Toward a Dynamic Model of Early Algebra Acquisition[1]

Benjamin A. MacLaren, Kenneth R. Koedinger

Human-Computer Interaction Institute
School of Computer Science
Carnegie Mellon University, Pittsburgh, PA 15213
E-Mail: maclaren@cmu.edu

Abstract: How does one go about creating quality cognitive models that capture the difficulties students have in learning complex skills? The answer we propose is to break a domain down into a number of dimensions (or difficulty factors) and use cognitive modeling and empirical work to better understand them. We demonstrate how this can be done for the domain of algebra. This type of analysis can not only lead to a better understanding of the domain for traditional instruction, but it can also serve as the foundation for the development of computer tutors.

Introduction

We are developing a cognitive model of quantitative problem solving skill. Our focus is particularly on skills near the transition between arithmetic and algebra, a domain math educators are now referring to as "early algebra." Our goals in characterizing these skills are (1) to provide guidance in principled design of instruction to help students acquire critically important algebraic reasoning skills, and (2) to set the stage for the creation of a developmental model of algebra learning.

Cognitive analysis is important to determine which beliefs behind the structure of today's instruction are true and which aren't. Nathan, Koedinger, and Tabachneck (1996) have found evidence that certain beliefs of math educators and teachers are inconsistent with the reality of student problem solving. It is a commonly held belief that mathematical story problems are more difficult than problems presented as equations. This belief was exhibited by math educators and teachers, when they were asked to rank the difficulty of problems like those in Figure 1. The verbal problems (in the first two rows) were consistently ranked as more difficult than the corresponding symbolic problems (row three). In contrast, we found that even after a high school algebra course, students were better able to solve verbal problems than the analogous symbolic problems.

Prior models of algebra story problem solving (e.g., Bobrow 1968; Mayer 1982; Lewis 1981) have assumed a two-step process. Story problems are converted into equations and the equations are then solved using symbolic algebra. Such a model predicts that performance on story problems must be worse than performance on equations (since equation solving is a subgoal of story problem solving), in contrast to the behavior of students in studies we have performed (Koedinger and Tabachneck, 1995). One goal of this paper is to provide a better cognitive model of early algebra problem solving that (1) characterizes these alternative strategies, (2) provides a possible explanation for students' surprising relative success on story problems, and (3) more generally captures the essential knowledge differences between good and poor early algebra students. First, we will review the empirical results on student problem solving, and then present a cognitive model that accounts for the observed student behavior.

Difficulty Factor Assessments of Student Problem Solving

To gain a better understanding about student problem solving, we developed several "Difficulty Factor Assessments" (DFAs). The goal of a Difficulty Factor Assessment is to provide for the systematic comparison of factors that may contribute to problem solving difficulty. Two such factors in early algebra are illustrated in Figure 1, unknown position and presentation type. The pair of problems in each row of Figure 1 differ in where the problem unknown is positioned. The problems in column 1 are called Result Unknown Problems because the unknown is the result of the process described. The problems in column 2 are Start Unknown Problems because the unknown is the start of the process described. Problems in the columns illustrate a second factor. They require the same underlying arithmetic, but differ in the representation in which they are presented. The "Story Problems" in the first row are presented verbally and include reference to a real world situation (e.g., wages). The "Word Equations" in the second row are also presented verbally but do not include a situation. The "Equations" in the third row are presented symbolically and have no situational information. Other factors we have looked at that are not illustrated in Figure 1 include number difficulty (integers versus non-integers) and cover story for story problems (e.g., the "waiter story" below, or purchasing a basketball).

[1] Paper accepted to EuroAIED, September 1996, Lisbon, Portugal.

Results from DFA studies

DFA studies 1 and 2 revealed large effects for unknown position, problem presentation and number difficulty (integers vs. decimals). Differences for unknown position and problem presentation are shown in Figure 2. Students were significantly better at solving word equations than solving equations in both studies ($p<.001$). In the first study, there was no significant difference in the students' ability to solve story problems and word equations ($p>.23$); in the second study, there was a significant difference ($p<.01$), but, as Figure 2 shows, the distance between the success rate on story problems and word equations was perceivably smaller than the distance between the success rate of word equations and equations.

