game-playing themes and featured two newly-developed tools: the unplugged Robot Turtles, and the robotic Ozobot. The activities embed programming concepts such as the order of operations, symbolic representations, and functional abstraction. The observations show that children understood and applied concepts such as sorting, sequential operations, and functional abstraction. However, children struggle with giving directions to the object which highlights a spatial awareness limitation. Finally, we link the observations to Piaget's theory and his limitations to thoughts for children in this age. We find that some of Piaget's limitations such as egocentrism can explain a few observed behaviors. However, a few limitations contradict our observations such as the irreversibility and transductive reasoning.

1. Introduction

Computational thinking education for children has been developing for decades (Papert, 1980; L. Morgado, Cruz, & Kahn, 2006). The aim is to make computer programming, a core aspect of computational thinking (Ambrosio, Almeida, Macedo, & Franco, 2014), widespread and more accessible. Many programming systems and languages were developed primarily for children as young as four, in schools and kindergartens (L. C. Morgado, 2005) The majority of these were computer aided (Kelleher & Pausch, 2005). For older children, Scratch has recently become a favored platform (Meerbaum-Salant, Armoni, & Ben-Ari, 2010; Hermans & Aivaloglou, 2017). Unplugged techniques, i.e., away from computers were also designed [1]. With these tools in place, the focus is still to a great extent on developing more systems and tools for children, and to incorporate them into school curriculum (Fessakis, Gouli, & Mavroudi, 2013). However, few studies investigate the children cognitive and social reactions to these educational tools (L. C. Morgado, 2005).

To that end, we provide programming education sessions to eleven children between four and six at a primary school in the Netherlands. We use both unplugged and robotics materials to teach concepts such as the order of operations, symbolic representation, and functional abstraction. By guiding the children throughout these sessions, we aim at observing their reactions while using these educational tools and what difficulties they face in completing the tasks. We then compare our observations to Piaget's theory (Piaget, 1964), one of the dominant theories in the psychology of educational development (Mitchell & Ziegler, 2013). Results show the children's ability to perform sorting and classification tasks, to order the operations sequentially and to use a basic level of abstraction. On the other hand, children have difficulties with spatial awareness needed to give directions to the objects within the games. For the teacher, it is recommended to establish an active learning environment with a high level of engagement with the children. Finally, we find that some Piaget's limitations to thought, such as egocentrism, can help in explaining some of the observations. In other cases there are opposing observations to a few limitations such as irreversibility and transductive reasoning.

[1] http://csunplugged.org/

2. Background: Preschoolers Education

In this section, we provide an overview of constructivism, one of the most established theories in educational and development psychology. We follow by highlighting a few aspects of Piaget's theory as a major influencer in constructivism.

2.1. Constructivism

In the psychology of learning, constructivism is one of the dominant theories (L. C. Morgado, 2005; Mitchell & Ziegler, 2013). Derived from the word construct, constructivists assert that students and learners acquire new knowledge by actively processing sensory data against their previous knowledge. Ben-Arie ((Ben-Ari, 1998) provides a comprehensive revision to constructivism and its application in education, with a focus on computer science. In summary, constructivism promotes an active learning environment combined with exploration and discovery activities. The role of the teacher is important to guide this process by designing the activities, and assisting the students especially when conflicts occur between the new data and previous knowledge. In his revision, Ben-Arie provides examples showing that constructivism is applied in mathematics and physics education, but fewer efforts in CS education.

2.2. Piaget's theory

There are a variety of theories under the umbrella of constructivism (L. C. Morgado, 2005). We highlight Piaget's theory (Piaget, 1964) as it is one of the major influences in constructivism. His theory focuses on the cognitive development of children. Piaget theory proposes four stages of cognitive development: sensori-motor, pre-operational, concrete operational and formal operational (Piaget, 1964). In his theory, new information directed towards the child is either assimilated according to the already existing cognitive models or accommodated by trying to build a new cognitive construct (McLeod, 2015). A state of equilibrium occurs in between these two processes where no new knowledge is required. Even though the child, through personal experience and social relations, is responsible for this mental process, the environment is still recognized as a playing factor in his theory (L. C. Morgado, 2005). The teacher as part of the environment is the motivator to the learning process by creating situations which cause the child to be in a disequilibrium state, and allow new cognitive constructs to be developed. The teacher's role expands to provide helping information, asking questions and making comparisons for example. In this paper, we investigate children in the preoperational stage of development, which Piaget defines between two and seven years. Children in this stage are expected to develop language, complete operations, solve one-step logic problems (McLeod, 2015). More complex logical thinking starts to develop only after the age of seven. During the preoperational stage, children suffer from many limitations to acquiring knowledge, which Piaget calls *"limitations to thoughts"* (L. C. Morgado, 2005). Table 1 summarizes these limitations in the preoperational stage.

