Computers & Education 57 (2011) 1907–1918

Contents lists available at ScienceDirect

Computers & Education

journal homepage: www.elsevier.com/locate/compedu

# The effect of simulation games on the learning of computational problem solving

Chen-Chung Liu*, Yuan-Bang Cheng, Chia-Wen Huang

*Graduate Institute of Network Learning Technology, National Central University, No.300, Jhongda Rd., Jhongli City, Taoyuan County 32001, Taiwan, ROC*

A R T I C L E I N F O

*Article history:*
Received 31 December 2010
Received in revised form
28 March 2011
Accepted 11 April 2011

*Keywords:*
Game-based learning
Problem solving
Simulation
Flow experience

A B S T R A C T

Simulation games are now increasingly applied to many subject domains as they allow students to engage in discovery processes, and may facilitate a flow learning experience. However, the relationship between learning experiences and problem solving strategies in simulation games still remains unclear in the literature. This study, thus, analyzed the feedback and problem solving behaviors of 117 students in a simulation game, designed to assist them to learn computational problem solving. It was found that students when learning computational problem solving with the game were more likely to perceive a flow learning experience than in traditional lectures. The students' intrinsic motivation was also enhanced when they learned with the simulation game. In particular, the results of the study found a close association between the students' learning experience states and their problem solving strategies. The students who perceived a flow experience state frequently applied trial-and-error, learning-by-example, and analytical reasoning strategies to learn the computational problem solving skills. However, a certain portion of students who experienced states of boredom and anxiety did not demonstrate in-depth problem solving strategies. For instance, the students who felt anxious in the simulation game did not apply the learning-by-example strategy as frequently as those in the flow state. In addition, the students who felt bored in the simulation game only learned to solve the problem at a superficial level.


## 1. Introduction

Problem solving is one of the integral approaches to achieving effective and meaningful learning (Jonassen, 2004). Learners, while solving a problem, have to understand the problem, devise a plan and test the plan to solve it (Polya, 1957). In other words, they have to analyze the strategies which can possibly solve the problems by themselves, and thus, are more likely to generate creative solutions and achieve effective learning (Bransford & Stein, 1984). As a result, problem solving has been extensively applied to many subject domains such as science (Linn, Clark, & Slotta, 2003), mathematics (Jonassen, 2003) and design (Jermann & Dillenbourg, 2008) as a means of promoting learning in these domains. In particular, computational problem solving, which involves the development of computer programs to solve a problem, is considered to be the core competency of computer science education because computer science involves broad problem solving skills, rather than purely technically centered activity (Kay et al., 2000). However, novice programmers suffer from a wide range of difficulties and deficits. For example, novice programmers tend to focus only on surface knowledge, lack elaborative mental models, and fail to integrate relevant knowledge to solve a problem (Robins, Rountree, & Rountree, 2003). In particular, learning problem solving with computer programs is difficult for novices due to the fact that they have to understand problem solving strategies such as looping, conditionals, recursion and coordination which are difficult for them to apply. Therefore, one of the major issues facing computer science educators is how to foster students' abilities to solve problems with computer programs.

Simulation games on computers, presenting students with embodied problem situations, may be helpful in fostering students' problem solving ability. Such games simulate a model of a system or a process, and thus allow students to experience the scientific discovery process such as hypothesis generation, experiment designs and data interpretation (de Jong & van Joolingen, 1998). Due to these advantageous features, simulation games have been applied to many subject domains such as health care (Woodward, Carnine, & Gersten, 1988), scientific exploration (Tan & Biswas, 2007) and mathematics (Lee & Chen, 2009) as a tool to develop students' problem solving abilities. In the current

* Corresponding author. Tel.: +886 3 4227151x35412; fax: +886 3 4221931.
*E-mail address:* ccliu@cl.ncu.edu.tw (C.-C. Liu).

0360-1315/$ – see front matter 
doi:10.1016/j.compedu.2011.04.002

stage, as more and more games are developed for supporting learning, a better understanding of students' gaming and learning behaviors has become necessary. Consequently, this study attempted to explore the effect of simulation games on students' problem solving with computer programs.

Games may have potential in improving learners' knowledge, skills, attitudes and behaviors (Papastergiou, 2009b). Therefore, computer simulation games may have multiple effects on problem solving with computer programs. For instance, Kiili (2005) has argued that games can be a vehicle for engaging students in a "flow". Flow is a consciousness state during which an individual "is in control of his actions, and in which there is little distinction between self and environment, between stimulus and response, or between past, present and future" (Csikszentmihalyi, 1975, p. 36). Flow may be considered as a useful construct for improving problem solving. Many studies have confirmed that experiencing a state of flow may foster students' learning, as well as their exploratory and participatory behaviors (Hoffman & Novak, 1996). In particular, the higher level of flow perceived by learners correlates positively with higher engagement in experimentation (Trevino & Webster, 1992) and flexible learning (Webster, Trevino, & Ryan, 1993). As game-playing is regarded as a problem solving process (Choi & Kim, 2004) and has potential to promote the flow experience, games provide "a means to offer possibilities to students to set personal goals, to actively handle and gather information, and monitor and evaluate problem solving processes" (Kiili, 2005, p. 17).

The above literature suggests that games may be a pathway to developing problem solving abilities. However, recent investigations into game-based learning yield divergent results regarding the effect of the games on learning. On one hand, the interactivity and challenge of games have a prevailing impact on facilitating active and self-directed learning experiences (Kim, Park, & Baek, 2009; Papastergiou, 2009a; Shih et al., 2010). On the other hand, many studies have found that no differences in student learning can be found between learning environments that involve games and those without game elements (Annetta, Minogue, Holmes, & Cheng, 2009; Gredler, 2003; Papastergiou, 2009b; Reiber, 2005). In the study by Rodrigo et al. (2008), they further found that students, even learning with games with unique features of fantasy and competition, do not sense significantly more flow than those who learn with learning systems without such features. Therefore, it is necessary to investigate the influence of simulation games on the learning experiences associated with computational problem solving.

Moreover, educators and researchers have confirmed that the strategies which students apply to solving problems profoundly influence the performance of the problem solving activity (Stubbart & Ramaprasad, 1990). For instance, it is found that using the learning-by-example strategy in the initial problem solving stage can help students gain skills to solve problems (Renkl, Atkinson, Maier, & Staley, 2002). In addition, expert problem solvers more often apply the analytical reasoning strategy, whereas novice problem solvers tend to use the trial-and-error strategy (Hong & Liu, 2003). Several studies (e.g. Skadberg & Kimmel, 2004; Webster et al., 1993) have indicated that a flow experience may lead to more exploratory activities in web-based learning systems. However, the relationship between learning experiences and problem solving strategies in simulation games still remains unclear in the literature. It is, thus, necessary to explore how simulation games influence students' learning experience and problem solving strategies, so that we can achieve a better understanding regarding how to apply games to promote students' problem solving ability.

In this vein, the study addresses this issue. At a macro level, this study attempts to investigate the effect of a simulation game on a problem solving activity. More specifically, this study attempts to investigate students' learning experiences in terms of Csikszentmihalyi's (1990) experience states, namely the flow, anxiety and boredom states in a simulation game for assisting computational problem solving. At a micro level, this study addresses the relationship between the learning experience states and the problem solving strategies by focusing on examining whether the learning experiences play a role in the exploratory behaviors of the problem solving activity. More specifically, this study aims to answer the following research questions:

- How do the simulation games influence students' learning experiences and motivation toward the computational problem solving activity?
- What strategies do students apply to solve problems in simulation games?; and
- How are the learning experiences related to the application of these strategies?

