# Better Learners Use Analogical Problem Solving Sparingly

**Kurt VanLehn**
Learning Research and Development Center
University of Pittsburgh
Pittsburgh, PA 15260
VanLehn@cs.pitt.edu
(412)624-7458

**Randolph M. Jones**
Artificial Intelligence Laboratory
University of Michigan
1101 Beal Av.
Ann Arbor, MI 48109-2110
rjones@engin.umich.edu
(313)764-0683

## Abstract

When solving homework exercises, human students often notice that the problem they are about to solve is similar to an example. They then deliberate over whether to refer to the example or to solve the problem without looking at the example. We present protocol analyses showing that effective human learners prefer not to use analogical problem solving for achieving the base-level goals of the problem, although they do use it occasionally for achieving meta-level goals, such as checking solutions or resolving certain kinds of impasses. On the other hand, ineffective learners use analogical problem solving in place of ordinary problem solving, and this prevents them from discovering gaps in their domain theory. An analysis of the task domain (college physics) reveals a testable heuristic for when to use analogy and when to avoid it. The heuristic may be of use in guiding multistrategy learners.

## 1 WHEN TO ANALOGIZE?

When doing homework exercises, human learners often notice that the problem they are about to solve is similar to an example, then deliberate over whether to refer to the example or to solve the problem without its aid. As one of our subjects said, "this looks very much like the one I had in the examples. Okay. Should I just go right to the problem, which I distinctly remember? Or should I try to do it without looking at the example?" A multistrategy machine learning program could face the same decision. The objective of this paper is to find out what heuristics good human learners use for deciding whether to do analogical problem solving, then determine when those heuristics would be good for a machine learning program to use.

Because we use protocol data, the only evidence we have of analogical problem solving is episodes where a person explicitly refers to an example, typically by flipping pages in a textbook in order to expose the page on which the example is printed. Thus, analogical problem solving, in this paper, means the process of referring to a written example rather than a mentally held one. As will be seen later, nothing in our conclusions relies on this restriction, so the results may apply to analogies that refer to mental examples (or cases?) as well as written ones.

The protocol data come from subjects learning Newtonian physics. The subjects worked with textbook physics problems and examples, such as the one in Figure 1. The protocol data were collected as part of a study by Chi, Bassok, Lewis, Reimann & Glaser (1989). The subjects were 9 college students selected to have similar backgrounds (Chi & VanLehn, 1991). The subjects first refreshed their mathematical knowledge by studying the first 4 chapters of Halliday & Resnick (1981), a popular physics textbook. They then studied the expository part of chapter 5, which introduces the basic principles of Newtonian mechanics, its history and some classic experiments. Student were tested at this point and had to re-study parts of the material that they did not understand. After they had mastered the mathematical prerequisites and the basic principles, they studied 3 examples and solved 19 problems while talking aloud. They were allowed to refer to the examples at any time while solving problems, but they were not allowed to refer to their own previous problem solutions. The 9 subjects' protocols, which averaged 5 hours each, are the raw data for the findings reported here. They contain many instances of analogical problem solving. The goal is to discover which ones helped learning and which ones hurt it.

We used a contrastive protocol analysis technique pioneered by Chi et al. (1989). The basic idea is to split the subjects into two groups—effective learners and ineffective learners—then determine what the effective learners did differently from the ineffective learners. Because the students were trained to have the same prerequisite knowledge, the scores on their problem solving reflect their learning rate during example

.

Figure 1: A physics example, with line numbers added

studying and problem solving. The 4 highest scoring subjects constitute the effective learners (called Good solvers by Chi et al.), and the 4 lowest scoring subjects constituted the ineffective learners (called Poor solvers). The middle subject's protocol was not analyzed (until later: see below).

The next section presents a learning mechanism and argues that it is the main source of learning by subjects in this study. The argument uses new protocol analyses as well as analyses published earlier. With this as background, the subsequent sections present the main result, which is that effective learners use analogical problem solving sparingly. A discussion section speculates on why this policy was better for human solvers in this experiment, and suggests conditions under which this policy would be good for any multi-strategy learner.