□	**Result Unknown Problems**	**Start Unknown Problems**
Story Problems	When Ted got home from his waiter job, he multiplied his hourly wage, $2.65, by the 6 hours he worked that day and added the $66 he received in tips. How much money did Ted make that day?	When Ted got home from his waiter job, he took the amount he made that day and subtracted the $66 he made in tips. He divided the resulting amount by the six hours he worked and got $2.65, his hourly wage. How much did Ted make that day?
Word Equations	If I multiply 2.65 by 6 and then add 66, I get a number. What number do I get?	Starting with some number, if I subtract 66 and then divide by 6, I get 2.65. What number did I start with?
Equations	$2.65 * 6 + 66 = X$	$(X – 66) / 6 = 2.65$

Figure 1: Examples Combinations of Difficulty Factors

Students relative success at verbal problems was due in part to their use of strategies other than formal algebra (students used formal algebra about 12% of the time on the start unknown problems in DFA1). The other strategies that they used included two informal strategies we called "guess-and-test" and "unwind." In guess-and-test, a value for the unknown is guessed at and that value is propagated through the known constraints. The guess is then adjusted and the process repeated until the correct answer is arrived at. Guess and test was used more than 22% of the time on the start unknown problems. By far the most common strategy, the informal "unwind" strategy, was used almost 40% of the time. Unwind is a verbally mediated strategy, where students work backwards from the given result value, inverting operators along the way, to produce the unknown start value.

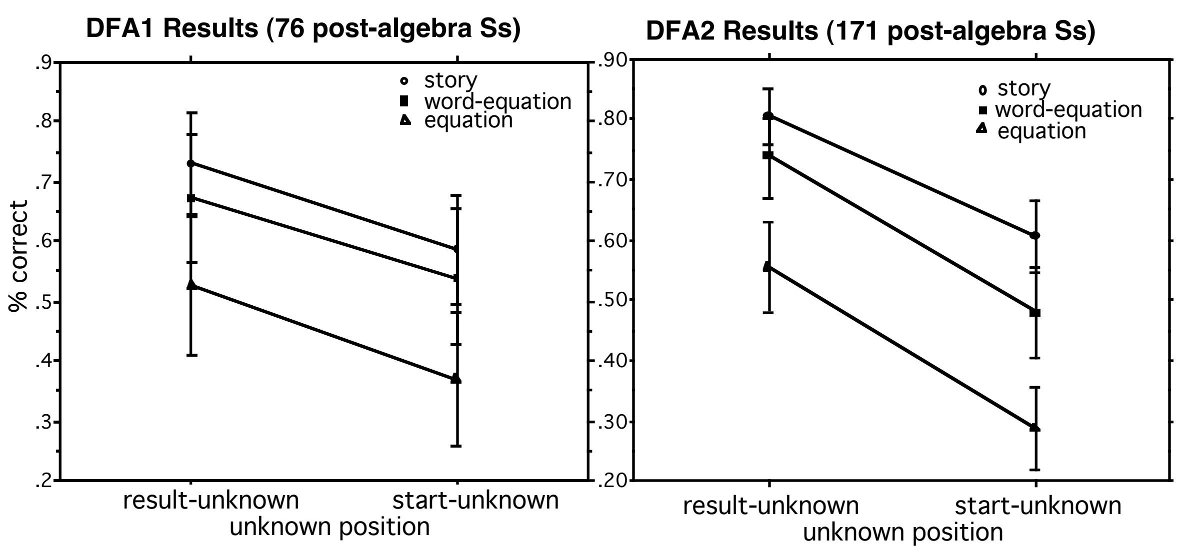


Figure 2: Summary of DFA Results

Figure 3 provides an example of a student using the unwind strategy on a start-unknown or "algebra" problem. Instead of writing down and manipulating an equation (e.g., $(X-66)/6=2.65$), the student works backwards through the problem statement, first inverting the final division and multiplying 2.65 by 6, then inverting the initial subtraction and adding the intermediate result 15.90 to $66. Surprisingly, students in DFA1 did better on Story Algebra problems than they did on analogous Symbolic Arithmetic problems. Students had difficulty interpreting symbolic problems, often giving up on them, and when they did attempt the they often made manipulation errors not made in the verbal problems.

