Research and Practice in Technology Enhanced Learning
Vol. 9, No. 3 (2014) 439－459

EXAMPLES AND TUTORED PROBLEMS: IS ALTERNATING EXAMPLES AND PROBLEMS THE BEST INSTRUCTIONAL STRATEGY?

AMIR SHAREGHI NAJAR

Intelligent Computer Tutoring Group
University of Canterbury, Christchurch, New Zealand
amir.shareghinajar@canterbury.ac.nz

ANTONIJA MITROVIC

Intelligent Computer Tutoring Group
University of Canterbury, Christchurch, New Zealand
tanja.mitrovic@canterbury.ac.nz

Previous studies have shown that learning from worked examples is superior to unsupported problem solving. Examples reduce the cognitive load on the learner's working memory, thus helping the student to learn faster or deal with more complex problems. Intelligent Tutoring Systems (ITS) support problem solving in many ways, adaptive feedback being one of them. Only recently researchers started comparing worked examples with ITSs, and several studies show the worked examples result in faster learning. We conducted a study to investigate the effects of studying Examples Only (EO) in comparison with Problems Only (PO) and Alternating Examples/Problems (AEP). Our results show that, in contrast to prior studies, learning solely from examples is not as effective as solving problems or a mixture of examples and problems. In our study, novices learned most from AEP, but advanced students learned the same from AEP and PO. Novices and advanced students learned less from EO than AEP and PO. Therefore, interleaving examples with supported problem solving is an optimal choice compared to using examples or supported problems only in SQL-Tutor.

Keywords: Worked examples; problem solving; self-explanation; intelligent tutors.

1. Introduction

In early stages of learning, learners benefit more from seeing worked-out examples (i.e. problems with solutions) than attempting to solve problems unaided. Numerous studies have found the worked example effect (Rourke & Sweller, 2009; Schwonke et al., 2009; Sweller, Ayres, & Kalyuga, 2011), when learners learn more from studying worked examples rather than solving problems. Sweller (2006) explains the worked example effect based on the Cognitive Load Theory (CLT). Examples provide step-by-step explanations and knowledge necessary to solve problems and thus decrease the load on the learner's working memory. Therefore, the example-based strategy is more helpful for novices who lack the necessary knowledge and have to deal with an enormous amount of cognitive load.

Researchers also compared learning from examples to learning with Intelligent Tutoring Systems (ITSs) (Kim, Weitz, Heffernan, & Krach, 2007; McLaren & Isotani, 2011; Schwonke et al., 2009). ITSs support problem solving by providing adaptive scaffolding in terms of feedback, guidance, problem selection and other types of help. The results of those studies show that examples result in faster learning in comparison with ITSs. However, little attention has been devoted so far to the difference between novices and advanced students in learning from examples and learning from supported problems solving.

Recently we have conducted a study that compared learning from examples only (EO), alternating examples and tutored problems (AEP), and tutored problems only (PO) in the area of specifying database queries in SQL. We scaffolded examples and problems with Self-Explanation (SE) prompts, which require the student to explain the principles necessary for solving a problem or explain how the solution was generated. Our hypothesis was that AEP condition would be superior to both PO ad EO conditions, and this hypothesis was confirmed (Shareghi Najar & Mitrovic, 2013).

In this paper, we turn our focus to the effect of the three conditions on novices and advanced students separately. We start by presenting an overview of related work in Section 2. Section 3 describes the material, participants and the procedure of the study, while Section 4 presents the results of the study. The discussion is presented in Section 5, followed by conclusions and the directions of future work in Section 6.

2. Related Work

In this section, we review prior work from two angles: studies that compared examples with unsupported problem solving (solving problems on paper), and studies that compared examples with ITSs. Research shows that self-explaining has an important impact on learning; therefore, this section starts by presenting a short overview of the self-explanation effect.

2.1. *Self-Explanation effect*

Self-Explanation (SE) is a metacognitive process in which the student explains the provided example to him/herself. Researchers have found evidence that students who generate explanations learn more than students who receive explanations (Brown & Kane, 1988; Chi, De Leeuw, Chiu, & LaVancher, 1994; Webb, 1989). Research shows that very few students self-explain spontaneously, but can be encouraged to self-explain with carefully designed prompts (Chi, Bassok, Lewis, Reimann, & Glaser, 1989; Chi et al., 1994).

Prior studies, either with a human teacher prompting self-explanations (Chi et al., 1994), or with an ITS prompting self-explanations (Weerasinghe & Mitrovic, 2006), show that self-explanation is an effective metacognitive strategy. Aleven and Koedinger (2002) investigate the effectiveness of self-explanation using the Geometry Cognitive Tutor, and show that students who engage in self-explanation acquire deeper knowledge.

According to Cognitive Load Theory (CLT), using examples decreases working memory load by reducing extraneous load (Sweller et al., 2011). Extraneous load is the information which is not related to learning (Clark, Nguyen, & Sweller, 2006). If the freed working memory loads with germane load, then it improves learning. In contrast to extraneous load, germane load is the information which is related to learning materials (Clark et al., 2006). One way of producing germane load is to prompt students to self-explain. In a study to teach concept mapping, Hilbert and Renkl (2009) show that students who gave self-explanations after they studied examples learned more than students who did not engage in self-explanation. In another study, Schworm and Renkl (2006) found that self-explanation is effective for studying worked examples and solved-example problems. Solved-example problems only provide the problem formulation and the solution while worked-out examples consist of a formulation, solution steps, and the final answer.

We believe that if self-explanations are designed specifically to complement problem solving and learning from examples, learning will be more effective. For instance, Schwonke et al. (2009) show that students who worked with examples had more conceptual knowledge than procedural knowledge, and students in problem-solving condition learned more procedural knowledge than conceptual knowledge. This suggests that different types of SE are needed to scaffold problem solving and examples.

SE prompts can be of different nature, according to the knowledge they focus on. For instance, Hausmann, Nokes, VanLehn, and Gershman (2009) compare justification-based prompts (e.g. “what principle is being applied in this step?”) and meta-cognitive prompts (e.g. “what new information does each step provide for you?”) with a new type called step-focused prompts (e.g. “what does this step mean to you?”). They found that students in the step-focused and justification conditions learned more from studying examples than students in the meta-cognitive prompts condition. In another study, Chi and VanLehn (1991) categorized SE as either procedural explanation (e.g. answer to “why was this step done?”), or derivation SE (e.g. answer to “where did this step come from?”).

To recap, SE has an effective impact on learning from examples and problems. However, learning from examples should be complemented with SE prompts focused on procedural knowledge, while problem solving should be complemented with SE prompts about the relevant domain principles.

