

An Intelligent Pedagogical Agent to Foster Computational Thinking in Open-Ended Game Design Activities

Özge Nilay Yalçin, Sébastien Lallé, Cristina Conati

▶ **To cite this version:**

Özge Nilay Yalçin, Sébastien Lallé, Cristina Conati. An Intelligent Pedagogical Agent to Foster Computational Thinking in Open-Ended Game Design Activities. IUI '22: 27th International Conference on Intelligent User Interfaces, Mar 2022, Helsinki, Finland. pp.633-645, 10.1145/3490099.3511124 . hal-03695879

HAL Id: hal-03695879
https://hal.science/hal-03695879

Submitted on 15 Jun 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Intelligent Pedagogical Agent to Foster Computational Thinking in Open-Ended Game Design Activities

ÖZGE NILAY YALÇIN, Department of Computer Science, University of British Columbia, CANADA
SÉBASTIEN LALLÉ, LIP6, Sorbonne University, CNRS, FRANCE
CRISTINA CONATI, Department of Computer Science, University of British Columbia, CANADA

Free-form Game-Design (GD) environments show promise in fostering Computational Thinking (CT) skills at a young age. However, such environments can be challenging to some students due to their highly open-ended nature. Our long-term goal is to alleviate this difficulty via pedagogical agents that can monitor the student interaction with the environment, detect when the student needs help and provide personalized support accordingly. In this paper, we present a preliminary evaluation of one such agent deployed in a real-word free-form GD learning environment to foster CT in the early K-12 education, Unity-CT. We focus on the effect of repetition by comparing student behaviors between no intervention, 1-shot intervention, and repeated intervention groups for two different errors that are known to be challenging in the online lessons of Unity-CT environment. Our findings showed that the agent was perceived very positively by the students and the repeated intervention showed promising results in terms of helping students make less errors and more correct behaviors, albeit only for one of the two target errors. Based on these results, we provide insights on how to improve the delivery of the agent's interventions in free-form GD environments.

CCS Concepts: • **Applied computing** → **Interactive learning environments**; • **Human-centered computing** → **User models**; • **Information systems** → *Association rules.*

Additional Key Words and Phrases: Pedagogical agent, Real-Time Support, Game Design, Computational Thinking, Open-Ended Learning Environments

ACM Reference Format:
Özge Nilay Yalçın, Sébastien Lallé, and Cristina Conati. 2022. An Intelligent Pedagogical Agent to Foster Computational Thinking in Open-Ended Game Design Activities. 1, 1 (February 2022), 21 pages. https://doi.org/10.1145/3490099.3511124

1 INTRODUCTION

Open-ended learning environments (OELEs) have been found to have great potential for fostering students' learning, by letting the students proactively explore and experiment with the learning material with minimal constraints [24]. However, it is known that some students may not learn well from this relatively unstructured and self-directed form of interaction, because they lack the skills to assess their progression and success [32]. Research has shown that this challenge can be alleviated by AI-driven help functionalities that are adaptive to the student needs, i.e., detect and respond to the learners' difficulties [29, 36, 39, 45], and there is increasing interest in investigating if/how Intelligent

Authors' addresses: Özge Nilay Yalçın, onyalcin@cs.ubc.ca, Department of Computer Science, University of British Columbia, 2366 Main Mall, Vancouver, BC, CANADA, V6T 1Z4; Sébastien Lallé, sebastien.lalle@sorbonne-universite.fr, LIP6, Sorbonne University, CNRS, 4 place Jussieu, Paris, FRANCE, 75005; Cristina Conati, conati@cs.ubc.ca, Department of Computer Science, University of British Columbia, 2366 Main Mall, Vancouver, BC, CANADA, V6T 1Z4.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2022 Copyright held by the owner/author(s). Publication rights licensed to ACM.
Manuscript submitted to ACM

Pedagogical Agents (IPAs) can facilitate students' learning in OELEs [9, 12, 38, 42], by offering this help in a more engaging and motivating manner.

Designing such IPAs for OELEs has two main challenges. One relates to student modeling, namely, how to capture and recognize students' behaviors that indicate the need for help. The second challenge relates to how to deliver the help effectively. This challenge has been identified in many different types of learning environments, due to issues with learners ignoring the help or misusing it [2, 4, 41], but it is exacerbated in OELEs, because help provision might interfere with the exploratory nature of the interaction, and be dismissed by learners as intrusive or annoying, especially if there are of younger ages.

This paper addresses the latter challenge by deploying and evaluating an IPA that provides assistance in Unity-CT, an OELE designed to foster the acquisition of computational thinking (CT). CT is defined as the ability to express problems and their solutions computationally, and in recent years there has been increasing interest in fostering CT in K-12 education [5]. Unity-CT was developed by a Vancouver-based company, UME Academy (*https://www.ume.academy*), to engage elementary school students with CT by leveraging free-form game-design (GD) activities, where students create video games without programming knowledge. Unity-CT is built on the Unity game engine and delivers a curriculum of 8 classes during which students are given "challenges" asking to design incremental game components that meet specific constraints, with the help of an instructor. UME Academy has been successfully running Unity-CT classes as part of after-school activities and camps since 2015. Unity-CT was originally designed for in-classroom settings and mostly used by local institutions, however, with the onset of the COVID pandemic, UME Academy developed a version for online classes. This online version has been used since November 2020 to deliver 800+ classes in North America, and it is the version we use in this paper.

Although free-form GD activities can increase student motivation and engagement [1, 5, 15, 28], Unity-CT introduces specific challenges to students as it requires them to learn how to operate in the complex Unity game engine and learn the CT material, simultaneously, and they often need timely personalized support from the instructor to overcome these difficulties. Although students can ask the instructor for help while working on a challenge, some students do not do it even if they are at an impasse. Instructors generally try to watch for these situations to provide help, but this is challenging without continuously monitoring what each student is doing on their computer. These challenges are exacerbated in online classrooms, both because the instructor cannot rely on the visual cues signaling the need for help that are available when seeing students in person, and because it is not possible to talk to students privately due to difficulties in implementing effective tools for one-to-one communication. To address these challenges, we have been working with UME Academy to add an IPA to Unity-CT, which can spontaneously provide help when it detects that a student would benefit from it based on the student's behaviors.

In previous work we examined how to build a data-driven student model for Unity-CT [31], showing the feasibility of inferring from data a variety of non-obvious suboptimal student behaviors that can benefit from the IPA unsolicited help. In this paper, we focus on the aspect of how to provide help, once the need for it has been established. We focus in particular on comparing the effectiveness of providing help on a specific issue only once, vs providing it again if the issue arises again. This is an important aspect of the IPA design because help repetition is found be effective in OELE for older students [29], but could be ineffective for younger students engaged in GD activities as they might be less experienced in using computers. We evaluated two versions of the IPA that differ mainly in whether they provide help on a repeated error or not, deployed in UME Academy's classes run from April to August 2021, and we compared these classes against classes who did not have the IPA, in terms of how the IPA help impacted student behaviors in Unity-CT.

Our results show that repeating the IPA intervention did lead to the students making significantly fewer errors in the rest of the class compared to both providing the help only once or providing no help at all. Furthermore, repeating the help also led to the students successfully using more of the corresponding correct behaviors in the rest of the class as compared to receiving no help. However, these positive results were only present for one type of error, which was harder to detect but easier to correct compared to the other. Altogether, these findings suggest that our version of the IPA with help repetition is promising toward supporting open-ended GD activities for CT, depending on the target error, an important finding since no prior work focused on delivering adaptive pedagogical content in this context. Furthermore, most students reported that they overall liked the IPA and found it helpful and not distracting, both when receiving one or repeated interventions. This shows that providing more interventions did not overall overwhelm the students, which can be a pitfall with repeated automated interventions. However, the repeated IPA generated more confusion in the students than the IPA that provides only one. Although the overall confusion rates were low, this suggests that there is still room to further refine the repeated IPA to avoid any possible confusion. Based on these results, we provide insights on how to improve the delivery of the IPA interventions in free-form GD environments.

Our work provides several novel contributions to research on IPAs for OELEs. First, we investigate the use of IPAs in free-form GD, a learning activity which has gotten increasingly popular in teaching CT to younger students [1, 5, 15, 28], but has not been examined before in IPA research. Second, our IPA is implemented in a real-world commercial OELE for remote learning, whereas previous work was limited to OELEs that are designed specifically for research purposes and evaluated in ad-hoc learning activities [9, 12, 38, 42]. Thus, our work is a further step toward showing the value of IPA for OELEs that are actively used in real-world education remotely, a setting that has become increasingly widespread with the Covid pandemic. Lastly, our work focuses on the effect of repetition in providing help content for a young audience who are not proficient in using computers, where previous work that focused on repeated interventions were targeting high-school or college students that do not experience this drawback [10, 11, 29].

