Assessing Implicit Computational Thinking in Zoombinis Gameplay: Pizza Pass, Fleens & Bubblewonder Abyss

Elizabeth Rowe
Jodi Asbell-Clarke
EdGE at TERC
2067 Massachusetts Ave
Cambridge, MA 02140
elizabeth_rowe@terc.edu
jodi_asbell-clarke@terc.edu

Kathryn Cunningham
School of Interactive Computing
Georgia Institute of Technology
Atlanta, GA 30332
kcunningham@gatech.edu

Santiago Gasca
EdGE at TERC
2067 Massachusetts Ave.
santiago_gasca@terc.edu

CHI PLAY'17 Extended Abstracts, October 15–18, 2017, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5111-9/17/10.
https://doi.org/10.1145/3130859.3131294

Abstract

Players can build implicit understanding of challenging scientific concepts when playing digital science learning games [1]. In this study, we examine implicit computational thinking (CT) skills among upper elementary and middle school students during *Zoombinis* gameplay. We report on the development of a human labeling system for gameplay evidence of four CT skills: problem decomposition, pattern recognition, algorithmic thinking, and abstraction. We define labels that identify use of these skills in three *Zoombinis* puzzles, based on analysis of video data from both CT novices (upper elementary and middle school students) and CT experts (computer scientists and expert *Zoombinis* players). Future work will involve the construction of detectors for implicit CT skills based on these human labels, in order to analyze gamelog data at scale and give feedback to teachers.

Author Keywords

Implicit learning; computational thinking; video analysis; learning games.

The Game: Zoombinis

Figure 1. Map of the Zoombinis' journey from Zoombini Isle to Zoombiniville.

Zoombinis puzzles develop concepts such as sets, logical relationships, dimensions, mappings, sorting, comparing, and algorithms [14]. All gameplay is based on the Zoombinis' attributes, or the attributes of other characters and props.

Figure 2. Zoombinis make their way through the puzzles in packs of 16. Zoombinis have four attributes (hair, eyes, nose, and feet), with five traits for each attribute.

ACM Classification Keywords

H.5.1 Information Interfaces and Presentation (e.g., HCI): Multimedia Information Systems—*Evaluation/methodology*; K.3.2 Computers and Education: Computer and Information Science Education—*Computer science education*; K.8.0 Personal Computing: General—*Games*

Introduction

Zoombinis [2] is an award-winning, popular learning game that elicits computational thinking. Players guide Zoombini characters on a journey away from the evil Blouts on Zoombini Isle, through a series of challenging puzzles, and to safety in Zoombiniville (see Figures 1 and 2). Situated in the mathematics necessary for computer programming and data analysis [3], the suite of 12 puzzles, each with four levels, provides scaffolded problem-solving for learners ages 8 and above. *Zoombinis* was re-released in 2015 for tablets, desktops, and, in 2017, for Chromebooks.

Currently, we are studying gameplay among students in grades 3-8 in order to understand how students implicitly learn computational thinking during *Zoombinis* gameplay and how their teachers can build upon that knowledge.

This paper reports on human-labeling of gameplay observations, which provides some of the relevant features for the data mining models and detectors. This paper builds on [4], which described the human labeling of three *Zoombinis* puzzles—Pizza Pass, Allergic Cliffs, and Mudball Wall. The data detectors developed by our work provide implicit *game-based learning assessments*, which may reveal knowledge that is evident in gameplay behaviors but which that may go unexpressed in typical assessments used in school and educational research [1].

Implicit Computational Thinking

Learners may demonstrate knowledge through behaviors that they are not yet able to express formally [5]. This is referred to as *implicit knowledge*. Game-based learning assessments (GBLA) show promise as a new method of assessing implicit knowledge by avoiding jargon, construct-irrelevant material, and test anxiety which can make traditional assessments challenging [6].

For research on *Zoombinis*, we defined a learning progression of computational thinking and problem-solving skills based on several definitions of Computational Thinking emerging in the field [7-11]. We used this learning progression to guide our labeling of strategies and behaviors in gameplay consistent with facets of the progression (Figure 3) [5]. This iterative learning progression of computational thinking that we hypothesize will be evident in *Zoombinis* gameplay are:

- **Problem Decomposition**: The reduction of ambiguity or complexity of a problem by breaking it into smaller, more manageable parts. This is comparable to isolating variables or systems to test.
- **Pattern Recognition**: The recognition that objects are arranged following a rule or rules. The identification of groups of solutions or characteristics of solutions that can be categorized.
- **Abstraction:** The removal of details to identify and extract relevant information to define main idea(s) or solutions.