2.2. *Learning from examples vs. unsupported problem-solving*

There has been no agreement on how much assistance should be provided to students. Kirschner, Sweller, and Clark (2006) show that maximum assistance (e.g. examples) is more efficient than minimal assistance which has been corroborated by prior studies (Atkinson, Derry, Renkl, & Wortham, 2000). Apart from the advantages of example-based strategy versus unsupported problem-solving, recently researchers focused on different example-based learning strategies. Van Gog, Kester, and Paas (2011) investigate the difference between worked examples only (WE), worked examples/problem-solving pairs (WE-PS), problem-solving/worked examples pairs (PS-

WE) and problem-solving only (PS) on novices. They used electrical circuits troubleshooting tasks, and the experiment run in four sessions. First, some general information was given to participants about the experiment, followed by a pre-test. Then participants started the condition-associated training tasks. They used mental scale rating to measure the actual cognitive load for each task, and these rates were indicated by participants in each task. Participants solved two problems (post-test) after the training task. The experiment was controlled for time. The result shows that the participants in WE and WE-PS had higher performance in the post-test than PS and PS-WE. Furthermore, the mental effort training and test rates in WE-PS and WE was lower than PS and PS-WE.

In a later study, Van Gog (2011) claimed that the previous results on WE-PS and PS-WE might be not sufficient. Examples which come after problems had different structure to the next problem; therefore, she opined that using identical pairs might lead to different result. She conducted a study using modeling examples in two conditions PS-ME-PS-ME and ME-PS-ME-PS in the Frog Leap game. In modeling examples, students have the opportunity to observe an expert or a peer performing the task (Van Gog & Rummel, 2010). After the two sequences of training, students worked on two tasks, of which the second one was not similar to training tasks. Results showed no difference in learning performance since the students learned most after studying the second worked example.

In another study, the student's prior knowledge has an important influence on the instructional formats. Formats which are efficient for some students might be not efficient for the student with different knowledge level (Kalyuga, 2007). In other words, if the additional information is not needed by the student, the expertise reversal effect is observed (Kalyuga, Chandler, Tuovinen, & Sweller, 2001); that is, if we provide the student with excessive information, this causes too much cognitive load which interferes with learning.

Most of the prior studies showed the example effect for well-defined tasks. Well-defined tasks are those for which there is an algorithm for solving problems (Mitrovic & Weerasinghe, 2009), such as mathematics and physics. Rourke and Sweller (2009) explain their two studies in which they investigated the example effect using ill-defined tasks. They hypothesized those students who learned how to identify the designer from observing examples of his/her work can identify their other works easier than students who learned from solving equivalent problems. Both studies consist of three phases. In the first phase, students participated in a design history lecture and afterward studied a worked example and solved a problem, or solved two problems, according to their group conditions. In the last phase, students answered visual recognition and short answer tests. The difference between the studies was in the participants' abilities. Students in the second study had a greater level of visual literacy skills than the students in the first study although both studies' participants had the same knowledge level on design history. In both studies, results show that the worked-example effect can be obtained in ill-defined tasks like well-defined tasks.

Kalyuga (2009) identifies five instructional designs, for the development of transferable knowledge and skill. These instructional designs are faded worked examples, worked-examples and tutored problems, worked examples and self-explanation, worked examples and visual mapping, and worked examples plus self-visualization. Hilbert and Renkl (2009) investigated the best structure of examples to teach concept mapping. They found that students learn more when the examples are presented with self-explanation than without it. They explained that when examples are presented alone, the amount of extraneous load decreases only; therefore, when the freed memory is allocated to germane load by using self-explanation, the learning gain improves.

In this section we presented a review of research that compared learning from worked examples to untutored problem solving. In the next section, we look at the example effect compared to tutored problem solving.

2.3. *Learning from examples vs. tutored problem solving*

Many prior studies addressed the advantages of example-based strategy over unsupported problem solving. Koedinger and Aleven (2007) criticized those studies because of the very different amounts of information provided to the two conditions (the unsupported problem-solving condition received no feedback upon submitting solution).

As the response to this criticism, Schwonke et al. (2009) compared an ITS (Geometry Tutor) with a new version which was enriched with faded worked examples. They conducted two experiments. In the first experiment, students in the problem-solving condition worked with pure problem-solving tasks, and students in the examples condition were working on fixed faded examples. The result revealed an improvement in learning time from using examples. In the second experiment, they used the think-aloud protocol in order to study relevant cognitive processes. This study also showed that learning from worked examples was more efficient than ITS. Salden, Koedinger, Renkl, Aleven, and McLaren (2010) reviewed a number of prior studies on worked examples (e.g. McLaren, Lim, and Koedinger (2008); Anthony, Mankoff, Mitchell, and Gross (2008)) and finally they bolster the idea that using worked examples in tutored problem-solving decreases learning time.

Learning from examples only is more efficient than learning from ITSs, especially for novices since they do not have adequate prior knowledge on the problem, and examples can help them obtain the needed information. Therefore, using a combination of examples and problem solving might lead to a better result.

McLaren et al. (2008) discuss three studies using the stoichiometry tutor in which they compared the problems condition to the examples condition. The students in the problems condition worked on solving problems while students in the examples condition observed worked examples, were prompted to self-explain, and solved isomorphic problems. Both groups had to take pre-test and post-test. In all three studies the examples condition resulted in faster learning, but there were no significant difference in the near transfer (far transfer was not measured). They suggest that one possible reason for no difference in the amount of knowledge learned is that students in the problems condition made initial problems into worked examples by clicking on the hint button, and then they tried to solve the next isomorphic problems. From the authors' point of view, that might be the reason of having the same near transfer for both groups.

In a recent study, McLaren and Isotani (2011) compared examples only, alternating worked examples/tutored problems, and all tutored problems. They conducted the study using Stoichiometry Tutor and modeling examples. Surprisingly, the result shows that the students benefit most by learning with worked examples only, at least with respect to the learning time. However, all examples were followed by self-explanation prompts while the problems were not. The authors indicate that this result is interesting at least in some domains, under some conditions. They also discovered that using interactive worked examples may sometimes be more beneficial than static worked examples and tutored problems. In interactive worked examples, students were asked about their understandings of the examples (e.g. comprehension questions).

Depending on how much information examples contain, they can be adapted to students' needs. Salden, Aleven, Renkl, and Schwonke (2009) compared fixed faded examples with adaptive ones. In fixed faded examples, the same steps of solutions were faded for all students, but in adaptive faded examples, the solution steps were faded with respect to the student's prior knowledge. They conducted two studies, one in a lab setting and the other in a classroom setting. Their main hypothesis was to see whether using adaptive examples combined with problem solving, compared to pure problem solving could lead to better learning. They tested three conditions: traditional problem-solving cognitive tutor (Geometry), to the versions of the cognitive tutor enriched with fixed examples and adaptive faded examples. The lab results indicate that adaptive examples led to better learning and higher transfer compared to the other conditions. In contrast, the classroom results depict no significant difference in immediate post-test, but in the delayed post-test students who used adaptive examples learned more. They believe that the difference in the lab and classroom's results might be caused by either inherent noise in the class compared to the lab, or using Cognitive Tutor's mastery learning criterion in the class (which led students to enjoy remedial problems for the concept they had not mastered yet).