2 RELATED WORK

IPAs have been previously shown to have great potential for improving an OELE's ability to provide AI-driven adaptive (just "adaptive" from now on) support to students [35], albeit to the best of our knowledge, there has been no prior work on devising such IPAs during CT learning activities based on free-form GD. Although studies have shown that providing adaptive hints can significantly increase the performance of the students in programming GD activities [36, 39], using IPAs for delivering this support has also not been examined yet.

The closest work to our research is by Basu et al. [6], who designed an IPA in the CTSiM environment to provide personalized support during learning of CT with interactive simulations for model building activities (e.g., modeling a car's speed based on its mass and engine force). The personalized hints delivered by an IPA are based on students' behaviors in the simulation and found to have a positive effect on their learning performance. We contribute to this work by showing that IPAs can be valuable to teach CT in another activity, free-form GD, which is arguably more engaging to a very young audience than model building, but also much more unconstrained. In particular, the IPA in CTSiM leverages expert models of desired solutions and strategies, which is not possible to generate in Unity-CT due to the extremely large solutions and behaviors space. Furthermore, we test our IPA as part of an OELE that has been used for years at schools and camps with a tight collaboration with industry, whereas CTSiM is a research software.

Several works studied the value of IPAs for OELEs that support learning domains others than free-form GD and CT, with results showing that these IPAs can increase learning outcomes and engagement. Namely, Biswas et al. [9] studied an IPA that provides textual feedback while learning about science topics in Betty's Brain. Bouchet et al. [12] designed 4

IPAs to scaffold self-regulated learning based on the students' behaviors in MetaTutor, a hypermedia that lets students freely browse biological content. Moreno et al. [38] designed an IPA that provides adaptive support in the Design-A-Plant microworld meant to engage college students into science topics. Crystal Island [42], is a narrative-centered OELE that teaches microbiology concepts with animated IPAs as students freely navigate in the 3D environment. Our work extends these previous works by considering IPAs for a novel learning domain (free-form GD to foster CT).

While all of the aforementioned OELEs are desktop applications, there has been a few works that focused on designing IPAs for remote, web-based OELEs. In particular, in virtual worlds [42, 44] for environmental engineering courses, simulation environments for medical students [27] and collaborative learning [43]. However, only one of them had evaluations which was not conducted in a remote class setting as intended [27]. Moreover, unlike in our work, interaction with the IPAs were not adaptive as it was initiated by the students.

In problem-solving learning environments where correctness of student solutions can be assessed, extensive work studied sequence of hints that usually gradually increase in specificity, e.g., [18, 22, 46]. In OELEs, Kardan and Conati [29, 30] found that repeating help twice on a particular issue led to more students complying with the help content, than when just receiving the help once at the university level. Borek et al. [10] compared the impact on learning of different strategies to modulate the amount of help in an OELE for high-school chemistry course, and found that the best strategy is to provide help at every student errors and on-demand. They, however, neither did explicitly control for nor report the amount of help provided for different errors. Bouchet et al. [11] found that adapting the amount of feedback provided in an OELE for a university-level biology course can increase student performance, but also hinder their perception of the quality of the feedback. We extend these works by examining both the value and possible added distraction of providing one versus several interventions. We also focus on younger elementary-school students who might react to help repetition differently than the high-school and college students examined in the above work.

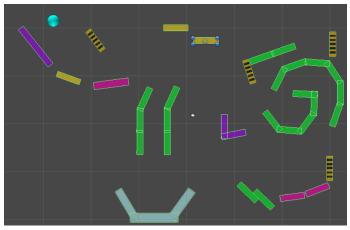


Fig. 1. Unity-CT environment.

3 UNITY-CT ENVIRONMENT TO FOSTER CT

Figure 1 shows a screenshot of Unity-CT. Unity-CT allows students to freely build games via an interactive scene view (Figure 1.1) in which they can manipulate game objects (organized in a hierarchy, Figure 1.2). Different manipulators (Figure 1.3) allow to interact with the objects in the scene (e.g., move, resize and rotate). For example, in Figure 1, the yellow inclined platform is selected with the rotate manipulator, and the student can directly perform the rotation with the mouse by dragging the white circle around the yellow platform. Object properties can also be directly modified in the inspector panel (Figure 1.4). Lastly, students can enter into Play Mode by clicking the Run button (Figure 1.5), to execute and test their game. They can exit the Play Mode using the same button. Unity-CT runs in the cloud and students access it through their Internet browser.

Table 1. Description of the challenge studied in this paper.

Challenge	Challenge instructions	Sample Solution
Challenge 1: Rube ramp	Create a ramp and bucket with different types of platform objects. Add a ball that must hit at least once each type of platform before landing in the bucket	

UME Academy's curriculum introduces CT skills that align with Brennan and Resnick [13] throughout the lessons, which are identified as suitable for young audiences. These skills are divided into higher-level problem-solving practices and perspectives that emerge during algorithmic and programming processes, and lower-level programming concepts employed for coding. Specifically, the first lesson focuses on two high-level practices: being incremental and iterative (to design algorithmic solutions step-by-step) and testing and debugging (trial and error processes to identify and remove malfunctions). Specifically, in the challenge, being incremental and iterative is fostered by the cycles of discovering the new objects at hand, imagining how to use them toward completing the challenge, and building a little bit of the solution before executing it. Testing and debugging is extensively encouraged by the instructors. Namely, the students usually have to repetitively modify an aspect of their game and test it in Play Mode, until they reach the desired outcome, e.g., in the challenge they may need to adjust the position of the different platforms many times until the ball lands in the bucket. This requires them to rotate certain objects in the scene correctly to allow the ball to reach its destination. In terms of programmatic concepts, the first lesson introduces variables, which are operationalized in Unity-CT via updating the value of specific properties of objects, such as their rotation angles which are displayed in the inspector when students perform a rotation in the scene (see Figure 1.4).

4 IPA FOR UNITY-CT

As a first step to assess how to effectively deliver adaptive help during open-ended GD activities, we designed an IPA that provides help on two errors the students frequently make in Unity-CT and created two versions of the IPA per each error via an iterative design process to examine the effect of help repetition. In one version, the IPA provides help

only once per error (1-Shot intervention version), whereas in the second version, the IPA provides help once more if the same error is repeated (Repeated intervention version).

This section provides details on the selection of these target behaviors, and the iterative design process for the two versions of the IPA.

4.1 Target Student Behaviors

The two student errors that we targeted for this study, Rotation in 3D (Rotation from now on) and Editing in Play Mode (Play Mode from now on), were identified by Unity-CT instructors and curriculum developers as being suitable test-beds for the provision of timely support during the first Unity-CT lesson.

The Rotation error happens when a student rotates an object in the third dimension within the 2D scene. This behavior distorts the image of the rotated object by displaying the 3D projection of the image in 2D. This happens because, although the Unity game engine supports 3D games, Unity-CT focuses on 2D games, deemed to be more appropriate for the pedagogical objectives of the course. The Play Mode error occurs when students edit their scene after entering "Play Mode" to execute their current game using the Play button in Unity-CT (shown in Figure 1.5). This behavior is problematic because Unity-CT does not record changes made to the scene while in Play Mode, so the changes are lost when Play Mode is exited. For both the Play Mode and Rotation errors, there are two different ways to correct them, listed in Table 2.

Table 2. Corrective Behaviors

Intervention	Corrective Behavior
Edit in Play Mode	• Exit play mode without further edits • Stay in play mode but stop editing the game
Rotation in 3D	• Undo the erroneous rotation with Ctrl+Z • Correctly rotate object

These two errors are fundamentally different in two aspects: 1) students' ability to immediately notice their errors and 2) the effort it takes to correct them. Firstly, Play Mode error is not immediately noticeable because of the minimal visual cues in the interface to remind the student they are in "Play Mode" (see image in Figure 2.A), but also the consequences of the error, i.e., not recording changes, are only discovered when the student exits Play Mode. In contrast, Rotation error is immediately visible to the students due to object distortion. Secondly, correcting the Play Mode error only requires the students to press the Play button and exit the Play Mode, which is the same action required to enter Play Mode. Contrarily, fixing the Rotation error requires the students to use a key combination (Ctrl+Z), which can be difficult for the young audience who are not proficient in using the keyboard, or have a different keyboard configuration. Moreover, correctly rotating the object also requires to perform a drag-and-drop as explained in Section 3, which is also challenging for the young audience according to the UME Academy's instructors.