3. Study Goal

In this paper, we describe our efforts to teach programming concepts to children between four and six years old. Our study aims first at exploring how children in this early age react to problem-solving activities in a programming context. Secondly, we target to observe difficulties that limit the children's ability to complete an activity or grasp a particular concept. These difficulties might be related to the internal mental structures already developed in children, or to the complexity of the concepts being taught. Finally, we want to compare the difficulty we observe to Piaget's limitations to thought. In summary, we want to answer questions such as:

- How do children react during these activities? For example: to what extent are they motivated and engaged? Do they enjoy the materials used?
- What limitations to thought from Piaget's theory on preoperational children still apply in activities related to programming education?
- What variations in performance are observed in concerning age difference?

Table 1 – Piaget's limitations to thought in the preoperational stage of development. The first three columns are taken from (L. C. Morgado, 2005)

Limitation	Description	Example	Category
Centration	The child focuses on a single aspect of the situation, disregarding any other.	John cries when his father gives him a biscuit broken in half. Since each half is smaller than the full biscuit, John thinks he's getting less.	**Agree**
Irreversibility	The child is unable to realize that an operation or action can be reversed.	John doesn't realize that both halves of the biscuit can be joined to make a full biscuit.	**Disagree**
Static thinking	The child is unable to realize the meaning of state transformations	In a conservation task, John fails to realize that the shape transformation of a liquid (from a glass to another) doesn't change the quantity.	**Not observed**
Transductive reasoning	The child doesn't employ either deduction or induction; going instead from one particular aspect to another, seeing a cause where there is none.	"I had ill thoughts about my brother.My brother got ill. So, I made him get ill." Or: "I misbehaved, so mum and dad divorced."	**Disagree**
Egocentrism	The child assumes that everyone thinks like he/she does.	Mary picks up a game and tells her mum, "This is your favorite." She's assuming that her mother likes the game as much as she does.	**Agree**
Animistic thinking	The child sees life in inanimate objects.	Mary thinks the clouds are alive because they're moving.	**Agree**
Inability to distinguish appearances from reality	The child mistakes appearances for reality.	John thinks that a sponge made to look like a rock is indeed a rock.	**Not observed**

4. Setup

To achieve the goals mentioned in the previous section, we provide five sessions to eleven children aged between four and six (five girls and six boys). The sessions include problem-solving activities in a game-playing theme. Our approach is influenced by the main principles of the constructivism: an active learning environment, with a guided exploration and immediate feedback. We follow an observational methodology where two to three supervisors guide and monitor the children's activities in each session. After the sessions, we discuss the observations and write the ones that are agreed upon by two supervisors. The sessions are held in a Dutch primary school. The children are part of an after-school club, and activities are performed in their daily classroom, a familiar and friendly environment. The children work in groups of two or three. When arranging the groups, we make different combinations in each session; sometimes based on gender or age differences and sometimes purely random.

5. Materials

We used both unplugged and robotic tools, in a game playing setup. Previous research showed the positive effect of game playing activities in the education of computational thinking and programming(U.S., 2011). In addition to the constant problem-solving theme, the exercises embed a variety of programming concepts, such as the order of operations, the symbolic representation, and functional abstraction. See Table 2 for the full list of the exercises, materials and their associated programming concepts. Following is a detailed description of the tools we use.

5.1. Sorting and Classification

This session was the first we did with the children. It was a session with light activities because we aimed at introducing ourselves to the children and getting to know each other. By doing this, we create a friendly environment where children feel free to act upon and express their thoughts. The session includes papers with printed animal pictures, and we ask the children to perform various sorting and classification activities. For example, to sort some animals based on size or color. Another example is to classify animals based on the habitation such as farm, wild and sea animals.

