Gamification of Joint Student/System Control over Problem Selection in a Linear Equation Tutor

Yanjin Long and Vincent Aleven

Human Computer Interaction Institute, Carnegie Mellon University,
5000 Forbes Avenue, Pittsburgh, PA, 15213
{ylong,aleven}@cs.cmu.edu

Abstract. Integrating gamification features in ITSs has become a popular theme in ITSs research. This work focuses on gamification of shared student/system control over problem selection in a linear equation tutor, where the system adaptively selects the problem *type* while the students select the individual problems. In a 2x2+1+1 classroom experiment with 267 middle school students, we studied the effect, on learning and enjoyment, of two ways of gamifying shared problem selection: performance-based rewards and the possibility to redo completed problems, both common design patterns in games. We also included two ecological control conditions: a standard ITS and a popular algebra game, *DragonBox 12+*. A novel finding was that of the students who had the freedom to re-practice problems, those who were not given rewards performed significantly better on the post-tests than their counterparts who received rewards. Also, we found that the students who used the tutors learned significantly more than students who used *DragonBox 12+*. In fact, the latter students did not improve significantly from pre- to post-tests on solving linear equations. Thus, in this study the ITS was more effective than a commercial educational game, even one with great popular acclaim. The results suggest that encouraging re-practice of previously solved problems through rewards is detrimental to student learning, compared to solving new problems. It also produces design recommendations for incorporating gamification features in ITSs.

Keywords: *DragonBox*, educational games, student control, shared control, intelligent tutoring systems, algebra, classroom evaluation, rewards.

1 Introduction

In recent years, Intelligent Tutoring System (ITS) researchers have started to investigate how to integrate game elements within a tutoring environment. The goal is typically to make the system more engaging for students, while maintaining its effectiveness in supporting learning. Empirical studies have been conducted to evaluate the effects of gamifying tutors on students' learning and motivation, as well as to explore the best design to incorporate game elements in tutors. Some studies have found that game-based learning environments can significantly enhance students' learning outcomes [3, 10] and can produce the same learning effects as nongame tutors [7]. However, gamification of ITSs is not always successful. For example, one study [5]

S. Trausan-Matu et al. (Eds.): ITS 2014, LNCS 8474, pp. 378–387, 2014.
© Springer International Publishing Switzerland 2014

found that tutor-like assistance led to better learning and interest as compared to game-like assistance in an educational game of policy argument. Therefore, gamification of ITSs should be done with care, where possible informed by empirical studies.

Student control over problem selection may be an interesting area for gamification. *Full* student control over problem selection tends to be detrimental for learning (see e.g., [2]). However, *shared* control between student and system has shown some promise. Simple forms of shared control, in which the system and the students share the responsibilities to select problems in the system, had led to comparable learning as full system control [4, 9]. However, these simple techniques may not be as engaging as they could be, nor do they take full advantage of ITSs' ability to make good problem selection decisions. In the current work, we focus on a form of shared control in which the system selects problem types and decides when students have mastered each problem type and may go on to the next, while the student selects individual problems from a certain problem type. We try to improve on this form of shared control by adding gamification features, and investigate whether the gamified shared control leads to higher engagement and better learning.

Commercial games provide plenty of ideas for gamification of problem selection. A feature found in many popular games (e.g., *Angry Birds, DragonBox*) is the possibility to re-do problems after they have been completed. This feature is often combined with rewards (such as a number of stars) that reflect performance on the given problem. One reason players may elect to re-do a problem is to increase the rewards. According to theories of autonomy in learning [6], allowing re-practice gives students more freedom, which could possibly enhance their engagement in learning. Moreover, re-practicing could lead to more efficient acquisition of problem-solving skills, although to the best of our knowledge that has not been established definitively in the cognitive science literature. On the other hand, frequent re-practice may reduce problem variability and therefore be detrimental for learning [11]. Empirical investigation of the effectiveness of these gamification features is therefore warranted.