## 2 GAP FILLING

Given that errors are used to determine when learning was not effective, a direct way to uncover dominant learning mechanisms is to examine the subjects' errors. If errors of a certain type are much less common among Good solvers than Poor solvers, we can assume that a learning mechanism employed by the Good solvers and not the Poor solvers is reducing those errors. From the characteristics of such errors we can infer the characteristics of the learning processes. We classified errors into 5 types, which are listed below:

- *Inappropriate analogies.* Sometimes subjects fetched an example that was inappropriate for the problem being solved. At other times, subjects fetched appropriate examples but applied them in inappropriate ways. Both types of errors are classified as inappropriate analogies.
- *Gap errors.* Subjects often lacked a piece of physics knowledge, such as the fact that the tension in a string is equal to the magnitude of a tension force exerted by that string. Sometimes errors would occur when the subject reached an impasse caused by their lack of knowledge, and used some ineffective repair strategy (VanLehn, 1990) to work around it. At other times, the gap would cause an error (such as a missing minus sign) without the subject ever becoming aware of the gap.
- *Schema selection errors.* All subjects knew several methods or schemas for solving physics problems. One method was to draw forces, generate equations and solve the equations. Another method was simply to generate equations that contained the sought and/or known quantities without considering what forces or other physical quantities might be present. On some problems, subjects chose the equation-chaining schema instead of the force schema, and this caused them to answer the problem incorrectly.
- *Mathematical errors.* A typical mathematical error was to confuse sine and cosine, or to drop a negative sign.
- *Miscellaneous errors.*

The error classification was done separately by two coders, with an intercoder reliability of 82%. Differences were reconciled by collaborative protocol analysis.

Table 1: Mean errors per subject for each error category

| Error type | Good | Poor |
|---|---|---|
| Inappropriate analogies | 1.00 | 2.25 |
| Gap errors | **0.25 | **7.75 |
| Schema selection | 0.50 | 1.75 |
| Math errors | 0.25 | 0.75 |
| Miscellaneous errors | 1.25 | 1.75 |
| Totals | **3.25 | **14.25 |

Table 1 shows the average number of errors of each type per subject. Although the Good solvers had fewer errors than the Poor solvers in every category, the difference was significant only for gap errors ($t(6) = 5.36, p < .01$). Moreover, the difference was quite large (3.8 standard deviations), and accounts for most (68%) of the difference in the total error rates of the Good and Poor solvers.

These results suggest that Good solvers were more effective learners than Poor solvers because they employed some kind of learning process that filled in the gaps in their knowledge.[1] There are many kinds of mechanisms in the literature that can detect and rectify incomplete domain theories. For handy reference, let us refer to the process(es) that Good solvers use as *gap filling* even though we do not know what it is. Table 1 suggests that gap filling is the main learning process that differentiates effective from ineffective learners in this study.

This suggestion is consistent with findings from Chi et al.'s (1989) analysis of the same data. They found that during example studying, Good solvers tended to thoroughly explain the examples to themselves, while the Poor solvers tended to read them rather casually. Further examination of the protocols suggested that self-explanation consisted of actually rederiving the lines of the solution (VanLehn, Jones & Chi, 1991). If the main learning process is gap filling, then this method of studying the example should cause the subjects to detect gaps in their knowledge. If a piece of physics knowledge is required for deriving a line of the example's solution, and students lack that knowledge, then they will be unable to fully explain the line. The resulting impasse might cause them to seek the missing knowledge and fill their gap. Thus, the gap-filling hypothesis is consistent with the finding that Good solvers self-explain examples more than Poor solvers. Moreover, it explains why self-explanation causes better learning (VanLehn & Jones, in press-b).