To sum up, most of the studies show that using worked examples in ITSs results in reduced learning time. Although there are some studies showing higher transfer performance for faded examples, most studies have found no difference in the amount learned. Using adaptive worked examples might be more helpful if it is reinforced with problem-solving approach. In addition, all the prior studies on using examples in ITSs were in Geometry, Chemistry and Algebra domains. All these tutors teach well-defined tasks. Well-defined tasks are those for which there is an algorithm for solving problems (e.g. mathematics, physics) (Mitrovic & Weerasinghe, 2009). Rourke and Sweller (2009) show that the worked-examples effect can be obtained in both ill- and well-defined tasks compared to unsupported problem solving. To the best of our knowledge, learning from examples has never been compared with ITSs for ill-defined tasks and therefore conducting a study in such an ITS is important for advancing the field.

3. Experiment Design

In our project, we focus on defining database queries using the Structured Query Language (SQL). This instructional task is more complex than the learning tasks used in prior research, and is also an ill-defined task (Mitrovic & Weerasinghe, 2009).

We performed an experiment with SQL-Tutor, a constraint-based tutor that teaches SQL (Mitrovic, 1998; 2003). SQL-Tutor is a complement to traditional lectures; it assumes that the student has already acquired some knowledge via lectures and labs, and the tutor provides numerous problem-solving opportunities. The system currently contains 280 problems defined on 13 databases. Figure 1 illustrates the problem-solving page in SQL-Tutor, showing the problem text at the top, as well as the schema of the selected database at the bottom of the screen. Additional information about the meaning and types of attributes is available by clicking on the attribute/table name. The student can specify his/her solution by filling the necessary clauses of the SQL select statement. Before submitting the solution to be checked, the student can select the level of feedback they want to receive. SQL-Tutor provides seven levels of feedback. The lowest level, *Positive/Negative* feedback, simply states whether the solution is correct or how many mistakes there are. The *Error Flag* feedback identifies the clause that is incorrect. The *Hint* level specifies the message corresponding to one violated constraint, while the *Detailed Hint* provides more information about the relevant domain principle. The *List All Errors* level provides the hint messages for all violated constraints. The *Partial*

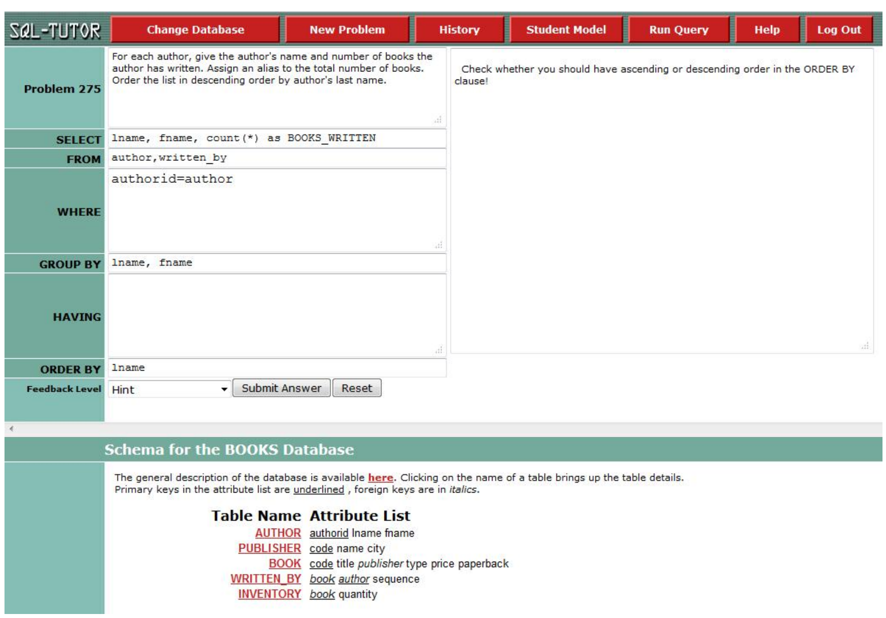

Figure 1. Screenshot of original SQL-Tutor.

Solution specifies the correct version of a clause that is incorrect in the student's solution, while the *Full Solution* provides the correct version of the SELECT statement. The feedback level automatically increases to the Detailed Hint level, while the student must explicitly request the higher levels. Students can attempt the same problem as many times as they want, and may switch to another problem at any time. The system selects the next problem based on the student model.

For this study, we developed three versions of SQL-Tutor which provided different combinations of worked examples and problems. Figure 2 shows the study design. In each condition, the student was given a set of 20 problems/examples arranged in pairs, so that the problems/examples in one pair were isomorphic. The Examples Only (EO) and Problems Only (PO) conditions presented isomorphic pairs of worked examples and problems consecutively, while the Alternating Examples Problems (AEP) condition presented a worked example followed by an isomorphic problem.

Previous research shows that worked examples increase conceptual knowledge more than procedural knowledge, while problem-solving produces results in higher acquisition of procedural knowledge (Kim et al., 2007; Schwonke et al., 2009). To compensate for this, we developed two types of SE: Conceptual-focused Self Explanation (C-SE) and Procedural-focused Self-Explanation (P-SE). C-SE prompts encourage students to reflect on concepts of the learning material (e.g. "what does the select clause in general do?"). P-SE prompts encourage students to self-explain the procedures of solutions (e.g. "what will happen if we don't use DISTINCT in this solution?").

Figure 3 shows a screenshot of a situation when a student has finished reading an example. The complete example was shown at the same time. Next, the system showed a P-SE prompt, located on the right side of the screen. The student gave a correct answer to the prompt, and the system provided positive feedback.

PO	**AEP**	**EO**
Pre-test		
20 problems in 10 isomorphic pairs	20 problems and examples in 10 isomorphic pairs	20 examples in 10 isomorphic pairs
1^{st} in each pair: problem 2^{nd} in each pair: problem	1^{st} in each pair: example 2^{nd} in each pair: problem	1^{st} in each pair: example 2^{nd} in each pair: example
Each problem followed by a C-SE prompt	Each problem followed by a C-SE prompt and each example followed by a P-SE prompt	Each example followed by a P-SE prompt
Post-test		

Figure 2. Design of study with three conditions.