Both these errors are prominent in the first lesson: based on historical data collected by UME Academy, 63% of students exhibit the Rotation error and 88% of students exhibit the Play Mode error, with 90% of students repeating the errors more than once. Instructors mentioned that these behaviors often confuse the students and result in them being

stuck. Both behaviors require timely interventions, but the instructors confirmed that they often are not able to detect them and respond in a timely manner.

We enabled the IPA to automatically track these behaviors by using a version of Unity-CT fitted to log students' actions. This data is synchronized in real time in a secure and password-protected database. Benchmarks were conducted to ensure that the logging mechanism has a negligible impact on Unity-CT's performance and caused no issue during data collection.

4.2 Design and Implementation of the IPA

The IPA and the format of its help interventions were co-created with the UME Academy's UX/UI team, to fit the configuration of Unity-CT while being effective in capturing the students' attention without being intrusive. Existing research shows that the perception and effectiveness of IPAs can vary depending on the visual indicators of gender, race, and level of realism, e.g., [7, 23]. We designed the IPA for Unity-CT to be non-gendered to avoid triggering any stereotypical assumptions, and a cartoon-like character to be engaging for the target age group. Figures 2, 3 and 4 show the final designs of the IPAs and its interventions when it is giving help for 1-Shot (Figure 2), and Repetition (Figure 3 & 4) groups. When the IPA is not giving help, only its hat is visible in the lower-right corner of the screen (see Figure 1.6).

The content of all the IPA interventions were determined via interviews with the Unity-CT instructors, to make sure the language was helpful and likeable for the age group without being confusing or too distracting. The interviews revealed that instructors tend to provide help incrementally, to encourage students to think about their problematic behaviors rather than just fixing them. Namely, instructors first flag the problem, following by an explanation of why the problem exists, and how to fix it if the student does not know what to do using the methods listed in Table 2. For instance, for the Rotate error they flag the problem by pointing out the distortion of the rotated object, provide an explanation of why the distortion happened, and finally suggest how to correct the error by undoing the faulty rotation and reminding the student of how to make a correct one. For the Play Mode error, when instructors detect the behavior, they warn the students that their changes will be lost because they are in play mode, followed by a suggestion to exit this Mode if they want to make further changes and not lose them.

4.2.1 Design of the 1-Shot IPA. To match the incremental way the instructors address the errors, in this version of the IPA we structured the help content provided for each of the two behaviors in the progression of speech bubbles shown in Figure 2 (A for Play Mode, B for Rotation), to be shown to the students only once, the first time they make the corresponding error. The speech bubbles are shown one at a time, where students can advance by clicking the buttons at the bottom of each bubble to see the next one. Note that for the Play Mode intervention (Figure 2.A), flagging the problem (i.e., changes will be lost) and explaining why the problem exists (editing in Play Mode) were combined in a single bubble, as instructors commented that decoupling the sequence would make it too fragmented.

Students can close the IPA intervention at any point with the exit button. The last bubble of every intervention, which tells the student how to fix the problem, also includes an animated image that visualizes the recommended action (see Figure 2.A, 2nd bubble, and Figure 2.B 3rd bubble). Specifically, the Play Mode animation (Figure 2.A) shows to the students that in Play mode the Run button is enabled in the control bar located above the scene (see Figure 1.5). The animation then shows that the students should click the Run button (highlighted in yellow) to exit play mode. The Rotation animation (Figure 2.B) shows how the selected object can be correctly rotated in 2D by using a drag-and-drop motion using the outer-most white circle that appears after being selected.[1]

[1]All animations are submitted as part of the supplementary material.

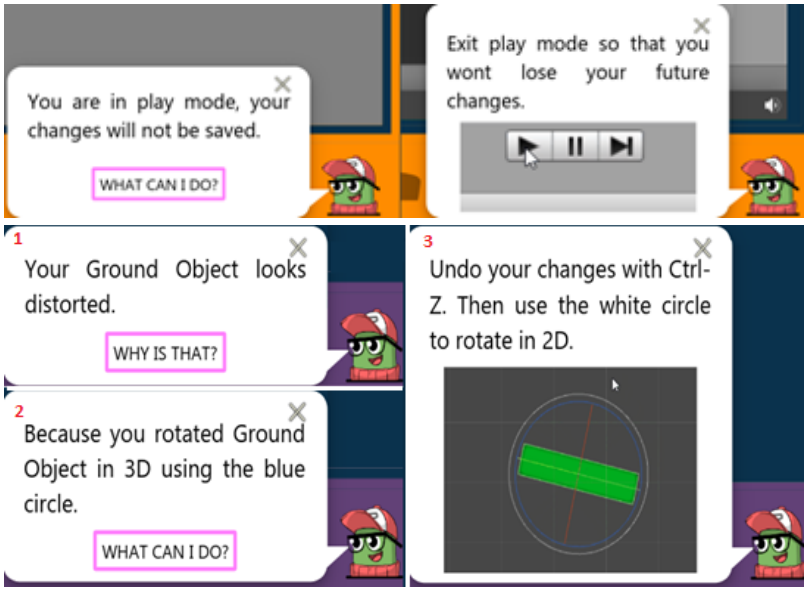


Fig. 2. Screenshots of the interventions for the 1-Shot intervention IPA. A) show the content of the Play Mode intervention; B) show the content of the Rotation intervention. Numbers in red indicate the order of speech bubbles, and are not visible to the students.

The 1-Shot IPA shows each intervention once per student, the first time the corresponding error was observed, for a maximum of two IPA interventions per student. We deployed the 1-Shot IPA in 13 classes with 56 students (more details will be provided in Section 5), and examined the usability of the IPA to ascertain whether there were any changes that would be needed to the design of the interventions for the second version of the IPA that provides repeated interventions. The initial feedback of the students about the perceived helpfulness of the IPA was very positive, but we identified two aspects for improvement. First, we observed that more than 75% of the students opted to close the interventions at the first speech bubble, meaning that less than a fourth of the students engaged into the entire flow and saw the last speech bubble with the IPA recommendation. We also observed that very few students who got to the last bubble in the Rotation intervention followed the recommendation to use Ctrl+Z to undo their rotation error (Figure 2.B, 3rd speech bubble), possibly because the animated gifs do not show this behavior. These findings were used to refine the design on the interventions in the version of the IPA agent that provides repeated help, described in the next section.

4.2.2 Design of the Repeated Intervention IPA. Unity-CT instructors commented that error repetition is common even when they provide help on the first occurrence of an error during classes, which they address this issue by re-iterating the help later in the class to strengthen student's knowledge. This behavior was confirmed when we deployed the 1-Shot IPA described in the previous section, where most students repeated each error after receiving the first and only intervention the IPA provides for that error. Thus, the Repeated Intervention version of the IPA delivers a second intervention to those students who repeat the same error after receiving an initial intervention. We limit the repetition to two because we want to ascertain how this amount of repetition works before trying longer sequences, and to avoid overwhelming the students with too many interventions.



(a) Play Mode (b) Rotation

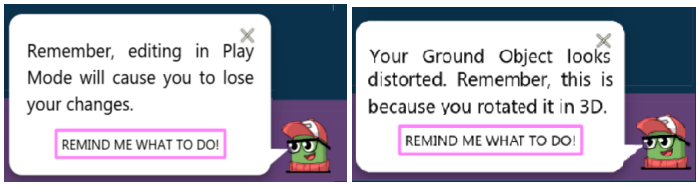
Fig. 3. Screenshots of the new design of the first interventions of Repeated intervention IPA for Play Mode (A) and Rotation (B) errors. Numbers in red indicate the order of speech bubbles, and are not visible to the students.