The computational sufficiency of gap-filling has been tested by implementing a simulation of human learning, called Cascade, that is based a particular gap filling mechanism and comparing Cascade's behavior to the protocols (VanLehn, Jones & Chi, 1991; VanLehn & Jones, 1991; in press-a). Gaps can cause Cascade to reach impasses (i.e., be unable to achieve a goal) while trying to solve problems or rederive examples. When Cascade's "official" domain knowledge is insufficient to achieve a goal, it tries to apply overly general knowledge that captures regularities common to many types of scientific and mathematical problem solving. For instance, one overly general rule is that scientific concepts often correspond roughly to common sense concepts. Cascade uses this rule during a problem where a block rests on a spring. Cascade lacks the knowledge that a compressed spring exerts a force on the objects at its ends, so it reaches an impasse. The overly general rule applies, because Cascade knows that springs push back when you push on them. The overly general rule justifies creating an instance of a scientific concept ("force") because it involves the same objects as an instance of a lay concept ("push back"). As a side-effect of the application of this overly general knowledge, a new domain rule is proposed: If a block rests on a spring, the spring exerts a force on it. If this rule is used successfully enough times, it becomes a full-fledged member of the domain theory. In this fashion, Cascade fills gaps in its domain knowledge.

Cascade's behavior compares well with both aggregate findings (VanLehn, Jones & Chi, 1991) and individual protocols (VanLehn & Jones, 1993). This establishes that with plausible assumptions about subjects' prior knowledge, there is enough information present in the environment to allow a gap filling process to learn everything that the Good solvers learn, to do so without implausibly large computations, and to generate outward behavior that is similar to the subjects' behavior. In short, gap filling is a computationally sufficient account for the Good solvers' learning.

None of the results show that gap filling is the *only* learning process going on. There could be others as well. However, Table 1 suggests that gap filling is the most important learning process, because it accounts for most of the difference in the learning of the Good and Poor solvers.

## 3 AVOIDING ANALOGY

There is already some evidence that the Good solvers avoid analogical problem solving. This section reviews those findings, then tries to ascertain whether this is a just a correlation or whether avoiding analogy actually causes more effective learning.

Chi et al. (1989) counted episodes of analogical prob-

[1] An alternative explanation is that the Good solvers never had the gaps because they learned the knowledge before studying the examples. Analyses of pre-test data (Chi et al., 1989), the instructional material (VanLehn, Jones & Chi, 1991) and the subjects' backgrounds (Chi & VanLehn, 1991) fail to support this interpretation of the data.

lem solving during the first 3 problems. They found that Good solvers used analogy only 2.7 times per problem, whereas the Poor solvers used analogy 6.7 items per problem. Thus, the Good solver use analogical problem solving less often than the Poor solvers. Chi et al. also found that the Good solvers used analogy in a more focused way. When the Good solvers referred to an example, they tended to jump into the middle of it and read only a few lines (1.6 lines per episode, on average). The Poor solvers tended to start at the beginning of the example and read until they found something they could use (13.0 lines per episode). This suggests that Good solvers are basically solving the problem on their own, but they occasionally use analogical problem solving to get specific information from the example. The Poor solvers, on the other hand, seem to use analogical problem solving instead of regular problem solving. These findings indicate that effective learning co-occurs with avoiding of analogy, but it is not clear which way the causality runs.

Our first hypothesis was that the Poor solvers used more analogical problem solving because they lacked domain knowledge so they had to refer to the example if they were to make any progress. Cascade embedded this hypothesis. It did analogy (called transformational analogy in earlier reports) only when it reached an impasse (VanLehn, Chi & Jones, 1991; VanLehn & Jones, in press-a). On this account, the Chi et al. correlation is due to ineffective learning causing analogy.

However, when we fitted Cascade to individual protocols, we found that we sometimes had to force it to do analogy even though it had the knowledge to do regular problem solving (VanLehn & Jones, 1993). While simulating all 9 subjects, Cascade used analogy 231 times, and 196 of these were caused by impasses while 35 (15%) were caused by our intervention. If we believe the modeling, then these 35 analogies were "optional" in that the subjects did not have to do them. They could have used their knowledge of physics principles instead. In most of these cases (30 of 35), the subjects copied the example's force diagram rather than generate their own. Copying the force diagram was also frequent among the 196 impasse-driven analogies.