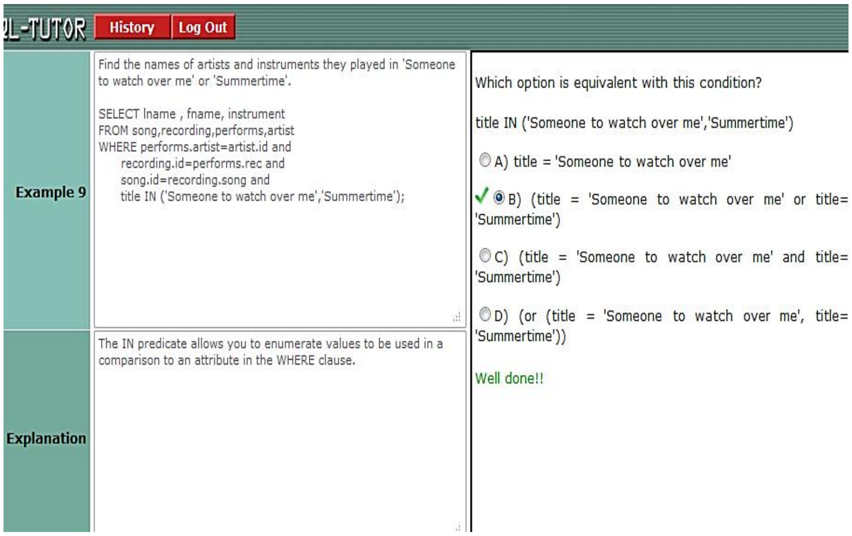

Figure 3. Screenshot of an example page followed by P-SE.

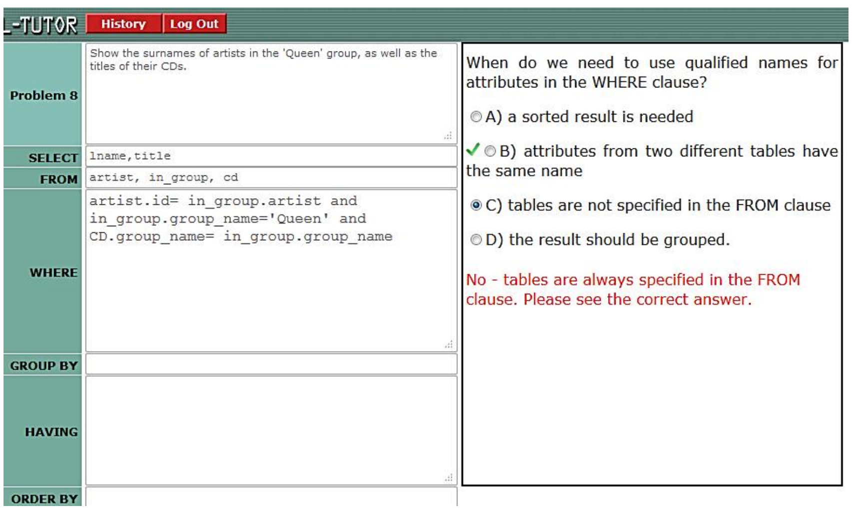

Figure 4. Screenshot of a problem solving page followed by C-SE.

Figure 4 shows a screenshot of a problem-solving task. In this situation, the student was given a C-SE prompt after s/he solved the problem. The student gave a wrong answer to the C-SE prompt, and because there is only one attempt per SE prompt, the system showed the negative feedback and revealed the correct answer. Once students received SE feedback, they could continue with the next task.

Participants were 34 students enrolled in the Relational Database Systems course at the University of Canterbury. They learned about SQL in lectures before-hand, and needed to practice in the lab. The students did not receive any inducements for participating in the study, but we told them that working with our system will help them learn SQL. We informed them that they would see ten pairs of problems, and that the tasks in each pair were similar.

The study was conducted in a single, 90-minute long session. At the beginning of the session, the students took a pre-test for 10 minutes. Once students logged in, SQL-Tutor randomly allocated them to one of the conditions (EO, PO, or AEP), giving sample sizes of 12 in PO, 11 in AEP and 11 in EO. The students then interacted with SQL-Tutor, and took the post-test at the end of the session.

The pre-test had five questions, three of which were multiple-choice and two were problem-solving questions. The first two multiple-choice questions measured conceptual knowledge students had, while the third question measured procedural knowledge. For the fourth and the fifth questions, students had to write SQL queries. These two questions measured procedural knowledge and the problem-solving skill of the students. The post-test was similar to the pre-test with one extra question about the difficulty of the tasks. We asked students to answer this question: “How difficult was it for you to complete the tasks in this study?” Students rated the complexity of the tasks on the Likert scale from 1 to 5 (*simple* to *difficult*). The maximum score on each test was 11.

4. Results

We calculated the average of scores in the pre-test and the post-test, and the time students spent on the system (Table 1). The students who had the pre-test scores lower than 45% were considered as novices and the rest were classified as advanced students.

We analyzed the data to find the answer to two questions. How did students learn from the three conditions? How did novices and advanced students benefit from different versions of the system? We start by explaining the results for all students followed by explaining the results for novices and advanced students.

4.1. *Results for all students*

The basic statistics about the study are presented in Table 2. There was no significant difference between the pre-test performances of the three groups. ANOVA revealed a significant difference between the post-test results ($p = .02$). The Tukey post-hoc test

Table 1. Basic statistics for all students (standard deviation given in brackets).

Number of students	34
Pre-test (%)	45 (14)
Post-test (%)	70 (17)
Learning time (min)	58 (20)

Table 2. Basic statistics for the three conditions (* denotes the mean difference significant at the 0.05 level).

	PO (12)	AEP (11)	EO (11)	p
Pre-test (%)	41.67 (13.82)	48.76 (13.19)	44 (14.63)	.48
Post-test (%)	72.73 (13.98)	77.69 (16.57)	58.68 (16.57)	*.02
Improvement	*p=.0, t=-9.8	*p=.0, t=-5.1	*p=.03, t=-2.4	
Pre/post-test correlation	*p=.01, r=.69	p=.49, r=.22	p=.43, r=.26	
Learning time (min)	69.67 (11.16)	65.91 (14.53)	38.45 (16.14)	*<.01
Number of attempted problems	14.58 (5.11)	14.09 (5.10)	18.63 (3.23)	.05
learning gain ^{N}	.54 (.19)	.55 (.31)	.21 (.35)	*.01
Problem solving gain ^{N}	.64 (.27)	.58 (.42)	.19 (.37)	*.01
Conceptual knowledge gain ^{N}	.29 (.39)	.77 (.41)	.54 (.47)	*.03
Procedural knowledge gain ^{N}	.59 (.22)	.48 (.42)	.13 (.40)	*.01
Perceived task difficulty	3.50 (.80)	3.27 (.90)	2.82 (.75)	

showed that the performance of the EO group was significantly lower than the AEP group ($p = .02$) and marginally significantly lower than the PO group ($p = .09$), thus confirming our hypothesis. The students in all three conditions improved significantly between the pre- and the post-test, as shown by the paired t-test reported in the *Improvement* row of Table 2. Correlations between the pre- and post-test scores are also reported in Table 2, but only the PO condition had a significant correlation ($p = .01$, $r = .69$).