In the current work, we investigate the effects of gamifying shared student/system control in our linear equation tutor, *Lynnette*. We investigated two gamification features: giving students the freedom to re-practice previously completed problems (not allowed e.g., in standard Cognitive Tutors) and rewards (stars) for each problem based on students' performance. These features are similar to *Angry Birds*' or *DragonBox*' problem selection and rewards systems. We hypothesize that 1) the possibility to re-practice problems, added to shared control over problem selection will enhance students' learning and engagement; 2) rewards based on students' performance on individual problems will also lead to better learning and engagement. Consequently, we created four experimental versions of *Lynnette* to evaluate the effects of the two gamification features. Moreover, we included two ecological control conditions in the study: a standard ITS and a commercial algebra game. The standard ITS is a control version of *Lynnette* without any gamification features and with full system control over problem selection (as is common in e.g. Cognitive Tutors). The algebra game is *DragonBox*, which has attracted substantial public attention for allegedly helping young children learn algebra in a very short period of time [8, 12]. Although *DragonBox* has been the subject of at least one research study [1], we are not aware of any studies that empirically investigated its effectiveness in teaching algebra. Given the publicity surrounding the game, it would be good to know how educationally effective and engaging it is, compared to technology proven to be effective in helping students learn (i.e., an ITS). We conducted a classroom experiment with 267 middle school students to investigate our hypotheses.

2 Methods

2.1 Lynnette and DragonBox 12+

***Lynnette* – Web-Based Linear Equation Tutor on Android Tablet.** *Lynnette* is a tutor for basic equation solving practice. It comprises five levels with increasingly difficult equations, starting with equations of the form x + a = b and their variations at Level 1 and ending with equations of the form a(bx + c) + d = e and their variations at Level 5. Students are required to explain some of their steps by indicating the main transformation (see Fig. 1). The problems in *Lynnette* do not require fractions and the tutor does not allow strategies that involve fractions along the way. Otherwise, it is flexible in the major and minor strategy variants that it recognizes. It also allows some suboptimal strategies, while warning students about them in the hint window (see Fig. 1), on the assumption that students can learn from seeing and being explicitly reminded of suboptimal strategies. It does not allow mathematically correct but useless transformations. *Lynnette* was designed to run on Android tablets but also runs on regular desktop computers. It was implemented as a rule-based Cognitive Tutor using the Cognitive Tutor Authoring Tools (http://ctat.pact.cs.cmu.edu/). Its cognitive model comprises 73 rules. *Lynnette* is the first CTAT-built tutor that runs on Android tablets and the first elaborate CTAT-built rule-based tutor used in classrooms.

Fig. 1. The problem solving interface of *Lynnette* on a Samsung Galaxy Tablet

***DragonBox 12+*.** We used the Android version of *DragonBox 12+* in the study, which is one of the two *DragonBox* games that targets middle and high school algebra. It has 10 progressive chapters, each with 20 problems, covering 24 algebraic rules [13]. The two sides of the screen represent the two sides of an equation. The game provides immediate step-by-step feedback. It starts by hiding the algebraic expressions and the players have to isolate a box on one side of the screen through moving cards (Fig. 2, leftmost). It gradually transitions to algebraic problems as the students progress in the game (Fig. 2, middle and rightmost). As claimed on its official site, students can learn basic algebra in one hour with *DragonBox*.

Fig. 2. Screenshots of *DragonBox* from its official site (©WeWantToKnow)

2.2 Experimental Design, Participants, Procedure and Measurements

We conducted an experiment with a 2x2+1+1 design with a total of six conditions. The 2x2 design varies two factors: 1) whether or not the students are able to access and re-practice completed problems; and 2) whether or not the tutor shows rewards to the students. The two "+1" conditions are a popular algebra game, *DragonBox 12+*, and a standard ITS.

Table 1. Experimental conditions in the study. RePr stands for Re-Practice, NoRePr stands for no Re-Practice, Rwd stands for Rewards, and noRwd stands for no Rewards.

