

Learning Trajectories Visualizations of Students Data on the Computational Thinking Context

José Michel Fogaça Vieira¹, Luciana A. M. Zaina²

¹Federal University of São Carlos
Campus Sorocaba – João Leme dos Santos Road, Km 110 – SP-264
– 18052780 Sorocaba, SP – Brazil

²Department of Computer Science - Federal University of São Carlos
Campus Sorocaba – João Leme dos Santos Road, Km 110 – SP-264
– 18052780 Sorocaba, SP – Brazil

`jmichelfv@gmail.com, lzaina@ufscar.br`

Abstract. *Learning trajectories are paths that students may follow in order to achieve learning goals. The visualization of learning trajectories of students can support teachers in tracking students evolution and identify difficulties. We propose visualizations of learning trajectories in a new and interactive way, representing different concepts of computational thinking and learning goals in concise or detailed manner, according to interactions of the user. To evaluate our proposal, we chose to represent a series of exercises found in code.org, a free and well known platform that introduces and exercises computational thinking through visual programming. These visualizations were evaluated by 20 elementary school teachers in usability perspective.*

1. Introduction

Learning trajectories are possible paths that students may follow in order to achieve their learning goals [Fortenbacher et al. 2013, Cai 2018]. These trajectories can be obtained from the data analysis of the students interaction with learning tools in general or Virtual Learning Environments (VLEs), where they are commonly approached [Tafner et al. 2012]. Recently, learning trajectories have been applied to investigating the computational thinking (CT) learning process [Rich et al. 2017]. Computational thinking is taking an approach to solving problems, designing systems and understanding human behavior that draws on concepts fundamental to computing [Wing 2008]. CT has been introduced in the elementary school curriculum in India, United States, Germany, Greece and several other countries around the world [Sáez-López et al. 2016, Rich et al. 2017]. While some authors used visualizations to represent concepts and learning goals of CT [Rich et al. 2017, Rich et al. 2018], in general papers do not approach visualization aspects of learning trajectories [de Borba et al. 2016].

Information Visualization has applications in several of different areas (e.g. e-health, e-learning). In the educational field it can ease the process of obtaining and understanding information through visual analysis of data sets by users with the help of computational resources [Nascimento and Ferreira 2005, da Silva 2014]. Visualizations can be defined as the process of transforming abstract data in images in such a way that they can be viewed and interpreted by human beings [Nascimento and Ferreira 2005]. One vital

yet often neglected part of any visualizations is the interaction, which and can greatly increase the capacity of the visualizations to convey information [Yi et al. 2007]. Visualizations can be useful in the educational area by helping monitor the learning process and represent the learning trajectories followed by students [Berland et al. 2013].

Considering the discussion above, we propose three interactive visualizations to represent learning trajectories linked to the context of computational thinking, each focusing on an different aspect of representing the students' learning trajectories: complete (Complete Trajectory), activity focused (Single Activity Trajectory) and general visualization (Condensed Trajectory). We evaluated our proposals with 20 teachers from elementary schools of Brazil from the perspectives of usability. Our evaluation looked at the teachers understanding of the proposed visualizations. The contributions of our work focus on the visualizations which allow the teachers to look at learning trajectories in three different ways and help them to gain insight on the learning process. Our proposal also provides visualization formats that can be included in tools that work with CT, since they can visually represent CT programming exercises, provided that the data is properly organized.

2. Fundamentals

Students' interactions can be used to gather data and construct learning trajectories or virtual learning paths [Rich et al. 2017, Rich et al. 2018]. Learning trajectories are possible paths students may follow during the learning process [Fortenbacher et al. 2013, Cai 2018]. The trajectories are dynamic and can be very flexible, so there are multiple ways leading to the learning process [Tafner et al. 2012, Ramos et al. 2015]. With support of the learning trajectories and their respective visualizations, education professionals (e.g teachers, educational program creators) may have a clearer view of the progress and trajectory followed by the students [Cai 2018].

Computational thinking (CT) is defined by Wing [Wing 2008] as an approach to solving problems, designing systems and understanding human behavior drawing on concepts fundamental to computing. It is the capacity of analyze, systematize, represent, and solve problems and is often seen as an important competence required in order to adapt to the future [Raabe et al. 2017]. Due to its importance, different approaches were created to offer support in teaching CT. Programming exercises based on different approaches can be adopted as CT teaching strategy, employing, for example, connecting graphic elements instead of the traditional textual programming languages. In such approaches, a program is designed by reusable blocks where the shape and color make it easier for users to identify their functions and whether they can fit together or not [Sáez-López et al. 2016]. *Code.org*¹ and *Scratch*² are instances of such platforms providing resources to teach CT from code programming while allowing teachers to collect data on the learning process.