Table 2 – Summary of exercises performed during the sessions showing the programming concepts which the game and the exercises collectively serve

Session Order	Programming Environment	Exercise	Overview	Concepts
1	Printed animal pictures	Arranging cards based on some criteria	Sort the animals based on size.,Classify the animals based on their habitation.	Data sorting and classification
2 and 3	Robot Turtle	1. An Empty maze with diamonds in the center	Give directions to the turtle in relation to the diamond	Problem analysis Symbolic representations Order/sequence of operations Abstraction (functional) Fault isolation and debugging Team working (pair programming)
		2. A maze with obstacles	Implement a workaround based on the obstacle's type	
		3. A maze with obstacles #2: Full path constructed	Without moving the turtle, the child need to complete the whole,path.	
		4. A maze without obstacles, only with diamonds, with the need to use the frog card	Use the frog card to help the frog to reach the diamond faster	
4 and 5	Ozobot	Explore with the Ozobot	Explore the basic concepts of the Ozobot: sensor-based autonomy following a line, reflections of colors, and the behavior at line branches	Problem analysis Symbol interpretation What-if analysis Alternative branch concept (visual) Team working (pair programming)
		Use the color codes in two tracks, so that the Ozobot can get from home to school, and from home to shop	Choose the appropriate color codes	

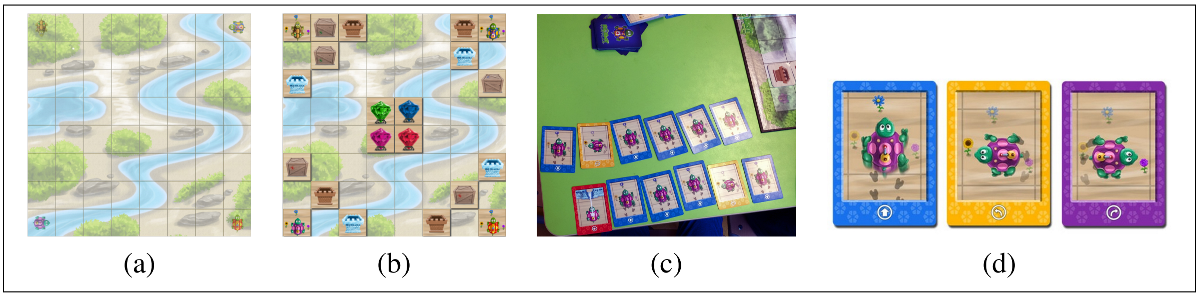
5.2. Robot Turtles

Robot Turtles is a board game that *"teaches programming to kids"*[2]. The board game, shown in Figure 1, was invented by Dan Shapiro, a software engineer. According to the game's website, it aims at allowing preschoolers to learn *"the fundamentals of programming while they are playing"*.

The game resembles a simple visual programming language, with the cards being ordered by the child player to direct the turtle through obstacles until it reaches the diamond. The adult who demonstrates the computer processor moves the turtle depending on the order of the cards. The game, as a result, features principles of simplicity, visualization, simulation and autonomy, which are recommended when considering children and novice programmers (Du Boulay, O'Shea, & Monk, 1999; Jenkins, 2002). In addition to the basic direction cards, the game has some special cards. The **laser card** melts down the ice obstacle, the **frog card** abstracts a function that can help the turtle performs repeatable actions, and the **bug card** by which a player breaks the execution and notifies the adult that a problem exists within the sequence of cards. Looking back at programming concepts, we notice that the Robot Turtles game focuses on some of them. In particular, the order of operations (through the sequence of cards), function abstraction (through the frog card), debugging (through the bug card) and code refactoring/optimization (through using shorter paths i.e., less amount of cards). We used the Robot Turtles for two sessions, during which we gradually introduced more game features and programming concepts. We additionally increased the difficulty of the maze the turtle has to solve. In total, we had four exercises where the children performed the tasks on two copies of the game in parallel. See Table 2 for an overview of the exercises.