	RePr +Rwd	**No-RePr+Rwd**	**RePr +noRwd**	**No-RePr+noRwd**	***Dragon-Box 12+***	**Control *Lynnette***
Re-practice	Yes	No	Yes	No		
Rewards	Yes	Yes	No	No		

We created four experimental versions of *Lynnette* and a control version (as listed in Table 1). The five *Lynnette* tutors all used the same interface for problem solving, shown in Figure 1. Also, all five tutor versions employed Bayesian Knowledge Tracing and Cognitive Mastery as part of their problem selection methods. The control version used it for full system control, as is customary in Cognitive Tutors. That is, in this version the tutor always selected the next problem for the student from level 1 to level 5. The four experimental versions used Bayesian Knowledge Tracing and Cognitive Mastery for shared control. In these versions, the students also had to do the levels in order. Within a level, they could select which problem to do next. The tutor decided when a level was complete (namely, when all skills were mastered).

The system presented one or two screens in-between problems, which vary according to the two experimental factors. All four experimental tutor versions had a problem selection screen, which lists the problems within the current level. On this screen, the student selected the next problem (Fig. 3, right). In the two Re-Practice conditions, the system "recommended" problems on this screen by displaying a flag next to them. These problems had unmastered skills, according to the tutor's Bayesian Knowledge-Tracing method, and had not been practiced yet by the given student. However, students were free to select a problem with or without a flag. Also in the two Re-Practice conditions, students could select any problem available on the given level, regardless of whether they had completed them previously. By contrast, in the No Re-Practice conditions, the previously-practiced problems were grayed out so they could not be selected again. In the two Rewards conditions, students saw an additional screen between problems (Fig. 3, left), a problem summary screen showing earned stars after completing each problem, based on the number of steps, hints and errors. A trophy could be earned for perfect performance. Further, in these conditions, the problem selection screen listed the rewards earned (see Fig. 3, right). After re-practice, the number of rewards would be updated.

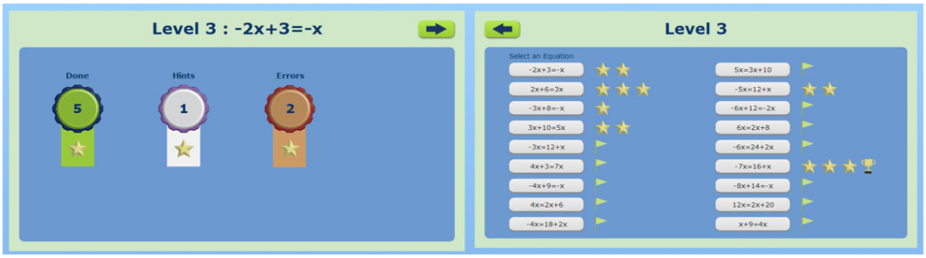

Fig. 3. Problem summary screen with rewards (left) and problem selection screen (right)

267 7th and 8th grade students participated in this study. They were from 15 classes of 3 local public middle schools, taught by 6 teachers. Students from each class were randomly assigned to one of the six conditions. All students completed a 20-minute paper pre-test on the first day of the study. They then worked for 5 42-minute class periods on consecutive school days either with one of the *Lynnette* versions or *DragonBox 12+* using Samsung Galaxy tablet PCs. All students took an immediate paper post-test after the five class periods. The pre- and post-tests were in the same format, which consisted of 6 equations that measured students' procedural skills of solving linear equations[1]. *Lynnette* only provides practice for a subset of problem types that are practiced in *DragonBox 12+*. Therefore, among the 6 equations, 4 were shared types of equations between *Lynnette* and *DragonBox 12+*, while 2 were types of equations practiced in *DragonBox 12+* only. Documentation of *DragonBox 12+* indicates that the algebraic rules that are needed to solve the 6 procedural items could

[1] The test forms also included items testing basic conceptual knowledge of algebra. However, because there was no improvement from pre-test to post-test on these items in any of the conditions (similar to what we saw in past studies), we do not report the results separately.

be practiced by Level 6 in the game [13]. We created two sets of equivalent test forms and administered them in counterbalanced order. We also included a 7-question questionnaire to measure students' enjoyment of using *Lynnette* or *DragonBox* along with the post-test. The questions were adapted from the interest/enjoyment subscale of the Intrinsic Motivation Inventory, and were all based on a 7-point Likert scale.