A visualization can be considered as an instance of a graphical representation of a certain data set aiming to provide visual representations of data sets designed to help people carry out tasks more effectively [Mazza 2009, da Silva 2014]. Interaction can be considered vital part of any visualization, as it can greatly multiply the capacity of the visualizations by adding to it a dynamic aspect absent in mere static images [Yi et al. 2007].

¹<https://studio.code.org/home>

²<https://scratch.mit.edu>

3. Related Work

Seiter and Foreman [Seiter and Foreman 2013] introduce the Progression of Early Computational Thinking (PECT) Model, a framework for understanding and assessing computational thinking in the primary grades. Grover and Pea [Grover and Pea 2013] article frames the current state of discourse on computational thinking in K–12 education by examining mostly recently published academic literature using Wing’s article as a springboard, identifying gaps in research and articulating priorities for future inquiries

Learning trajectories are approached by Yang et al. [Yang et al. 2015] using the *Scratch* tool. They measure amount learned, speed of learning and previous knowledge potential and also the vocabulary increase of 3852 users. In a systematic literature review Ramos et al. [Ramos et al. 2015] investigate the learning trajectories representation on e-learning systems. Rich et al. [Rich et al. 2017] present an analysis of 108 papers, categorizing and clustering learning goals into learning trajectories. Rich et al. [Rich et al. 2018] also presents learning trajectories used for decomposition in a context of computational thinking and computer science. Borba et al. [de Borba et al. 2016] analyses 25 different articles in the English language and tackles the several research questions, concluding that in general no visualizations of the learning trajectories are generated. Carmo, Gasparini and Oliveira [Carmo et al. 2019] use learning trajectories to analyze students navigational patterns.

Some works in the literature [Keim 2002, da Silva 2014, Nascimento and Ferreira 2005, Fohu et al. 2012, Yi et al. 2007] approach visualization and interaction while others approach learning trajectories [Rich et al. 2017, Rich et al. 2018, Ramos et al. 2015, de Borba et al. 2016, Carmo et al. 2019] however, the visual aspects of learning trajectories are not approached in majority of papers [de Borba et al. 2016]. This work aims to fill such gap and differs from the related papers in the visual approach of the learning trajectories, linking data collected from CT teaching into newly generated visualizations of learning trajectories.

4. Learning Trajectories Visualizations

As data visualization is tightly coupled to the context from which the data comes, we decided to analyze data types available in Code.org (due to having access of readily available data from previous works [de Melo et al. 2018]), a well-known platform commonly used to teach computational thinking. Code.org offers an environment where the students can construct solutions using code programming with reusable blocks. It provides lessons based on playful themes (e.g. Start Wars, Angry Birds) where each lesson is composed by a set of activities with different difficulty levels. Our proposal can also be extended to other platforms provided the data can be presented in a similar manner as Code.org platform (e.g time stamps of attempts, what code or block was sent, student identifier) in order to compose the learning trajectories. We proposed three forms of learning trajectories visualizations: *Complete Trajectories*, *Single Activity Trajectories*, and *Condensed Trajectories*. Here we provide a general view of how the visualizations occupy the *visualization spaces*, while explaining each visualization in detail in further paragraphs.

The three visualizations data is distributed in *visualization spaces*). Figure 1 is divided into a diagrammatic view of the visualization spaces of our proposal. The *visualization space (1)* presents the area where the *learning trajectories* are displayed where

teachers can visually explore the activities visited by the students and their progress as the students worked on the learning of computational thinking concepts. The *visualization spaces* (2), (3), and (4) serve as auxiliary interaction spaces that allow teachers to control and choose which student, activity and concepts will be shown on the *visualization space* (1). The *visualization space* (2) contains all the activities and allows the teachers to choose the activities. The *visualization space* (3) allows teachers to visually connect CT concepts with activities and filter CT concepts. In this space we showed the concepts of computational thinking which are categorized in CT concepts presented in the paragraph below. The *visualization space* (4) controls the input of data on the learning trajectories through the interaction of the user, the action of bringing a student chart over the other ones, the action of adding one student, filter, hide or show students and delete added students.