## 2. Related works

### 2.1. Computer simulation for supporting problem solving

Computer simulation provides an opportunity for students to learn by doing because students can learn through the interactions with the problem involved in the simulation and the specific task information (Anzai & Simon, 1979). Computer simulations are thus being increasingly applied to foster problem solving abilities in several scientific subject domains because they can provide a simulated real world context for enhancing classroom science teaching, and may be helpful in improving conceptual understanding of science (Kumar & Sherwood, 2007). More specifically, the computer simulations allow the students to experience the scientific discovery process, and thus, can potentially enhance scientific discovery learning (de Jong & van Joolingen, 1998). In addition, the computer simulations support multiple ways of visualizing and interacting with the complex information involved in a problem. In other words, the computer simulations provide an external framework which facilitates the meaning making process among learners (Roschelle & Teasley, 1994).

Due to the aforementioned features, computer simulations are increasingly being applied as a tool to enhance problem-based learning in science education. For instance, a computer simulation application was designed to facilitate medical science students to analyze information, formulate working hypotheses and identify medical learning issues (Rendas, Rosado Pinto, & Gamboa, 1999). Additionally, simulated patients (Bergin & Fors, 2003; Cook & Triola, 2009) were proposed to develop medical science students' clinical reasoning ability. It is shown that such simulations, on the one hand, can be helpful in improving the students' understanding of complex concepts (Holzinger, Kickmeier-Rust, Wassertheurer, & Hessinger, 2009; Kumar & Sherwood, 2007). On the other hand, they can improve inquiry strategies and self-learning abilities (Rendas et al., 1999) because they can enable students to directly apply their learning to understand complex systems (Holzinger et al., 2009). Through simulations, students can gain knowledge and promote their self-confidence, critical thinking and problem solving skills (Jeffries, 2005; Rivers & Vockell, 1987; Woodward et al., 1988). Students are thus more likely to transfer knowledge acquired from computer simulations to other problems (Kumar & Sherwood, 2007).

However, the above effectiveness of computer simulations is not conclusive because students often interact with simulations simply on a superficial and playful level (Mayer, 2004; Swaak & de Jong, 2001). Such superficial interaction is partly due to the fact that most students cannot solve problems without instructional support (Leutner, 1993; Yamen, Nerdel, & Bayrhuber, 2008). In other words, students need further assistance to acquire the knowledge and reduce the complexity of the simulations. Such a claim is consistent with the finding of Holzinger et al. (2009). Their study found that although simulations can be helpful in improving the understanding of complex concepts, students may not know how to interact with sophisticated simulations in order to solve a problem. Thus, to effectively apply computer simulations for educational purposes, educators need to provide additional instructional supports. In addition, students need to acquire a certain amount of knowledge so that they can interact with the computer simulations to achieve their learning goal.

### *2.2. Problem solving in games*

Computer games can be an effective approach to providing instructional supports in computer simulations to help students solve problems, because both the computer games and simulations address the problem solving activity in authentic contexts. In addition, games promote students to apply logic, memory, visualizations and problem solving in such authentic contexts, and, thus, can enhance learning (Kim et al., 2009). Moreover, games have a significant impact on learning experiences. Many studies (e.g., Chen, Wigand, & Nilan, 1999; Hoffman & Novak, 1996) have indicated that the integral features of games, such as focused attention and appropriate challenge, can lead to the "flow" state proposed by Csikszentmihalyi (1990). As negative learning experiences such as boredom and frustration are more likely to remain for a long period of time (Baker, Rodrigo, & Xolocotzin, 2007), computer games can provide a pathway to transforming the experiences into positive states so that students are more likely to engage in meaningful strategies to solve problems.

Due to the aforementioned effects on learning, computer games have been increasingly applied as a vehicle to enhance problem solving in education. They are effective in enhancing the learning of problem solving in computer sciences (Papastergiou, 2009a), mathematics (Lee & Chen, 2009), science (Tan & Biswas, 2007), and reasoning (Shih, Shih, Shih, Su, & Chuang, 2010). For example, the study by Shih et al. (2010) found that games featuring clear goals, rules, challenges and a sense of achievement can enhance collaboration among students. Lee and Chen (2009) also confirmed the positive effect of games on problem solving. Their study associated with non-mathematical problems found that the clear feedback of games can enhance students' reasoning performance. In the study by Tan and Biswas (2007), they directly incorporated simulation with games to support science learning. The results of their study support the argument that students while gaming have a better opportunity to construct their own learning experiences and thereby acquire a deeper understanding of science and problem solving skills.

The findings of the above studies support the claim made by Kiili (2005) that games with immediate feedback, clear goals and challenges can constitute an approach to creating positive learning experiences. However, recent empirical studies related to games have yielded different interpretations of the usage of games in educational contexts. Gredler (2003), Reiber (2005) and Rodrigo et al. (2008), for example, indicated that game-based learning and learning without games made few differences in terms of learning experiences and cognitive gains. In a more recent literature review by Papastergiou (2009b), she found that there is still limited empirical evidence regarding the effectiveness of games in the domain of health education and physical education. In other words, although games may have some potential in improving learners' knowledge, skills and attitudes, how the game-based learning experience plays a role in students' learning behaviors and strategies is still unclear. Therefore, it is necessary to better understand how the learning experience states can influence students' learning behaviors and strategies, so that we can apply games to enhance the learning of problem solving.

### *2.3. Simulation games for supporting computational problem solving*

Programming is often considered as a pathway to enhancing the learning of problem solving skills as it is associated with the formulation of an explicit computational model of ideas to solve some problems (Wenger, 1986). Such a computational problem solving approach to learning largely relies on Papert's constructionism (1972), asserting that it is more important to help learners to construct and debug theories through constructing a project than it is to teach them knowledge. Based on the notion of constructionism, many technology-enabled learning applications have been proposed to realize active learning experiences. For instance, Logo, a programmable robot, was used as an experiment simulator that allows children to manipulate and construct concepts. Such applications engage students in the construction of a simulated world, encouraging them to solve problems by themselves.

In the same vein, many program development environments such as Scratch (Monroy-Hernández & Resnick, 2008), Alice (Dann, Cooper, & Pausch, 2006), and the Greenfoot system (Kulling & Henriksen, 2005), provide students with visualization and interaction techniques which enable them to construct simulation games by themselves. Such contextualized construction experience has become increasingly prominent in science education (Sung, 2009) because it presents the students with a real-life situation by stimulating them to discover the knowledge of problem solving. Such game-based development environments are now increasingly leveraged in computer science education courses because games help engage students in solving computational problems (Bayliss, 2007). More specifically, games are used as programming assignments which involve students in designing games using the concepts they have to learn (Barnes, Richter, Powell, Chaffin, & Godwin, 2007; Lewis & Massingill, 2006). Furthermore, they can serve as a tool to help students link the abstract concept to the concrete game experience in an attempt to solve a problem (Gestwicki, 2007).

The use of games may enhance the learning of computational problem solving. The enhancement is limited to only investigating the learning experience states. For instance, in many studies such as Gestwicki (2007), Bayliss (2007) and Barnes et al. (2007), students' responses to such problem solving experiences were positive, and the games were shown to be able to motivate students to solve problems. However, few formal evaluations of learning behaviors can display how the games may be associated with their problem solving strategies. In particular, students may apply profoundly different strategies to learn to solve problems. Such differences in problem solving strategies may lead to different problem solving patterns (Stubbart & Ramaprasad, 1990). Thus, there is an imperative need to have a better understanding of the impacts of games on problem solving strategies. To this end, we have conducted an empirical study to investigate the influence of simulation games on problem solving in terms of both learning experience states and problem solving behaviors. The goal of this empirical study is to obtain a clearer picture of the problem solving strategies adopted by students learning with simulation games.