Upon reflection, it occurred to us that some of these supposedly impasse-driven analogies may not actually be caused by trying to generate forces, failing, and reaching an impasse. If this were the case, then one would expect the analogy to yield new knowledge about the missing force (or whatever the missing knowledge was), thus filling the gap and allowing the person to draw their own force diagram the next time it was needed. We examined all 196 cases of analogy and found no cases were this kind of learning occurred. If a person had an gap that caused an impasse-driven analogy, then they would use analogy on every subsequent occasion (if any) when that piece of knowledge was required. It could be that what people learn from such an impasse is that "analogy works here," so they continue to use it. However, it could also be that our modeling was incorrect, and they never had such any impasses for that gap. Instead, when they go to certain sections of the problem (typically, the force diagram), they would use analogy without ever considering using their domain knowledge. Perhaps some of those 196 cases of impasse-driven analogy were really optional analogies. Indeed, two of the subject never tried to draw a force diagram on their own—they always copied an example's diagram.

While investigating the gap-filling hypothesis, we discovered additional support for this conjecture. According to the gap-filling hypothesis, gaps in the textbook become gaps in the student's domain knowledge, which cause errors until they are detected and remedied. In order to check this story, we carefully analyzed the first 5 chapters of Halliday and Resnick (1981) and discovered 9 pieces of knowledge that are required by the problems and are not in the text (VanLehn & Jones, in preparation). Using Cascade, for each of the 9 subjects, we located the places in the protocols where the 9 pieces of knowledge could appear if they were known, or cause errors if they were unknown. For each of the 9 pieces of knowledge, we created a chart, such as the one shown in Table 2, that summarizes what happened at each possible occurrence of the gap. The particular piece of knowledge referenced by Table 2 is "Projecting a vector onto the negative portion of an axis yields a negative formula." This piece of knowledge is relevant 5 times during example studying and 16 times during problem solving. At each place, for each subject, we classified the protocol fragment into one of the categories shown below (the symbol in parentheses corresponds to the code used in Table 2).

- (E) The subject omited use of the knowledge, which resulted in an error.
- (O) The subject omited use of the knowledge, but no error occurred. For instance, one sign error might compensate for another.
- (blank) During example studying, the subject did not explain the part of the example where this piece of knowledge would be used. During problem solving, the subject used analogical problem solving to avoid the line of reasoning that would use the piece of knowledge.
- (U) The subject used the piece of knowledge without hesitation or other signs of unusual processing.
- (L) The subject seemed to learn the knowledge. Episodes received this code if the subjects expressed puzzlement or commented on their lack of knowledge, but eventually came up with the right action (e.g., writing a negative sign). For instance, subject P2 overlooked the first minus

Table 2: Places where the negative-projection rule could be used

| Subj. | Examples | | | | | Problems | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| P2 | | L | | | U | ? | ? | U | U | U | U | U | U | U | U | U | U | U | U | U | U |
| S101 | | | | | | | | L | L | | | O | U | | | U | | | | | U |
| P1 | L | L | U | O | U | U | | U | U | | U | U | | U | U | | U | | U | U | |
| S110 | | | | | | | | O | | | | | U | O | O | U | O | | | U | R |
| S102 | | L | | | | U | | | | | | | U | U | O | | U | | | | |
| S109 | | | | | | | | | | | | | | | | | | | | | |
| S105 | U | | | | | | | U | | | U | | U | | U | | U | | | | E |
| S103 | | | | | | | | | E | | | | | O | E | | | | | | |
| S107 | U | | | | | | | E | R | | | U | | U | U | O | U | | | | |