When Ted got home from his waiter job, he took the money he earned that day and subtracted the $66 he received in tips. Then he divided the remaining money by the 6 hours he worked and found his hourly wage to be $2.65.
How much did Ted earn that day?

```
 3 3            1
 2.65          66.00
 x   6       - 15.90        The answer is $81.90
 ____        _______
 15.90        81.90
```

Figure 3: A Student Solution for a Hard Story Algebra Problem (49% correct on DFA1)

Figure 4 illustrates a student's solution to a symbolic result unknown or "arithmetic" problem. Here we see a common error in the second decimal arithmetic operation, whereby the student miss-aligns the decimal point. This error was not observed in the Story problems where apparently the situational support of understanding the difference between dollars and cents helped students to avoid such an error. However, as we'll see later this error is just one of the reasons that students at this level do better on verbal problems – it doesn't account, for instance, for their better performance on the situation-less word equations over the symbolic equations.

Solve for X: 2.65 * 6 + 66 = X

```
 3 3          15.90
 2.65          + 66         The answer is 16.56
 x   6        _____
 ____         16.56
 15.90
```

Figure 4: A Student Solution for a Hard Arithmetic Equation (33% correct on DFA1)

An ACT-R Cognitive Model

We developed a model in ACT-R (Anderson, 1993) of how each of the most common strategies observed functions to successfully solve a problem, and how common errors arise for different combinations of a subset of the difficulty factors we have looked at. The model is capable of solving more complex problems than the ones illustrated in Figure 1 (see Tabachneck, Koedinger & Nathan 94), but here we will focus on the aspects that relate directly to the DFA1 data. We will first give a broad description of the model, two example traces of the model solving problems from DFA1, and then describe how we tuned the model to fit the DFA1 data.

The model proposes that students begin by searching for a relevant strategy. If none is found, they give up. After selecting a strategy, the model enters into a cycle of extracting arguments and an operator, determining what to do (e.g., should it invert the operator) and performing the necessary arithmetic until the problem has been solved. The full model has upwards of 150 productions in it, but the subset we focus on here has 30 productions (6 control, 1 giveup, 6 argument extraction, 4 arithmetic, 5 operator interpretation/inversion, 8 translation/update/exit productions).

We model two types of errors: arithmetic and conceptual. Conceptual errors include things like forgetting to change the sign when removing an operator in the verbal representation or confusing the order of operations in the symbolic representation. There are also productions to give up during a solution, resulting in a conceptual error. For arithmetic errors, we model bugs (miss-alignment of decimal places in doing arithmetic) and slips (e.g., 2 * 3 = 5). Bugs and slips are each modeled by a single production (abstracting over detailed arithmetic errors, such as carry errors and borrowing from zero). As stated before, the model can also give up on a problem, leaving no trace.

Now let us consider how to model the two problems shown in Figures 3 and 4. The trace in Figure 5 illustrates how we can walk the model through a solution by hand to match a particular student solution like that shown in Figure 3 (below we describe how we parameterized ACT-R's conflict resolution mechanism so the model makes these choices). The student's use of the verbal unwind strategy is modeled in cycle 1. In cycles 2 and 3, the model interprets "divided the remaining money by 6 and got 2.65" and inverts the operator. The arithmetic procedure that results is then performed (cycle 4). The remaining cycles repeat these steps, working backward through the initial subtraction operation in the problem.

```
    --- Options ---
    1. I could try verbal unwinding.
    2. I could try algebra.
    3. I'll just give up the problem.
    What should I do? 1
>>> Cycle 1: CONTROL*SELECT*UNWIND
  Let's try unwinding...
>>> Cycle 2: VERBAL*ALG*EXTRACT-ARGUMENTS-4
   DIVIDED-BY 6 gives 2.65...
>>> Cycle 3: VERBAL*INVERT-OP
   So 2.65 * 6...
    --- Options ---
    1. 2.65  *  6 = 15.9          (correct with situational support)
    2. 2.65  *  6 = 159           (bug)
    3. 2.65  *  6 = 0.16          (slip)
    What should I do? 1
>>> Cycle 4: ARITH*ARITH-PROCEDURE*SITUATIONAL-ASSIST
   2.65  *  6 is 15.9
>>> Cycle 5: VERBAL*ALG*EXTRACT-ARGUMENTS-4
   MINUS 66 gives 15.9...
>>> Cycle 6: VERBAL*INVERT-OP
   So 15.9 + 66...
    --- Options ---
    1. 15.9  +  66 = 81.9         (correct with situational support)
    2. 15.9  +  66 = 16.56        (bug)
    3. 15.9  +  66 = 67.59        (slip)
    What should I do? 1
>>> Cycle 7: ARITH*ARITH-PROCEDURE*SITUATIONAL-ASSIST
   15.9  +  66 is 81.9
>>> Cycle 8: VERBAL*DONE-TRANS*KNOW-EVERYTHING-DONE
```