## 3. Method

### 3.1. Participants

This study conducted an experiment to investigate the effect of simulation games on computational problem solving. The participants of this study were 117 first-year students in a university in northern Taiwan, who were enrolled in the "Introduction to Computer Sciences" course. They were novice programmers who did not have rich experience in programming. One of the goals of the course is to develop the students' computational problem solving ability. To achieve this goal, this study designed a simulation game for the students to learn and use their programming knowledge to solve some contextualized problems which are related to the transportation control of a railway system. The programming knowledge involved in the game-based learning activity includes basic object-oriented concepts, conditions, iteration, and object communication. Because the students were novices in programming, investigating their behaviors involved in the game could help us clarify the effect of simulation games on the learning experience states and the problem solving strategies associated with programming knowledge.

### 3.2. The simulation game

There are many types of simulation games, and they may differ considerably from each other in their effect on problem solving. This study thus focuses on construction games that allow students to construct an artifact through a game. This type of game follows Papert's constructionism view of learning (Harel & Papert, 1991) as they create a micro-world for students to explore computation and scientific thinking. Examples of such games include those developed by Dann et al. (2006), McNerney (2004), and Monroy-Hernández and Resnick (2008). To obtain details of the students' problem solving behaviors, this study thus developed a simulation game, namely TrainB&P (Train: Build and Program it). The game, as illustrated in Fig. 1, simulates the railway construction system. With this game, the students can construct rail systems and design the transportation behaviors of the trains on the rails. With the game, this study would be able to log each step that the students took to solve a problem. More specifically, the design of this game follows the principles of learning through construction and simulation of embodied experiences, as described below:

- Learning through construction: Constructionism, extending the view of constructivism and the ideas of manipulative materials, proposes that learning is most effective when learners construct a meaningful product. A micro-world that allows learners to be architects, and to think with and test ideas, is necessary to support learning-by-making (Kafai, 2006). To provide such a micro-world, TrainB&P provides a 3D environment, as displayed in Fig. 1, for students to design and build their own rail systems. The students can use several building blocks such as straight tracks, curved tracks, branch tracks and bridges to build their system. Such construction activity simulates a railway model building game which is supported by many commercial toys such as Tomica and Hornby railway models. The

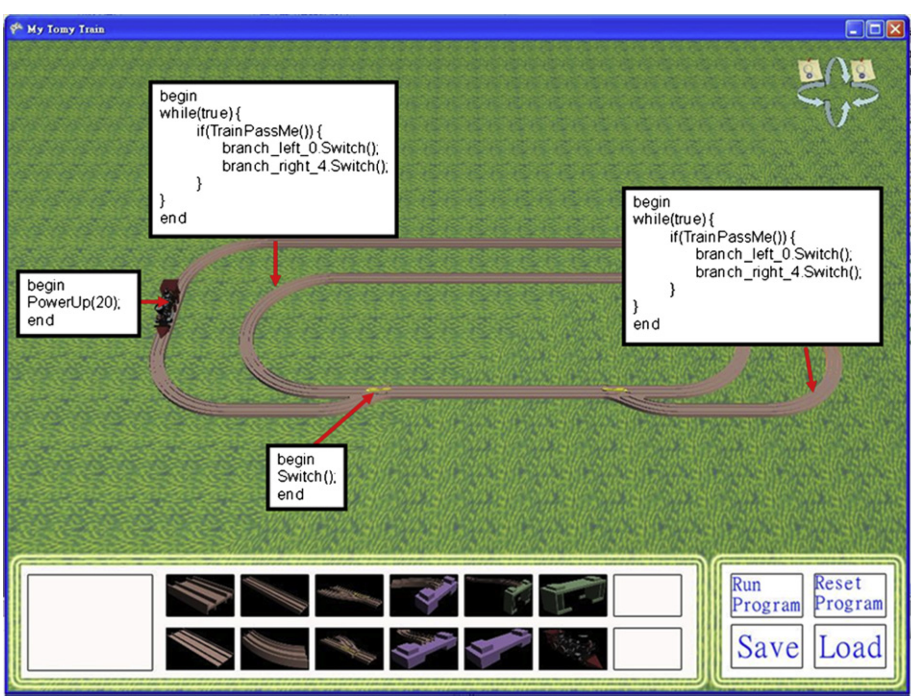


**Fig. 1.** The 3D railway model contruction environment in the TrainB&P.

students could freely use the building blocks to create a railway model in which they may encounter problems that they need to solve with programs in the 3D environment.

- Simulation of embodied experiences: Constructionism also addresses the importance of using manipulative materials for constructions and the simulation of ideas with the materials. For instance, MOOSE Crossing (Bruckman, 2006), Tangible Avatars (Liu, Liu, Wang, Chen, & Su, in press) and Logo (Turkle, 1995) all simulate the actions that learners perform on the mini-worlds to help them understand the effect of these actions. With such simulations, players can carefully consider how the actions will or will not solve a problem. To help the students of this study reflect upon their construction, TrainB&P allows students to program the transportation behaviors of the railway model and simulate the programs in the 3D environment. In particular, TrainB&P was developed with a physics engine. The physics engine could simulate the physics phenomena, such as gravity, speed, acceleration, and friction, to simulate the real behaviors of railway systems in the real world. Therefore, the students had to program the transportation behaviors of the railway model according to their experience in the real world. For instance, when encountering an ascent, the students had to increase the power of the train in the program so that it could climb up the ascent. On the other hand, they had to lower the speed of the train when it was rounding a curve to avoid derailing. To solve these problems, the students needed to learn programming knowledge and think scientifically to generate a railway model.

Since the students were novice programmers, a tutorial was provided to support them in learning the above computational problem solving knowledge. The tutorial consists of two main parts. One part of the manual describes the knowledge about the generic computational problem solving knowledge, while the other illustrates some examples of railway models. The example models exemplify the application of the knowledge related to the generic computational problem solving knowledge and that related to the building blocks. The students could refer to the tutorial at any time when they had problems in developing their railway models.

### 3.3. Procedures

The goal of this study is to understand the learning experience and exploratory behaviors in the problem solving activity in the context of learning with a simulation game. The experiment was conducted in the middle of a semester. Before the experiment the students had 1.5-months experience of learning programming knowledge in traditional lectures where the teacher lectured on the knowledge and the students had to complete certain assignments. To understand the students' perceptions of the learning experiences in the traditional lectures, they were asked to fill in a learning experience survey which probes students' flow states in the learning context (described later). After the traditional lectures, the students took part in the game-based learning activity. The activity lasted two weeks. The students were introduced to TrainB&P, and were then required to develop programs to make a train in a railway model (Fig. 2) go three rounds and then stop where it set off. To achieve this goal, the students needed to learn basic object-oriented concepts, conditions, iteration, and object communications which are general constructs of computational problem solving. In addition, they had to apply these constructs to find an appropriate solution to make the train go over the bridge and around the curved tracks as quickly and as stably as possible.

Because the simulation game involved extensive computation, this experiment was conducted in a computer classroom where each student was provided with a computer. The programs the students developed and the process of the students' problem solving activity were logged in activity logs for further analysis. After the experiment, the students also filled out the learning experience survey so that we could obtain their learning perceptions of the simulation game (described later). It was hoped that a complete view of how the simulation game affected the learning of computational problem solving skills could be obtained from both the activity logs and the learning experience survey.