Computational Thinking Learning Progression

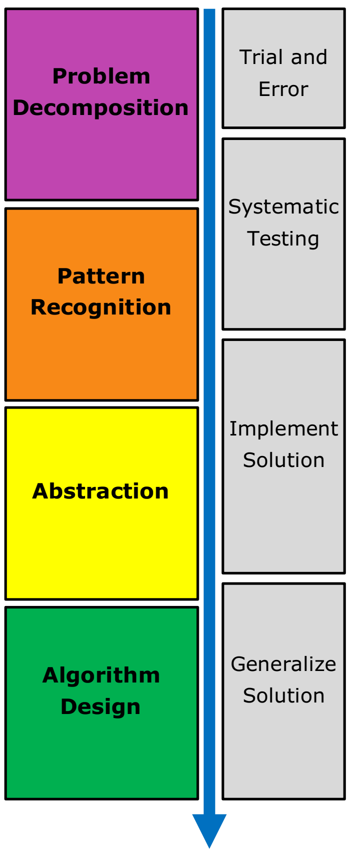

Figure 3. A learning progression of computational thinking operationalized in *Zoombinis* gameplay.

- **Algorithm Design**: The creation of an ordered list of instructions for solving a problem or doing a task. The creation or explication of general solutions to a problem or family of problems.

We have defined specific, iterative phases of problem solving that are intertwined with expressions of CT (Figure 3):

1. **Planning and preparation:** Using techniques that supersede puzzle play to make more efficient and effective problem-solving.
2. **Trial & Error:** No evidence of testing hypotheses in an ordered, planned way. Actions are independent of prior actions.
3. **Systematic Testing:** Testing hypotheses about underlying rule in an ordered, planned way. Next action depends on previous action. Goal of this phase is finding a working solution to implement.
4. **Systematic Testing with a Partial Solution:** Testing hypotheses about a second dimension of the underlying rule when the first dimension is known.
5. **Implementing a Partial Solution:** Completing a pattern according to one dimension of the underlying rule, while other dimension(s) remain unsolved.
6. **Implementing a Full Solution:** Completing the pattern once a working solution for all dimensions of the puzzle has been found.

These facets of CT are demonstrated in their progression from **Trial and Error**, where there is no systematic pattern to the behaviors, towards **Systematic Testing**, typically involving problem decomposition. When players recognize patterns in solutions to the smaller problems, they abstract that towards general rules so that they can then move to **Implement a Solution** to the larger puzzle. When learners encounter new puzzles that require similar solutions, they may begin to **Generalize Solutions** leading toward algorithm design. The central question this research addresses is:

What are the behavioral indicators of implicit computational thinking in Zoombinis gameplay that humans can reliably label?

Methods

We are building automated tools that can use *Zoombinis* gameplay data to provide information about players' implicit CT learning in the game using the same process we did for the physics game, *Impulse* [1]. To do this we:

1. Video record and then human label *Zoombinis* gameplay from beginners and expert players, children and adults, to capture the variety of strategies used to solve the puzzles.
2. Merge human labels with gamelog data
3. Distill log data into features useful for measuring strategies that appear in the videos, focusing on the strategies that are consistent with CT.
    1. Build detectors of players' CT strategies in the gameplay log, grounded in human labeling.
    2. Validate the detectors as formative assessments of implicit CT by comparing the performance of learners on external pre/post assessments of similar content.

Figure 4. Zoombinis must present pizzas and sundaes with certain toppings to appease trolls at Pizza Pass.

Figure 5. Only Zoombinis with matching attributes can lure Fleens off their branch in Fleens!

Figure 6. To cross Bubblewonder Abyss, Zoombinis must be launched in the right order, depending on their attributes.

This paper reports details about human labeling system developed from analysis of 87 videos as the first step in the development of GBLA of implicit computational thinking.

Sample and Procedures

Playtesting sessions last approximately 1 hour, and involve gameplay across multiple puzzles, as well as concurrent 'thinking aloud' by participants. ScreenFlow [12] is used to screen capture players' gameplay as well as video of their facial expressions and gestures and audio of their think-alouds.

