Providing Adaptive Support for Meta-Cognitive Skills to Improve Learning

Kasia Muldner and **Cristina Conati**
Department of Computer Science
University of British Columbia
2366 Main Mall Vancouver, B.C., V6T 1Z4, CANADA
{kmuldner,conati}@cs.ubc.ca

Abstract

We describe a computational framework designed to provide adaptive support for learning from problem solving activities that make worked-out examples available. This framework targets several meta-cognitive skills required to learn effectively in this type of instructional setting, including explanation-based-learning-of-correctness and min-analogy. The generated interventions are based on an assessment of a student's knowledge and meta-cognitive skills provided by the framework's student model, and thus are tailored to that student's needs.

Introduction

Many cognitive science studies have shown that students rely on worked-out examples, especially in the early and intermediate phases of learning (e.g., (Anderson & Fincham 1994; Anderson, Fincham, & Douglas 1997; Atkinson *et al.* 2000; Chi *et al.* 1989; Reed, Dempster, & Ettinger 1985; Reed 1987; Reed & Bolstad 1991; VanLehn 1996; 1998; 1999). In fact, examples facilitate the learning process during problem-solving activities better than other instructional materials, such as general procedures (Reed & Bolstad 1991). The potential downside of using examples is that many students do not possess the necessary analogical reasoning skills needed to use them effectively, resulting in diminished learning gains (e.g., (Chi *et al.* 1989; Chi & VanLehn 1992; Novick 1988; 1995; Cooper & Sweller 1987; Sweller & Cooper 1985; VanLehn, Jones, & Chi 1992; VanLehn & Jones 1993; VanLehn 1998; 1999)). Some of these skills are meta-cognitive in that they are not domain dependent, and therefore are an important aspect of a student's ability to learn in general.

Since computers are becoming more commonplace in instructional settings, there is growing interest in developing computational learning environments that improve various aspects of the learning process. In the past, most of these environments have focused on targeting cognitive traits, and in particular, domain knowledge (e.g., (Anderson *et al.* 1995)). Lately, new emphasis has been placed on the benefits of environments that also provide support for domain-independent, meta-cognitive skills. For example, a number of computational tutors have been developed that support the meta-cognitive skill of self-explanation, i.e., the process of explaining and elaborating instructional material to oneself (e.g., (Conati & VanLehn 2000; Mitrovic 2003)). However, we are not aware of an adaptive learning environment offering support for the types of meta-cognitive skills that are required to learn effectively from problem-solving activities when worked out examples are also available. In this paper, we describe how support for these meta-cognitive aspects can be realized through a computational framework, referred to as the E-A (Example-Analogy) Coach. This framework aims to maximize learning for different types of students by encouraging those behaviors which are beneficial to learning, while discouraging those that are not.

We begin by describing how students use examples and the related meta-cognitive skills needed to use them effectively. Next, we present and discuss the overall E-A framework. Finally, we describe how adaptive support for the relevant meta-cognitive skills can be realized in this framework.

APS Phases & Related Meta-Cognitive Skills

The process of using worked-out examples during problem-solving activities (also referred to as analogical problem solving, APS) can be characterized by two main phases: 1) retrieval of the example, and 2) application of the example solution to the target problem (e.g., (VanLehn 1996)).

Retrieval Phase

The retrieval phase involves the selection of an example that is similar to the target problem. This problem-example similarity may be assessed by using the various kinds of features that characterize the problem and example. Typically, these features are grouped into two categories (e.g., (Chi, Feltovich, & Glaser 1981; Novick 1988)): 1) *superficial* features, i.e. features not part of the ideal domain knowledge needed to solve the problem, such as the actions and objects making up the problem specification and/or its solution, and 2) *structural* features, i.e. the domain principles needed to generate the solution.

Research suggests that retrieval is a function of expertise, where novice students, unlike experts, typically make example selections on the basis of superficial similarity between the example and the problem (e.g., (Chi, Feltovich, &

A block resting on a plane pushes on it (commonsense reasoning). This push is a physics force acting on the plane (overly-general rule), causing the plane to push back on the block (by Newton's Third Law) - this push is the normal force.