### 3.4. The assessment of learning experiences

One of the goals of this study is to explore the role of simulation games on the learning experience associated with computational problem solving. To achieve this goal, this study examined how the students perceived the learning experience with the simulation games compared with that of the traditional lectures. Several well-known assessment schemes can be utilized to infer learning experience. Among them, perceptions of skills and challenges have been used as the theoretically meaningful reference for the presence or absence of flow (Massimini & Carli, 1988; Novak & Hoffman, 1997). Learners will be more likely to experience flow when the challenge of an activity matches

**Fig. 2.** The railway model involved in the computational problem solving activity of this study.

their skill (Massimini, Csikszentmihalyi, & Delle Fave, 1988). In other words, a perception of challenge that matches a learner's skill is an important indicator of the presence of flow (Chen et al., 1999). Therefore, this study used the learning experience survey proposed by Pearce, Ainley, and Howard (2005) to obtain the students' flow states. The survey consists of two question items with a five-point Likert scale that asks the level of "perceived skill" and "perceived challenge" of the problem solving activity. Because the two variables constitute the basic representation of flow, they have been applied as the most important elements to measure flow state in many studies (Ainley, Enger, & Kennedy, 2008; Konradt, Filip, & Hoffmann, 2003; Pearce et al., 2005). Therefore, this study follows this approach to assess the presence of flow.

Csikszentmihalyi's flow is based on the balance between perceived skills and challenges during an activity. As a result, the 3-channel flow model (Csikszentmihalyi, 1975), in which experience states are categorized into three states, namely, flow state, anxiety state and boredom state, is widely used to obtain the experience states (Pearce et al., 2005). Researchers such as Massimini and Carli (1988) and Novak, Hoffman, and Yung (2000) have refined this 3-channel model and proposed models that consist of more than 3 channels to include more flow states such as apathy and relaxation. However, this study applied the 3-channel model because of the limited number of students. More specifically, this study categorized the learning experiences only into the three states to avoid over-fitting (Mitchell, 1997). In other words, a balance of perceived challenge and skill would be considered as flow. Higher perceived challenge with lower perceived skill would be classified as anxiety. On the contrary, lower perceived challenge with higher perceived skill would be regarded as boredom.

### 3.5. Survey for learning motivations

In order to obtain a better understanding of the research question of this study, we also explored the students' motivation during the game-based learning activity. The Motivated Strategies for Learning Questionnaire (MSLQ) (Pintrich et al., 1991) was used to evaluate the students' motivation because it is widely used in contexts associated with learning. The MSLQ contains eight questions with a five-point Likert scale concerning the extrinsic and intrinsic motivations associated with learning. Example question items include "I want to get better grades in this class than most other students" and "In a class like this, I prefer course material that really challenges me so I can learn new things". The former queries the students' extrinsic motivation, while the latter concerns their intrinsic motivation. The Cronbach reliability (alpha) regarding the extrinsic and intrinsic motivation in the subjects of this study ranges from .67 to .83, indicating that the questionnaire was adequately reliable.

The MSLQ and the learning experience survey were used to obtain the students' motivations and learning experience states associated with both the traditional lectures and the simulation game. In other words, the students responded to the two surveys before and after the game-based learning activity. With the students' feedback to the surveys, we can understand the difference in their motivation and learning experiences between the learning context with the simulation game and that of the traditional lectures.

### 3.6. Activity logs

In order to explore problem solving behaviors in the simulation game, it is necessary to record and extract the potentially behavioral attributes for each student. The students may perform many types of learning and problem solving behaviors including solution development, experimenting, solution review, solution reuse, and reading the tutorial. TrainB&P records each of these behaviors in activity logs, as detailed below:

- Solution development: the students typed the codes or modified the codes in the program panel.
- Experiment: the students applied the simulation function of the game to verify the behavior of the programs they developed. The simulation game would execute the programs and visualize the effect in the 3D environment.
- Solution review: the students opened the program panel to review the program they developed without typing or modifying any of the program code.
- Solution reuse: the students copied code segments in the tutorial or in the programs, which they had already developed, to generate new solutions.
- Reading tutorial: The students retrieved existing examples, knowledge related to generic computational problem solving, or information about the building blocks in the tutorial.

The raw activity logs were recorded in the form of web access logs. The activity logs were, thus, pre-processed to obtain the above behavioral attributes that show how each student performed the five types of problem solving behaviors.

### 3.7. Data analysis

The activity logs and the results of the learning experience states and MSLQ were used to investigate the effect of the simulation games on the students' computational problem solving. More specifically, the learning experience states and motivations associated with the traditional lectures and in the simulation game approach were compared. Regarding the learning experience states, the distribution of students in the three states associated with the traditional lectures and the simulation game setting were compared with chi-square analysis. The motivations of the students obtained by the MSLQ were also analyzed so that we could gain a complete understanding of the effect of the simulation game on the students' motivations.

The students' problem solving behaviors were analyzed based on the activity logs. How frequently they applied the five types of problem solving in a specific learning experience state was extracted from the logs to investigate the influence of the learning states on their learning behaviors. Moreover, this study also conducted a sequential pattern analysis to reveal the students' problem solving strategies in the simulation game because sequential pattern analysis is widely applied in diverse application areas, such as communication pattern analysis (Yamauchi, Yokozawa, Shinohara, & Ishida, 2000) and web traverse analysis (Liu, Fan Chiang, Chou, & Chen, 2010) to understand users' behavior patterns. More specifically, this study applied a transition diagram to conduct the sequential pattern analysis to reveal how the

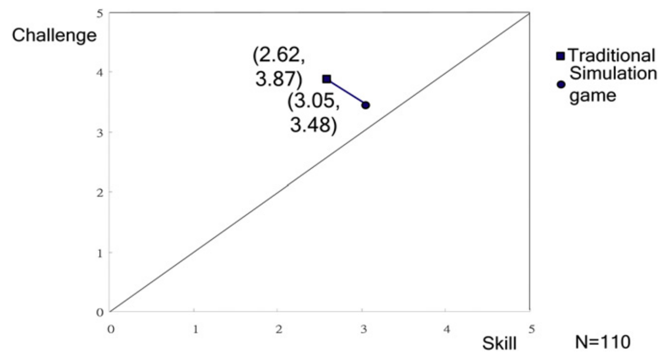


**Fig. 3.** The flow states in the traditional and simulation game settings.

students developed solutions through the five types of problem solving behaviors. The series of behaviors were coded into sequences according to the time order in which the behaviors were performed. The transition probability from behavior $a$ to behavior $b$ ($a \rightarrow b$) was calculated based on the percentage of transition $a \rightarrow b$ among all transitions to $b$. The probability of the transition $a \rightarrow b$ reflects the tendency of performing behavior $b$ after behavior $a$. The transition probability was represented as the value attached to an arc between two behaviors in the transition diagram. Such analysis of problem solving patterns was performed separately for the students who demonstrated the flow state, and for those who exhibited the anxiety and boredom states. Therefore, a comparative result can be shown from the sequential pattern analysis that possibly reveals the difference in problem solving strategies between the students with the three learning experience states.

## 4. Results

### 4.1. Results of the learning experience survey

Fig. 3 shows the students' learning experiences in the skill-challenge scatter plot. It should be noted that 115 students completed the learning experience surveys for the traditional setting and 112 students for the simulation game setting. In other words, 7 students did not complete the learning experience surveys for either the traditional setting or the simulation game setting. Therefore, data for only 110 students are included in the results. The results shown in Fig. 3 reveal that the learning experiences in the simulation game setting approached the flow state, i.e. the diagonal. More specifically, in comparison with traditional lectures, the level of challenge which the students perceived in the simulation game setting was significantly reduced ($t = 3.9$, $p < .01$), but the perceived level of skill was profoundly increased ($t = -4.53$, $p < .01$) in the simulation game setting. Such results reveal that the problem solving tasks given in the traditional lectures did not match the novice programmers' skill because the students in the traditional lecture setting perceived a high level of challenge (mean = 3.87, S.D. = .79) but a low level of skill (mean = 2.62, S.D. = .88). According to the 3-channel flow model, such a mismatch between the level of challenge and skill indicates that the students were in an anxiety state in which the level of challenge profoundly exceeds that of the level of skill. On the other hand, the students' feedback to the learning experiences in the simulation game setting reveal that the level of skill (mean = 3.05, S.D. = .71) is closer to the level of challenge (mean = 3.48, S.D. = .69), although the learning tasks were still challenging. Such results indicate that the students may be more likely to have experienced a flow state when they learned the computational problem solving in the simulation game setting.