3 Results

A total of 190 students were present on each day of the study and completed the pre- and post-tests. Given that the sample was nested in 15 classes, 6 teachers, and 3 schools, Hierarchical Linear Modeling (HLM) was used to analyze the test data. We constructed 3-level models in which students (level 1) were nested in classes (level 2), and classes were nested in teachers (level 3; 4-level models indicated little variance on the school level, so we built 3-level models). Specifically, for the learning effects from pre- to post-tests, we used both pre- and post-test scores as dependent variables to fit this model: $score_{ij} = test_j + student(class)_i + class(teacher)_i + teacher_i$, where $score_{ij}$ was $student_{ij}$'s score on $test_j$, and $student(class)_i$, $class(teacher)_i$ and $teacher_i$ indicated the nested sources of variability in the hierarchical model. To evaluate the main effects and interaction effect across the conditions on the post-test, we modified the model and used $student_i$'s pre-test score pre_i as co-variate: $post\text{-}score_i = pre_i + tutor_j + rewards_k + re\text{-}practice_l + rewards_k{*}re\text{-}practice_l + student(class)_i + class(teacher)_i + teacher_i$, with $tutor_j$ being whether the condition learned with a tutor or *DragonBox 12+*, $rewards_k$ being whether the tutor condition received rewards, $re\text{-}practice_l$ being whether the condition allowed re-practice, and $rewards_k{*}re\text{-}practice_l$ being the interaction between the two factors. We report Cohen's *d* for effect sizes. An effect size *d* of .20 is typically deemed a small effect, .50 a medium effect, and .80 a large effect.

Table 2. Means and SDs of all conditions on pre- and post-tests for the shared procedural items, game (*DragonBox)* only procedural items, and the overall test scores

	RePr+Rwd	NoRePr+ Rwd	RePr+ noRwd	NoRePr+ noRwd	*Dragon-Box 12+*	Control *Lynnette*
Pre-shared	.364 (.249)	.327 (.279)	.327 (.257)	.364 (.313)	.321 (.209)	.386 (.277)
Post-shared	.467 (.291)	.491 (.276)	.497 (.364)	.471 (.311)	.366 (.289)	.538 (.347)
Pre-game	.324 (.345)	.266 (.359)	.318 (.350)	.318 (.344)	.331 (.382)	.288 (.330)
Post-game	.352 (.320)	.281 (.358)	.313 (.307)	.300 (.323)	.310 (.410)	.297 (.356)
Pre-overall[2]	.439 (.178)	.413 (.142)	.403 (.183)	.477 (.172)	.422 (.133)	.418 (.155)
Post-overall	.463 (.160)	.491 (.173)	.520 (.203)	.503 (.167)	.438 (.161)	.477 (.190)

[2] Pre-overall and Post-overall include the conceptual items along with the 6 procedural items.