In addition to including a second intervention for each error, in this version of the IPA we also changed some aspects of the overall intervention design to address the two issues uncovered by the deployment of the 1-Shot IPA. To address the fact that students showed limited access to speech bubbles beyond the first in an intervention, we merged the first two of the three speech bubbles in the Rotation intervention, so that one single speech bubble now flags the problem and provides the reason for it (Figure 3.B). We also considered a design where all the information (problem, reason for it and how to fix it) is provided in one single speech bubble for each intervention, but decided against this design following the advice of both the Unit-CT instructors and UX designers, who said that children would be unlikely to process this much information at once. To address the issue that most students did not follow the 1-Shot IPA's advice to use Ctrl+Z to undo a rotation error, we included a new animated image in the last speech bubble for the rotation intervention, which exemplifies how to undo the last action with Ctrl+Z (Figure 3.B). For the Play Mode intervention, we changed the content of first speech bubble to include an exclamation, "Look out!", to better convey a sense of urgency in the hopes of better capturing students' attention and persuade them to engage in the intervention flow (Figure 3.A). Conveying this sense of urgency is justified by the fact that the Play Mode error can cause the students to lose several of their changes, which can be very discouraging and frustrating. We also reversed the content of the bubble by mentioning the issue ("Your changes will not be saved") before the explanation ("because you are in Play Mode"), so that the students focus directly on the risk of losing their work, which flows better after the "Look out!" warning.

The wording of the first speech bubbles in the repeated interventions was slightly changed from the first intervention (see Figure 4.A, 4.B), to highlight that the IPA is aware that it is showing the same suggestion again. Namely, we included the word "Remember" when describing the cause of the issue and changed the button to access the second speech bubble to "remind me what to do".

5 USER STUDY

To compare the two versions of the IPA interventions described in the previous section, we deployed them in 27 of the first classes on the Unity-CT curriculum held online by UME Academy from April 13th to August 4th, 2021. The classes were held as extracurricular activities and were open for registration for students ranging from grades 4 to



(a) Play Mode (b) Rotation

Fig. 4. Screenshots of the new design for first speech bubbles of the second interventions of Repeated intervention IPA for Play Mode (A) and Rotation Error (B). Second speech bubbles are same as Figure 3 for both error types and are not included.

6 and living in North America. The 1-Shot IPA was deployed in 13 classes held between April 13th and April 23rd, involving 56 students (M(SD)= 4.3 (1.75) students per class). These students form the 1-Shot Intervention group for our analysis. The Repeated Intervention IPA was deployed in 14 classes held between July 5th and August 4th, to match the number of students that worked with the 1- Shot IPA (M(SD)= 4 (1.75) students per class). These students form the Repeated Intervention group in our analysis. To evaluate the effectiveness of adaptive interventions compared to having no intervention at all, we created a No-Intervention group by selecting 56 students from 13 of the regular UME Academy's classes, to match both the number of students and average number of students per class in 1-Shot and Repeated Intervention groups (M(SD)= 4.3 (1.65) students per class). This was done to ensure there will be no effect of having classes with too many or too few students, which might impact how the instructors interact with students. The classes that were included in the No-Intervention group were selected from the dates between March 5th and 27th 2021, when the IPAs were not deployed. To maintain full ecological validity, we did not control for the demographics of the students in each group as part of the recruiting process, nor we prevented any students from participating in the classes. We ensured however that the UME Academy's enrolling process was the same across the groups and targeted the same key population (grade 4-6 North American students as said above). The inclusion of the IPA did not alter the lesson

Table 3. Content of the questionnaires.

Liking	
1) Did you like me	5-star rating
Perceived usefulness:	
2) Was I helpful?	5-star rating
3) Did I distract you?	Never/Sometimes/Always
4) Did I confuse you?	Never/Sometimes/Always
Intention to Reuse	
5) Would you like to see me again?	Yes/No

progression. The lessons took place in fully ecological settings, where the instructors proceeded as they normally do (described in Section 3), and no researcher attended the remote classrooms nor interacted with the students in any way. The students received no training nor specific instructions about the IPA. As detailed in Section 4.1, we logged all actions performed by the students and IPA interventions during the 30-minutes challenge phase.

At the end of each class, a short survey appeared on the screen of students who received at least one IPA intervention, to collect their perception of the IPA. The survey included 5 items to evaluate the levels of liking, perceived usefulness, and intention of future use of the students. Perceived usefulness is captured by three items that gauge if the IPA is perceived to be "helpful", "distracting" and "confusing" [34]. Similar items were used in related work on IPAs and

Table 4. Mean (SD) for the total number of actions of students for each intervention group.

Intervention Group	M (SD) Total Actions
No-Intervention	1704 (± 1138)
1-Shot	1453 (± 1032)
Repeated	1116 (±985)

adaptive support, e.g., [8, 29, 33]. Previous research shows that younger children should be presented with simple "Yes/No" questions and frequency-type Likert scales whenever possible [37]. Moreover, younger students are found to benefit from iconic representations for numbered scales (e.g., number of stars) [40]. We fine-tuned the wording of the survey items as well as the display of the scales, in accordance with these principles and in collaboration with UME Academy's instructors and UX/UI designers. The final questionnaire items and their corresponding scales are shown in Table 3.

6 DATA ANALYSIS AND RESULTS

In this section, we investigate the performance of the two IPA versions, by examining:

- objective measures of the impact of the IPA interventions, also compared to receiving no interventions, via the analysis of log data
- subjective measures of students' perception of the interventions, based on their answers to the usability survey described in Section 5.

For the analysis, we only look at students who made at least one error in the two groups with the IPA. This is because students who make no errors in these groups never see the IPA, thus there is no impact of the IPA's interventions to be measured. For consistency in the student samples, we excluded students who made no error in the No-Intervention group. We also had to remove data from 8 students from the 1-Shot intervention group and 10 students from the Repeated intervention group, who made errors but received no IPA help due to technical difficulties. After this removal of problematic students, we were left with a total of 123 students: 51 students in No-Intervention, 37 students in 1-Shot, and 35 students in Repeated. These students generated on average 873 actions per student (SD = 891), i.e., 0.48 action/second, during the class included in the study. Table 4 shows the average number of total actions per intervention group. As the differences in the groups may influence some of our analyses, we will be using the number of total actions as a covariate for the analyses that will be discussed in Section 6.1.2.

The students in the two IPA groups received an average of 1.38 interventions in the 1-Shot group, and 2.17 interventions in the Repeated group. Figure 5 shows the total number of first level interventions for the two groups, and of second level interventions for the Repeated Intervention group.

In each analysis that we perform, we remove outliers that are three standard deviations from the mean [20]. We will report the corresponding numbers in each section.

6.1 Objective Measures for evaluating the IPA

In OELEs such as Unity-CT it is challenging to objectively evaluate students' behavior by using classical performance measures such as test scores or percentage of completion, due to the lack of standardized solutions in creativity tasks such as Game Design activities. We therefore focused on objective measures that related to the impact of the IPA's interventions (or lack thereof) on the targeted error behaviors, Play Mode and Rotation. To do so, we analyzed the

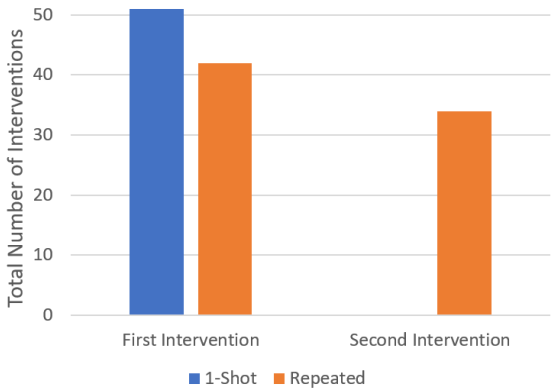


Fig. 5. Number of interventions received in total in the 1-Shot and Repeated group.

log data collected from the study to compare the amount of errors, error correction and correct behaviors in the three study group (No-Intervention, 1-Shot, Repeated). We isolate the impact of the first intervention and of the second intervention by looking at:

- their immediate impact on error correction behaviors of the students' first and second errors (Section 6.1.1).
- their longer-term impact on students' correct and incorrect behaviors after making their second errors (Section 6.1.2).

6.1.1 The Immediate Effect of the IPA interventions. To understand the immediate effect of the IPA intervention, we look at the students' immediate response to receiving help on an error, namely if they correct the error or not, compared to whether they correct the error spontaneously when they receive no intervention. We analyze the effect of interventions for the first error and the second separately, to gain specific insights on the impact of providing one hint, and of repeating it.

We mine the Unity-CT logs for the occurrence of the corrective behaviors mentioned in Section 4.1 (see Table 2) within the 20 actions that follow the reception of a corresponding intervention [2]. In the rest of this section, we evaluate the differences among the three groups in terms of whether students corrected their first and second errors using Logistic Regression, which is appropriate for modelling dichotomous dependent variables (corrected / not corrected) [20] [3].