sign in the first example, but on the second minus sign she said, "Hmm, why is [it] minus? Uh Huh.... Because these axis are starting here so this is minus." She then went back to the first minus sign and said, "How about the X's. It should also be a minus. Yah, that was a minus." Subject S102 paused after seeing the second minus and said, "Negative W.... It's because it's going in a negative direction it points...they give it a negative value [if] it's below the Y-axis. I mean the X-axis." Subject P1 (quoted at length in VanLehn, Jones & Chi 1991), took several garden paths before discovering the correct rule for explaining the minus sign, which is clear evidence of her lack of knowledge. However, her verbal behavior at the time of the discovery was just as brief and cryptic as the verbal behavior of P2 and S102. Such limited verbal evidence is typical of discovery events in protocol data (VanLehn, 1991; Siegler & Jenkins, 1989). They nonetheless seem to reliably mark transitions in the subjects' knowledge.

- (R) The subjects' verbal behavior indicates that they are learning the piece of knowledge, but they used it at least one before. We believe these are cases of relearning.
- (?) Protocol missing.

The blanks in the problem solving part of Table 2 shows that for this piece of knowledge, many gaps are not detected because the student used analogical problem solving. When we constructed similar analyses for all 9 gaps and all 9 subjects, we found that of the 81 (= 9x9) cases where a piece of knowledge could be learned, in 44 cases (54%) the subject avoided *all* places where the gap could be detected (as did S109 in Table 2). This analysis clearly indicates that analogical problem solving is thwarting gap filling by avoiding lines of reasoning that would cause the gap to be detected.

This finding makes intuitive sense. Most problems can be solved by a 4 step process: select some objects as the "bodies" (line 1 of Figure 1), draw a diagram for each body showing the forces acting on it (lines 2 and 3 of Figure 1), produce a set of equations (line 6 of Figure 1), then solve the equations for the sought quantity (omitted in Figure 1). The last step cannot usually be replaced by analogy because the problems seldom seek the same quantities. However, the first 3 steps can often be achieved by analogy. The student can find an analogous problem and copy either its force diagram, its equations or both. A student who does this avoids using the force laws (which generate forces) and Newton's laws (which generate the equations). Missing physics knowledge can remain undetected as long as one uses analogy to copy force diagrams and equations. To put it bluntly, analogical problem solving often preserves ignorance.

We can now understand part of Chi et al.'s finding about the use of analogy by Good and Poor solvers. The Poor solvers displayed more episodes of analogical problem solving and read more lines during each episode because they generally avoided generating their own forces and equations by copying them from the examples. This is not just a coincidence, but seems to have caused them to learn much less than they would otherwise. On the other hand, the Good solvers generally tried to generate their own forces and equations and only referred sporadically and briefly to the examples. Thus, they could still detect their gaps and remedy them.

## 4 USING ANALOGY SPARINGLY

We suspect that the Good solvers' use of analogy does more than just allow gap-filling to operate. It may actively aid gap-filling by helping to both detect and fill gaps. This section presents a few pieces of protocol data to support our conjecture. However, more data are clearly required.

There were 6 cases in the protocols where, according to the analysis above, knowledge was learned during problem solving. During 4 of the 6 episodes, the subject clearly referred to an example. These 4 cases, discussed below, are different enough that they begin to show that analogy can assist gap filling in several ways.

Subject P2 reached an impasse and filled it with the aid of analogy. The subject needed to know whether water exerted a force on a block floating on it. She referred to an example where a string held up a block instead of water. The close similarity of the problems convinced her that water can exert a force. This conjecture both filled a gap in the subject's knowledge and resolved a problem solving impasse. ACT* (Anderson, 1990) and other theories claim that analogy is often used to resolve impasses and thereby acquire new knowledge.