We introduce the game to the whole group: describe the basic cards with their action. We then work

[2] http://www.robotturtles.com/



(a) (b) (c) (d)

Figure 1 – The Robot Turtle game: (a) an empty board (b) a board with a maze (c) one solution to a Robot Turtle exercise in the classroom (d) the basic movement cards

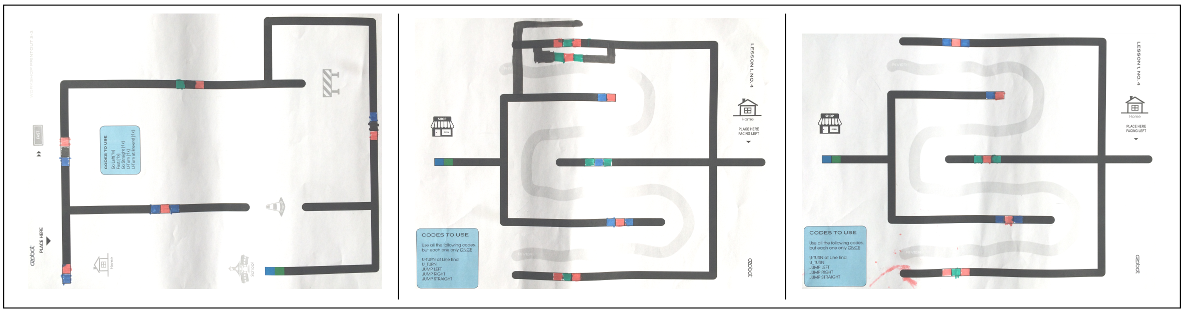


Figure 2 – Ozobot exercises performed by children throughout the sessions

closely with the children to repeat some information. We perform the same collective description for each new exercise; describing the goal and introducing new cards. Within the same exercise, the difficulty level was unified: children have the same mazes. We do not know in advance what are the limitations per child; in fact, this is an issue we want to observe.

5.3. Ozobot

The Ozobot [3] is *"a miniature smart robot that can follow lines or roam around freely, detect colors, and can also be programmed"* [4]. The Ozobot can be programmed through a set of color codes, to perform a variety of actions: speed, direction, and funny moves. The Ozobot is designed to randomly choose directions when facing a line branch or a junction. According to the Ozobot website, the Ozobot empowers the STEM (Science, Technology, Engineering, and Mathematics) education, and can be used to teach subjects such as programming. Regarding age, the Ozobot is designed for all levels of the elementary school. For younger children, it is not advised without the accompany of an adult, primarily because of its small size. Similar to other studies which used robotics in programming education (Magnenat, Shin, Riedo, Siegwart, & Ben-Ari, 2014; Kaloti-Hallak, Armoni, & Ben-Ari, 2015), Ozobot can be used to develop certain programming concepts. These concepts include the analytical problem solving, sequential programming skills, and understanding of computer machine autonomy. See Table 2 for the concepts associated with the Ozobot. We used five Ozobots in two sessions. Each Ozobot was controlled by two to three children. We used the Ozobot Basic Training lessons and resources provided by Ozobot [5]. The first session was more exploratory to the features and nature of the Ozobot. Children learned more about its tracking the lines and colors, but also about the randomness and indeterministic decision at a junction. Also, they learned about the color codes, and they explored their possibilities in movement, speed, and directions. In the second session, the children were asked to solve two exercises which require choosing specific color codes in order to get the Ozobot from a source to a destination. Figure 2 shows three solved exercises stemmed by children in our sessions.