There was also a significant difference between the mean learning times of the three groups ($p < .01$). The Tukey post-hoc test revealed that the EO group spent significantly shorter time than students in the AEP group and the PO group (both $p < .01$). The EO group participants were free to work with the system for the whole session, but spent much less time than the other two groups. This shows that the EO condition did not engage students like AEP and PO did. One potential explanation for this is that students overestimated their learning based on worked examples, and finished the tasks in a very short time.

There was a marginally significant difference between the three groups in the number of examples/problems they attempted ($p = .05$). The Tukey post-hoc test revealed that the EO group attempted more tasks than PO ($p = .1$) and the AEP group ($p = .07$).

The three groups also differed significantly in the normalized learning gain[1] ($p = .01$). The Tukey post-hoc test revealed that the EO group learned significantly less than students in the AEP group ($p = .02$) and the PO group ($p = .03$). When we analyzed normalized learning gains on the problem-solving questions only (questions 4 and 5), we found a significant difference between the groups ($p = .01$). As we expected, the students

[1] Normalized learning gain = (Post test - Pre test) / (Max score - Pre test); in the tables, (^{N}) represents normalized gain of a variable.

in the PO and AEP conditions performed significantly better than the students in the EO condition on problem-solving questions (Tukey post-hoc test: EO and PO p = .01, EO and AEP p = .04), because students in the EO condition were not given any problem-solving tasks during the learning phase.

We also analyzed how students acquired conceptual and procedural knowledge separately. Questions 1 and 2 in the tests measured conceptual knowledge, while the remaining three questions focused on procedural knowledge. There was a significant difference on both conceptual and procedural normalized learning gain. The Tukey post-hoc test reveals that the AEP group learned significantly more conceptual knowledge than the PO group (p = .02). Examples helped the AEP students to acquire conceptual knowledge. The students in the AEP condition acquired the most conceptual knowledge since they saw both examples and C-SE prompts. That was the only significant difference revealed by the Tukey post-hoc test. There was also a significant difference in the procedural knowledge gain (p = .01); the Tukey post-hoc test revealed a significant difference was between the PO and EO conditions (p = .01), and a marginally significant difference (p = .06) between the AEP and EO conditions.

In the post-test we also asked students about the perceived task difficulty. The Man-Whitney U test indicated that the PO group ranked the problems as more difficult in comparison to the ranking by the EO group; the difference is marginally significant (p = .053). This result was expected as problems impose more cognitive load on the working memory than examples (Sweller et al., 2011).

We calculated the effect size based on the normalized learning gain using Cohen's d, reported in Table 3. The effect sizes for both the AEP and PO conditions are large in comparison to the EO condition.

The participants received C-SE prompts after problems and P-SE after examples. Therefore, the AEP group saw half of the C-SE prompts that PO students received, and also half of the P-SE prompts that EO participants were given. We also analyzed the SE success rates for the three conditions, which are reported in Table 4. We found no

Table 3. The effect size on normalized learning gain between the groups.

Conditions		Effect size
AEP	PO	.04
AEP	EO	1.01
PO	EO	1.15

Table 4. SE prompts analysis.

	PO	AEP	EO	p
C-SE success rate (%)	88.50 (7.5)	92.84 (10.36)	N/A	.26
P-SE success rate (%)	N/A	77.69 (19.74)	71.36 (11.20)	.37

Table 5. Maximum hint level analysis.

	PO	AEP	
Second problem in pairs	1.08 (1.68)	1.54 (1.69)	p = .51
First problem in pairs	1.33 (1.56)		
	p = .39		

significant difference between AEP and PO in C-SE, and also no significant difference in P-SE success rate for the students in EO and AEP.

The students in the PO and AEP groups could select the feedback level when they submitted their solutions, up to the complete solution (the highest level of feedback). Therefore, the participants could transform a problem-solving task to a worked example by asking for the complete solution. For that reason, we analyzed help requests submitted for the problems given to the PO and AEP conditions.

Table 5 shows the mean number of problems for which the participants requested complete solutions. Looking at the second problem in each pair (the first row of Table 5) there was no significant difference in this respect between the PO and AEP conditions. Moreover, we did not see a significant difference in the number of times the PO students requested complete solutions for the first/second problem of each pair (p = .39). This result shows the participants from the PO/AEP groups have not converted their problems to worked examples often.

Plotting learning curve is a method to investigate how the students in AEP and PO learned SQL concepts in terms of constraints. For this, the number of times the constraints are relevant was plotted against the occasions when they were used incorrectly. A good fit to the power curve should result if the constraint measured is being learned (Martin & Mitrovic, 2005).

For this analysis, we excluded the EO group as they worked with examples only; consequently, they did not violate any constraints. Constraints are violated when a student submits an incorrect solution to a problem. Figure 5 shows the learning curve for

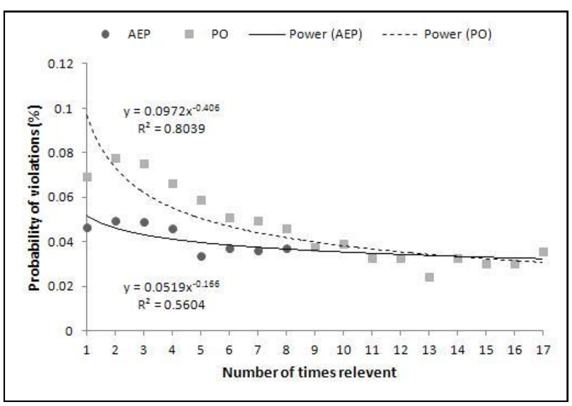

Figure 5. The probability of constraint violations for PO and AEP.

Table 6. The number of constraints learned.

	PO	AEP	
Constraints learned	7.75 (2.45)	3.09 (3.78)	p = .17

the PO and AEP conditions. The PO graph has a good fit to the power curve (R^2 = 0.80), but the fit to the AEP graph is poor (R^2 = 0.56). As shown by the slope, the learning rate is higher for the PO trend line. A possible explanation for this result is that students in PO violated fewer constraints after each attempt than students in AEP. Note that students in AEP saw an isomorphic example before solving a problem; therefore, they violated fewer constraints (as they had learned from the example).