Learning Effects of *Lynnette* and *DragonBox*. Table 2 shows the average test scores for all conditions on the 4 shared procedural items, the 2 *DragonBox/*game only procedural items, and the overall test scores including the conceptual items. Students in the *DragonBox* condition completed an average of 140 equations in the game by the end of the 5th period, which is equivalent to finishing Level 7. Students from all five *Lynnette* conditions completed an average of 36 equations. All five *Lynnette* conditions together improved significantly on the shared procedural items ($t(300)=4.543$, $p<.001$, $d=.52$) as well as the overall test scores ($t(300)=3.305$, $p=.001$, $d=.38$), but did not improve on the game only items. The best tutor condition, RePr+noRwd also improved significantly on the shared items ($t(41)=2.392$, $p=.021$, $d=.75$), and the overall test scores ($t(41)=3.088$, $p=.004$, $d=.96$). By contrast, the *DragonBox* students did not show significant improvement on any of the three categories of test items from pre- to post-test. When comparing the post-test scores between the *Lynnette* conditions and *DragonBox*, the five *Lynnette* conditions together significantly outperformed the *DragonBox* condition on both the shared items ($t(167)=2.118$, $p=.036$, $d=.33$) and all 6 procedural items together (i.e. shared items + game-only items, $t(167)=1.986$, $p=.049$, $d=.31$). The RePr+noRwd condition also significantly outperformed the *DragonBox* condition (shared items: $t(37)=2.214$, $p=.033$, $d=.73$; all 6 procedural items: $t(37)=2.295$, $p=.027$, $d=.75$). We also compared students' post-test scores between the control *Lynnette* and the experimental *Lynnette* tutors. There were no significant differences on any of the categories of test items.

Effects of Re-Practice and Rewards. We tested the main effects and interaction of the two factors with the four experimental *Lynnette* tutors. Neither re-practice nor rewards showed a significant main effect. The interaction between the two was significant for the overall test scores ($t(104)=-2.287$, $p=.024$). Post-hoc analysis revealed that for the two Re-Practice conditions, students who did not see rewards (i.e., RePr+noRwd) performed significantly better than students who received rewards (i.e., RePr+Rwd, $t(41)=-2.311$, $p=.026$, $d=.72$). On the other hand, there was no significant difference between the two No-Re-Practice conditions (i.e., NoRePr+Rwd and NoRePr+noRwd). To explore the mechanism behind the difference between the two Re-Practice conditions, we investigated how often the students re-practiced the completed problems. Seven out of 31 (22.58%) students in RePr+noRwd re-practiced a total of 9 problems start-to-finish, whereas 16 out of 33 (48.48%) students in RePr+Rwd re-practiced 37 problems start-to-finish. We also investigated the number of times students re-started a problem they had solved before, regardless of whether they actually finished it. Specifically, we calculated the ratio of (number of re-starts)/(number of total problem visits) for each student in the two Re-Practice conditions. The average ratio was .196 (SD=.172) for RePr+Rwd and .115 (SD=.074) for RePr+noRwd, with a significant difference between the two ($t(42)=2.858$, $p=.007$, $d=.88$). In other words, students in RePr+Rwd re-started significantly more problems than students in RePr+noRwd. Moreover, the correlation between the ratio of re-starts and students' post-test performance was -.277 ($p=.028$), controlling for the overall pre-test score. The more times the students re-started problems, the less they learned.

Enjoyment. Table 3 shows the average ratings of enjoyment from the intrinsic motivation questionnaire handed out with post-test. The *DragonBox* students provided significantly higher ratings of enjoyment while playing with the game, as compared to all the *Lynnette* conditions taken together (t(168)=-3.315, *p*=.001, *d*=.51). No significant main effects or interaction effect of re-practice and rewards were found for enjoyment among the experimental *Lynnette* tutors. The difference between the experimental *Lynnette* tutors and the control *Lynnette* was not significant either.

Table 3. Means and SDs of the enjoyment ratings across all 7 questions for all conditions

	RePr+ Rwd	NoRePr+ Rwd	RePr+ noRwd	NoRePr+ noRwd	*Dragon-Box 12+*	Control *Lynnette*
Enjoy-ment	3.815 (1.627)	3.884 (1.572)	4.166 (1.398)	4.372 (1.528)	5.099 (1.448)	4.138 (1.483)