[3] http://ozobot.com/

[4] https://education.microsoft.com/Story/Lesson?token=qgSYB

[5] http://portal.ozobot.com/lessons

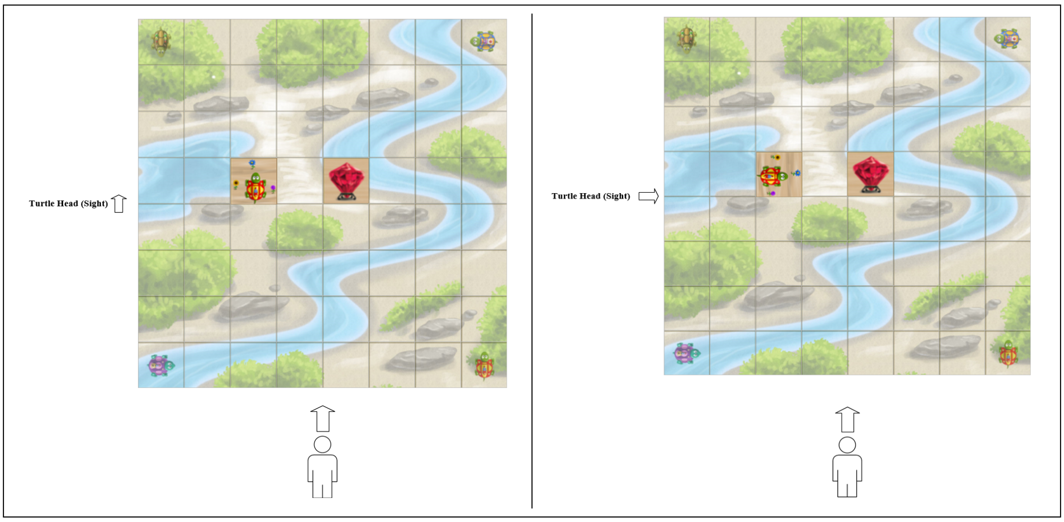


Figure 3 – Spatial reasoning arises when a child has a different view than the controlled object

6. Observations

In this section, we list six observations: behaviors of the children during the sessions. For a behavior to qualify into an observation, it needs to be demonstrated by at least two children and observed by two of the supervisors.

Observation 1: Children suffer when giving directions especially with different viewpoints
One of the most recurrent mistakes in both games is the choice of a wrong direction to the turtle or the Ozobot. Typically, the direction decision involves two objects: where the child is looking initially, and where to turn/look finally. However, in these exercises, the moving objects were not the child; they were the turtle and the Ozobot. Figure 3 shows two different boards where the turtle has a similar and a different viewpoint. This behavior is associated with a computational cognitive skill known as spatial reasoning, and it defines *"the ability to recognize and view figures, and the ability to rotate or follow the movements of figures"*(Ambrosio et al., 2014). In their study, they considered it as one of *"the crucial dimensions for introductory programming"*.

Observation 2: Children get derailed from a task by fantasies they connect to
Children find themselves immediately drawn to joys and fantasies familiar to them found in these games. For example, quite a few children (mostly boys) felt very enthusiastic when we introduced the laser card in the robot turtles exercises. They started to point it to each other making a sound imitating the laser hitting other objects. One boy later in the exercise asked if he can put the laser card before reaching the ice obstacle because the laser can transfer and melt from a distance. In the Ozobot exercises, it was more tempting for children to get away from the original task because of the many temptations involved: robots, funny and fast moves, and coloring. Almost all of the children were amazed when we introduced the Ozobot. Some kept the amusement for the overall two sessions. They enjoyed the idea of having a moving robot, and tried to control it by voice commands like *"move"* or *"go left"* and ordering it to be colored. Some children got their hands on the coloring markers quickly and immediately started coloring on their own. When it comes to color codes, many children mostly boys wanted the Ozobot to go fast and make funny moves all the time regardless of the problem in question.

Observation 3: Immediate execution helps children better grasp game's rules and identify faults
The immediate feedback of the computer (the adult in Robot Turtles and the Ozobot) was essential for children to understand the symbolic meaning behind the games' material. In the case of Robot Turtles, we simulated sounds, when the turtle moves, to send confirming messages to the children about the meaning of some actions and obstacles. For example, the turtle can push a box, but not a wall. Even though we told the children about the meaning verbally, some children made mistakes and tried to move

forward through a wall. By making a special sound and visual appearance of difficulty to move a wall, children made fewer faults in the next tasks. The exercise of the full path creation at once showed that the performance, when the turtle starts to move, motivates the child to anticipate the fault in the card ahead. As soon as the turtle moves, we can see that some of the children detect the error just before the adult executes the subsequent action. They then try to intervene and fix it with the correct card. Adding the right card, however, did not cause the children to adjust the sequence of cards which follow after the fault. They again waited for the simulation of the turtle and then react, and so on.