The results above suggest that the simulation game may be helpful in transforming the learning of computational problem solving from an anxiety state to a flow state. A further analysis was made to clarify individual students' learning experience in the two learning settings. Table 1 describes the student distribution among the three learning experience states based on the definitions proposed in Section 3.5. It should be noted that all the students' feedback to the learning experience surveys was used in this analysis. Therefore, the results in Table 1 were obtained from the feedback of 115 students in the traditional setting and of 112 students in the simulation game setting. It is shown that the student distribution in the three learning experience states changed significantly (Chi-square = 26.58, df = 2, $p < .01$) in the two settings. For instance, a large number of students ($n = 78$) reported an anxiety state in the traditional setting. However, the number of anxious students reduced and a significant portion of students ($n = 64$) indicated a flow state in the simulation game setting. Such a result also supports the argument that the two main features of the simulation game of this study, namely learning through construction and the simulation of embodied experiences, may be helpful in promoting the positive experience of computational problem solving.

### 4.2. Motivations

The MSLQ was used to understand the students' motivation associated with the learning activities in the two settings. As shown in Fig. 4, the students' intrinsic motivations associated with the learning in the simulation game was 3.95 (S.D. = .6), which was higher than those in

**Table 1**
The distribution of students on flow states.

| | Flow | Anxiety | Boredom | Total |
|---|---|---|---|---|
| Traditional | 30 | 78 | 7 | 115 |
| Simulation | 64 | 38 | 10 | 112 |

Chi-square = 26.58, df = 2, $p < .01$.
Notes: Flow, challenge = skill; Anxiety, challenge > skill; Boredom, skill > challenge.

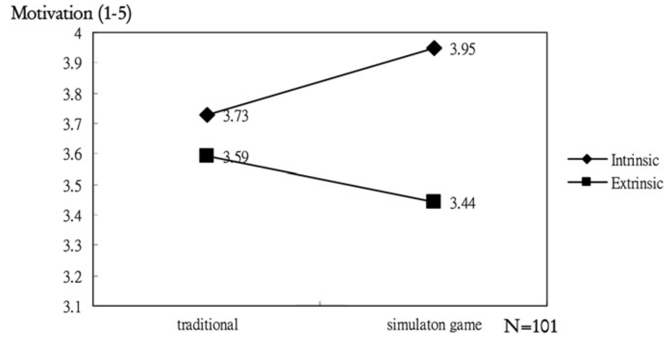


**Fig. 4.** The students' motivation in the traditional and simulation game setting.

the traditional setting (mean = 3.73, S.D. = .52). The difference is statistically significant ($t = -2.94$, $p < .01$). On the contrary, the students' extrinsic motivations (mean = 3.44, S.D. = .59) in the simulation game setting were significantly lower than those in the traditional setting (mean = 3.59, S.D. = .74). The difference regarding extrinsic motivations in the two settings is also significant ($t = -2.27$, $p < .05$). Such results are consistent with the findings in the previous sub-section, supporting that the students in the simulation game setting were more likely to learn in a flow state because they took active actions to solve the problems based on intrinsic motivation.

### *4.3. Problem solving behaviors and patterns*

A detailed analysis of the activity logs in the simulation game was undertaken to investigate how the students behaved to solve the problem. The activity logs were also analyzed in relation to each student's learning experience state, so that we could understand how a student in a specific state solved the problem. Because only 10 students perceived a boredom state in the simulation game setting, this study only analyzed the problem solving behaviors of the students who perceived a flow state and those who indicated an anxiety state in the simulation game. Table 2 displays the frequencies of the problem solving behaviors for the two groups of students. It is shown that students in flow and those with anxiety demonstrated quite similar frequencies of solution review, reading tutorial, solution development and experiments. However, the two groups solved the problems differently in the reuse of available solutions. The students in a flow state demonstrated higher frequency of solution reuse behaviors than those who felt anxious during the learning in the simulation game. The difference almost approaches significance ($t = 1.97$, $p = .052$). In others words, the students who reported a flow state were more likely to be aware of the available solutions and to apply the solutions to new problems.

The students' problem solving behaviors were further analyzed to reveal the problem solving patterns associated with the students who demonstrated a specific learning experience state. The results are shown in Figs. 5–7, displaying the transition possibilities among the five types of problem solving behaviors. Among these transitions, the transition with small probabilities (transition probability $< 1/5$, the random probability between any two of the five problem solving behaviors under uniform distribution) were eliminated so that only frequent transitions were displayed to illustrate the students' problem solving patterns.

Figs. 5–7 reveal that the learning experience states may influence the problem solving patterns in the simulation game. As shown in Fig. 5, the students who reported a flow state started to solve the problem from reading the tutorial where they found reusable examples, and experimented on these examples. Such a problem solving pattern can be found in the transition path of reading tutorial → solution reuse-→ experiment. In other words, they tended to adopt a learning-by-example strategy. Moreover, they often applied a trial-and-error strategy by which they developed a solution and experimented with the solution. Based on the experiment result, they reviewed and refined the solution. Such applications of trial-and-error strategy were revealed in the transition path of solution development → experiment → solution review → solution development. However, the trial-and-error strategy was not the only strategy the students in the flow state adopted to solve a problem. They also demonstrated the frequent transition path of solution development → solution review. In other words, they often analyzed a solution before experimenting with it, demonstrating analytical reasoning for solving problems. Such results indicate that the students in the flow state demonstrated a problem solving strategy that mixed learning-by-example, trial-and-error, and analytical reasoning strategies.

**Table 2**
Comparison of problem solving behaviors of students in flow and anxiety state.

| | | N | Mean | Std | t-value | *p*-value |
|---|---|---|---|---|---|---|
| Solution development | Flow | 64 | 119.92 | 54.38 | −.53 | .6 |
| | Anxiety | 38 | 126.53 | 69.71 | | |
| Experiment | Flow | 64 | 165.06 | 74.7 | −.95 | .34 |
| | Anxiety | 38 | 180.31 | 84.13 | | |
| Solution review | Flow | 64 | 84.42 | 54.97 | .60 | .55 |
| | Anxiety | 38 | 77.03 | 67.8 | | |
| Solution reuse | Flow | 64 | 64.28 | 48.35 | 1.97 | .052 |
| | Anxiety | 38 | 47.13 | 38.58 | | |
| Reading tutorial | Flow | 64 | 110.38 | 69.6 | −.68 | .50 |
| | Anxiety | 38 | 120.76 | 81.7 | | |