We also investigated the number of constraints learned by students in AEP and PO. For each constraint in the student model, we considered the first five attempts and the last five attempts when the constraint was relevant. The constraint was considered to be learned if the probability of violating a constraint was reduced by 70% during the last five attempts (Weerasinghe, Mitrovic, Van Zijl, & Martin, 2010). We used the t-test to compare the number of constraints students learned in AEP and PO. Table 6 shows the result. We found no significant difference between AEP and PO in the number of constraints students learned. A possible reason is that AEP might learn constraints from examples; thus, they did not violate the constraints when they solved the subsequent problems.

4.2. *Results for novices and advanced students*

In this section, we present results for novices and advanced students in PO, AEP and EO. We compared novices, and then advanced students in the three conditions.

Table 7 presents some statistics about the novices. The Kruskal-Wallis 1-way ANOVA test did not reveal a significant difference on the pre-test performance of novices from the three conditions; therefore, our groups were comparable. Using the

Table 7. Dependent variables for novices (^{N} Normalized).

	PO	AEP	EO	p
Number of students	6	5	5	
Pre-test score (%)	31 (11)	36 (5)	33 (10)	.79
Post-test score (%)	65 (12)	73 (18)	53 (14)	.14
Improvement pre- to post-test	p = .03*	p = .04*	p = .07	
Learning gain ^{N}	.50 (.14)	.56 (.30)	.29 (.20)	.13
Learning time (min)	67 (12)	70 (12)	46 (15)	.06
Multiple choice questions ^{N}	.33 (.27)	.60 (.09)	.20 (.68)	.15
Problem solving questions ^{N}	.56 (.22)	.55 (.44)	.24 (.35)	.28
Conceptual knowledge ^{N}	.42 (.38)	1.00 (0)	.70 (.45)	.04*
Procedural knowledge ^{N}	.52 (.20)	.45 (.36)	.18 (.29)	.12

same test, we saw no significant difference between the groups on the post-test. The Wilcoxon signed ranks tests show that novices in PO and AEP condition improved significantly between the pre-test and the post-test while EO shows a marginally significant improvement. There are no significant differences between the three conditions on the normalized learning gain, but there is a marginally significant difference in learning time ($p = .06$). The Mann-Whitney test shows that novices in EO spent significantly less time than novices in AEP ($p = .03$) and PO ($p = .05$). The table also indicates a significant difference in the normalized conceptual knowledge gain ($p = .04$), and the Mann-Whitney test revealed that novices learned significantly more conceptual knowledge from AEP than PO ($p = .01$).

The participants received C-SE prompts after problems and P-SE after examples. Therefore, the AEP group saw half of the C-SE prompts that PO students received, and also half of the P-SE prompts that the EO participants were given. The SE success rates for novices are reported in Table 8. The Kruskal-Wallis 1-way ANOVA test shows a significant difference between novices in the three conditions on the overall success rate. The Mann-Whitney test reveals that novices in PO and AEP scored significantly higher than novices in EO ($p < .01$ and $p = .03$). Moreover, the Mann-Whitney test indicates a significant difference in P-SE success rate on SE prompts ($p = .3$); thus, novices in AEP performed significantly better than novices in EO who saw the same type of SE prompts (P-SE).

Overall, the analyses of the pre-test, post-test and SE performances confirm our hypothesis that novices benefit more from AEP or PO than using EO. The ITS engaged novices with both examples and problems while examples could not provide any rehearsal opportunity. On the other hand, AEP novices learned significantly more conceptual knowledge than PO. Since novices in the PO condition did not have a chance to improve their conceptual knowledge (apart from C-SE prompts), the AEP novices outperformed PO by acquiring significantly more conceptual knowledge due to studying examples.

Students who scored more than average in the pre-test were classified as advanced students, and their performance is reported in Table 9. The Kruskal-Wallis 1-way ANOVA reveals that there was no significant difference between the pre-test performances of the three groups; thus, our groups were comparable. Although the table shows no significant difference between the three conditions in the post-test, the Wilcoxon signed ranks tests revealed that advanced students in EO did not significantly improved between the pre-test and the post-test ($p = .42$). The table shows a marginally

Table 8. Analysis of SE performance for novices.

	PO	AEP	EO	p
SE success rate (%)	88 (7)	87 (12)	67 (8)	.02*
C-SE success rate (%)	88 (7)	90 (14)	N/A	.26
P-SE success rate (%)	N/A	85 (12)	67 (8)	.03*

Table 9. Dependent variables for advanced students ([N] Normalized).

	PO	AEP	EO	p
Number of Students	6	6	6	
Pre-test (%)	52 (6)	59 (7)	55 (10)	.16
Post-test (%)	80 (13)	82 (15)	64 (18)	.16
Improvement pre- to post-test	p =.03*	p = .03*	p = .42	
Learning gain	.59 (.24)	.55 (.36)	.15 (.46)	.23
Learning time (min)	73 (10)	63 (17)	32 (15)	<.01*
Multiple choice questions [N]	.17 (.26)	.50 (.44)	-.03 (.82)	.34
Problem solving questions [N]	.72 (.32)	.61 (.45)	.16 (.42)	.08
Conceptual knowledge [N]	.17 (.40)	.58 (.49)	.42 (.49)	.28
Procedural knowledge [N]	.66 (.26)	.52 (.50)	.08 (.50)	.12

Table 10. Analysis of SE prompts for advanced students.

	PO	AEP	EO	p
SE success rate (%)	89 (8)	83 (12)	75 (13)	.18
C-SE success rate (%)	89 (8)	95 (6)	N/A	.22
P-SE success rate (%)	N/A	72 (24)	75 (13)	.94

significant difference on the problem-solving, and a significant difference in learning time between the groups. The Mann-Whitney test shows a significant difference between EO and PO on problem-solving ($p = .04$), and learning time ($p < .01$). This result is in line with those studies show advanced students learn more from problem-solving only than reviewing examples only. The Mann-Whitney test also shows a significant difference between EO and AEP on learning time ($p = .02$). Note, that the result shows insignificant improvement between pre-test and post-test for students who studied examples only while students spent less time than the other groups on the system. That may be caused by illusion of understanding.

We analyzed the performance of advanced students on SE prompts, summarized in Table 10. Kruskal-Wallis 1-way ANOVA test shows no significant difference between the three groups. A possible explanation is that the difficulty of the self-explanation prompts was not suitable for the advanced students. The SE prompts gradually become more complex, but advanced students might not have difficulty understanding the prompts as they have more domain knowledge.

Overall, we found that novices improved the most from the AEP condition in comparison to the other two conditions. Moreover, advance students did not improve when learning from examples only; therefore, EO was not an appropriate approach for them.