Figure 1: EBLC example

Glaser 1981; Novick 1988; Silver 1979; Schoenfeld & Hermann 1982)). This has the potential to diminish what students learn from the example in several ways. First, it may result in selection errors, where the example is not appropriate (typically because its solution corresponds to a different principle than the problem's) (e.g., (Chi, Feltovich, & Glaser 1981; Novick 1988)). Second, examples exhibiting certain types of superficial similarity do not encourage students to use those meta-cognitive skills that benefit learning, as we will discuss shortly.

Application Phase

The application phase involves applying the example solution (for example, by copying it). The learning outcomes resulting from this phase are heavily influenced by a number of meta-cognitive skills, which can be classified among two dimensions: 1) the preferred style of problem-solving in the presence of examples (min/max analogy dimension) and 2) the reasoning mechanism of choice (the reasoning dimension).

Min/Max Analogy Dimension A relevant APS meta-cognitive skill is min-analogy, which characterizes a student's preferred style problem solving when examples are available (e.g., (VanLehn & Jones 1993; VanLehn 1998)). Specifically, students who choose to solve problems on their own, without the help of an example, are classified as min-analogy students. These students tend to refer to examples only to check their solutions, or when at an impasse. Max-analogy students, on the other hand, prefer to replace regular problem-solving by coping from examples, even if they have the knowledge required to generate the problem solution without the help of the example.

There are two ways in which min/max behaviors impact what a student learns (VanLehn & Jones 1993; VanLehn 1998). Unlike max-analogy, min-analogy provides an opportunity for students to 1) uncover knowledge gaps (i.e. through impasses encountered during problem solving), and 2) strengthen their knowledge through practice. Thus, min-analogy is good for learning and max-analogy is not (VanLehn & Jones 1993; VanLehn 1998).

Reasoning Dimension Once an example is selected, the student may choose to think about its solution. This may be motivated by several factors, including the desire to transfer an example solution line requiring some adaptation to make it suitable to the target problem, and/or the desire to learn or confirm a rule embedded in the example solution (e.g, (Chi *et al.* 1989; Reed, Dempster, & Ettinger 1985; VanLehn, Jones, & Chi 1992; VanLehn 1999)). A relevant meta-cognitive skill which can accomplish both these goals is explanation-based-learning-of-correctness (EBLC) (Chi & VanLehn 1992; VanLehn, Jones, & Chi 1992; VanLehn 1998).

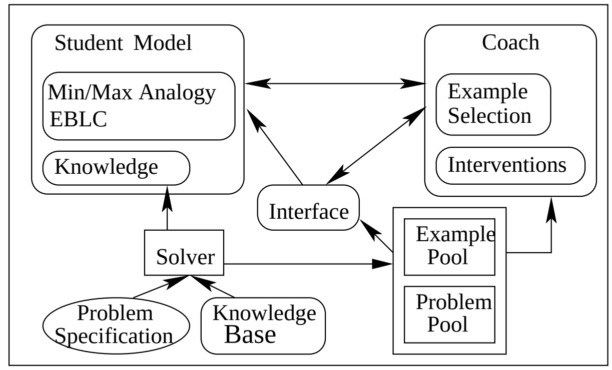


Figure 2: E-A Architecture

EBLC is a form of self-explanation used by students to overcome impasses when existing domain knowledge is insufficient. This process involves using common sense knowledge about properties and values (instead of principled domain knowledge), in conjunction with general rules, to derive new rules. Fig. 1 shows an example (borrowed from (VanLehn 1999)) of how EBLC can be used to derive a rule about the existence of a normal force, needed to generate the solution to a physics problem of the type shown in Fig. 3. Unfortunately, many students employ other processes instead, which either do not result in learning, or result in shallow forms of knowledge (e.g., (Reed, Dempster, & Ettinger 1985; VanLehn 1998; 1999)).