Observation 4: Children comprehend functional abstraction in Robot Turtles faster than expected
We introduced the concept of the frog card as a helper to the turtle, to move faster through jumping some ordered actions. We tried to be very clear in the explanation, giving them examples. We found the children to be reasonably fast in understanding the application of a frog (function) card in the main sequence. Having said so, we highlight some concerning observations in this area. All of the children chose to apply a single-type operation for the frog function, which is the moving forward card. They wanted the frog to move the turtle two or three steps ahead. In response, we provided thought-provoking questions after they completed the task like *"what other cards you could have used"*. Another concern is that some children placed the cards for the frog i.e., defined the function, but they did not use it within the main sequence of cards. When notified by the supervisor *"Oh I don't see a frog!"* the children eventually placed the card, only in one location in the main sequence. We provided them with additional suggestions, and only then few children got the idea that another place in the main sequence is possible for a second frog card call. Finally, we observed some variances in the speed of getting the idea and the quality of its application among the children. These differences relate primarily to the age group of four years old and will be discussed later in Section 7.

Observation 5: Adult-supported and active learning environment is a necessity
Limiting the role of the adult in such experiments to the activity designer, and imitator of knowledge does not suit the young children. These children have different interpretations to a single information outspoken by adults. They might not understand it or might accommodate it with the incomplete knowledge they already have. Thus it is important for adults to discuss individually or in small groups of the children what they think about a particular aspect of a task or a concept. We often did this by asking questions in the form of *"What do you think… "*, *"Why did the object behave in a specific way… "* and *"What if you did this action? what will happen?"* The environment of the classroom was an additional support to children: it is their daily place to learn and play, they feel very confident and relaxed moving around and using available resources. We saw little influence to learning from peers. Time was not sufficient for one group to be more knowledgeable than the other group. In one occasion a child who finished his robot turtle exercise tried to give his peer group the answer to their next card. Unfortunately this was the wrong direction card, and fortunately, the recipient boy thought for a second and chose to stick to his original card. The influence, in this case, would have been adverse to the solution of the problem. We suspect this is because the intervening child lacked the proper spatial awareness, and so he rushed with the wrong card.

Observation 6: Children often do not express their sense of difficulty or lack of understanding
Children will go directly into action after explaining the task. In one case only we saw a six years old girl approaching us saying *"what do you mean by this"* in referring to the final Ozobot exercise. This question came after explaining the task to all children. In another area, children seem to prematurely assess the difficulty of a game. In one case, a boy of six years old, when introduced with the Robot Turtle exercise in the second session shouted *"oh no, not this game again. It is so easy"*. However, this boy took the longest time to complete the exercise which followed. In fact, other children said the word easy so often when playing with the Robot Turtle, combined with their fingers naturally pointing the route the turtle should follow to reach the diamond. However, their route building procedure using cards usually did not come as smooth as the path pointing. We cannot tell the exact reason. Research show that young children tend to prematurely evaluate the explanation of an issue based on incomplete aspects (U.S., 2011). In our case, this means that their mind considers the pointing as the challenge, and

then judge the easiness of the game based on it. It can be other things such as a mechanism to avoid embarrassment or showing off among peers. It can be that these children did not fully understand how to build the routes by cards, despite being able mentally to point the route.

7. Observation Analysis

In this section, we provide an analysis of the observations by looking back into Piaget's theory in general, and Piaget's limitations of thought for preoperational children in particular. We classify the limitations depending on whether it is spotted in one of the observations or not. Additionally, this classification includes other core aspects of Piaget's theory which we believe worth mentioning.

7.1. Piaget's theory in agreement with the observations

7.1.1. Egocentrism

Piaget's definition of egocentrism is not about improper social behavior. It primarily focuses on the child's ability to considering the perspectives of others (L. C. Morgado, 2005). He showed, using the three mountains experiment (Mitchell & Ziegler, 2013), that children fail to recognize the viewpoint of an object different than their own. The egocentrism of the children in this age is still an aspect the educators need to consider. In our sessions, wrong direction choices were the most frequent mistakes. In the Robot Turtle, the child has an extra support by colors and flower painting on the cards. Each move is associated with a color, as shown in Figure 1d: blue card is forward, the yellow card is left, and the purple card is right. Also, each card has three different flowers matching the color scheme mentioned above. Despite this kind of visual support, children still suffered in spatial reasoning. The problem is observed more when the child perspective is different than the perspective of the turtle or the Ozobot (see Figure 3). The child will choose the card that agrees with his position in relation to the goal (the diamond or the branch respectively).