As we mentioned before, students could transform a problem-solving task to a worked example by asking for the complete solution. Therefore, we analyzed the help requests submitted for the problems given to the PO and AEP conditions. Similar to the result for all students (Table 5), there was no significant difference, in number of requests for complete solution, between novices and advanced students in PO and AEP.

5. Discussion

Prior research shows that students, particularly novices, learn more from examples than unsupported problem solving. On the other hand, most of the studies that compared examples to ITSs indicate that students learn the same from worked examples and ITSs, in domains with well-defined tasks. This encouraged us to observe the examples effect in a domain with ill-defined tasks (SQL). We compared students' performance in three conditions: alternating example/problems, problems only and examples only. We analyzed the data to find how students in general benefit from different versions of the system, and how novices and advanced students improve on each condition.

We found no significant difference between PO and AEP in the normalized learning gain and learning time. However, the AEP group acquired significantly more conceptual knowledge than the PO group. Consequently, the best instructional condition in our study for all students was AEP, and our hypotheses were confirmed. The AEP participants learned from the worked examples (the first task in each pair); when they were presented with isomorphic problems, they were already primed and did not have to deal with many unfamiliar details like students in the PO group.

The results show that novices who worked with alternating examples and problems, and problems only outperformed novices who worked with examples only. This suggests that novices benefit most when they were engaged in tutored problem solving. On the other hand, the results show that novices in alternating examples and problems outperformed problems only in conceptual knowledge acquisition; thus, alternating examples and problems is the best learning strategy for novices. The difference between alternating examples and problems and the other two groups was that the novices were able to increase their initial learning by studying examples and then use what they have learned to tackle isomorphic problems.

In addition, advanced students did not significantly improve in the examples only condition. This is an expected result, since advanced students had enough prior knowledge, so what they need was practicing that knowledge in solving new problems. The EO condition did not get problem-solving opportunities. Moreover, examples might cause the expertise reversal effect for advanced students (Kalyuga, Chandler, and Sweller, 1998). Expertise reversal effect indicates that worked examples are more convenient in the early stages of learning while students could benefit more from problem solving in later stages (Salden et al., 2009).

6. Conclusion and Future Work

The results show that students who worked with examples did not learn the same as students who worked with problems only and alternating examples/problems. Our result is in contrast with the findings presented in (McLaren & Isotani, 2011). There are three main differences between the two studies. First, in our study the participants were given self-explanation prompts after examples and after problems, not only after worked examples as in (McLaren & Isotani, 2011). Moreover, we designed self-explanation prompts to complement problem solving and examples. We provided procedural-focused self-explanation prompts after examples, as examples have been shown to reinforce conceptual knowledge more than procedural knowledge. We also provided conceptual-focused self-explanation prompts after problem solving to reinforce the acquisition of conceptual knowledge. Therefore, both types of self-explanation prompts were designed to complement the type of learning provided by the main activity (problem solving or learning from examples). The second difference is in the instructional domain used in each study. The instructional task in the McLaren and Isotani's study was simpler, consisting of simple algebraic equations and basic chemistry concepts, while in our study the participants were specifying SQL queries. Thirdly, our constraint-based tutor provided feedback on demand while the Stoichiometry tutor used in McLaren and Isotani (2011) provided immediate feedback.

Why are worked examples not as effective as supported problem solving? Worked examples alone do not engage students as much as problem solving, and over time some students become less motivated to put enough effort into learning. Moreover, supported problem solving in contrast with unsupported problems avoid impasses, and is thus less frustrating and more effective. Examples may also induce an illusion of understanding. For instance, students may think they have already learned the example while they have not; consequently, they pass over the example very fast without spending enough time to process it which causes shallow learning. One potential approach to scaffold learning from worked examples is to provide support for self-assessment. For example, Roll, Aleven, McLaren, and Koedinger (2011) describe the Self-Assessment Tutor which is an ITS to improve the accuracy of the students' judgments regarding their own knowledge.

A limitation of our study is the small number of participants. It would therefore be interesting to see the results of a larger study. Moreover, it could be argued that this result is due to differences between conceptual-focused and procedural-focused self-explanation. As discussed previously, we used two different types of self-explanation prompts in order to reinforce examples and problems with the most suitable prompts. For instance, it is not appropriate to reinforce examples with conceptual-focused self-explanation prompts because examples have been shown to increase conceptual knowledge (Kim et al., 2007; Schwonke et al., 2009).

Sweller and Cooper (1985) explained a two-step learning process. First, examples are suitable approach for students, particularly for novices, since examples reduce the cognitive load and increase the initial learning. Second, students use the knowledge they learned from studying examples in solving similar problems. Our results are in line with two-step learning process. However, using ITS instead of examples leads to a higher performance, because ITS provides students with a variety of supports. In general, our study justified a learning strategy that helps students in early stages (novices) and in later stages (advanced students). This strategy suggests using a combination of examples and ITS for novices, then when student's knowledge increases, the system can continue giving them a mix of examples and problem solving, or gradually switches to ITS only. The result suggests that for a long learning time, problems only may even outperform alternating examples and problems condition since advanced students do not need any more knowledge, and what they need is applying knowledge in solving new problems. Nevertheless, in our study, novices learn the most from AEP since SQL is not completely novel to them (all the students in our study attended several lectures about SQL a week before the study). Thus, if we adapt the proportion of examples and problems to students' needs, the system may work more efficient than EO, AEP and PO at any stage of learning.

In future work, we plan to test an adaptive model, which chooses the best assistance level for students. We will compare this model with a fixed sequence of examples/problems like the AEP condition we explained in this paper. We expect that students who will study in the adaptive condition will improve more than students who work in the non-adaptive condition.

References

Aleven, V., & Koedinger, K. R. (2002). An effective metacognitive strategy: Learning by doing and explaining with a computer-based Cognitive Tutor. *Cognitive Science*, *26*(2), 147–179.

Anthony, L., Mankoff, J., Mitchell, T., & Gross, M. (2008). *Developing handwriting-based intelligent tutors to enhance mathematics learning*. Doctoral dissertation, Carnegie Mellon University, USA.

Atkinson, R. K., Derry, S. J., Renkl, A., & Wortham, D. (2000). Learning from examples: Instructional principles from the worked examples Research. *Review of Educational Research*, *70*(2), 181–214.

Brown, A. L., & Kane, M. J. (1988). Preschool children can learn to transfer: Learning to learn and learning from example. *Cognitive Psychology*, *20*(4), 493–523.

Chi, M. T. H., & VanLehn, K. A. (1991). The content of physics self-explanations. *The Journal of the Learning Sciences*, *1*(1), 69–105.

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, *13*(2), 145–182.