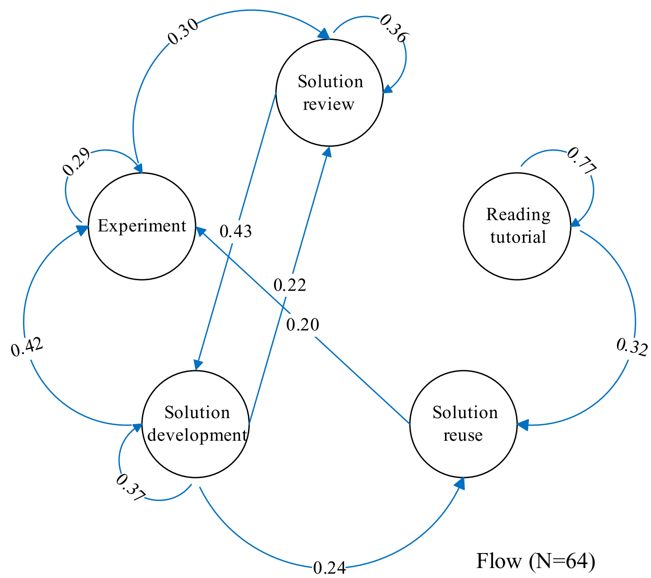


**Fig. 5.** The problem solving pattern associated with flow experience.

The students who reported boredom or anxiety experiences demonstrated different problem solving patterns. As shown in Fig. 6, the students who felt bored in the simulation game also frequently demonstrated the transition path of solution development → experiment → solution review → solution development and reading tutorial → solution reuse → experiment. Such results reveal that the students also adopted the strategies of trial-and-error and learning-by-example to solve a problem. However, this type of student did not often take an in-depth analytical reasoning strategy in the problem solving activity because there is not a frequent transition path of solution development → solution review in their problem solving pattern. Such results reveal that the students might have only learned to solve the problem at a superficial level, and did not conduct in-depth analyses of their solutions.

The students who felt anxious about the problem solving activity in the simulation game also exhibited a different problem solving pattern from that demonstrated by the students in flow. As illustrated in Fig. 7, this type of student did not often apply the learning-by-example strategy (i.e., the transition path of reading tutorial → solution reuse → experiment), but frequently demonstrated the trial-and-error strategy (i.e., the transition path of solution development → experiment → solution review → solution development) and the analytical reasoning strategy (i.e., the transition path of solution development → solution review). More specifically, there is not a frequent transition from solution reuse to experiment. Such a result implies that learning-by-example may be one of the prominent strategies to alleviate anxiety and facilitate a flow experience.

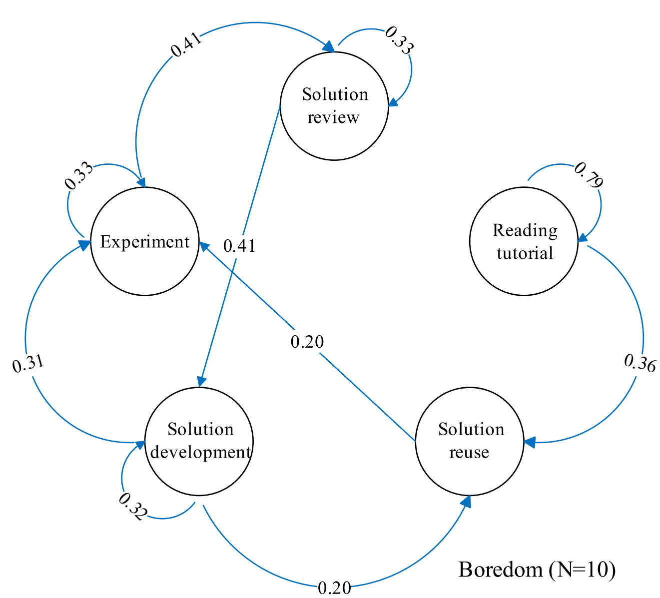


**Fig. 6.** The problem solving pattern associated with boredom experience.

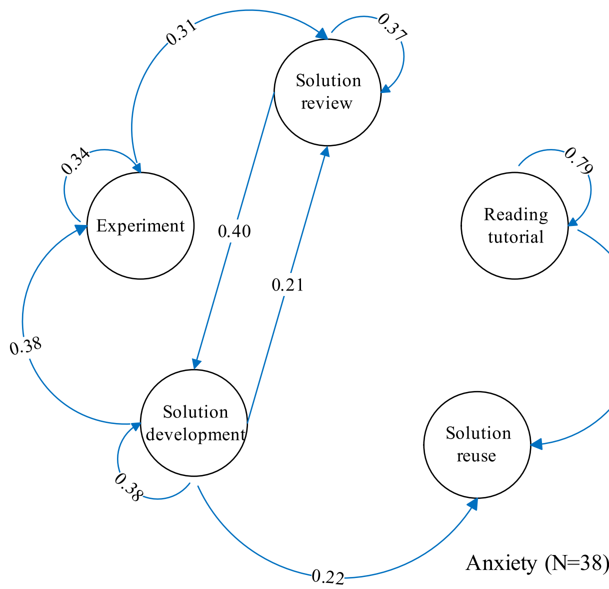


**Fig. 7.** The problem solving pattern associated with anxiety experience.

## 5. Conclusion and implications

Educators have highlighted the importance of problem solving competence. Consequently, many approaches have been proposed to enhance such competence. This study proposes an approach which utilizes a simulation game to assist students in developing their computational problem solving abilities. Such simulation games address the features of problem solving through construction and the simulation of embodied experience. From the analysis of the students' feedback and activity logs, this simulation game is helpful in creating a flow experience in which the students are motivated to apply trial-and-error, learning-by-example, and analytical reasoning strategies to learn the computational problem solving skills.

The simulation game of this study helps the students to learn by construction in a simulated embodied experience. It is found that students, when learning computational problem solving with the game, are more likely to perceive a flow learning experience than they are in traditional lectures. Such findings suggest that simulation games based on Paperts' constructionism may improve problem solving. This might be because constructing a system is helpful in retaining students' focused attention on the problem solving process, and provides appropriate challenge, and thus, can cause a flow state (Chen et al., 1999; Skadberg & Kimmel, 2004). Educators may find it helpful to use other constructionist approaches to enhance the learning of computational problem solving skills. From the results of the learning experience survey, it is found that the students' intrinsic motivation was enhanced when they learned with such constructivist approaches.

Although recent empirical studies do not reveal consistent results regarding the effectiveness of games (Papastergiou, 2009b; Reiber, 2005; Rodrigo et al., 2008), the results of this study found a close association between the students' learning experience states and their problem solving strategies. In other words, students may apply different problem solving strategies in a simulation game according to their learning experience states. Most of the students in the study perceived a flow state when they solved the problem of the simulation game. It is found that this type of student was more likely to adopt a learning-by-example strategy, as the simulation game allowed the students to directly apply their learning to understand the complex system (Holzinger et al., 2009). Consequently, the students extensively applied and tested the existing solution to obtain a solution for a new problem. Moreover, through the simulation of their solutions, the students critically reviewed their solutions and thus could adopt an analytical reasoning strategy during the problem solving process. Such a finding is consistent with the claim of previous studies (e.g., Jeffries, 2005; Rivers & Vockell, 1987; Woodward et al., 1988), indicating that students' critical or analytical thinking and problem solving skills can be fostered through simulations. Therefore, they can achieve deep learning of computation problem solving. However, such an effect may only appear in the problem solving process of the students who perceived a flow state.

Educators have emphasized the effect of simulation games on improving inquiry strategies and self-learning abilities because they can demonstrate the complexity of the real world (Rendas et al., 1999). However, a certain portion of students who were in boredom and anxiety states did not demonstrate in-depth problem solving strategies. In particular, the students who felt anxious in the simulation game did not apply the learning-by-example strategy as frequently as did those in flow. In other words, they may have encountered heavy cognitive load in the simulation game. Such findings echo the findings of previous studies, indicating that students may not be able to solve the problems to reduce the complexity of the simulations (Holzinger et al., 2009; Leutner, 1993; Yamen et al., 2008). Consequently, for the students who feel anxious about simulation games, it is necessary to provide instructional support to alleviate their anxiety. The result of this study suggests that learning-by-example may be an effective potential problem solving strategy that may transform anxious learning experiences into flow experiences because such a strategy was extensively applied by the students in flow. Therefore, it is suggested that educators may help the students to apply learning-by-example to reduce their anxiety. Regarding the students who felt bored in the simulation game, they only learned to solve the problem at a superficial level. Educators may need to apply strategies to engage the students in in-depth reasoning of their solutions. For instance, the teacher may increase the complexity of the problem according to the ability of each student so that the student may need to analyze the solution critically in order to solve the problem.