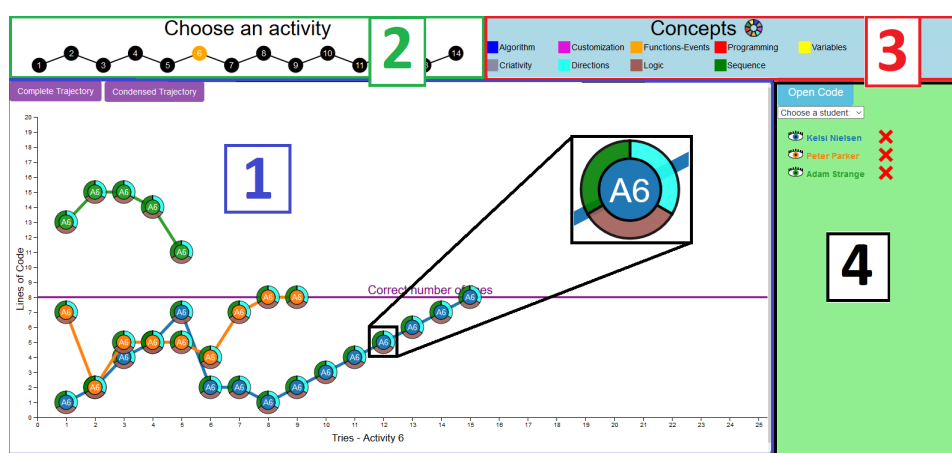


Figure 1. Diagrammatic view of the *visualization spaces* - Example of *visualization spaces* from *Single Activity Trajectory* of activity 6 of 3 students at the *visualization space* (1)

To construct the visualizations we used a data set gathered during a study³ with 48 students from two elementary schools in Brazil [de Melo et al. 2018]. The students performed a set of *Star Wars*⁴ themed activities in the *Code.org* platform. This process generated a database containing the code submitted by the participants, participant name, time stamp of each submission, and the activity the code belonged to along with the code that corresponded to the correct answer to each of the activities. In our previous work, the data had associated with each activity some CT concepts such as *algorithm*, *function-events*, *logic*, *programming*, *sequence*, *variables* along with other concepts related to the activities such as *creativity*, *customization* and *directions*, categorized according to the descriptions provided by *Barefoot*⁵, an online platform containing lessons, online guides, computational thinking workshops and programming workshops. The proposed visualizations make use of colors, color changes and tooltips triggered by user interactions to link which concepts are used in each activity. The CT concepts are represented as a segmented ring around each vertex of the visualizations as shown in the zoomed portion of Figure 1. All the students' names we used in our representations are fictitious, keeping

³This study was previously conducted by our research group [de Melo et al. 2018].

⁴<https://studio.code.org/s/starwarsblocks>

⁵<https://barefootcas.org.uk>

the real students anonymous.

Taking into account the interactive aspect of visualizations, our proposal comprises three layers: *representational*, *navigation* and *exploration*. These layers are independent from data origin, data types or platform. They were created to prevent visual overwhelming of the user, as the details of data are presented only as the user requests them. For each *visualization space* (see Figure 1), we implemented the features of these layers. In the *representational layer*, data is displayed through visual elements like charts and icons. The *navigation layers* covers functionalities from which teachers can interact with the visualizations in a more shallow way without exploring data in-depth. For instance, from this layer teachers can see summary of student's data through tooltips when *navigating* the learning trajectories. Finally, the *exploration layer* requires more actions of the teachers to access more detailed data and define which data will be shown (i.e., add, exclude or show data from the visualizations). By interacting with this layer, for example, teachers can access the code written by students along with the right answer to each activity. In a desktop environment with interaction from mouse, the *navigation* and *exploration* layer could be implemented by using resources of "mouse hover" and "mouse click" respectively. In the same way, in a touch interface, these layers could apply the resources of "single tap" and "double tap" respectively. We illustrate in two videos how the *navigation*⁶ and *exploration layer*⁷ work. We built functional prototypes of the three visualizations using the *visualization spaces* and the data collected from *Code.org* on the *star wars* activities.

The *Complete Trajectory* (Figure 2a) allows teachers to look at the entire learning trajectory of the students in the course of the activities. It contains all of the available data, and is the most complete of the three visualizations, providing a thorough view of the general path each student followed and allowing the teachers to analyze the differences and similarities between the trajectories created by the students learning process. It is represented by a graph inside a scatter plot. We chose the graph shape based on the works of Ramos et al. [Ramos et al. 2015] which concluded that graphs are the most common representation of learning trajectories. Each vertex corresponds to one *attempt* made by the student while trying to solve the activity denoted in each vertex label. We consider as an *attempt* of completing the activity the act of a student piecing together a sequence of lines of code and clicking the "Run" button on the *Code.org* platform. The code of each *attempt* can be accessed by interaction with the vertices (Figure 2b). Each vertex of the graph has a label "AN" which denotes the "Activity ". We adopted this labeling way by following Simon's [Simon 1995] view of learning trajectories. "A1" stands for "Activity 1", "A2" is "Activity 2" and so on. Each vertex is linked by an edge to give the idea of connection between the attempts the student did, creating a long linear graph. This graph is also put inside a scatter plot chart, so it uses the axis *x* and *y* to convey more information to teachers. The *y* axis represents how many lines of code the students used while trying to solve an activity while the *x* axis shows the number of times the student attempted to solve an activity. Around each vertex, there is also a donut chart containing all the concepts the activities aimed to teach. The representation of the CT concepts on the vertices was inspired on the works of Rich et al. [Rich et al. 2017] where the vertices