Adaptive Support for APS

The previous discussion highlighted meta-cognitive skills relevant to APS activities. We will now describe an overall framework designed to provide support for these activities, and in particular, to encourage those behaviors which are beneficial for learning while discouraging those that are not. This support is realized in a computer tutor, referred to as the E-A (Example-Analogy) Coach, which is embedded in the Andes infrastructure (Conati & VanLehn 2000). Andes is a tutoring system for Newtonian Physics, which is also the target domain for the E-A Coach. Currently, Andes does not provide support for the use of examples *during* problem-solving activities, which is needed to ensure that students learn effectively, as discussed in the previous section. We are thus working on designing and implementing a framework capable of providing this support computationally, and now describe the structure of this support.

E-A Coach Architecture

We begin by describing the overall architecture of the E-A Coach, shown in Fig. 2. The system contains two data bases of problems: worked-out examples (*example pool* in Fig. 2) and problems for the students to solve (*problem *pool* in Fig. 2). The solutions to these problems and examples are automatically generated by the *solver*, using the problem specification and the rules found in the *knowledge base* component. The E-A *interface* component allows students to solve problems from the problem pool and to refer to worked-out examples in the example pool. The *student model* component provides an assessment of both a student's domain knowledge, and her meta-cognitive behaviors related to a) min/max analogy and b) EBLC tendencies during the application phase. This assessment is used to generate tailored interventions by the *coach* component, as well as to update the student model. The coach component is also responsible for selecting examples for students tailored to their needs, based the student model's assessment.

We will now describe how this framework can be used to support APS in more detail. This description is organized according to the two phases of APS, example retrieval and application.

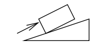
Support for Phase 1 of APS: Example Retrieval

One of the reasons students decide to look for an example during problem solving activities is because they lack some piece of knowledge needed to generate the solution for the target problem. Unfortunately, as we pointed out above, these knowledge gaps also cause students to experience difficulties selecting appropriate examples at a point in the learning process when examples are the most useful (e.g, (Chi, Feltovich, & Glaser 1981; Novick 1988; Schoenfeld & Herrmann 1982; Silver 1979)). A solution to this predicament is to have the framework's Coach select examples for students. This is a task very well-suited for an adaptive learning environment, since it would not be realistic to expect a teacher to perform tailored example selections for each student.

To find an appropriate example, it is crucial that the system can reason about the impact of the similarity between an example and the target problem on a given student's knowledge and meta-cognitive skills. We therefore begin by describing this impact.

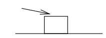
Impact of Similarity To help make the subsequent discussion more concrete, we will refer to Fig. 3, which shows a problem and a small example pool (in the interests of space, only a portion of the solution for the problem and example 1 is shown) The example pool in this figure consists of examples which are structurally identical to the problem (i.e. require the same knowledge to generate the solution). However, the example specifications are superficially different from the problem and from each other, as are some of their solution elements, demonstrating that even in this restricted scenario, structurally identical examples can appear as superficially different. We will now discuss the impact of each type of similarity (structural, superficial).

Since structural features correspond to domain principles needed to generate the solution, structural differences between a problem and example may impact the usefulness of the example. This is particularly the case if a difference corresponds to a student's knowledge gap, since it means that the student can not reconcile the difference. Thus,



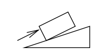
Problem. A 5kg block is being pushed up a ramp inclined 40 degrees, with an acceleration of 3m/s^2. The force is applied to the block at 40 degrees to the horizontal, with a magnitude of 100N. Find the normal force on the block.

Solution (only first portion shown):
[1] We will apply Newton's 2nd Law.
[2] First, we choose the *block* as the body.
[3] One force acting on the block is the normal force
[4] It is directed perpendicular to the ramp and away from it.

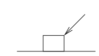


Example 1. A workman pushes a 50 kg. crate along the floor. He pushes it hard, with a magnitude of 120 N, applied at an angle of 25 degrees. The crate is accelerating at 6 m/s^2. What is the normal force on the crate?

Solution (only first portion shown):
[1] We will apply Newton's 2nd Law.
[2] First, we choose the *crate* as the body.
[3] One force acting on the crate is the normal force
[4] It is directed straight up.