7.1.2. Animistic Thinking

According to Piaget, the child in this age gives life to non-living objects (L. C. Morgado, 2005). Observation 2 includes behaviors in common with this limitation, as some fantasies the children brought during the sessions fit under this definition. However, this is not necessarily a limitation towards the cognitive development. In some cases, it is a part of the game theme of the activity, such as making the laser sound and playing with each other using the laser card. While in the case of giving voice orders to the Ozobot, it can be considered as a challenge to the child's ability to understand the nature of forces controlling the Ozobot movement, and then his ability to apply color codes in the process. It is worth noting that the Ozobot has some similarities with living objects, especially the moving. However, children are also affected by previous knowledge about robots. To handle distractions caused by this limitation, we suggest to allow free time playing with the materials, making it clear that afterward attention shall be given to the exercises.

7.1.3. Centration

Centration is defined as the preoperational child tendency to focus on a single aspect of the problem disregarding the other aspects. Observation 3 includes behaviors that can explained by this definition. For example, the children were more engaged and reacted faster when the turtle was moving, as they introduced the next move's card quickly. On the other hand, building the full path required an extra step of imagination from the child. Most of the children committed more mistakes in the full path part than when the turtle moved step by step.

7.2. Piaget's limitations in disagreement

7.2.1. Irreversibility

As stated in Table1, the irreversibility limitation in preoperational children describes the inability to understand that a particular operation or action is reversible. This kind of behavior was not observed during our sessions. We found the children motivated to amend any fault immediately, sometime in runtime. As mentioned in Observation 3, children start to put correct cards instead of the wrong ones when the turtle starts moving in an unintended direction. In Ozobot, less reversible actions were possible because of the permanent coloring. However, some children managed to create extra branches for the

Ozobot in order to overcome one wrong color code, see Figure 2. Nevertheless, this behavior was limited and occurred following a hint from the supervisor.

7.2.2. Transductive reasoning

Piaget defined it as the child's inability to deduce or induce causes and effects correctly (see Table 1). We found opposing observations to this limitation in our sessions. We note, however, the careful and precise explanations we provided to the children prior to making the reasoning. One example from the Ozobot is related to the branch decision. The ozobot would choose a direction randomly when no color code is provided ahead of the branch. First, we asked the children: *"Where do you think the ozobot goes on a branch?"* We received rushing contradicting answers *"always red"*, *"always left"* or *"always straight"*. We then let them explore with two branches where we ask them where do you think the Ozobot will go before it moves. Children observed how the Ozobot is not predictable: sometimes they get it right by chance, and many other times they do not. After a few runs with these exercises, we asked the first question again. We got a strange moment of silence, followed by a four-year-old saying exactly what we wanted to hear *"We do not know!"*. As long as we do not tell the Ozobot, by color codes, where to go, we do not know its decision for sure. This is aligned with findings of (Du Boulay et al., 1999; Fay & Klahr, 1996) who described this as the indeterminate problem. Their results concluded that children in this age should be able to distinguish between deterministic and indeterministic situations following some exercises.

7.2.3. Abstraction, sorting, and classification

Piaget theory believes that preoperational children are unable to develop sorting, classification and abstraction thinking until the following stages: the concrete operational and formal operational stages. Children in our case showed a high level of skill in sorting animals based on attributes as size, color, and habitation. While in the Robot Turtle's frog card, children grasped the concept of the helping card rather quickly. Before the exercise, we presented the frog card usage as a helper to the turtle and gave examples on how to use it. The application of the frog card into the main sequence, however, was limited. Only one type of movement cards was used in building the functions, the forward card, and it was called in one place in the main sequence.

7.3. Age factor

Despite having a group who are very close in age, we noticed a variation between four and six-year-old children. From what we observed, this difference is primarily related to their focus span being shorter than older children. As a result, four years old children had more tendency to lose track of the task, and get easily distracted by other joyful activities like drawing and coloring. Besides affecting their level of understanding the concepts presented, this sometimes negatively affected their teammates in the group.