First Error Correction. After the first error of each type (Play Mode and Rotation), students in both IPA groups receive an intervention. There is, however, a difference in what these two groups see, related to the changes in the design of both interventions described in section 4.2.2. Thus, comparing error correction after the first error for the three study groups will ascertain possible effects of both providing an intervention vs not, and of the design changes we made to the interventions. A total of 123 students made at least one error and are included in the analysis of corrections for the first error (51 students (91%) in No-Intervention, 37 students (77%) in 1Shot Intervention, 35 students (80%) in

[2]We chose this action window because we were not sure how long it would take students to comply, given that generally they do not need to fix the error right away to continue with the challenge. These 20 actions are performed on average in 58 seconds (SD = 48 sec).
[3]Throughout the paper we report statistical significance at the 0.05 level. For effect sizes, we used Cohen's [16] convention for standardized effects in pairwise comparisons, and report the effect sizes as large for d > 0.8, medium for d > 0.5, and small otherwise [20].

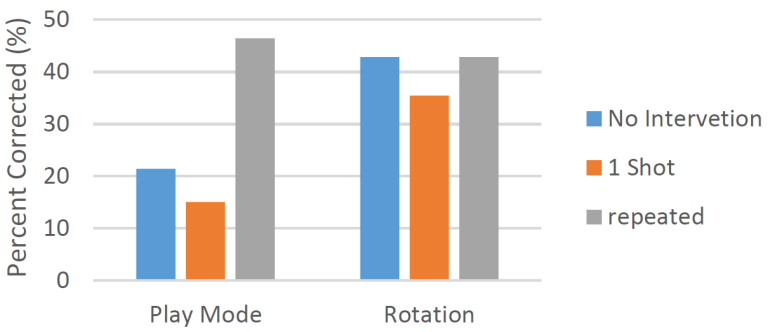


Fig. 6. Percent of students who corrected their first errors for each error type and intervention group.

Repeated Intervention). 90 (73%) of these students made the Play Mode error, and 73 (59%) of them made the Rotation error.

To formally compare the differences in error correction among groups, we fit two Logistic Regression models, one per error type (Rotation, Play Mode), with First Error Correction (two levels: corrected, not corrected) for the corresponding error type as the dependent variable, and Group (three levels: No-Intervention, 1-Shot, Repeated) as the factors. Figure 6 reports the proportions of students who corrected the 1st error per intervention group and error type.

For the Play Mode error, the logistic regression analysis indicated a significant main effect of Intervention Group ($\tilde{\chi}^2(2, N=90) = 7.12$, $p = .03$). The pairwise comparisons using the Tukey's HSD yielded no significant differences among the groups after adjustment of the p-values. There are, however, marginally significant pairwise comparisons (p-values between 0.05 and 0.1), with large effect sizes, indicating that more students corrected their errors in the Repeated intervention group than in the one 1-Shot group (Z=2.17, p=.07; d=1.59), and in the No-Intervention group (Z=2.16, p=.07; d=1.16). These effects with large effect sizes reflect the clear trend shown in Figure 6 (left), that the proportion of corrected first errors (46%) is more than two times higher than the 1-Shot (15%) and the No-Intervention (21%) group for Play Mode. It is very likely that this difference of Repeated group vs 1-Shot and No-Intervention is captured by the significant main effect yielded by the Logistic Regression model. Altogether, these results suggest that the Repeated intervention likely has a positive effect on first error correction for Play Mode, which could be further verified in future analysis with a larger sample size. No significant effect of group was found for Rotation Error on error correction of the first error ($\tilde{\chi}^2(2, N=73) = 0.40$, $p=.8$).

Second Error Correction. We perform a similar analysis for the immediate effect of the repeated intervention, by looking at how students in the three groups reacted after making their second errors. Here, only the students in the Repeated intervention group receive an intervention for their second error, but we keep both the other two groups in the analysis in case there is an effect on the second error behaviors of having received an intervention (1-Shot group) vs not (No-Intervention) after the first error. A total of 114 students (92% of the 123 students that made at least one error) continued to make the second error and are included in the analysis of corrections for the second error. The distribution of students who repeated their errors is similar in the groups (49 students (96%) in No-Intervention, 35 students (94%) in 1-Shot Intervention, 30 students (86%) Repeated Intervention). 76 (67%) of these students repeated the Play Mode error, and 60 (53%) of them repeated the Rotation error.

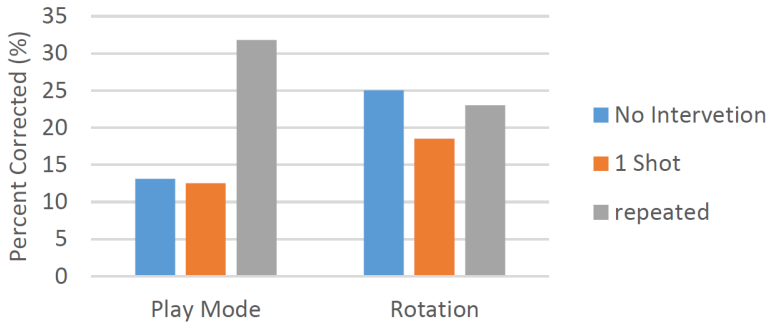


Fig. 7. Percent of students who corrected their second errors per interaction group and error type.

A logistic regression analysis (similar to First Error Correction above) with Second Error Correction as the dependent variable indicated no significant effect of groups neither for Play Mode ($\tilde{\chi}^2$(2, N=76) = 5.50, p = .06), nor for Rotation errors ($\tilde{\chi}^2$(2, N=60) = 0.30, p=.8). However, the close to significant results for the Play Mode error is worth exploring more with additional data. Figure 7 indicates the proportions of students who corrected their 2nd error per intervention group and error type.

6.1.2 Longer-term effect of IPA interventions . To ascertain whether there is any lasting effect of the IPA interventions after they are no longer provided, we also look at the overall number of errors, number of correct behaviors, and error correction rates after the second error for each error type. Number of errors [4] and of correct behaviors are complementary because together they give a sense of how much a student tries and succeeds (or fails) to use the two behaviors targeted in the study. Error correction rate shows whether the students can spontaneously fix the errors they make after the second one. This is to measure whether the interventions are successful in teaching the students how to recognize and fix their errors when they occur.

For the analysis of error counts, one student was removed as an outlier (from Repetition group, Rotation Error), with a value of 30 errors. For the analysis of correct behavior count, four student data were removed as outliers from the analysis (1 No-Intervention, 2 1-Shot and 1 Repetition group, all behaviors related to Rotation error). For the error correction rates, there were no outliers.

To investigate the differences among the three study groups in terms of error counts and correct behavior counts, we used Negative Binomial Regressions (NBR), which is suitable for count data that is overdispersed [14] [5]. Namely, we fit two NBR models for each error type, with Error Count and Correct Behavior Count as the dependent variables, Intervention Group as a fixed effect (three levels) and the number of total actions as a covariate, to account for the differences in total actions performed across the groups as reported above in Table 3. For error correction rates, we use a Beta regression model, which is suitable for analysis of proportional data [19]. Namely, we fit one Beta Regression models for each error type, where Correction Rate is included as a dependent variable, and Group as the independent

[4]We use counts and not rates over the total number of actions performed during the lesson because this number is very large compared to the number of error/correct behaviors for the two target actions. The resulting very small rates would not allow us to get a good sense of how much the students are using the behaviors. However, we use the total actions as a covariate in all our count analyses to account for possible differences in the total number of actions performed by each student.

[5]To test whether these distributions were negative binomially distributed, a Chi-Squared goodness-of-fit test was run on error count and total correct behavior count for all levels of condition. All results were statistically non-significant, indicating non-detectable deviations from a negative binomial distribution for all groups for both dependent variables.

variable. To measure the correction rate, we use the exact same approach we described in Section 6.2.1. We report the outputs of these models and subsequent post-hoc analysis next, starting with error correction rate.

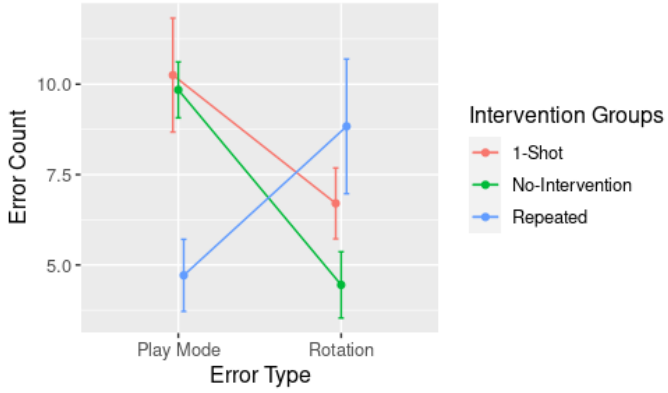


Fig. 8. Error Counts after the second error for each intervention and error type.