⁶Example of the navigation layer: <https://bit.ly/2ZetUFR>

⁷Example of the exploration layer: <https://bit.ly/2Vrg29H>

also represented concepts. Each slice represents a different concept with a different color. The representation of the concepts is very important in this visualization because it shows all the activities, so the teacher can locate which activities have a target concept while viewing the complete learning trajectory. The teachers can also find which activities have a target CT concept by interacting with the concepts visualization space (Figure 1), as it will highlight⁸ the activities that have the concept that the cursor is hovering over.

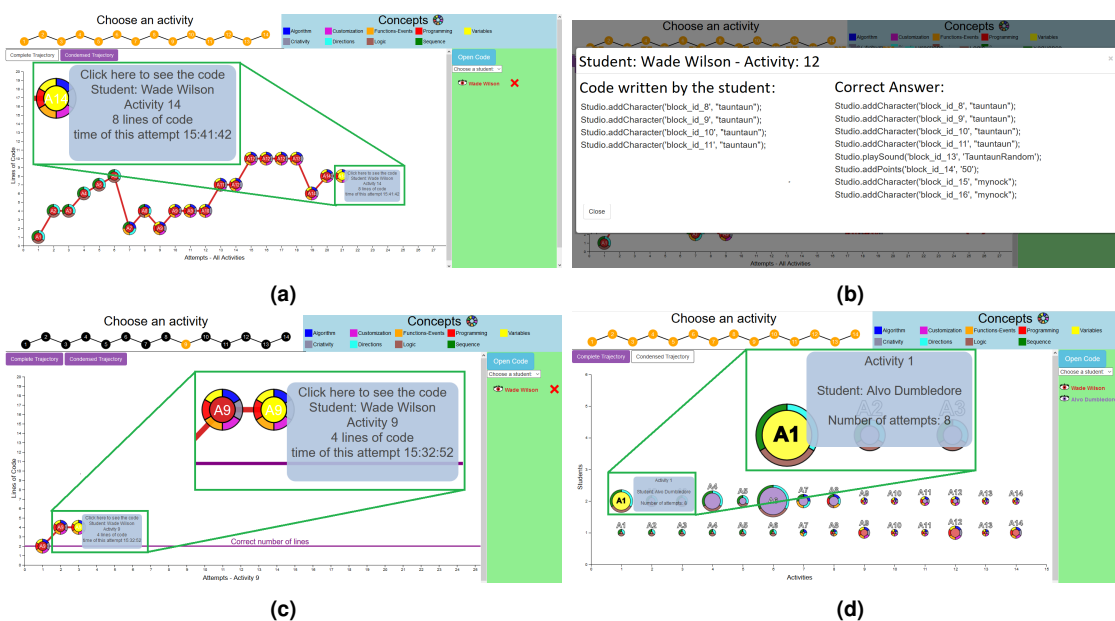


Figure 2. (a) - Complete Trajectory visualization of one student with tooltip from the navigation layer. (b) - modal window showing the code written by the student and the correct code besides it. (c) - Single Activity Trajectory. (d) - Condensed Trajectory.

The *Single Activity Trajectory* (Figure 2c) is a fragment of the *Complete Trajectory*. It represents the trajectory of a single activity and allows better analysis between trajectories of different students in a specific activity. It also aligns the attempts of the students of the chosen activity that were most likely misaligned on the *Complete Trajectory* visualization, making the visual analysis more friendly. Additionally, this visualization shows the correct number of lines of code as a purple line with a legend on the *visualization space* (1). The *Single Activity* visualization can be accessed by *exploring* the vertices of the graph on *visualization space* (2), named in Figure 2c as *Choose an activity*.