7.4. Motivation

Overall the children were motivated and engaged in the activities. This motivation was shown during the free time given to them just before leaving. In the robot turtles session, they engaged in building a maze for each team trying to challenge each other. For the Ozobot they created by drawing and coloring their routes and problems for the Ozobot to overcome. Also, they expressed to their parents the wish to continue playing these games at home.

8. Conclusions

In this paper, we report our experience to provide five programming lessons to children between four and six. We present six observations related to the children reaction and understanding of the programming concepts embedded in the activities. Among these observations, we highlight the children ability to perform sorting and classification tasks, to order the operations sequentially and to use a basic level of functional abstraction. On the other hand, children had difficulties with spatial awareness needed to give directions to the objects within the games. We compared our observations to Piaget's limitations to thought and showed the egocentrism helps in explaining the spatial awareness observations, while transductive reasoning and irreversibility were opposed by our observations.

9. Acknowledgements

We would like to thank Martine Noesen on her efforts in manging the children and co-supervising during the sessions.

10. References

Ambrosio, A. P., Almeida, L. S., Macedo, J., & Franco, A. H. R. (2014). Exploring core cognitive skills of computational thinking. In *Psychology of programming interest group annual conference (ppig) 2014.*

Ben-Ari, M. (1998). Constructivism in computer science education. *ACM SIGCSE Bulletin*, *30*(1), 257-261. doi: 10.1145/274790.274308

Du Boulay, B., O'Shea, T., & Monk, J. (1999). The black box inside the glass box: presenting computing concepts to novices. *International Journal of Human-Computer Studies*, *51*(2), 265-277. doi: 10.1006/ijhc.1981.0309

Fay, A. L., & Klahr, D. (1996). Knowing about guessing and guessing about knowing: Preschoolers' understanding of indeterminacy. *Child Development*, *67*(2), 689. doi: 10.2307/1131841

Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5-6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, *63*, 87-97. doi: 10.1016/j.compedu.2012.11.016

Hermans, F., & Aivaloglou, E. (2017). Teaching software engineering principles to k-12 students: A mooc on scratch. In *Tud-serg-2017-003. issn 1872-5392.*

Jenkins, T. (2002). On the difficulty of learning to program. In *Proceedings of the 3rd annual conference of the ltsn centre for information and computer sciences* (Vol. 4, pp. 53–58).

Kaloti-Hallak, F., Armoni, M., & Ben-Ari, M. M. (2015). Students' attitudes and motivation during robotics activities. *Proceedings of the Workshop in Primary and Secondary Computing Education - WiPSCE '15*. doi: 10.1145/2818314.2818317

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys*, *37*(2), 83-137. doi: 10.1145/1089733.1089734

Magnenat, S., Shin, J., Riedo, F., Siegwart, R., & Ben-Ari, M. (2014). Teaching a core cs concept through robotics. *Proceedings of the 2014 conference on Innovation & technology in computer science education - ITiCSE '14*. doi: 10.1145/2591708.2591714

McLeod, S. (2015). *Preoperational stage.* Retrieved from `Retrievedfromhttps://www.simplypsychology.org/preoperational.html`

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. M. (2010). Learning computer science concepts with scratch. *Proceedings of the Sixth international workshop on Computing education research - ICER '10*. doi: 10.1145/1839594.1839607

Mitchell, P., & Ziegler, F. (2013). *Fundamentals of developmental psychology* (1st ed.). Taylor and Francis.

Morgado, L., Cruz, M., & Kahn, K. (2006). Radia perlman: A pioneer of young children computer programming. *Current developments in technology-assisted education. Proceedings of m-ICTE*, 1903–1908.

Morgado, L. C. (2005). Framework for computer programming in preschool and kindergarten. *Ph.D Thesis*. Retrieved from `http://hdl.handle.net/10348/5344`

Papert, S. (1980). *Mindstorms* (1st ed.). Basic Books.

Piaget, J. (1964). Part i: Cognitive development in children: Piaget development and learning. *Journal of Research in Science Teaching*, *2*(3), 176-186. doi: 10.1002/tea.3660020306

U.S., N. R. C. (2011). *Report of a workshop of pedagogical aspects of computational thinking (computational thinking)* (1st ed.). National Academies Press.