Figure 5: A Model Trace for a Hard Story Algebra Problem

Figure 6 shows part of a trace of the solution in Figure 4. The productions for strategy selection, argument extraction etc. achieve the same goals as those applied to the Story problem (Figure 5), but their if-parts access a different representation, symbolic rather than verbal. It is this difference that captures a key difficulty students have with symbolic problems. Reflecting students' lesser experience with symbols than words, the productions that interpret symbolic forms are weaker than those that interpret verbal forms. We assume the productions for arithmetic are largely the same for verbal and symbolic problems. The only difference is the ARITH*ARITH-PROCEDURE*SITUATIONAL-ASSIST production (used in cycle 7 of Figure 5) that models the use of "situational semantics" (e.g., of money) to aid the decimal alignment. The use of this rule is the only difference between the model's approach to story problems and word equations.

```
   --- Options ---
    1. I will perform the arithmetic in (2.65 * 6 + 66 = X).
    2. I'll just give up the problem.
    What should I do? 1
>>> Cycle 0.0: CONTROL*SELECT*ARITH-EQ
    --- Options ---
    1. Simplify 2.65 * 6        in (2.65 * 6 + 66 = X).
    2. Simplify 6 + 66          in (2.65 * 6 + 66 = X).        (order-of-ops bug)
    What should I do? 1
. . .
>>> Cycle 5.0: SYMBOLIC*SIMP*EXTRACT-ARGUMENTS-1
   To simplify 15.9 + 66 on the left side.
>>> Cycle 6.0: SYMBOLIC*ALG*DONT-INVERT
   So 15.9 + 66
    --- Options ---
    1. 15.9  +  66 = 81.9         (correct)
    2. 15.9  +  66 = 16.56        (bug)
    3. 15.9  +  66 = 225          (slip)
    What should I do? 2
>>> Cycle 7.0: ARITH*ARITH-PROCEDURE*BUG
   15.9  +  66 is 16.56           (bug)
>>> Cycle 8.0: SYMBOLIC*DONE-TRANS*KNOW-EVERYTHING-DONE
```

Figure 6: A Model Trace for a Hard Arithmetic Equation

Parameter setting in the ACT-R model

After developing a knowledge-level model that could be guided through the space of decisions, we set conflict resolution parameters to stochastically select productions consistent with the "average student" from DFA1 data. Part of ACT-R includes a "rational" control mechanism based on decision theory, which uses parameters such as the likelihood that executing a production will eventually satisfy the current goal and the cost of executing a production. Also, ACT-R predicts that gaussian noise will sometimes cause a production to be selected other than the one with the highest estimated utility. These features enabled us to model the average student, by setting the noise and parameters to make errors with the same frequency as the group did.

To get the data for these estimations, we broke the problems from DFA1 down according to three of the difficulty factors: position of the unknown (arithmetic/result-unknown or algebra/start-unknown), representation (story problems, word equations, and equations), and the difficulty of the numbers (integer or non-integer). This led to 12 groups of problems as shown in the left-most columns of Tables 2 and 3.

In setting the parameters, we started with the simplest group (result-unknown integer verbal arithmetic) that contains "core-productions" that are common to every group, then working towards tuning parameters for productions that address difficulties introduced along each of the three dimensions. For instance, moving to non-integers involved tuning the parameters for buggy arithmetic, moving to start-unknown involved tuning parameters for unwind productions, and moving to symbolic problems involved parameters to interpreting symbols.

Table 1 shows the central productions in the model that we tuned (in the left-most column) and for each problem type (along the top) it shows what productions apply for that type. For example, for easy story arithmetic (Arth Easy Stry) there are two strategy selection productions, Select*Verbal-Arithmetic and GiveUp-Problem. For argument extraction, there is only the correct Verbal*Extract-Arguments and the buggy Verbal*Incomplete. Since no operator inversion is required for arithmetic, only the arithmetic productions apply, and because the arithmetic is easy, only the correct production Arith-Procedure and the simple Arith-Proc*Slip applies. In contrast to the simplest problem type, hard algebra equations on the far right have several more productions that apply.