The *Condensed Trajectory* consists in a modified bubble chart that shows the learning trajectory of the students, but in a more concise way (see Figure 2d). This visualization facilitates the search for outliers or skipped activities, since it shows one *bubble* for each activity. If the student skipped one activity, there will be no *bubble* for that activity, and a greater number of attempts in a particular activity is easily perceived due to the bigger size of the *bubble* representing such activity. The *Condensed Trajectory* compresses the *Complete Trajectory* and its vertices into *bubbles*, one for each activity. The more attempts made to answer the activity, the bigger the *bubble* will be. The *x* axis represents the number of the activity and the *y* axis representing the number of students inserted on

⁸concept highlight example at <https://bit.ly/2Vf9PAB>

the chart. The *AN (Activity N)* labels inside or above the *bubbles* denotes which *bubble* represents which activity. The segmented ring around each bubble represents the concepts exercised in each activity. As this visualization also shows all the activities, the CT concepts representation is important so as to guide the teachers to find which activities have which concepts. By interacting with with the Concepts diagram, the teachers can highlight which activities have a target CT concept.

5. Usability Evaluation

We conducted an evaluation of our proposed visualizations with 20 teachers from elementary schools in Brazil. Complying with strict ethic and scientific standards, a research project (Ethic Appreciation Presentation Certificate - EAPC 18606019.2.0000.5504) was approved by the Ethics committee of research in human beings of the University of So Carlos. We followed the Lazar et al. [Lazar et al. 2017] instructions to the planning, execution and analysis of our study.

The participants were recruited through groups of emails and posts on social media in the education area. It is worth noting that the evaluation was done before the COVID-19 pandemic. We had a total of 23 participants in sessions with one or two participants at a time which were given a quick slide explanation about the evaluation, the visualizations, code programming in blocks and computational thinking. We had 12 sessions with teachers from three different elementary schools in Brazil. We clarified any doubts before the start the of the visualizations evaluation. No interactions between the participant and the researchers happened while the study was occurring. After the completion of the tasks, the participants answered a questionnaire about their experience with the visualizations (two open questions where the participants reported suggestions and problems).

The evaluation assessed the understanding, usability and reception of the teachers regarding the proposed visualizations through four artifacts: a *profile questionnaire* to gather the participants data, a *tasks questionnaire* (Table 1 , video and audio recordings of the teachers interactions with the visualizations and a two questions questionnaire regarding suggestions and comments on the visualizations. The *tasks questionnaire* contained nine tasks the teachers had to complete using the learning trajectories visualizations. The tasks evaluated the interactions and the visualization types needed to complete each task. We used notebooks equipped with mice and microphones where the participants could interact with the proposed visualizations and answer the online questionnaires. Video and audio recording were done with the software streamLabsOBS,⁹ a free and open-source software, which is capable of recording the screen of the computer and the audio at the same time.

We assessed four types of validity threats, as determined by Lazar et al. [Lazar et al. 2017]: internal, external, construction and conclusion. The threats to internal validity was mitigated by taking into consideration the fatigue aspect, making the test session take no longer than an hour in order to exclude this aspect of our evaluation. The issues with regards the external validity was minimized by the gathered participants presenting a wide interval of ages, background and teaching experience which allowed to be confident in relation to our findings generalization. We also recruited participants with and without experience in CT to better reflect real world conditions. The measures that we

⁹<https://streamlabs.com/streamlabs-obs>

took into account allowed us to have focus on the investigation goal and to mitigate possible issues on construct validity, we discussed the study tasks with an HCI specialist and minimized the carry-over effect from one task to another [Forsell 2010]. We also made sure to elucidate all doubts of the participants regarding the understanding of the tasks. Finally, to draw accurate conclusions from observation (i.e. conclusion validity), we used the triangulation in our analysis which provided a more confidence on the findings.

6. Results and Discussion

The profile questionnaire resulted in a varied set of answers. From the participants, 15% were less than 30 years old, 35% were 30-40 and a majority of 50% was 41+ years old. The participants were mostly experienced teachers with 80% of having 9+ years of experience. 90% had degrees in Pedagogy with 15% having post graduation titles, and all of them have knowledge of learning trajectories concepts. The profile questionnaire results also shows that 90% of the participants accessed the internet 9+ times/week with 95% having used at least one online tool in their classes, and and 90% used at least two kinds of electronic devices regularly (computer, tablet or smartphone). Regarding experience in teaching computational thinking, 65% had some experience while the other 35% had never taught the subject. In the field of knowledge of charts, 65% knew 3 or more kinds of charts.