Example 2. Jake pushes his sled up the hill, which is inclined 15 degrees. The sled weighs only 3 kg, and he pushes it with a force of 50 N, applied at 15 degrees wrt the horizontal. The sled is accelerating at 1.5 m/s^2. What is the normal force on the sled?



Example 3. Bob has decided to replace his refrigerator. He pushes his old one (which has a mass of 50 kg) along the kitchen floor. He's giving it his all, pushing with a force of 75 N applied at an angle of 23 degrees. The refrigerator is accelerating at 7 m/s^2. What is the normal force on the refrigerator?

Figure 3: Scenario Problem and Examples

systems that select examples for students aim to minimize the structural differences between the problem and example (e.g., (Weber 1996)).

Unlike the case with structural differences, the impact of superficial differences between the target problem and an example is less clear. There exists some research in the cognitive science community demonstrating that superficial similarity affects students' ability to select appropriate examples (e.g., (Bassok, Wu, & Olseth 1995; Chi, Feltovich, & Glaser 1981; Reed 1987)). However, what still remains unclear is the overall impact of the various kinds of differences on the relevant meta-cognitive processes, and thus learning, from superficially different examples. In the process of exploring this issue, we realized that we needed a finer-grained classification of superficial similarity than what is presently found in the literature. Thus, we have formulated our own, based on comparing structurally identical solution elements, and classifying any superficial differences as either trivial or non-trivial:

- *trivial* differences correspond to elements that appear *both* in the example's specification and its solution, and have a corresponding element in the problem specification

- *non-trivial* differences corresponds to elements that do not appear in the example/problem specifications.

For illustrative purposes, we will classify two of the superficial differences between the problem and example 1 shown in Fig. 3. One trivial difference between the two corresponds to the object chosen as the body in Step 2 of the two solutions: *block* and *crate* problem and example solution elements respectively. This constant appears both in the example's specification and its solution, and can be mapped to the corresponding constant in the problem specification (i.e. block), thus satisfying the conditions required for a trivial difference. Thus, one way that this difference can be reconciled to generate a correct problem solution is by replacing the example constant with the problem constant. A non-trivial difference between this problem/example pair corresponds to the direction of the normal force (Step 4, Fig. 3) Unlike the trivial case, the elements corresponding to this difference (i.e. in force directions) do not appear in the problem and example specifications, and so the student can not reconcile the difference by substituting example and problem elements.

Given this classification, the question that still needs to be addressed concerns the impact of these two kinds of superficial similarity on students' meta-cognitive skills and thus learning. There is some indication that students generally do not have difficulty reconciling trivial differences, but do not necessarily learn from doing so (Reed, Dempster, & Ettinger 1985; VanLehn 1998). One possible explanation for this finding is that problems and examples sharing only trivial superficial differences allow students to fall back on more shallow, syntactic processes to generate the answer (e.g., such as transformational analogy (VanLehn 1998)). In other words, these differences do not force students to use those meta-cognitive skills which are beneficial to learning. Thus, superficial similarity has the potential to impact students who lack these skills, and have low knowledge corresponding to the transferred elements. For these types of students, example 2 from the example pool presented in Fig. 3 would therefore be a poor choice, since unlike examples 1 and 3, the only differences between it and the problem are superficially-trivial ones. Although there is no direct evidence with regards to the impact of non-trivial differences, our hypothesis is that these do have the potential to encourage EBLC and min-analogy, since these are the only processes that will reconcile the superficial difference between the problem and the example, thus allowing the student to correctly continue problem-solving (assuming that the student is given correctness feedback). Of course, a potential limitation of imposing these kinds of differences is that some students may not be capable of reconciling them. For these students, more direct support may be required.

E-A Example Selection To summarize the above discussion, we propose that a number of factors should play a role when the system selects an example for a student, including: 1) similarity, both superficial (including a consideration of trivial and non-trivial differences) and structural, 2) student knowledge and 3) student meta-cognitive skills. We are working on combining these factors in a principled manner, that will allow the system to select the most appropriate example for a given student.