4 Discussion and Conclusion

Gamifying ITSs to foster higher engagement and perhaps even better learning outcomes has become a popular theme in the ITS community. However, what gamification features are beneficial and how to integrate them with existing tutor features remains a challenging question. Our study found that gamification of shared student/system control was a partial success. The two gamification features held up well in the classroom but did not foster the expected higher enjoyment or learning gains. We did not find a significant difference between the experimental (gamified) *Lynnette* tutors and the control *Lynnette* with respect to enjoyment or learning. One of the gamified conditions (RePr+noRwd) had the highest learning gains, with a greater pre/post effect size (*d*=.96) than that for all *Lynnette* tutors (*d*=.38), but was not reliably better on any measure than the control tutor. Thus, gamifying tutors by incorporating common game design patterns does not automatically make them more effective. This finding is not uncommon. As discussed in the introduction, efforts at gamifying tutors frequently do not result in greater learning gains. Nonetheless, our findings may have practical value: students may have come to expect the problem selection features they know from games. Our study shows they can be added to a tutor (though with the caveat noted below) with relatively low implementation cost while maintaining the tutor's effectiveness.

An interesting finding was that the students who could re-practice completed problems and received rewards performed significantly worse than their counterparts who could re-practice problems but did not receive rewards. The same difference was not found between the two conditions that could not re-practice. To the best of our knowledge, this is a novel finding: we are not aware of studies showing a detrimental effect of re-practice in (tutored) problem solving. A possible explanation is that the urge to earn more stars pushed the students to re-practice, yet re-practicing previously-seen problems is not an optimal strategy for learning as compared to practicing new problems. (In standard ITSs, it is common practice that students practice new problems targeting the same skills, instead of re-practicing problems they have completed before.) Further data analysis supports this explanation: there were significantly more re-starts of problems in the RePr+Rwd condition and there was a significant negative correlation between the re-start ratio and students' post-test scores. This finding affirms that performance-based rewards can influence students' study choices but it also highlights the need to ensure that students are guided in making optimal choices. Although the combination of re-practicing with performance-based rewards is a very common design pattern in games, its implementation in tutors should be handled with care. For example, instead of giving rewards for individual problems, one could consider adding to the tutor data visualizations that help students analyze and summarize their performance, and provide rewards on an aggregated level. Also, instead of allowing students to re-practice problems they have seen before, the system might afford them freedom to select remedial *new* problems to earn more rewards.

Lastly, the experiment illustrated that an ITS can help students learn more effectively than a commercial educational game, even one with high popular acclaim. The students in the tutor conditions had greater learning gains than students who worked with *DragonBox,* in spite of the fact that the *DragonBox* students solved, on average, four times as many problems. In fact, our results indicate that *DragonBox* is ineffective in helping students acquire skills in solving algebra equations, as measured by a typical test of equation solving. This test is a fair test of *DragonBox'* effectiveness; on average, the students who worked with *DragonBox* reached Level 7 in the game, and thus covered the necessary algebraic rules to solve the equations on this test. Although *DragonBox* was more engaging than the tutor, where it falls short may be in using a concrete context to hide equations during much of the game, without a clear connection to standard algebraic notation and transformation rules. To be fair, *WeWantToKnow*, the company that markets *DragonBox* has recognized the need for supplemental instruction outside of the game and provides a document that teachers can use to help transfer. It is not known how effective this additional instruction is. It is not that there is no learning in *DragonBox* - there is plenty of it, as evidenced by students' progression through the game levels. However, the learning that happens in the game does not transfer out of the game, at least not to the standard equation solving format. Much of the publicity surrounding *DragonBox* seems to have focused on progression through the game levels as an indicator of learning, perhaps because this measure is so readily observable. This, in our opinion, is a profound mistake. What matters is not within-game learning, but out-of-game transfer of learning, and the two cannot be equated. We hope that our study will contribute to more careful consideration in the popular media of out-of-game transfer of learning as a key criterion when judging the educational value of games. Incidentally, our study should not be interpreted as questioning the educational potential of games in general, just that of one game in particular. We see educational games and gamification of ITSs as promising approaches to developing effective and enjoyable advanced learning technologies.