Subject S105 used analogy to detect a knowledge gap. While drawing forces for an inclined plane problem, the subject omitted one force (the surface normal—a notoriously unintuitive force). He had never produced that force in earlier problem solving, nor had he self-explained the line in the examples that was intended to teach it. Thus, we assume he had a knowledge gap. After finishing his force diagram, the subject fetched an example, viewed its force diagram, said, "That's the force I wasn't thinking of," and drew a normal force on his diagram. Thereafter, the subject regularly drew normal forces on his diagrams. In this case, analogy was used to *check* a step in the problem solving, and that revealed a knowledge gap. Whereas the preceding case illustrates how analogy can help in *filling* gaps, this case illustrates how analogy can help in *detecting* gaps. There was a second case of analogy-based checking causing detection of a gap, but it will not be presented here. These cases are consistent with a finding of Chi et al. (1989), who classified analogical episodes as either reading, checking or copying. Good solvers had many fewer episodes of reading and copying than Poor solvers, but they actually had more episodes of checking.

Subject 101 learned a rule while engaged in a mixture of analogical problem solving and self-explanation. While trying to project forces onto axes, the subject decided to see what a similar example had done. He could partially explain the example's equations, but he could not self-explain the negative signs. He gave up, copied the negative signs and said, "Well let's see if I can just push these in.... This is called copying too much from the book. I hate that." This is clearly a case of analogical problem solving of the worst kind. However, after solving the equations and producing a negative formula, he said, "So, according to this, my x-component is equal to $-9$, which means, okay. That makes sense. That makes sense. One of these has to always be negative, doesn't it?" Apparently, the subject figured out why there is a negative sign. His reasoning appears to be based on some kind of symmetry argument, which is an overly general line of mathematical reasoning, but not the best one for learning this rule. A bit later in the protocol, he failed to use his new knowledge, but corrected his oversight a few lines later (we coded this as a learning event in Table 2). After that, he used the rule fairly consistently. This case illustrates how analogy can combine with self-explanation to produce learning via a kind of justified analogy (Kedar-Cabelli, 1985).

We believe that these cases are just a few of the many ways that analogy can support learning. Note that all these uses of analogy only briefly interrupt regular problem solving in order to achieve a specific meta-goal, such as detecting a gap or filling one. Wholesale analogy, the kind used by the Poor solvers, avoids detecting gaps and thus tends to retard learning. Thus, we hypothesize that learning is most effective when analogy is used sparsely, as an augmentation to regular problem solving rather than a replacement.

## 5 DISCUSSION

The preceding sections showed that in one study, effective human learners used analogical problem solving sparingly. With a little bit of computational common sense, we can generalize this result and formulate a heuristic for when such sparse analogical problem solving should be effective.

There are four steps to solving a Newtonian mechanics physics problem:

1. *Define a system.* One must (a) select an idealization of the physical world consisting of idealized bodies that have idealized relationships to other objects and move in idealized trajectories, and (b) decide whether to base the analysis on forces, energies or momenta. In Figure 1, line 1 corresponds to part *a* (albeit, tersely), and part *b* is missing because the text has only introduced one type of analysis (forces) at this point.
2. *Explicate physics quantities.* For each body in the system, one notes the forces, energies or momenta associated with that body. Often, a diagram is drawn to help one remember them. In Figure 1, this occurs during line 2.
3. *Generate equations.* The equations are instances of general principles. In Figure 1, the equations are produced on line 6. They are instances of Newton's first law.
4. *Solve the equations for the sought quantity.*

Although logically distinct, these steps are often intermingled in an solver's work. Solvers have no trouble learning this basic procedure. It is often printed in the textbook. It is a specialization of the general 3-step procedure (define a system, formulate a mathematical model, solve it) that is used for all mathematical analysis problems, from lowly arithmetic and algebraic word problems to esoteric branches of science and engineering (see any textbook on systems theory, e.g., Shearer, Murphy & Richardson, 1971).

The system definition step is quite different from the others. There are no real principles for defining a system. Sometimes a chain is treated as a single body, sometimes as an infinite sequence of infinitely small bodies, and sometimes as two bodies. Textbooks give few heuristics for defining systems. Although the problems used in our data are too simple to reveal how the subjects learn about system definition, it is plausible that system-defining can be learned by analogical problem solving and/or and case-based reasoning.