Support for Phase 2 of APS: Application

The first stage of providing support for APS is through the selection of an appropriate example. Once an example is provided, however, further support may be necessary to encourage students to use the appropriate meta-cognitive skills, including min-analogy and EBLC. Since not all students have the same meta-cognitive skills, it is vital that E-A Coach rely on the student model's assessment of each student so that these interventions can be tailored to a particular student's needs. Below, we discuss how such tailored support for min-analogy and EBLC can be provided.

Support / Assessment of Min/Max Analogy In order to discourage max-analogy, the E-A Coach needs to provide interventions targeting students who have an overall tendency to transfer (i.e. copy) from examples. The simplest form that these interventions could take is text-based hints, but we are also investigating other possibilities, including exploring ways in which interface design affects students' tendency to transfer from examples.

As we pointed out above, these interventions should be tailored to an individual student's needs, and thus be based on the E-A model's assessment of the corresponding skills. We will begin by discussing how the student model could obtain information needed to assess min/max analogy and then discuss how this information can be used by the model to generate an assessment of a particular student's skills.

To assess min/max analogy, the model needs information about copying behaviors. There are two obvious sources of such information: visual attention and similarity. The simplest approach for using these sources of information involves having the model assume that whenever an example is open, and student input corresponds to elements found in the example solution, the student is copying. The downside of this approach is that the model may over-estimate some students' max-analogy tendency, since an open example does not guarantee a student is transferring from it. To improve accuracy, the student model needs information regarding visual attention to allow it to deduce if and where the student is looking at the example. One way to accomplish this is to make the interface more restrictive. For instance, we could adopt the SE-Coach interface design (Conati & VanLehn 2000). With this design, example solutions are covered (and can be uncovered by moving the mouse over them), which allows the system to track which line the student is looking at. This information can be used in conjunction with subsequent student input to the problem solution to identify transferred elements. A third alternative involves using eye-tracking technology, which provides the most ac-

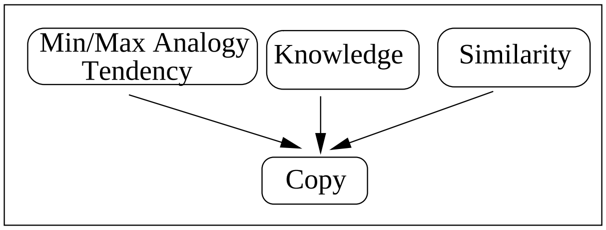


Figure 4: Assessment of Copying Behavior

curate information regarding visual attention. For an initial prototype of our system, we are investigating the trade-offs between the first two options by exploring the loss of accuracy that occurs if the model does not have direct information about students' visual attention.

To generate its assessment of a student's tendency for min/max analogy, the E-A student model relies on the Andes approach, which 1) uses the Solver to automatically generate a solution graph from the problem specification, which consists of all the steps and corresponding knowledge needed to solve the problem, as well as paths between these steps, 2) converts this solution graph into a Bayesian network, and 3) uses the network in real-time to perform knowledge assessment (Conati *et al.* 1997). In particular, each time a student enters a solution to a problem, the student model uses this input as evidence to update its belief in that student's knowledge. Notice that this model assesses student knowledge, which is also useful in the E-A framework. To see why this is so, consider the fact that an assessment of knowledge is needed to fully assess the impact of a particular copy. This is because one of the major downsides of max-analogy is that it allows students to keep their knowledge gaps intact (VanLehn 1998) (when steps corresponding to these gaps are copied). Given an accurate assessment of *both* knowledge and min/max analogy tendency allows the system to intervene at appropriate times. The Andes student model, however, does not account for transfer from examples, needed in the E-A framework to generate an accurate assessment of knowledge and min/max analogy. Thus, to make it appropriate for assessing APS activities we are working on extending this model.