The results shown in this study demonstrate that simulation games constitute an effective approach to assisting novice programmers to learn computation problem solving skills. However, this study was still a small-scale investigation. Further work is needed to be undertaken with a larger sample to provide additional evidence. In particular, the subjects of this study were first-year students in a university. It would be interesting to see how the simulation game could influence the learning of problem solving of students at different ages. Moreover, this study applied the simulation game in the subject domain of computer programming. Researchers and educators may find it useful to apply this approach in some other domains. For instance, it may be interesting to know how the simulation game can promote problem solving in science discovery because the simulation game involved a physics engine. Students may apply the engine in the game to conduct a physics experiment. In addition, this study only investigated students' learning experiences and problem solving strategies in an individual learning setting. It would be worthwhile to investigate the learning experiences and problem solving strategies associated with a group of students who work together on the problem. For instance, Liu and Tsai (2008) have analyzed the peer interaction patterns among student groups who solved programming problems together. Such analyses of peer interaction patterns can extend the current understanding of simulation games from an individual learning perspective to a collaborative perspective. Gathering information on these issues through further studies can help to obtain a thorough understanding of this instructional innovation and thereafter to design a game approach to enhancing students' problem solving abilities in a broader context.

## Acknowledgments

This research was partially funded by the Research Center for Science and Technology for Learning at National Central University and the National Science Foundation under 98-2511-S-008-004-MY3, 99-2631-S-008-001-, 99-2631-S-008-004- and 99-2631-S-008-003. The authors would like to thank Chin-Chung Tsai for valuable comments.

## References

Ainley, M., Enger, L., & Kennedy, G. (2008). The elusive experience of 'flow': qualitative and quantitative indicators. *International Journal of Educational Research, 47*(2), 109–121.

Annetta, L. A., Minogue, J., Holmes, S. Y., & Cheng, M. (2009). Investigating the impact of video games on high school students' engagement and learning about genetics. *Computers & Education, 53*, 74–85.

Anzai, Y., & Simon, H. A. (1979). The theory of learning by doing. *Psychological Review, 86*, 124–140.

Baker, R. S. J. D., Rodrigo, M. M. T., & Xolocotzin, U. (2007). The dynamics of affective transitions in simulation problem-solving environments. *Lecture Notes in Computer Science, 4738*, 666–677.

Barnes, T., Richter, H., Powell, E., Chaffin, A., & Godwin, A. (2007). Game2Learn: building CS1 learning games for retention. In *Proceedings of ACM conference on innovation and technology in computer science education* (pp. 121–125).

Bayliss, J. D. (2007). The effects of games in CS1-3. In *Proceedings Of Microsoft Academic Days Conference on Game Development in Computer Science Education* (pp. 59–63).

Bergin, A. R., & Fors, G. H. U. (2003). Interactive simulation of patients – an advanced tool for student-activated learning in medicine & healthcare. *Computers & Education, 40*(4), 361–376.

Bransford, J. D., & Stein, B. S. (1984). *The ideal problem solver: A guide for improving thinking, learning, and creativity*. New York: Freeman.

Bruckman, A. (2006). Learning in online communities. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 461–472). New York: Cambridge University Press.

Chen, H., Wigand, R., & Nilan, M. S. (1999). Optimal experience of web activities. *Computers in Human Behavior, 15*, 585–608.

Choi, D., & Kim, J. (2004). Why people continue to play online games: in search of critical design factors to increase customer loyalty to online contents. *Cyberpsychology and Behavior, 7*(1), 11–24.

Cook, D. A., & Triola, M. M. (2009). Virtual patients: a critical literature review and proposed next steps. *Medical Education, 43*(4), 303–311.

Csikszentmihalyi, M. (1975). *Beyond boredom and anxiety*. San Francisco: Jossey-Bass Publishers.

Csikszentmihalyi, M. (1990). *Flow: The psychology of optimal experience*. New York: Harper and Row.

Dann, W., Cooper, S., & Pausch, R. (2006). *Learning to program with Alice*. Upper Saddle River, NJ: Prentice Hall.

de Jong, T., & van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research, 68*(2), 179–201.

Gestwicki, P. V. (2007). Computer games as motivation for design patterns. In *Proceedings of the 38th SIGCSE technical symposium on Computer Science Education* (pp. 233–237).

Gredler, M. E. (2003). Games and simulations and their relationships to learning. In D. Jonassen (Ed.), *Handbook of research for educational communications and technology* (2nd ed.). Mahwah, NJ: Lawrence Erlbaum Associates.

Harel, I., & Papert, S. (1991). *Constructionism*. Norwood, NY: Ablex Publishing Corporation.

Hoffman, D., & Novak, T. P. (1996). Marketing in hypermedia computer-mediated environments: conceptual foundations. *Journal of Marketing, 60*, 50–68.

Holzinger, A., Kickmeier-Rust, M. D., Wassertheurer, S., & Hessinger, M. (2009). Learning performance with interactive simulations in medical education: lessons learned from results of learning complex physiological models with the HAEMOdynamics SIMulator. *Computers & Education, 52*(2), 292–301.

Hong, J. C., & Liu, M. C. (2003). A study on thinking strategy between experts and novices of computer games. *Computers in Human Behavior, 19*(2), 245–258.

Jeffries, P. (2005). A framework for designing, implementing, and evaluating simulations used as teaching strategies in nursing. *Nursing Education Perspectives, 26*, 96–103.

Jermann, P., & Dillenbourg, P. (2008). Group mirrors to support interaction regulation in collaborative problem solving. *Computers & Education, 51*(1), 279–296.

Jonassen, H. D. (2003). Designing research-based instruction for story problems. *Educational Psychology Review, 15*(3), 267–296.

Jonassen, D. H. (2004). *Learning to solve problems: An instructional design guide*. San Francisco, CA: John Wiley & Sons.

Kafai, Y. B. (2006). Constructionism. In R. K. Sawyer (Ed.), *The Cambridge handbook of the learning sciences* (pp. 35–46). New York: Cambridge University Press.

Kay, J., Barg, M., Fekete, A., Greening, T., Hollands, O., Kingston, J., et al. (2000). Problem-based learning for foundation computer science courses. *Computer Science Education, 10*, 109–128.

Kiili, K. (2005). Digital game-based learning: towards an experiential gaming model. *Internet and Higher Education, 8*(1), 12–24.

Kim, B., Park, H., & Baek, Y. (2009). Not just fun, but serious strategies: using meta-cognitive strategies in game-based learning. *Computers & Education, 52*(4), 800–810.

Konradt, U., Filip, R., & Hoffmann, S. (2003). Flow experience and positive affect during hypermedia learning. *British Journal of Educational Technology, 34*(3), 309–327.

Kulling, M., & Henriksen, P. (2005). Game programming in introductory courses with direct state manipulation. In *Proceedings Of the 10th Annual SIGCSE Conference on Innovation and Technology in Computer Science Education* (pp. 59–63).

Kumar, D. D., & Sherwood, R. D. (2007). Effect of a problem based simulation on the conceptual understanding of undergraduate science education students. *Journal of Science Education and Technology, 16*(3), 239–246.