All the steps except system definition are governed by well-known principles, such as the force laws (for step 2), Newton's laws (for step 3) and mathematical transformations (for step 4). Moreover, once the system has been defined, the analysis is completely determined. In step 2 (explication of physics quantities), one produces all the forces (or energies or momenta) acting on the system's bodies. In step 3 (equation generation), one produces all the equations implied by those physics quantities.[2] In step 4, the equations are solved mechanically. The point here is that in physics and many other mathematical analysis task domains, the most important decisions are made during system definition, and the rest of the analysis follows more or less deterministically from those choices. Although analogical problem solving would be useful for learning search control, search control is not very important for steps 2 and 3.

Although principles can be used for steps 2 and 3, it may be that case-based reasoning is more efficient than principle-based reasoning. If so, analogical problem solving would be an effective way to master those steps. The output of step 2 is completely determined by the output of step 1, and case-based reasoning appears to be the most effective way to define systems (step 1), so case-based reasoning should be just as effective for delineating forces and other quantities (step 2). However, generation of equations (step 3) depends only only on the earlier steps, but on the exactly which quantities are sought and which are given. Even when problems have very similar systems, they rarely have similar givens and soughts, so case-based reasoning should seldom be helpful generating equations. In addition, it is much easier to remember the diagrams produced by steps 1 and 2 than the systems of equations produced by step 3.

It is worth recalling that our top-level goal is to determine when analogical problem solving is advisable for effective learning. So far, we have argued that case-based reasoning and/or analogy should be used during the system defining step and possibly during the quantity delineation step, whereas principle-based reasoning should be used instead of analogy during the equation generation step (we ignore the equation solving step, since it involves skills acquired prior to studying the task domain). We next consider how principles can be learned during steps 2 and 3.

Suppose the learner already knows many principles, so the learning problem is to detect a gap (missing principle) and fill it, rather than to learn a whole batch of principles at once. In order to detect a gap, one must try to use principles instead of analogies to achieve goals, for otherwise the body of knowledge containing the gap will not be referenced and detecting the gap would be impossible.

Although impasses in principle-based problem solving will uncover some gaps, not all gaps cause impasses. Thus, it is good to check intermediate results because early detection of an error will facilitate locating the gap that caused it. Analogy is one way to check intermediate results.

By considering the nature of the task domain, one can predict which intermediate results should be checked. In physics, missing knowledge of physics quantities will cause step 2 to produce too few forces, energies, etc., but this will not causes impasses until much later, if at all. Consequently, it is good to use analogical problem solving to check the results of step 2 before moving on to step 3. This is just what subject S105 did in the case mentioned earlier.[3] In general, when the goal is "generate all X that you can think of," where X in S105's case is "forces," then gaps will not cause impasses, so it is wise to use analogy to check the outcome of generate-all-X goals. Experienced learners of mathematical analysis may know this heuristic for increasing their learning rate. They may know other heuristics as well.

Once a gap is detected, analogy is certainly one possible way to fill it, but it is not easy to predict whether one should use analogy or some other technique, such as instantiating an overly general rule (VanLehn, Jones & Chi, 1991) or explanation pattern (Schank, 1986). A heuristic for this decision would be hard to formulate.

We have arrived finally at our goal, which are heuristics for deciding when to use analogical problem solving. The heuristics are:

1. If the task domain, or some part of the task domain (e.g., steps 2 and 3), has principles, and they are more effective knowledge than cases, and they require little search control, then the target knowledge should be principles.

[2] Actually, there are choices to make during step 3 regarding how to rotate the coordinate axes or whether to omit certain equations. These choices affect the difficulty of step 4, but not the ultimate outcome.

[3] In order to fill such gaps, Cascade 3 used analogy essentially as a check of step 2, although the implementation was rather baroque (VanLehn & Jones, 1991; in press-a).

2. If the target knowledge is principles, they should be acquired by gap filling, which implies:
   (a) Gap detection: Try to use principles instead of analogies, as a gap may show up as an impasse. Use analogy to check the intermediate results derived via principles, as this may uncover gaps that do not produce impasses.
   (b) Gap filling: Analogical problem solving is one possible technique for filling gaps. Others should be considered as well.
3. If the target knowledge is not principles but cases or search control, then analogical problem solving may be useful.