Fifty-two elementary students in grades 3-5, 28 middle school students, and seven computer scientists have participated in playtesting. Of these 87 participants, 40 were female and 47 were male. Participants were recruited from local schools and clubs, as well as after-school programs.

The Puzzles

Three *Zoombinis* puzzles are the focus of this paper: Pizza Pass, Fleens, and Bubblewonder Abyss in Figures 4-6, respectively.

PIZZA PASS

The Zoombinis' path is blocked by one or more trolls that demand a meal (pizza, or pizza and sundae) with a specific set of toppings. The player selects a combination of toppings via buttons on a machine, and a Zoombini delivers the meal to the troll(s). However, the troll(s) only say whether (a) they want more toppings, (b) don't like at least one of the toppings, or (c) the meal is perfect. The troll throws incomplete meals to the side of the path, while meals that all trolls reject are thrown into a pit. Once all trolls are satisfied, they (noisily) eat their pizzas and let the remaining Zoombinis through.

FLEENS

The Zoombinis' path is blocked by Fleens, creatures who possess the same attributes as Zoombinis (eyes, noses, hair, legs) but with different traits (e.g. purple hair, rocket feet). Each Fleen trait corresponds to a Zoombini trait, so there's a unique Fleen for each Zoombini. In order to pass, three Fleens must be lured off a tree branch by their corresponding Zoombinis.

BUBBLEWONDER ABYSS

The Zoombinis must cross an abyss—split into a 13x13 grid with rocky ledges at the corners—by travelling inside floating bubbles. The bubbles only travel in directions determined by instructions marked on the grid. An arrow on the cliff ledge marks a starting point on the grid. Placing a Zoombini here will trigger the bubble machine to enclose the Zoombini and their journey begins in the direction of that arrow. The grid contains wormholes into which Zoombinis may fall (and are lost) if they are on the wrong path.

Human Labels of Implicit Computational Thinking

The human labeling system was developed collaboratively by the four authors. The authors have varying levels of computer science and *Zoombinis* play experience. We started with the definitions described in the 'Implicit Computational Thinking' section and an initial set of behavioral indicators. We iteratively watch 2-3 videos independently, discuss our labeling as a team, and revise the labeling system to incorporate emergent gameplay behaviors.

While the labels are consistent across puzzles, the behavioral indicators used to identify each vary by puzzle. Tables 1-3 describe sample gameplay indicators of computational thinking skills for the Pizza Pass, Fleens, and Bubblewonder Abyss puzzles, respectively. In all cases, if a sequence of behaviors is repeated across multiple attempts to solve the puzzle, it is considered evidence of Algorithm Design.

Gameplay Behaviors	***Labels***
Selecting one pizza or ice cream topping at a time. After all toppings have been tried, placing all those the troll likes on one pizza.	**Systematic Testing**—testing one topping at a time **Problem Decomposition**—isolating toppings **Pattern Recognition**—selecting toppings the troll accepts
Selecting one pizza or ice cream topping at a time until they find one a troll likes. They retain the desired topping on all future pizzas and add new toppings one at a time.	**Systematic Testing**—testing one topping at a time **Problem Decomposition**—isolating toppings **Pattern Recognition**—selecting toppings the troll hasn't rejected

Table 1. Gameplay behavioral indicators of implicit computational thinking skill in Pizza Pass [4]

Implications and Next Steps

We have found evidence of implicit computational thinking in two new *Zoombinis* puzzles. With the labeling systems proposed in [4], five of the twelve *Zoombinis* puzzles can be labeled for CT. Our next steps are to establish inter-rater reliability for the labeling system with a small sample of videos. We will then apply labels to the remainder of the sample,

Gameplay Behaviors	***Labels***
Selecting Zoombinis with different traits (e.g. different feet) until a match is found with a Fleen on the branch. Then, selecting Zoombinis with that trait until the targeted Fleen is knocked off the branch.	**Systematic Testing**—testing one Zoombini trait at a time **Problem Decomposition**—isolating traits **Pattern Recognition**—identifying a match between Zoombini and Fleen traits **Abstraction**—one-to-one correspondence between Zoombini and Fleen attributes
Luring all three Fleens off the branch within 5 moves. Such efficiency requires counting Zoombini traits and comparing them to counts of Fleen traits to find matches.	**Planning and Preparation**—collecting and analyzing relevant information before gameplay **Problem Decomposition**—isolating traits **Abstraction**—one-to-one correspondence between Zoombini and Fleen attributes