Table 1: Summary of Parameter Setting Strategy and Results

Productions	R	Cost A+B	Arth Easy Stry	Arth Easy Wrd	Arth Easy Eq	Arth Hrd Stry	Arth Hrd Wrd	Arth Hrd Eq	Alg Easy Stry	Alg Easy Wrd	Alg Easy Eq	Alg Hrd Stry	Alg Hrd Wrd	Alg Hrd Eq
<u>Strategy Selection</u>														
GiveUp-Problem	.01	0.0	X	X	X	X	X	X	X	X	X	X	X	X
Select*Vrb*Arith	.91	**16.4**	XX	XX	-	X	X	-	-	-	-	-	-	-
Select*Sym*Arith-Eq	.51	**9.4**	-	-	XX	-	-	XX	-	-	-	-	-	-
Select*Vrb*Unwind	.67	**12.3**	-	-	-	-	-	-	XX	XX	-	X	X	-
Select*Vrb*Alg	.62	**12.4**	-	-	-	-	-	-	XX	XX	-	X	X	-
Select*Sym*Unw-Eq	.56	**10.8**	-	-	-	-	-	-	-	-	XX	-	-	X
Select*Sym*Alg-Eq	.50	**10.0**	-	-	-	-	-	-	-	-	XX	-	-	X
<u>Argument Extraction</u>														
Sym*Extract-Args	**.25**	4.0	-	-	X	-	-	X	-	-	XX	-	-	X
Sym*Order-of-ops-bug	.01	0.0	-	-	-	-	-	-	-	-	X	-	-	X
Vrb*Extrct-Args	**.30**	4.0	XX	XX	-	XX	XX	-	XX	XX	-	XX	XX	-
Vrb/Sym*Incomplete	**.05**	0.0	X	X	XX	X	X	XX	X	X	XX	X	X	XX
<u>Operator Interp/Inv</u>														
Vrb*Invert-Op	**.32**	4.0	-	-	-	-	-	-	XX	XX	X	X	X	X
Vrb*Unwind-Error	.01	0.0	-	-	-	-	-	-	X	X	X	X	X	X
<u>Arithmetic</u>														
Arith-Procedure	**.81**	4.0	XX	XX	XX	X	X	X	X	X	X	X	X	X
Arith-Proc*Sit-Assist	.81	**0.9**	-		-	XX	-	-	-	-	-	X	-	-
Arith-Proc*Slip	.63	3.4	X	X	X	X	X	X	X	X	X	X	X	X
Arith-Proc*Bug	.63	**2.1**	-	-	-	XX	XX	X	-	-	-	X	X	X

Table 1 also shows for each production we tuned, what group of problems we tuned it for (XX) and what group it also applies to (X). Finally, it also shows the resulting parameters: the estimated probability for success if that production fires (R) and the sum of the production cost and estimated cost-to-goal after firing that production, A+B (measured in seconds). The values shown in bold were the actual parameters we tuned.

Model-Data Fit

The results of our parameter tuning can be seen in Tables 2 and 3 below. The comparison is presented as sets of triples: first the model, then the DFA1 data, and then the difference. Table 2 shows the results for arithmetic (result unknown) problems. Table 3 shows the results for algebra (start unknown) problems broken down into formal and informal strategies.

Table 2: Result Unknown Problems (Arithmetic): Model vs. DFA data

Representation	Correct Model	Correct DFA	Correct diff	Arithmetic Errors Model	Arithmetic Errors DFA	Arithmetic Errors diff	Conceptual Errors Model	Conceptual Errors DFA	Conceptual Errors diff	No Answer Model	No Answer DFA	No Answer diff
Easy Story	80	77	3	4	1	3	12	17	-5	5	5	0
Easy Word	79	84	-5	7	5	2	10	5	5	5	7	-2
Easy Equation	65	65	0	3	7	-4	7	12	-5	27	16	11
Hard Story	69	63	6	13	17	-4	12	11	1	7	9	-2
Hard Word	49	42	7	33	36	-3	12	21	-9	7	0	7
Hard Equation	35	33	2	28	24	4	9	9	0	29	33	-4