Error Correction Rates. The Beta regression test showed no significant main effects of intervention group, neither for Play Mode ($\tilde{\chi}^2$(2, N=66) = 2.50, p=.28), nor the Rotation error ($\tilde{\chi}^2$(2, N=54) = 1.95, p=.37), indicating no measurable lasting effect of the interventions on error correction.

Error count. Figure 8 shows the number of errors made by each intervention group, with the error count averaged across students in each group, for both the Play Mode (left) and Rotation Error (right). The NBR for error count indicated a significant main effect of group ($\tilde{\chi}^2$(2, N=75) = 23.28, p < .001) for Play Mode. Pairwise comparisons using Tukey's HSD reveal the following effects. For Play Mode, the error counts were significantly lower for 'Repetition' group (M(SD)=4.71(4.56)) compared to both 'No Intervention' (Z = 4.83, p < .001; d=1.46) and '1-Shot Intervention' (Z =2.97, p < .01; d=1.03) groups, with large effect sizes. No significant difference was found between 'No Intervention' (M(SD)=9.84 (4.77)) and '1-Shot' (M(SD)=10.25 (6.30)) groups (Z=1.45, p=.3; d=0.4) in the Play Mode error. For Rotation error, no significant effect of group was observed ($\tilde{\chi}^2$(2, N=59) = 5.17 p = .07).

Correct behavior count. [6]Figure 9 shows the number of correct behaviors made by each intervention group, with the correct behavior counts averaged across students in each group, for both the Play Mode and Rotation Error. The negative binomial regression for count of correct behaviors indicated a main effect of intervention group for Play Mode error ($\tilde{\chi}^2$(2, N=75) =9.35, p < .01). Pairwise comparisons using Tukey's HSD test indicated that the counts of correct behaviors were significantly higher for 'Repetition' group (M(SD)= 7.67 (8.92)) compared to 'No Intervention' (M(SD)= 2.55 (3.66)) group (Z = 2.98, p < .01; d=1.09) with a large effect size, but not between '1-Shot' group (M(SD)= 4.25 (8.23)) and others with small to medium effect sizes. For Rotation error, no significant effect of intervention group was observed ($\tilde{\chi}^2$(2, N=57) = 0.27, p =.8).

6.2 Subjective Measures for Evaluating the IPA

Of the 72 students who received IPA interventions (37 from 1-Shot Intervention group, 35 from Repeated Intervention group), 29 (40%) completed the usability survey displayed at the end of the class (see Section 5). This response rate is

[6]For completeness of analyses, we also looked at the correct behaviors between 1st and 2nd errors but found no significant differences across the groups.

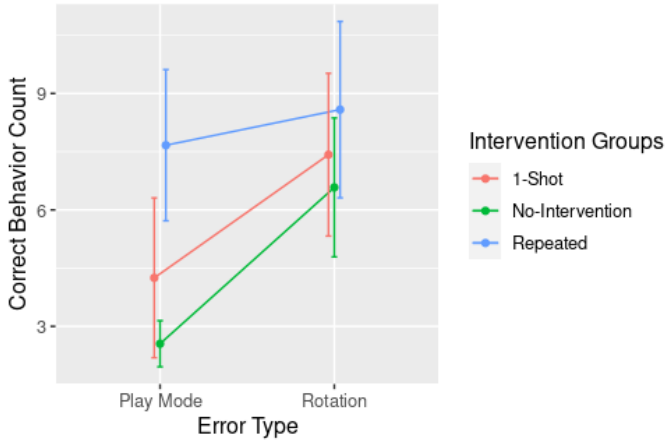


Fig. 9. Correct Behavior Counts after the 2nd errors for each intervention group and error type.

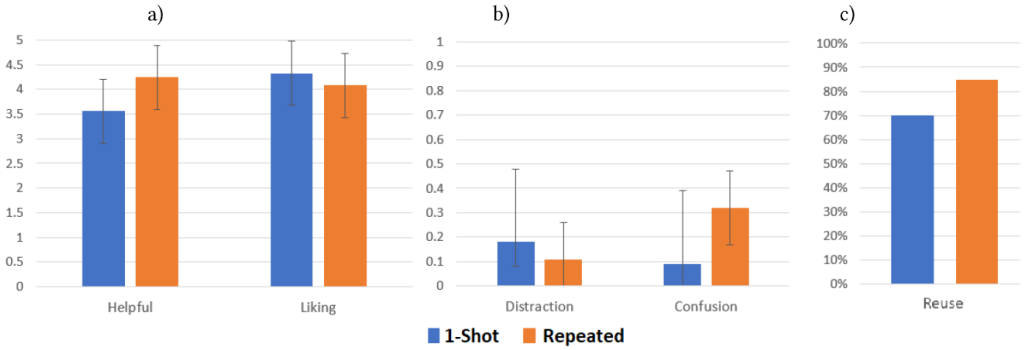


Fig. 10. Summary statistics over the survey answers.

consistent with what is usually observed in user studies with remote, online surveys, as it is well-known that most users don't bother filling them [17]. This said, the number of students who provided feedback fully satisfies the sample size recommendations for usability studies [3, 26]. All students who completed the survey received interventions for both error types.

Figure 10 shows descriptive statistics of the students' answers to the usability survey. To generate statistics over the categorical items in the survey for Confusion and Distraction, we code the students' answers as 0 for "Never", 0.5 for "Sometimes", and 1 for "Always", thus obtaining scores ranging from 0 to 1. We formally compare the Likert-scale ratings among the 1-Shot and Repeated group using Ordinal Logistic Regression (OLR), which is appropriate to handle Likert scale dependent variables [20]. Namely, for each item on the survey except the Reuse item (see below), we fit an OLR model with the students' answers to that item as the dependent variable, and group (two levels: 1-Shot, Repeated) as the factor. For the Reuse item, where students rated their preferences on seeing the IPA again in the next lesson in a binary way rather than with a Likert scale, we used Fisher's exact test [21] to compare a categorical dependent variable

("Yes", "No") by a categorical independent variable (two levels: 1-Shot, Repeated), which is suitable is appropriate for binary dependent variables with small sample sizes [21, 25].

Figure 10 indicates that the students' perception of the IPA intervention was very positive for both IPA groups. Almost all students liked the IPA (1-Shot M(SD)= 4.33 (±1.32), Repeated M(SD)= 4.08 (±1.31)), and found it helpful (1-Shot M(SD)= 3.56 (±1.94), Repeated M(SD)= 4.25 (±1.36)) (see Figure 10-a). No significant differences were observed for helpfulness ($\tilde{\chi}^2(1) = 0.64$, p = .4) or liking ($\tilde{\chi}^2(1) = 0.49$, p = .5) items across the groups. Students also did not find the IPA to be distracting (mean scores of 0.1 for Repeated, 0.18 for 1-Shot, which corresponds to "never distracting" in the survey), and there was no statistical difference among the groups ($\tilde{\chi}^2(1) = 0.89$, p = .3), indicating that the delivery of the intervention was well calibrated in both groups (see Figure 10-b).

However, students found that the IPA in the Repeated Intervention group to be significantly more confusing (M(SD)= 0.32 (±0.3)) than the students from the 1-Shot Intervention group (M(SD)= 0.09 (± 0.3)) ($\tilde{\chi}^2(1) = 4.198$, p = .04).

Lastly, 82% of the students rated that they would like to reuse the IPA again in future lessons (70% 1-Shot, 85% Repeated Intervention groups, see Figure 10-c). Fisher's exact test indicated no significant differences between groups in terms of reuse preference ($\tilde{\chi}^2(1, N=25) = 0.09$, p=.7).

7 DISCUSSION

We discuss the main takeaways from our analysis in this section, starting with the findings on the subjective measures, followed by the findings on objective measures. We also discuss ways to improve the IPA based on the findings, which could be the target of future evaluation of the IPA.