This conclusion was suggested by human data. There is evidence that the Poor solvers use analogy "wholesale," as a replacement for principle-based solutions to steps 1, 2 and 3. There is also evidence that Good solvers avoid analogical problem solving in general. There is some evidence, albeit only a few protocol excerpts, that when Good solvers do use analogy, it is used as an aid to gap filling. In this last section, we have reflected on the human data and, supported by common sense, derived a heuristic for when analogical problem solving should aid learning. This heuristic could have several uses. (1) It helps explains why the Good solvers learned more than the Poor solvers. (2) It is a prescription for effective learning that could perhaps be taught to human students. (3) It could be embedded in a multi-strategy learning.

Clearly, all these implications are testable in their own fashions. A good next step would be to augment Cascade so that we can experimentally test whether the heuristic does increase learning. If this succeeds, one could try teaching human students the heuristic and see if their learning increases as predicted.

## Acknowledgements

This research was supported by grant N00014-88-K-0086 from the Cognitive Sciences Division of the Office of Naval Research. We appreciate Joel Martin's and Ted Rees' critiques of the manuscript.

## References

Anderson, J.R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Lawrence Erlbaum Assoc.

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, 145-182.

Chi, M.T. H. & VanLehn, K. (1991) The content of physics self-explanations. *Journal of the Learning Sciences*, 1(1), 69-106.

Halliday, D. & Resnick, R. (1981). *Fundamental of physics, Second edition.* New York: Wiley.

Kedar-Cabelli, S. (1985). Toward a computational model of purpose-directed analogy. In A. Prieditis (Ed.) *Analogica.* San Mateo, CA: Morgan Kaufman. Reprinted in J. W. Shavlik & T. G. Dietterich (Eds.) *Readings in machine learning.* San Mateo, CA: Morgan Kaufman.

Schank, R.C. (1986). *Explanation patterns: Understanding mechanically and creatively.* Hillsdale, NJ: Lawrence Erlbaum Assoc.

Shearer, J.L., Murphy, A.T. & Richardson, H.H. (1971). *Introduction to systems dynamics.* Reading, MA: Addison-Wesley.

Siegler, R.S. & Jenkins, E. (1989). *How children discover new strategies.* Hillsdale, NJ: Lawrence Erlbaum Assoc.

VanLehn, K. (1990) *Mind bugs: The origins of procedural misconceptions.* Cambridge, MA: MIT Press.

VanLehn, K. (1991) Rule acquisition events in the discovery of problem solving strategies. *Cognitive Science, 15*, 1-47.

VanLehn, K., Jones, R.M. & Chi, M.T.H. (1991). A model of the self-explanation effect. *Journal of the Learning Sciences, 1*, 69-106.

VanLehn, K. & Jones, R.M. (1991). Learning physics via explanation-based learning of correctness and analogical search control. In L. Birnbaum & G. Collins, (Eds.), *Machine Learning: Proceedings of the eighth international workshop.* San Mateo, CA: Morgan Kaufman.

VanLehn, K. & Jones, R.M. (1993). Learning by explaining examples to oneself: A computational model. In S. Chipman & A. L. Meyrowitz (Eds.) *Foundations of knowledge acquisition: Cognitive models of complex learning.* Boston: Kluwer Academic Publishing.

VanLehn, K. & Jones, R.M. (in press-a). Integration of analogical search control and explanation-based learning of correctness. In S. Minton & P. Langley (Eds.) *Machine learning methods for planning and scheduling.* Los Altos, CA: Morgan Kaufman.

VanLehn, K. & Jones, R.M. (in press-b). What mediates the self-explanation effect? Knowledge gaps, schemas or analogies? To appear in M. Polson (Ed.) *Proceedings of the 15th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

VanLehn, K. & Jones, R.M. (in preparation). Self-explanation and analogy.