Table 2. Gameplay behavioral indicators of implicit computational thinking skill in Fleens

Gameplay Behaviors	***Labels***
Launching a Zoombini with a common trait on both launchpads. Using outcomes to guide future launches.	**Systematic Testing**—testing outcomes for a specific trait **Problem Decomposition**—isolating traits
Ordering the launch of Zoombinis based on their traits, so they successfully cross even while grid instructions change.	**Implementing a partial solution**—completing one dimension of the puzzle **Problem Decomposition**—isolating traits **Pattern Recognition**—understanding the pattern of traits required

Table 3. Gameplay behavioral indicators of implicit computational thinking skill in Bubblewonder Abyss

synchronize with the game log data, and develop automated detectors of implicit computational thinking.

The measurement of implicit computational thinking may enable the assessment of a broad array of diverse learners, even those who are unable to express their knowledge on a traditional exam. This work shows and example of using data mining methods to measure implicit learning through behaviors exhibited in digital environments, which may be particularly important for learners with cognitive differences [13]. The ability to measure computational thinking through behavior-based data generated by digital environments might provide novel forms of assessments leading to an inclusive STEM education and workforce opportunities.

Acknowledgements

We are grateful for NSF/EHR/DRK12 grant#1502282. We are thankful for the study participants and the many contributions of our wonderful EdGE colleagues without whom the study could not have been conducted

References

1. Elizabeth Rowe, Jodi Asbell-Clarke, and Ryan S. Baker. 2015. Serious game analytics to measure implicit science learning. In *Serious Game Analytics: Methodologies for Performance Measurement, Assessment, and Improvement*. C. S. Loh, Y. Sheng, and D. Ifenthaler (Eds.). Springer Science+Business.
2. TERC. 2015. *Zoombinis*. Game [Android, iOS, MacOS, Windows, Web]. (7 August 2015). TERC, Cambridge, MA.
3. Chris Hancock and Scot Osterweil. 1996. Zoombinis and the Art of Mathematical Play. *Hands On!* 19, 1 (Spring 1996), 1,17-19.
4. Elizabeth Rowe, Jodi Asbell-Clarke, Santiago Gasca, and Kathryn Cunningham. 2017. Assessing Implicit Computational Thinking in Zoombinis Gameplay. In *Proceedings of the International Conference on the Foundations of Digital Games* (FDG '17).
5. Michael Polanyi. 1966. *The Tacit Dimension*. University of Chicago Press, Chicago, IL.
6. Valerie J. Shute, Iskandaria Masduki, Oktay Donmez, Vanessa P. Dennen, Yoon-Jeon Kim, Allan C. Jeong, and Chen-Yen Wang. 2010. Modeling, Assessing, and Supporting Key Competencies Within Game Environments. In *Computer-Based Diagnostics and Systematic Analysis of Knowledge*. D. Ifenthaler, P. Pirnay-Dummer, & N. M. Seel (Eds.). Springer US, Boston, MA, 281-309.
7. Valerie Barr and Chris Stephenson. 2011. Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads* 2, 1 (February 2011), 48-54.
8. Jeannette M. Wing. 2006. Computational thinking. Commun. ACM 49, 3 (March 2006), 33-35.
9. Google. 2016. CT Overview. Retrieved from https://edu.google.com/resources/programs/exploring-computational-thinking/#!ct-overview.
10. CSTA. 2017. CSTA K-12 Computer Science Standards. Retrieved from http://www.csteachers.org/?page=CSTA_Standards.
11. Shuchi Grover and Roy Pea. 2013. Computational Thinking in K–12: A Review of the State of the Field. *Educational Researcher* 42, 1, 38-43.
12. Telestream. Screenflow for Mac 6.0. 2016. Software [MacOS]. https://www.telestream.net/screenflow/overview.htm
13. Thomas M. Haladyna and Steven M. Downing. 2004. Construct-irrelevant variance in high-stakes testing. *Educational Measurement: Issues and Practice,* 23, 1, 17-27.