Subjective measures. The subjective measures to evaluate both IPA versions showed students had an overall positive perception of the agents. We found that students liked both agents, found them helpful and would prefer seeing the IPA again in future lessons. The results for distraction were also very encouraging, because a known pitfall of real-time interventions is that they can be seen as disruptive during the task. This is even more crucial because in the Repeated group students could receive up to four interventions (2 per error type), which could have generated some levels of distraction. Here, our results show that it was not the case, thus confirming that our strategy for providing the repeated interventions (see Section 4) was effective. However, students found the Repeated IPA more confusing than the 1-Shot, which suggests that even though there might be room for delivering more interventions, it is important to control for the possible confusion that the additional interventions could generate, if not done carefully. This said, the confusion remained on the low side even for the Repeated group (0.3 on average is in-between "Never confusing" and "Sometimes Confusing"), suggesting that the reasons for the confusion were overall not too severe, and future work could focus on identifying what caused the increase in confusion.

Objective measures. The objective results allowed us to see more fine-grained differences in the effectiveness of the IPAs on improving student behaviors. Our results showed that, overall, 1-Shot Intervention failed to significantly improve any of the students' behaviors related to both errors, compared to the No-Intervention group. On the other hand, the Repeated Interventions had promising results in effectively improving student's both immediate and long-term behaviors, albeit only for the Play Mode error. Table 5 summarizes the outcomes of the analyses of objective measures for intervention groups with significant main effects.

The results on the immediate effects of the IPA intervention on error correction show that the changes that were made to the Repeated intervention groups' first intervention for Play Mode error (see Section 4.2.2) had a positive effect on the corrections students made after the first intervention, as compared to the 1-Shot and No-Intervention groups, although the effect was not significant. Namely, our revisions on the content of the first speech bubble to convey a

Table 5. Summary of Pairwise Comparisons for the analyses of objective measures per each error type. The ">" sign indicates trends. Trends that are underlined are not statistically significant. Results with non-significant main effects are not included.

Objective Measures	Error Type	Pairwise Comparisons
First Error Correction	Play Mode	Repetition group>No-Intervention group>1-Shot group
Error Count	Play Mode	No-Intervention group > 1-Shot group > Repetition group
Count of Correct Behavior	Play Mode	Repetition group > 1-Shot group > No-Intervention group

sense of urgency (see Section 4.2.2) seems to have been successful in helping students to correct their first errors. This finding is interesting as it shows that it is important to carefully examine the effects of different wording may have on the student's compliance and engagement with the interventions. However, both IPAs were not successful enough to affect students' error correction behaviors better than the No-Intervention group for both the 2nd error and long-term corrections. Moreover, the interventions for the Rotation error failed to improve the students' error corrections in both the first and second intervention groups. These results could be due to the fundamental difference between the two error types (see Section 4.1) in terms of their corrections, where the interventions were not as effective due to either the difficulty of the corrective actions for Rotation error, or the content of intervention failing to convey the intended message. These results also show the importance of evaluating each intervention individually, to identify those that are not effective and need to be refined, which is consistent with previous findings on IPAs in OELEs [29].

Our analyses showed that repeating the Play Mode intervention helped with making significantly fewer Play Mode errors after the second intervention, as compared to both No-Intervention and 1-Shot intervention (first row in Table 5). This shows that for Play Mode error, the 1-Shot intervention was not sufficient to discourage this error and the repeated intervention was necessary for the students to avoid this error. The Repeated group also performed significantly more correct Play Mode behaviors than the No-Intervention group, showing that the repeated interventions were useful in helping the students use the Play Mode correctly. This is perhaps because, as mentioned in Section 4.1, there is no obvious visual cues that they are editing in Play Mode and might lose their changes, and thus it was useful to provide a reminder to always check for the status of the Run button. It is also possible that without the IPA intervention in the No-Intervention group, some students might not even notice that they are losing some of their changes, and thus reiterating the issue in the Repeated group might have helped understanding this error.

However, the effect was not present for Rotation errors, namely neither of the interventions could not significantly reduce the number of errors made after the second error than no intervention. This is possibly due to the complexity of the Rotation action, especially for young students who are not familiar with using a mouse and keyboard. Indeed, as elaborated in the paper, detecting the Rotation error is more straightforward than the Play Mode, even for the No-Intervention group due to obvious object distortion. However, correcting and performing a correct rotation is challenging to operate for this young audience by using a complex keyboard configuration and performing precise drag-and-drops. Even if the IPA interventions showcase how to perform these corrective behaviors, perhaps it was not sufficient for the students to master them. In future work, it might be worth experimenting with other ways to show how to rotate, such as directly highlighting in the Unity scene which of the outer circle they must drag-and-drop to perform the rotation. It is also possible that students understand the error but are not interested in fixing it because it does not prevent them from completing the challenge, and they can just leave the distorted object in their scene. Future

work could thus also focus on better conveying why it is important to learn how to rotate, for instance by providing students with dedicated activities where they must use a correct rotation before they can go back to building their game.

Overall, our results point towards the importance of reiterating intervention content for reducing the number of errors and increasing the number of correct behaviors made by the students, depending on the target error type. However, the lack of strong evidence for long-term error correction rates for both errors might suggest that, having the corrective action hint in the last speech bubble for both interventions might have caused the students to miss the hints. A future study with IPAs only providing corrective action hints for their repeated interventions might help investigating this issue further. Although we tested the IPA on two specific errors that students commonly make when working with Unity-CT, as opposed to targeting the full set of sub-optimal behaviors identified by the data-driven student model in [31]. This approach allows us to reduce confounds due to inaccuracies in the student model or other possible issues related to providing a large variety of feedback. For future work, we aim to utilize the AI-driven student models we built in [31] by applying to Anon-CT a data-driven framework for user modeling and adaptation (FUMA, [29]) and showing that FUMA could infer from data a variety of non-obvious suboptimal student behaviors that call for personalized help.

8 CONCLUSION

In this paper, we examined the factors that affect providing automated interventions with IPAs in free-form GD environments. We particularly focused the effect of repetition in providing such interventions by comparing two versions of an IPA, one of which provides a single intervention (1-Shot) and the other, two interventions (Repeated). We do so by iteratively designing and deploying the two versions of the IPA in a real-world online OELEs that is aimed to foster CT in early K-12 education using free-form GD: Unity-CT. We evaluated the effectiveness of each IPA by comparing the behaviors of the students who received both 1-Shot and Repeated interventions to the students who did not receive any interventions. Our results showed that students overall rated both versions of the IPA positively in their subjective evaluations. We also found promising results on the effect of repeating an intervention on lowering error counts, and increasing the correct behaviors and error corrections for one error type (Play Mode); where 1-Shot interventions failed to do. No such effects of providing either one or two interventions were found for the other error type we consider (Rotation). Overall, we found that our results are highly dependent on the nature of the errors, and the design of the intervention content to be effective, calling for careful examination of these aspects in future work on adaptive IPA in OELEs. Altogether, our work is a step towards leveraging the great potential of IPAs in facilitating learning with OELEs for a young audience, in a remote setting that has become the new normal worldwide with the Covid pandemic.

Based on our findings we provide ideas on how to refine the design of the IPA, which we plan to test in future work. We also envision to design data-driven interventions based on complex behaviors learned by mining the student's data, as we did for instance in [31]. Lastly, we plan to study students' willingness to spontaneously request the IPA's support, and how this would impact learning.

ACKNOWLEDGMENTS

Research reported in this paper was supported by Mitacs (award number IT13336) and UME Academy Ltd.

REFERENCES

[1] Mete Akcaoglu and Matthew J. Koehler. 2014. Cognitive outcomes from the Game-Design and Learning (GDL) after-school program. *Computers & Education* 75 (2014), 72–81. Publisher: Elsevier.

[2] Vincent Aleven. 2013. Help seeking and intelligent tutoring systems: Theoretical perspectives and a step towards theoretical integration. (2013), 311–335.

[3] Roobaea Alroobaea and Pam J Mayhew. 2014. How many participants are really enough for usability studies?. In *2014 Science and Information Conference*. IEEE, 48–56.

[4] Otávio Azevedo, Felipe de Morais, and Patricia A Jaques. 2018. Exploring gamification to prevent gaming the system and help refusal in tutoring systems. In *European conference on technology enhanced learning*. Springer, 231–244.

[5] 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 (2011), 48–54. Publisher: ACM New York, NY, USA.

[6] Satabdi Basu, Gautam Biswas, and John S. Kinnebrew. 2017. Learner modeling for adaptive scaffolding in a Computational Thinking-based science learning environment. *User Modeling and User-Adapted Interaction* 27, 1 (March 2017), 5–53.

[7] Amy L Baylor. 2009. Promoting motivation with virtual agents and avatars: role of visual presence and appearance. *Philosophical Transactions of the Royal Society B: Biological Sciences* 364, 1535 (2009), 3559–3565.