We recorded the voice of the participants in order to employ the "Think Aloud" [Lewis and Rieman 1993] protocol and later transcribe it to apply the *open coding* technique, an analytic process through which concepts are identified and their properties and dimensions are discovered in data [Strauss and Corbin 1998]. We used *triangulation* of about 7 hours of video video and audio together with the questionnaires to make a qualitative analysis of the collected data [Lazar et al. 2017]. From the triangulation and *open coding* we found out a total of 85 codes. We divided these codes results in negative and positive issues and split the negative issues again into two other categories: *obstacles* and *barriers*. *Obstacles* were issues that expressed solvable problems in the process of understanding the visualizations. On the other hand, *barriers* represented insurmountable problems that effectively halted their progress in a specific activity or aspect [Lazar et al. 2017]. We assigned a color for each classification. From 85 codes found, 46 (54.1%) were expressions of understanding (green), 32 (37.6%) obstacles (orange) and 7 (8.2%) barriers (red). Participants had more than half of their verbal expressions (codes) showing understanding. This shows that in spite of several obstacles, the participants were usually able to work around or understand the problems they found. The 7 barriers show that the visualizations can still be improved by removing these barriers or at least making them easier for the users to transpose. Table 2 shows an example of each classification.

The analysis on the videos led to the exclusion of 3 participants on the account that they answered the task questionnaire without looking or interacting with the visualizations. While we watched the videos we could analyze the process leading to the answers of the task questionnaire and see the most and least understood interactions and visualizations since we mapped each interaction and visualization needed by each task (Table 1). We also investigated possible reasons why some of the participants could not answer some tasks correctly (if it was due to visualization, interaction or interpretation problems). Lastly, we analyzed the suggestions of the participants.

Table 1. Example of Tasks, Visualization Actions, Interactions and Visualization Types

Title\Tasks	Visualization Actions	Interaction	Visualization Type
1 - Looking at the trajectories of the students "Homer Simpson", "Barry Allen" and "Harry Potter", which of them completed the activity 6 correctly?	identify the student's code and the answer's code, use the "correct number of lines" line	select the students, select a specific activity, click on the graph's vertices	<i>Single Activity Trajectory</i>
2 - How many tries did the student "Harry Potter" use in order to complete the task 1?	select the student, identify the number of submissions using the graph's axis on the <i>Complete Trajectory</i> or <i>Single Activity Trajectory</i> graphs, or locate the correct bubble on the <i>Condensed Trajectory</i>	Select the student, select the specified activity	<i>Single Activity Trajectory</i>
3 - Question - Check all the concepts present in activity 9.	identify the concepts in the right vertex or bubbles	select a student, select a specific activity	<i>Single Activity Trajectory</i>
4 - Question - Considering the students "Barbara Kean", "Bucky Barnes" and "Harry Potter", which of them tried to answer all the activities?	identify activities on the <i>Complete Trajectory</i> , or check every activity separately or use the <i>Condensed Trajectory</i> and find missing bubbles	select the students, use any of the three charts	<i>Complete Trajectory</i>
5 - Question - How many tries did the student "Mickey Mouse" do in order to complete all activities?	interpret the "tries" axis in the <i>Complete Trajectory</i> or <i>Single Activity Trajectory</i> graphs, or use <i>Condensed Trajectory</i> tool tips	select student, use the charts	<i>Complete Trajectory</i>
6 - Question - What was the time between the last try on activity 6 and the first try on the activity 6 of the student "Mickey Mouse"?	check the time on the edge of the <i>Complete Trajectory</i> graph or check the last and first tries of activities 5 and 6 separately	select students, use the tooltips with mouse hover	<i>Complete Trajectory</i>
7 - Question - In which activity did the student "Adam Strange" make the most tries?	identify which activity had the most tries.	select the student, use any of the charts	<i>Condensed Trajectory</i>
8 - Question - Which of the three students had less tries on the activity 10?	identify activity 10, identify which of the three students had less tries on activity 10	select the students, use sight and interaction any of the three charts to locate the activity 10 and compare the number of tries.	<i>Condensed Trajectory</i>
9 - Question - What are the concepts used on the activity on which the student "Peter Parker" made the most tries?	identify the student "Peter Parker" and which activity he tried the least and then identify the concepts of that activity	select student, use the different charts and mouse interactions to identify the student, the activities, number of tries and concepts	<i>Condensed Trajectory</i>

Table 2. Understanding, Obstacles and Barriers from open coding process

Category	Speech
Understanding	Bubble chart..oh, here it shows all of the activities...
Obstacle	I am trying to understand these lines of code...
Barrier	Why did this only appears here?