Chi, M. T. H., De Leeuw, N., Chiu, M. H., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science*, *18*(3), 439–477.

Clark, R. C., Nguyen, F., & Sweller, J. (2006). *Efficiency in learning: Evidence-based guidelines to manage cognitive load*. San Francisco: Jossey-Bass.

Hausmann, R., Nokes, T., VanLehn, K., & Gershman, S. (2009). The design of self-explanation prompts: The fit hypothesis. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 2626–2631).

Hilbert, T. S., & Renkl, A. (2009). Learning how to use a computer-based concept-mapping tool: Self-explaining examples helps. *Computers in Human Behavior*, *25*(2), 267–274.

Kalyuga, S. (2007). Expertise reversal effect and its implications for learner-tailored instruction. *Educational Psychology Review*, *19*, 509–539.

Kalyuga, S. (2009). Instructional designs for the development of transferable knowledge and skills: A cognitive load perspective. *Computers in Human Behavior*, *25*(2), 332–338.

Kalyuga, S., Chandler, P., & Sweller, J. (1998). Levels of expertise and instructional design. *Human Factors and Ergonomics Society*, *40*, 1–17.

Kalyuga, S., Chandler, P., Tuovinen, J., & Sweller, J. (2001). When problem solving is superior to studying worked examples. *Journal of Educational Psychology*, *93*(3), 579–588.

Kim, R. S., Weitz, R., Heffernan, N. T., & Krach, N. (2007). Tutored problem solving vs. “Pure” worked examples. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society*. *Austin, TX: Cognitive Science Society* (pp. 3121–3126).

Kirschner, P. A., Sweller, J., & Clark, R. E. (2006). Why minimal guidance during instruction does not work: An analysis of the failure of constructivist, discovery, problem-based, experiential, and inquiry-based teaching. *Educational Psychologist*, *41*(2), 75–86.

Koedinger, K., & Aleven, V. (2007). Exploring the assistance dilemma in experiments with cognitive tutors. *Educational Psychology Review*, *19*(3), 239–264.

Martin, B., & Mitrovic, A. (2005). Using learning curves to mine student models. In L. Ardissono, P. Brna & A. Mitrovic (Eds.), *Proceedings of 10th Int. Conf User Modeling* (pp. 79–88). Springer.

McLaren, B., & Isotani, S. (2011). When is it best to learn with all worked examples? In G. Biswas, S. Bull, J. Kay & A. Mitrovic (Eds.), *Proceedings of 15th Int. Conf AIED* (pp. 222–229). Berlin/Heidelberg: Springer.

McLaren, B. M., Lim, S.-J., & Koedinger, K. R. (2008). When and how often should worked examples be given to students? New results and a summary of the current state of research. In *Proceedings of Conf. Cognitive Science Society* (pp. 2176–2181). Austin, TX: Cognitive Science Society.

Mitrovic, A. (1998). Experiences in implementing constraint-based modeling in SQL-Tutor. In B. Goettl, H. Halff, C. Redfield & V. Shute (Eds.), *Proceedings of Int. Conf. ITS* (pp. 414–423).

Mitrovic, A. (2003). An intelligent SQL tutor on the Web. *International Journal of Artificial Intelligence in Education*, *13*(2-4), 173–197.

Mitrovic, A., & Weerasinghe, A. (2009). Revisiting ill-definedness and the consequences for ITSs. In V. D. R. Mizoguchi (Ed.), *Proceedings of 14th Int. Conf. on AIED* (pp. 375–382). Amsterdam, The Netherlands: IOS Press.

Roll, I., Aleven, V., McLaren, B., & Koedinger, K. (2011). Metacognitive practice makes perfect: Improving students self-assessment skills with an intelligent tutoring system. In *Proceedings of Int. Conf. AIED* (pp. 288–295).

Rourke, A., & Sweller, J. (2009). The worked-example effect using ill-defined problems: Learning to recognise designers’ styles. *Learning and Instruction*, *19*(2), 185–199.

Salden, R., Aleven, V., Renkl, A., & Schwonke, R. (2009). Worked examples and tutored problem solving: Redundant or synergistic forms of support? *Topics in Cognitive Science*, *1*(1), 203–213.

Salden, R. J. C. M., Aleven, V., Schwonke, R., & Renkl, A. (2009). The expertise reversal effect and worked examples in tutored problem solving. *Instructional Science*, *38*(3), 289–307.

Salden, R., Koedinger, K., Renkl, A., Aleven, V., & McLaren, B. (2010). Accounting for beneficial effects of worked examples in tutored problem solving. *Educational Psychology Review*, *22*(4), 379–392.

Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., & Salden, R. (2009). The worked-example effect: Not an artefact of lousy control conditions. *Computers in Human Behavior*, *25*, 258–266.

Schworm, S., & Renkl, A. (2006). Computer-supported example-based learning: When instructional explanations reduce self-explanations. *Computers and Education*, *46*, 426–445.

Shareghi Najar, A., & Mitrovic, A. (2013). Examples and tutored problems: How can self-explanation make a difference to learning? In K. Yacef et al. (Eds.), *Proceedings of 16h Int. Conf. AIED* (pp. 339–348). Memphis, USA: Springer, Heidelberg.

Sweller, J. (2006). The worked example effect and human cognition. *Learning and Instruction*, *16*(2), 165–169.

Sweller, J., & Cooper, G. A. (1985). The use of worked examples as a substitute for problem solving in learning algebra. *Cognition and Instruction*, *2*, 59–89.

Sweller, J., Ayres, P., & Kalyuga, S. (2011). *Cognitive load theory*. Springer.

Van Gog, T. (2011). Effects of identical example-problem and problem-example pairs on learning. *Computers & Education*, *57*(2), 1775–1779.

Van Gog, T., & Rummel, N. (2010). Example-based learning: Integrating cognitive and social-cognitive research perspectives. *Educational Psychology Review*, *22*(2), 155–174.

Van Gog, T., Kester, L., & Paas, F. (2011). Effects of worked examples, example-problem, and problem-example pairs on novices' learning. *Contemporary Educational Psychology*, *36*(3), 212–218.

Webb, N. M. (1989). Peer interaction and learning in small groups. *International Journal of Educational Research*, *13*(1), 21–39.

Weerasinghe, A., & Mitrovic, A. (2006). Facilitating deep learning through self-explanation in an open-ended domain. *International Journal of Knowledge-based and Intelligent Engineering Systems*, *10*(1), 3–19.

Weerasinghe, A., Mitrovic A., Van Zijl, M., & Martin, B. (2010). Evaluating the effectiveness of adaptive tutorial dialogues in database design. In *Proceedings of Int. Conf. on Computers in Education* (pp. 33–40). Putrajaya, Malaysia.