In sum, our study represents progress in our understanding of the value of gamification in ITSs. We demonstrated ways of gamifying shared problem control in an ITS with no detrimental effects, though we would have liked to see gains at minimum in enjoyment and preferably also in learning. Further, we discovered that the combination of performance-based rewards and the freedom of re-practicing, both common game design patterns, is detrimental for learning when imported into an ITS. The comparison between the tutors and *DragonBox* affirms that an intelligent tutor can be highly effective in helping students learn. It illustrates also that an educational game can foster high enjoyment and gain great popularity without helping students learn.

We continue to see great potential for incorporating gamification features in ITSs to enhance students' learning and engagement, although as our study illustrates importing popular game design patterns into ITSs needs to be done with care. There may be no substitute for careful evaluation studies.

Acknowledgements. We thank Jonathan Sewall, Octav Popescu, Martin van Velsen, Kristen Chon, Gail Kusbit and Kate Souza for their kind help with this work. We also thank the participating teachers and students. This work is funded by an NSF grant to the Pittsburgh Science of Learning Center (NSF Award SBE083612).

References

1. Andersen, E., Gulwani, S., Popovic, Z.: A Trace-based Framework for Analyzing and Synthesizing Educational Progressions. In: Proc. of the SIGCHI Conference on Human Factors in Comp. Systems, pp. 773–782. ACM, New York (2013)
2. Atkinson, R.C.: Optimizing the Learning of a Second-Language Vocabulary. Journal of Experimental Psychology 96, 124–129 (1972)
3. Boyce, A., Barnes, T.: BeadLoom Game: Using Game Elements to Increase Motivation and Learning. In: Proc. of Foundations of Digital Games, FDG 2010, pp. 25–31 (2010)
4. Corbalan, G., Kester, L., Van Merriënboer, J.J.G.: Selecting Learning Tasks: Effects of Adaptation and Shared Control on Efficiency and Task Involvement. Contemporary Educational Psychology 33(4), 733–756 (2008)
5. Easterday, M.W., Aleven, V., Scheines, R., Carver, S.M.: Using Tutors to Improve Educational Games. In: Biswas, G., Bull, S., Kay, J., Mitrovic, A. (eds.) AIED 2011. LNCS, vol. 6738, pp. 63–71. Springer, Heidelberg (2011)
6. Grolnick, W.S., Ryan, R.M.: Autonomy in Children's Learning: An Experimental and Individual Difference Investigation. Journal of Personality and Social Psychology 52, 890–898 (1987)
7. Jackson, G., McNamara, D.: Motivation and Performance in a Game- based Intelligent Tutoring System. Journal of Educational Psychology 105(4), 1036–1049 (2013)
8. Liu, J.: DragonBox: Algebra Beats Angry Birds. Wired (2012), http://www.wired.com/geekdad/2012/06/dragonbox/all/
9. Long, Y., Aleven, V.: Supporting Students' Self-Regulated Learning with an Open Learner Model in a Linear Equation Tutor. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 219–228. Springer, Heidelberg (2013)
10. Meluso, A., Zheng, M., Spires, H.A., Lester, J.: Enhancing 5th Graders' Science Content Knowledge and Self-Efficacy through Game-based Learning. Computers and Education 59(2), 497–504 (2012)
11. Paas, F.G.W.C., Van Merriënboer, J.J.G.: Variability of Worked Examples and Transfer of Geometrical Problem-Solving Skills: A Cognitive-Load Approach. Journal of Educational Psychology 86(1), 122–133 (1994)
12. Shapiro, J.: It Only Takes about 42 Minutes To Learn Algebra with Video Games. Forbes (2013), http://www.forbes.com/sites/jordanshapiro/2013/07/01/it-only-takes-about-42-minutes-to-learn-algebra-with-video-games/
13 Where Is the Math in DragonBox 12+?, http://wewanttoknow.com/resources/DragonBox/Math_In_DragonBox.pdf