These extensions involve supplementing the Bayesian network representing the model with additional nodes representing the various states which need to be assessed, as well as links representing relations between these nodes. For instance, to assess copying behaviors in the absence of direct evidence, the model can take a number of factors into account, including student knowledge, the similarity between the problem and the example and the student's tendency for max-analogy. Fig. 4 shows the high-level relations between these factors and the possibility of a copy. Incorporating these factors allows the model to predict, for example, that a student with a tendency for max-analogy is more likely to copy than a student with a min-analogy tendency, and that the former is more likely to copy steps corresponding to trivial differences than steps sharing other kinds of similarity. Knowledge plays a role because this configuration encodes the assumption that students are more likely to copy steps that they do not have the knowledge to generate.

Support / Assessment of EBLC When students reach an impasse during problem solving and decide to refer to an example, they have a choice: copy from the example and maintain the knowledge gap that caused the impasse, or reason via EBLC, fill the gap, and use the newly acquired knowledge from the EBLC process to continue problem solving. Unfortunately, there is some indication that not all students possess the meta-cognitive skills needed to reason effectively (Chi & VanLehn 1992). Thus, it is important that the system provide support to students who have a low tendency for EBLC and who are at an impasse in their problem solving. This feedback, delivered by the E-A Coach component, could take a number of forms, including 1) text-based hints and 2) specially designed tools for those students who require the additional scaffolding, as in (e.g., (Bunt, Conati, & Muldner 2004; Conati & VanLehn 2000)). These tools could be mirrored on the design of the SE-Coach (Conati & VanLehn 2000), which allows students to self-explain solution steps via provided dialog boxes by re-deriving the rule which generated the solution step. However, these tools need to be extended to incorporate support for the common-sense reasoning required by EBLC.

To allow the Coach to tailor these interventions to a student's behaviors, as is the case with encouraging min-analogy, it needs to rely on the student model's assessment. Designing a model capable of assessing EBLC presents a number of challenges. The first concerns the question of which student actions the model should use as evidence of EBLC. The above mentioned interface tools are one source of evidence, since they force students to explicitly demonstrate that EBLC took place. It would be too restrictive, however, to always force students to use these tools, especially since some students have an inherent tendency for EBLC (Chi & VanLehn 1992). Unfortunately, in the absence of tool usage, the model has very little information about which type of reasoning (if any) the student is engaging in when referring to and transferring from an example. One factor that could provide some indication of EBLC is information regarding a student's overall tendency for EBLC, as suggested in (Bunt, Conati, & Muldner 2004). An additional factor that the model could take into account is the superficial similarity between the problem and the example, since as we proposed previously, certain kinds of similarity, including non-trivial superficial differences, may encourage EBLC.

Given this information, the next question to be addressed is how the model should use it to perform its assessment of EBLC. We are planning to further extend the student model discussed in the previous section to also account for EBLC actions, which will allow it to generate an assessment of this meta-cognitive skill. We also intend to have the model use its appraisal of EBLC to influence its assessment of student knowledge of the corresponding concept, thus modeling student learning as a consequence of the interaction.

Summary

We have described a general framework aimed at encouraging those meta-cognitive skills which are beneficial to learning during APS activities. This framework relies on a model of relevant student meta-cognitive behaviors, and uses the assessment generated by this model to tailor interventions to individual student's needs. We are working on completing the design and implementation of this framework, and plan to validate it with evaluations upon doing so.

Acknowledgments We would like to thank Andrea Bunt, Benjamin Phillips and the anonymous reviewers for their helpful comments.

References

Anderson, J., and Fincham, J. 1994. Acquisition of procedural skills from examples. *Journal of Experimental Psychology: Learning, Memory and Cognition* 20(6):1322–1340.

Anderson, J. R.; Corbett, A. T.; Koedinger, K.; and Pelletier, R. 1995. Cognitive tutors: Lessons learned. *The Journal of Learning Sciences* 4(2):167–207.

Anderson, J.; Fincham, J.; and Douglas, S. 1997. The role of examples and rules in the acquisition of a cognitive skill. *Journal of Experimental Psychology: Learning, Memory and Cognition* 23(4):932–945.

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

Bassok, M.; Wu, L.; and Olseth, K. 1995. Judging a book by its cover: Interpretive effects of content on problem-solving transfer. *Memory and Cognition* 23(3):354–367.