In order to correctly complete each task, the teachers had to use and understand specific visualizations, visualization aspects and interactions for each task (as shown in Table 1), so we considered the task questionnaire score of the participants as a direct representation of the teachers understanding of the visualizations and its interactions (e.g a teacher who correctly complete all the tasks must have understood all the needed aspects of the visualizations to do so). The average task score result was 72%, directly translating to 72% of general understanding of the visualizations (Table 3). Most of the errors while answering the tasks were caused by a lack of visual interpretation of the data presented on the visualizations, as most of the users were able to successfully access the data but some were not able to make the connections between the information displayed in order to answer the tasks correctly. The tasks 1 and 4 were the least understood tasks, showing, according to Table 3 that the *Single Activity Trajectory* and *Complete Trajectory* posed more difficulties than the *Condensed Trajectory* visualization.

Regarding the participants opinion, we prepared two questions for each participant to answer. *Question 1: "Would you want to change anything in the visualizations?"* and *Question 2: "Do you have any comments regarding the visualizations?"*. The questions were optional, so the participant could left them in blank if they did not have any opinion.

Table 3. Results of the task questionnaire.

Participant	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Tasks	1	0	1	0	0	0	1	1	1	0	0	1	0	0	0	0	1	1	1	1
	2	0	1	0	1	0	0	1	1	1	1	1	0	1	0	1	1	1	1	1
	3	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	4	0	0	0	0	0	0	1	0	1	1	1	1	1	0	0	1	1	1	1
	5	1	1	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1
	6	0	1	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
	7	1	1	0	1	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1
	8	0	1	0	1	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1
	9	1	1	0	1	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1
Understanding Score	3	8	0	5	2	7	8	8	8	8	9	7	6	1	7	8	9	9	9	

The *Question 1* got a total of 5 answers¹⁰ suggesting changes on the visualizations. The teachers suggested modifications in the control aspect of the visualization (scroll bars) and in the total number of attempts to make it more visible. The *Question 2* got a total of 16 answers¹¹. Some participants had mixed feelings expressed on their comments. From total of 16 comments, we had 12 positive, 6 suggestions and 4 negative. Some comments contained more than one category, hence the sum of the results exceeds 16. The positive comments complimented the visualizations (e. g. *"I found it interesting."*, *"It is very didactic."*, *"I like it!"*). The suggestions approached visual aspects like overlapping learning trajectories and the impossibility to choose two activities at the same time. The negative comments spoke of difficulty to see the charts, doubts about the interpretation of the charts and the overwhelming visual aspect that inserting several students at the same time had on the visualizations.

7. Conclusion and Future work

This paper proposed three interactive visualization of learning trajectories representing the actions of students in CT exercises. An usability and user experience evaluation was conducted with 20 teachers to assess the proposed learning trajectories visualizations. The evaluation results led us to the conclusion that the teachers could for the most part understand the data of the CT exercises with the proposed visualizations of learning trajectories, despite some minor issues. The main contribution of our paper is to present proposed interactive learning trajectories visualizations that can be used in multiple platforms that introduce students to CT. As far as we know, our paper presents an innovative proposal of visualization of students' learning trajectories regarding CT activities. For future work we plan to implement a few of the teachers' suggested modifications and explore new ways of representing the students learning trajectories.

Acknowledgments

We thank the financial support of Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) Finance Code 001.