Table 3: Start Unknown Problems (Algebra): Model vs. DFA data

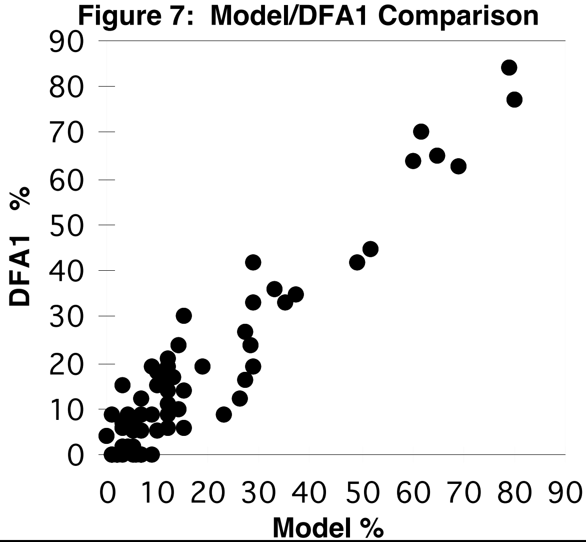
	Informal Strategy									Formal Strategy									No Answer		
	Correct			Arith Error			Conc Error			Correct			Arith Error			Conc Error			(Giveup)		
Rep.	Mod	DFA	diff	Mod	DFA	diff	Mod	DFA	diff	Mod	DFA	diff	Mod	DFA	diff	Mod	DFA	diff	Mod	DFA	diff
Easy St	60	64	-4	5	2	3	12	14	-2	6	6	0	1	0	1	3	2	1	15	14	1
EsyWd	62	70	-8	5	0	5	12	19	-7	6	0	6	1	0	1	4	2	2	12	9	3
EasyEq	37	35	2	2	0	2	9	19	-10	19	19	0	3	0	3	4	9	-5	29	19	10
HardSt	52	45	7	14	10	4	14	24	-10	0	4	-4	9	0	9	3	1	2	10	15	-5
HrdWd	27	27	0	23	9	14	15	30	-15	3	6	-3	2	0	2	1	9	-8	12	18	-6
HardEq	26	12	14	15	6	9	6	18	-12	12	6	6	7	0	7	3	15	-12	29	42	-13

Currently, the model does a good job of capturing the main effects of the three difficulty factors on solution correctness. It also does a reasonable job with arithmetic and conceptual errors. The difference under the correct columns in Tables 2 and 3 are small (less than 8%) for the most part. Similarly, the productions for the error categories are quite close. In general, the complexity inherent in the 66 data points in Tables 2 and 3 is well captured by the model through the setting of only 13 free parameters (shown in bold in Table 1).

As much can be learned, however, from the weaknesses of the current model. Looking at the correctness columns, the biggest deviation is on hard number algebra equations where the model is more often using both the informal strategy (26% vs. 12%) and the formal strategy (12% vs. 6%). Part of this overprediction of success is caused by the model's relative lack of conceptual errors on algebra problems, perhaps, because we have implemented too few buggy conceptual rules. Another problem is that students appear much more likely to give-up on hard algebra equations than the model does (42% vs. 29%). The current model does not consider number difficulty in picking the initial strategy (the failure of which results in giving up). However, it appears that subjects may be anticipating downstream arithmetic difficulties and thus are giving up earlier.

In order to provide a better sense of the overall goodness-of-fit for the model, Figure 7 shows a scatter plot of the 66 data points, where each point is a data category from Table 2 and Table 3 (e.g., arithmetic errors on easy algebra story problems). The percent occurrence predicted by the model is plotted against the actual percent occurrence in DFA1.

Figure 7: Model/DFA1 Comparison



Conclusions

The importance of the DFAs and cognitive modeling work is in how they help to generate and refine precise hypotheses about student thinking. DFAs provide an empirical basis for knowledge acquisition – indicating distinctions between good and poor problem solvers that should be captured in learner models. Building models within a unified theory like ACT-R helps to connect results across many domains in an integrated and coherent way. We have used ACT-R to shed light on students' difficulties in early algebra. Empirical studies have shown that symbolic algebra is much more like a foreign language for beginning students than many educators and teachers suspect (Nathan, et. al, 1996; Koedinger & Tabachneck, 1995). The model presented here accounts for students' better performance on verbal problems both through the greater use of more familiar alternative strategies and through the higher acquired utility (at this level of student experience) of productions that interpret verbal forms over those that interpret symbolic forms.

There are some areas for potential improvement in the model. The current model implements giving-up as an explicit choice, but it is more likely that giving up is an error of omission produced by a failure to retrieve a relevant production. ACT-R provides the ability to model errors of omission by introducing a cutoff on the latency of memory retrievals. Similarly, arithmetic slips and errors we coded as "unknown" could be modeled more accurately if we took advantage of the partial matching feature of ACT-R (Lebiere, Anderson & Reder, 1994). Finally, we have begun to wonder whether the productions for strategy selection are necessary. Rather than viewing strategies as monolithic wholes that get selected and pursued to completion, it may be that different strategies are epiphenomenal consequences of single production rule differences or small representational changes that have large behavioral consequences. Instead of explicitly selecting strategies, the model might go directly to productions that interpret the problem statement and perform translations or transformations. This could mean a reduction in the number of productions and parameters needed.