Bunt, A.; Conati, C.; and Muldner, K. 2004. Scaffolding self-explanation to improve learning in exploratory learning environments. In *Seventh International Conference on Intelligent Tutoring Systems*, 656–667. Springer.

Chi, M., and VanLehn, K. 1992. The content of physics self-explanations. *The Journal of the Learning Sciences* 1(1):69–105.

Chi, M.; Basssok, M.; Lewis, M.; Reimann, P.; and Glaser, R. 1989. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science* 13:145–182.

Chi, M.; Feltovich, P.; and Glaser, R. 1981. Categorization and representation of physics problems by experts and novices. *Cognitive Science* 5(2):121–152.

Conati, C., and VanLehn, K. 2000. Toward computer-based support of meta-cognitive skills: a computational framework to coach self-explanation. *International Journal of Artificial Intelligence in Education* 11:389–415.

Conati, C.; Gertner, A.; VanLehn, K.; and Druzdzel, M. 1997. On-line student modeling for coached problem solving using bayesian networks. In Jameson, A.; Paris, C.; and Tasso, C., eds., *Proceedings of the sixth International Conference in User Modeling*, 231–242. Springer-Verlag.

Cooper, G., and Sweller, J. 1987. Effects of schema acquisition and rule automation on mathematical problem-solving transfer. *Journal of Educational Psychology* 79(4):347–362.

Mitrovic, A. 2003. Supporting self-explanation in a data normalization tutor. In Aleven, V.; Hopppe, U.; J. Kay, R. M.; Pain, H.; Verdejo, F.; and Yacef, K., eds., *International Conference on Artificial Intelligence in Education, Supplementary Proceedings*, 565–577.

Novick, L. 1988. Analogical transfer, problem similarity and expertise. *Journal of Experimental Psychology: Learning, Memory and Cognition* 14(3):510–520.

Novick, L. 1995. Some determinants of successful analogical transfer in the solution of algebra word problems. *Thinking and Reasoning* 1(1):1–30.

Reed, S., and Bolstad, C. 1991. Use of examples and procedures in problem solving. *Journal of Experimental Psychology: Learning, Memory and Cognition* 17(4):753–766.

Reed, S.; Dempster, A.; and Ettinger, M. 1985. Usefulness of analogous solutions for solving algebra word problems. *Journal of Experimental Psychology: Learning, Memory and Cognition* 11(1):106–125.

Reed, S. 1987. A structure-mapping model for word problems. *Journal of Experimental Psychology: Learning, Memory and Cognition* 13(1):124–139.

Schoenfeld, A., and Herrmann, D. 1982. Problem perception and knowledge structure in expert and novice mathematical problem solvers. *Journal of Experimental Psychology: Learning, Memory and Cognition* 8(5):484–494.

Silver, E. 1979. Student perceptions of relatedness among mathematical verbal problems. *Journal for Research in Mathematics Education* 10(3):195–210.

Sweller, J., and Cooper, G. 1985. The use of worked examples as a substitute for problem solving in learning algebra. *Cognition and Instruction* 2(1):59–89.

VanLehn, K., and Jones, R. 1993. Better learners use analogical problem solving sparingly. In Utgoff, P. E., ed., *Proceedings of the Tenth International Conference on Machine Learning, San Mateo, CA*, 338–345. Morgan Kaufmann.

VanLehn, K.; Jones, R.; and Chi, M. 1992. A model of the self-explanation effect. *The Journal of the Learning Sciences* 2(1):1–59.

VanLehn, K. 1996. Cognitive skill acquisition. *Annual Review of Psychology* 47:513–539.

VanLehn, K. 1998. Analogy events: How examples are used during problem solving. *Cognitive Science* 22(3):347–388.

VanLehn, K. 1999. Rule-learning events in the acquisition of a complex skill: An evaluation of cascade. *The Journal of the Learning Sciences* 8(1):71–125.

Weber, G. 1996. Individual selection of examples in an intelligent learning environment. *Journal of Artificial Intelligence in Education* 7(1):3–33.