¹⁰The complete Q1 table of answers can be found at: <https://bit.ly/2COPkBX>

¹¹The complete Q2 table of answers can be found at: <https://bit.ly/2COPkBX>

References

- Berland, M., Martin, T., Benton, T., Smith, C. P., and Davis, D. (2013). Using learning analytics to understand the learning pathways of novice programmers. *Journal of the Learning Sciences*, 22(4):564–599.
- Cai, R. (2018). Adaptive learning practice for online learning and assessment. In *Proceedings of the 2018 International Conference on Distance Education and Learning, ICDEL '18*, pages 103–108, New York, NY, USA. ACM.
- Carmo, E., Gasparini, I., and Teixeira de Oliveira, E. (2019). Captura e visualização das trajetórias de aprendizagem como ferramentas para a análise do comportamento dos estudantes em um ambiente adaptativo educacional. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação - SBIE)*, 30(1):309.
- da Silva, C. G. (2014). Visualização de informação: introdução e influências de IHC. In Boscarioli, C., Bim, S. A., Leitão, C. F., and Maciel, C., editors, *Companion Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems, IHC 2014, Foz do Iguaçu, Brazil, October 27-31, 2014*, pages 81–82. ACM.
- de Borba, J. E., Gasparini, I., Lichtnow, D., Pimenta, M. S., and de Oliveira, J. P. M. (2016). Captura e visualização da trajetória de aprendizagem do aluno: um mapeamento sistemático. *TISE*, 12:105–111.
- de Melo, A. N. B., Zaina, L. M., Martinelli, S. R., and Sakata, T. (2018). Visualização de informações para acompanhamento do ensino do pensamento computacional: uma proposta baseada em design centrado no usuário - technical report. Universidade Federal de São Carlos.
- Forsell, C. (2010). A guide to scientific evaluation in information visualization. In *2010 14th International Conference Information Visualisation*, pages 162–169.
- Fortenbacher, A., Beuster, L., Elkina, M., Kappe, L., Merceron, A., Pursian, A., Schwarzrock, S., and Wenzlaff, B. (2013). Lemo: A learning analytics application focussing on user path analysis and interactive visualization. In *2013 IEEE 7th International Conference on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS)*, volume 02, pages 748–753.
- Fouh, E., Akbar, M., and Shaffer, C. A. (2012). The role of visualization in computer science education. *Computers in the Schools*, 29(1-2):95–117.
- Grover, S. and Pea, R. (2013). Computational thinking in k–12: A review of the state of the field. *Educational Researcher*, 42(1):38–43.
- Keim, D. A. (2002). Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics*, 8(1):1–8.
- Lazar, J., Feng, J. H., and Hochheiser, H. (2017). In Lazar, J., Feng, J. H., and Hochheiser, H., editors, *Research Methods in Human Computer Interaction (Second Edition)*, page 508. Morgan Kaufmann, Boston, second edition edition.
- Lewis, C. and Rieman, J. (1993). *Task-centered User Interface Design: A Practical Introduction*. hcibib.org.
- Mazza, R. (2009). *Introduction to Information Visualization*. Springer Publishing Company, Incorporated, 1 edition.

- Nascimento, H. A. D. and Ferreira, C. B. R. (2005). Visualização de informações – uma abordagem prática. In *XXV Congresso da Sociedade Brasileira de Computação*, pages 1262–1313. Universidade Federal de Goiás.
- Raabe, A. L. A., Zorzo, A. F., Frango, I., Granville, L. R. L. Z., Salgado, L., da Cruz, M. K., Bigolin, N., Cavalheiro, S. A. C., and Fortes, S. (2017). Referenciais de formação em computação: Educação básica. *Sociedade Brasileira de Computação*.
- Ramos, D. B., de Oliveira, E. H. T., Ramos, I. M. M., and Oliveira, K. M. T. (2015). Trilhas de aprendizagem em ambientes virtuais de ensino-aprendizagem: Uma revisão sistemática da literatura. In *Anais do XXVI Simpósio Brasileiro de Informática na Educação*, pages 182–190. CBIE-LACLO.
- Rich, K. M., Binkowski, T. A., Strickland, C., and Franklin, D. (2018). Decomposition: A k-8 computational thinking learning trajectory. In *Proceedings of the 2018 ACM Conference on International Computing Education Research*, ICER '18, pages 124–132, New York, NY, USA. ACM.
- Rich, K. M., Strickland, C., Binkowski, T. A., Moran, C., and Franklin, D. (2017). K-8 learning trajectories derived from research literature: Sequence, repetition, conditionals. In *Proceedings of the 2017 ACM Conference on International Computing Education Research*, ICER '17, pages 182–190, New York, NY, USA. ACM.
- Seiter, L. and Foreman, B. (2013). Modeling the learning progressions of computational thinking of primary grade students. In *Proceedings of the Ninth Annual International ACM Conference on International Computing Education Research*, ICER '13, page 59–66, New York, NY, USA. Association for Computing Machinery.
- Simon, M. (1995). Reconstructing mathematics pedagogy from a constructivist perspective. *Journal for Research in Mathematics Education*, 26.
- Strauss, A. and Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*, volume 4. Thousand Oaks, CA: Sage, 2 edition.
- Sáez-López, J.-M., Román-González, M., and Vázquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: A two year case study using “scratch” in five schools. *Computers & Education*, 97:129 – 141.
- Tafner, E. P., Tomelin, J. F., and Müller, R. B. (2012). Trilhas de aprendizagem: uma nova concepção nos ambientes virtuais de aprendizagem-ava. In *Congresso Internacional de Educação a Distância*.
- Wing, J. (2008). Computational thinking and thinking about computing. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 366:3717–25.
- Yang, S., Domeniconi, C., Reville, M., Sweeney, M., Gelman, B. U., Beckley, C., and Johri, A. (2015). Uncovering trajectories of informal learning in large online communities of creators. In *Proceedings of the Second (2015) ACM Conference on Learning @ Scale, L@S '15*, pages 131–140, New York, NY, USA. ACM.

Yi, J. S., ah Kang, Y., Stasko, J., and Jacko, J. A. (2007). Toward a deeper understanding of the role of interaction in information visualization. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1224–1231.