[8] Timothy Bickmore, Ha Trinh, Michael Hoppmann, and Reza Asadi. 2016. Virtual agents in the classroom: experience fielding a co-presenter agent in university courses. In *International Conference on Intelligent Virtual Agents*. Springer, 154–163.

[9] Gautam Biswas, James R. Segedy, and John S. Kinnebrew. 2013. Smart Open-Ended Learning Environments That Support Learners Cognitive and Metacognitive Processes. In *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data (Lecture Notes in Computer Science)*, Andreas Holzinger and Gabriella Pasi (Eds.). Springer, Berlin, Heidelberg, 303–310.

[10] Alexander Borek, Bruce M. McLaren, Michael Karabinos, and David Yaron. 2009. How much assistance is helpful to students in discovery learning?. In *Proceedings of the 4th European Conference on Technology Enhanced Learning*. Springer, Nice, France, 391–404.

[11] François Bouchet, Jason M Harley, and Roger Azevedo. 2013. Impact of different pedagogical agents' adaptive self-regulated prompting strategies on learning with MetaTutor. In *International Conference on Artificial Intelligence in Education*. Springer, 815–819.

[12] François Bouchet, Jason M Harley, and Roger Azevedo. 2016. Can adaptive pedagogical agents' prompting strategies improve students' learning and self-regulation?. In *International conference on intelligent tutoring systems*. Springer, 368–374.

[13] Karen Brennan and Mitchel Resnick. 2012. New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada*, Vol. 1. 25.

[14] A Colin Cameron and Pravin K Trivedi. 2013. *Regression analysis of count data*. Vol. 53. Cambridge university press.

[15] Nur Akkuş Çakır, Arianna Gass, Aroutis Foster, and Frank J. Lee. 2017. Development of a game-design workshop to promote young girls' interest towards computing through identity exploration. *Computers & Education* 108 (2017), 115–130. Publisher: Elsevier.

[16] Jacob Cohen. 2013. *Statistical power analysis for the behavioral sciences*. Routledge.

[17] Elisabeth Deutskens, Ko De Ruyter, Martin Wetzels, and Paul Oosterveld. 2004. Response rate and response quality of internet-based surveys: an experimental study. *Marketing letters* 15, 1 (2004), 21–36.

[18] Sidney D'Mello and Art Graesser. 2012. Emotions during learning with AutoTutor. *Adaptive technologies for training and education* (2012), 169–187.

[19] Silvia Ferrari and Francisco Cribari-Neto. 2004. Beta regression for modelling rates and proportions. *Journal of applied statistics* 31, 7 (2004), 799–815.

[20] Andy Field, Jeremy Miles, and Zoë Field. 2012. *Discovering statistics using R*. Sage publications.

[21] Ronald A Fisher. 1922. On the interpretation of χ 2 from contingency tables, and the calculation of P. *Journal of the Royal Statistical Society* 85, 1 (1922), 87–94.

[22] Sebastian Gross and Niels Pinkwart. 2015. Towards an integrative learning environment for Java programming. In *2015 IEEE 15th International Conference on Advanced Learning Technologies*. IEEE, 24–28.

[23] Agneta Gulz and Magnus Haake. 2010. Challenging gender stereotypes using virtual pedagogical characters. In *Gender Issues in Learning and Working with Information Technology: Social Constructs and Cultural Contexts*. IGI Global, 113–132.

[24] Michael J Hannafin, Craig Hall, Susan Land, and Janette Hill. 1994. Learning in open-ended environments: Assumptions, methods, and implications. *Educational Technology* 34, 8 (1994), 48–55.

[25] Avijit Hazra and Nithya Gogtay. 2016. Biostatistics Series Module 4: Comparing Groups - Categorical Variables. *Indian Journal of Dermatology* 62, 4 (2016), 385–392.

[26] Wonil Hwang and Gavriel Salvendy. 2010. Number of people required for usability evaluation: the 10±2 rule. *Commun. ACM* 53, 5 (2010), 130–133.

[27] W Lewis Johnson, Erin Shaw, Andrew Marshall, and Catherine LaBore. 2003. Evolution of user interaction: the case of agent adele. In *Proceedings of the 8th international Conference on intelligent User interfaces*. 93–100.

[28] Yasmin B. Kafai and Quinn Burke. 2015. Constructionist gaming: Understanding the benefits of making games for learning. *Educational psychologist* 50, 4 (2015), 313–334. Publisher: Taylor & Francis.

[29] Samad Kardan and Cristina Conati. 2015. Providing adaptive support in an interactive simulation for learning: An experimental evaluation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. 3671–3680.

[30] Sébastien Lallé, Cristina Conati, Roger Azevedo, Nicholas Mudrick, and Michelle Taub. 2017. On the Influence on Learning of Student Compliance with Prompts Fostering Self-Regulated Learning. *International Educational Data Mining Society* (2017).

[31] Sébastien Lallé, Özge Nilay Yalçın, and Cristina Conati. 2021. Combining Data-Driven Models and Expert Knowledge for Personalized Support to Foster Computational Thinking Skills. In *LAK21: 11th International Learning Analytics and Knowledge Conference*. 375–385.

[32] Susan M Land. 2000. Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research and Development* 48, 3 (2000), 61–78.

[33] Christine Lisetti, Reza Amini, Ugan Yasavur, and Naphtali Rishe. 2013. I can help you change! an empathic virtual agent delivers behavior change health interventions. *ACM Transactions on Management Information Systems (TMIS)* 4, 4 (2013), 1–28.

[34] Arnold M Lund. 2001. Measuring usability with the use questionnaire12. *Usability interface* 8, 2 (2001), 3–6.

[35] Ati Suci Dian Martha and Harry B Santoso. 2019. The design and impact of the pedagogical agent: A systematic literature review. *Journal of Educators Online* 16, 1 (2019), n1.

[36] Kathryn S. McCarthy, Micah Watanabe, Jianmin Dai, and Danielle S. McNamara. 2020. Personalized learning in iSTART: Past modifications and future design. *Journal of Research on Technology in Education* 52, 3 (2020), 301–321. Publisher: Taylor & Francis.

[37] David Mellor and Kathleen A Moore. 2014. The use of Likert scales with children. *Journal of pediatric psychology* 39, 3 (2014), 369–379.

[38] Roxana Moreno, Richard Mayer, and James Lester. 2000. Life-like pedagogical agents in constructivist multimedia environments: Cognitive consequences of their interaction. In *EdMedia+ Innovate Learning*. Association for the Advancement of Computing in Education (AACE), 776–781.

[39] Thomas Price, Rui Zhi, and Tiffany Barnes. 2017. Evaluation of a Data-Driven Feedback Algorithm for Open-Ended Programming. *International Educational Data Mining Society* (2017). Publisher: ERIC.

[40] Laura Reynolds and Robert Johnson. 2011. Is a picture is worth a thousand words? Creating effective questionnaires with pictures. *Practical Assessment, Research, and Evaluation* 16, 1 (2011), 8.

[41] Ido Roll, Vincent Aleven, Bruce M McLaren, and Kenneth R Koedinger. 2011. Improving students' help-seeking skills using metacognitive feedback in an intelligent tutoring system. *Learning and instruction* 21, 2 (2011), 267–280.

[42] Jonathan Rowe, Bradford Mott, Scott McQuiggan, Jennifer Robison, Sunyoung Lee, and James Lester. 2009. Crystal island: A narrative-centered learning environment for eighth grade microbiology. In *workshop on intelligent educational games at the 14th international conference on artificial intelligence in education, Brighton, UK*. 11–20.

[43] Leonid Sheremetov and Adolfo Guzman Arenas. 2002. EVA: an interactive Web-based collaborative learning environment. *Computers & Education* 39, 2 (2002), 161–182.

[44] Mohamed Soliman and Christian Guetl. 2013. Implementing Intelligent Pedagogical Agents in virtual worlds: Tutoring natural science experiments in OpenWonderland. In *2013 IEEE Global Engineering Education Conference (EDUCON)*. IEEE, 782–789.

[45] John Stamper, Tiffany Barnes, Lorrie Lehmann, and Marvin Croy. 2008. The hint factory: Automatic generation of contextualized help for existing computer aided instruction. In *Proceedings of the 9th International Conference on Intelligent Tutoring Systems Young Researchers Track*. 71–78.

[46] Kurt VanLehn, Collin Lynch, Kay Schulze, Joel A Shapiro, Robert Shelby, Linwood Taylor, Don Treacy, Anders Weinstein, and Mary Wintersgill. 2005. The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence in Education* 15, 3 (2005), 147–204.