Lee, C. Y., & Chen, M. P. (2009). A computer game as a context for non-routine mathematical problem solving: the effects of type of question prompt and level of prior knowledge. *Computers & Education, 52*, 530–542.

Leutner, D. (1993). Guided discovery learning with computer based simulation games: effects of adaptive and non adaptive instructional support. *Learning and Instruction, 3*(2), 113–132.

Lewis, M. C., & Massingill, B. (2006). Graphical game development in CS2: a Flexible infrastructure for a semester long project. In *Proceedings Of the 37th SIGCSE Technical Symposium on Computer Science Education* (pp. 505–509).

Linn, M. C., Clark, D., & Slotta, J. D. (2003). Wise design for knowledge integration. *Science Education, 87*(4), 517–538.

Liu, C. C., Fan Chiang, S. H., Chou, C. Y., & Chen, S. Y. (2010). Knowledge exploration with concept association techniques. *Online Information Review, 34*(5), 786–805.

Liu, C. C., Liu, K. P., Wang, B. H., Chen, G. D., & Su, M. C. Applying tangible story avatars to enhance children's collaborative storytelling. British Journal of Educational Technology, in press, doi:10.1111/j.1467-8535.2010.01146.x.

Liu, C. C., & Tsai, C. C. (2008). An analysis of peer interaction patterns as discoursed by on-line small group problem-solving activity. *Computers & Education, 50*(3), 627–639.

Massimini, F., & Carli, M. (1988). The systematic assessment of flow in daily experience. In M. Csikszentmihalyi, & I. S. Csikszentmihalyi (Eds.), *Optimal experience: Psychological studies of flow in consciousness* (pp. 266–287). Cambridge, New York: Cambridge University Press.

Massimini, F. C., Csikszentmihalyi, M., & Delle Fave, A. (1988). Flow and bio-cultural evolution. In M. Csikszentmihalyi, & I. S. Csikszentmihalyi (Eds.), *Optimal experience: Psychological studies of flow in consciousness* (pp. 60–81). Cambridge: Cambridge University Press.

Mayer, R. E. (2004). Should there be a three-strikes-rule against pure discovery learning? A case for guided methods of instruction. *American Psychologist, 59*, 14–19.

McNerney, T. (2004). From turtles to tangible programming Bricks. *Personal and Ubiquitous Computing, 8*(5), 326–337.

Mitchell, M. T. (1997). Decision tree learning. In M. T. Michell (Ed.), *Machine learning* (pp. 52–80). New York: The McGraw-Hill Companies, Inc.

Monroy-Hernández, A., & Resnick, M. (2008). FEATURE: empowering kids to create and share programmable media. *Interactions, 15*(2), 50–53.

Novak, T. P. & Hoffman, D. L. (1997). Measuring the flow experience among web users. Paper Presented at the Interval Research Corporation.

Novak, T. P., Hoffman, D. L., & Yung, Y. F. (2000). Measuring the customer experience in online environments: a structural modeling approach. *Marketing Science, 19*(1), 22–42.

Papastergiou, M. (2009a). Digital game-based learning in high school computer science education: impact on educational effectiveness and student motivation. *Computers & Education, 52*(1), 1–12.

Papastergiou, M. (2009b). Exploring the potential of computer and video games for health and physical education: a literature review. *Computers & Education, 53*(3), 603–622.

Papert, S. (1972). Teaching children thinking. *Programmed Learning and Educational Technology, 9*(5), 245–255.

Pearce, J. M., Ainley, M., & Howard, S. (2005). The ebb and flow of online learning. *Computers in Human Behavior, 21*(5), 745–771.

Pintrich, P. R., Smith, D., Garcia, T., & McKeachie, W. (1991). *A manual for the use of the motivated strategies for learning questionnaire (MSLQ)*. Ann Arbor Michigan: The University of Michigan.

Polya, G. (1957). *How to solve it: A new aspect of mechanical method* (2nd ed.). Garden City, NJ: Doubleday.

Reiber, L. P. (2005). Multimedia learning in games, simulations, and microworlds. In R. E. Mayer (Ed.), *The Cambridge handbook of multimedia learning* (pp. 549–568). New York, NY: Cambridge University Press.

Rendas, A., Rosado Pinto, P., & Gamboa, T. (1999). A computer simulation designed for problem-based learning. *Medical Education, 33*, 47–54.

Renkl, A., Atkinson, R. K., Maier, U. H., & Staley, R. (2002). From example study to problem solving: smooth transitions help learning. *Journal of Experimental Education, 70*, 293–315.

Rivers, R. H., & Vockell, E. (1987). Computer simulations to stimulate scientific problem solving. *Journal of Research in Science Teaching, 24*, 403–415.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: a review and discussion. *Computer Science Education, 13*(2), 137–172.

Rodrigo, M. M. T., Baker, R. S. J. D., D'Mello, S., Gonzalez, M. C. T., Lagud, M. C. V., Lim, S. A. L., et al. (2008). Comparing learners' affect while using an intelligent tutoring systems and a simulation problem solving game. In *Proceedings Of the 9th International Conference on Intelligent Tutoring Systems* (pp. 40–49).

Roschelle, J., & Teasley, S. (1994). The construction of shared knowledge in collabrative problem solving. In C. E. O'Malley (Ed.), *Computer supported collaborative learning*. Heidelberg: Springer-Verlag.

Shih, J. L., Shih, B. J., Shih, C. C., Su, H. Y., & Chuang, C. W. (2010). The influence of collaboration styles to children's cognitive performance in digital problem-solving game "William Adventure": a comparative case study. *Computers & Education*, 1–12.

Skadberg, Y. X., & Kimmel, J. R. (2004). Visitors' flow experience while browsing a web site: its measurement, contributing factors and consequences. *Computers in Human Behavior, 20*, 403–422.

Stubbart, C. I., & Ramaprasad, A. (1990). Conclusion: the evolution of strategic thinking. In A. Huff (Ed.), *Mapping strategic thought*. NY: John Wiley and Sons.

Sung, K. (2009). Computer games and traditional CS courses. *Communications of the ACM, 52*(12), 74–78.

Swaak, J., & de Jong, T. (2001). Learner vs system control in using online support for simulation-based discovery learning. *Learning Environments Research, 4*(3), 217–241.

Tan, J., & Biswas, G. (2007). Simulation-based game learning environments: building and sustaining a fish tank. In *Proceedings Of the First IEEE International Workshop on Digital Game and Intelligent Toy Enhanced Learning* (pp. 73–80).

Trevino, L. K., & Webster, J. (1992). Flow in computer-mediated communication: electronic mail and voice mail evaluation and impacts. *Communication Research, 19*(5), 539–573.

Turkle, S. (1995). *Life on the screen: Identity in the age of the internet*. New York: Simon & Schuster.

Webster, J., Trevino, L. K., & Ryan, L. (1993). The dimensionality and correlates of flow in human-computer interaction. *Computers in Human Behavior, 9*, 411–426.

Wenger, E. (1986). *Artificial intelligence and tutoring systems: Computational approaches to the communication of knowledge*. Los Altos: Morgan Kaufmann Publishers.

Woodward, J., Carnine, D., & Gersten, R. (1988). Teaching problem solving through computer simulations. *American Educational Research Journal, 25*(1), 72–86.

Yamauchi, Y., Yokozawa, M., Shinohara, T., & Ishida, T. (2000). Collaboration with lean media: how open-source software succeeds. In *Proceedings of computer supported cooperative work Conference* (pp. 329–338).

Yamen, M., Nerdel, C., & Bayrhuber, H. (2008). The effects of instructional support and learner interests when learning using computer simulations. *Computers & Education, 51*(4), 1784–1794.