Future Work

Good instruction steps students through zones of proximal development (e.g., Brown, 1994; Vygotsky, 1978). A zone of proximal development is a characterization of a student's current understanding of a set of concepts in a domain. Students move from one zone to a more sophisticated one by learning to overcome difficulties along one or more dimensions (Carpenter and Fennema, 1992). We want to identify students' zones of proximal development (ZPDs) and, more generally, have a set of dimensions for characterizing what they are and how progressive movement between them might occur. Analysis of these difficulty factors, and cognitive modeling based on that analysis, provides a method for identifying ZPDs. Our hypothesis is that ZPDs can be characterized within a space of problem difficulty factors, which we have used as a heuristic for our cognitive modeling efforts.

Our current model provides one possible zone of proximal development for algebra word problem solving, a snapshot in the development of algebraic problem solving knowledge. However, a snapshot can't substitute for a portrait over time. We are now looking into modeling the transition of weaker students into stronger ones, which will provide us with a dynamic portrait. ACT-R provides automatic production-rule tuning mechanisms, such as learning to predict the utility of a particular strategy in a given context, for this future work.

References

Anderson, J. R. (1993). Rules of the Mind. Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1990). The Adaptive Character of Thought. Hillsdale, NJ: Erlbaum.

Bobrow, D. G. (1968). Natural language input for a computer problem-solving system, in Semantic information processing. Cambridge, Mass.: MIT Press, 146-226.

Brown, A. L. (1994). The advancement of learning. *Educational Researcher, 23, 8*, 4-12.

Carpenter, T. P., & Fennema, E. (1992). Cognitively Guided Instruction: Building on the knowledge of students and teachers. In W Secada (Ed.), Curriculum reform: The case of mathematics in the United States. Special issue of the International Journal of Educational Research, (pp 457-470). Elmsford, N.Y.: Pergamon Press.

Koedinger, K. R., & Anderson, J. R. (1993). Effective use of intelligent software in high school math classrooms. Artificial Intelligence in Education: Proceedings of the World Conference on AI in Education, AACE: Charlottsville, VA.

Koedinger, K.R., & Tabachneck, H.J.M. (1994). Two strategies are better than one: multiple strategy use in word problem solving. Presented at the annual meeting of the American Educational Research Association, New Orleans, LA.

Koedinger, K.R., & Tabachneck, H.J.M. (1995). *Verbal reasoning as a critical component in early algebra*. Paper presented at the 1995 annual meeting of the American Educational Research Association, San Francisco.

Lebiere, C., Anderson, J. R., & Reder, L. M. (1994). Error modeling in the ACT-R production system. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Lewis, C. H. (1981). Skill in algebra. In J. R. Anderson (Ed.), Cognitive Skills and Their Acquisition (pp. 85-110).

Mayer, R. E. (1982). Different problem-solving strategies for algebra word and equation problems. Journal of Experimental Psychology: Learning, Memory, and Cognition, 8, 448-462.

Nathan, M.J., Koedinger, K.R., Tabachneck, H. T., Difficulty Factors in Arithmetic and Algebra: The Disparity of Teachers Beliefs and Students Performance. Paper prepared for The 1996 American Education Research Association Annual Meeting, New York.

Newell, A., & Simon, H. A. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice-Hall.

Polk, T.A. (1992) Verbal Reasoning. Dissertation, Carnegie Mellon School of Computer Science #CMU-CS-92-178.

Tabachneck, H. (1992). Computational differences in mental representations: Effects of mode of data presentation on reasoning and understanding. Doctoral Dissertation. Carnegie Mellon.

Tabachneck, H. J. M., Koedinger, K. R., & Nathan, M. J. (1994). Toward a theoretical account of strategy use and sense-making in mathematics problem solving. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Tabachneck, H. J. M., Koedinger, K. R., & Nathan, M. J. (1995). A cognitive analysis of the task demands of early algebra. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*. Hillsdale, NJ: Erlbaum.

Vygotsky, L. S. (1978). Mind in society. Cambridge, MA: Harvard University Press