16. Ohlsson, S.: Computer simulation and its impact on educational research and practice. International Journal of Educational Research. 12, 5-34 (1988)
17. Platt, J.: Social traps. American Psychologist. 28, 641-651 (1973)
18. Plötzner, R., Spada, H., Stumpf, M., & Opwis, K.: Learning qualitative and quantitative reasoning in a microworld for elastic impacts. European Journal of Psychology of Education. 4, 501-516 (1990)
19. Schroeder, D.A., Jensen, T.D., Reed, A.J., Sullivan, D.K., & Schwab, M.: The actions of others as determinants of behavior in social trap situations. Journal of Experimental Social Psychology. 19, 522-539 (1983)
20. Spada, H., & Ernst, A.M.: Wissen, Ziele und Verhalten in einem ökologisch-sozialen Dilemma. In: Umwelt und Verhalten (K. Pawlik & K.-H. Stapf, Hrsg.). S. 83-106. Bern: Huber 1992
21. Spada, H. & Opwis, K.: Ökologisches Handeln im Konflikt: Die Allmende-Klemme. In: Umwelt und Handeln (P. Day, U. Fuhrer, & U. Laucken, Hrsg.). S. 63-85. Tübingen: Attempto 1985
22. Spada, H., Opwis, K., Donnen, J., Schwiersch, M., & Ernst, A.: Ecological knowledge: Acquisition and use in problem solving and decision making. International Journal of Educational Research. 11, 665-685 (1987)
23. VanLehn, K.: Problem solving and cognitive skill acquisition. In: Foundations of cognitive science (M.I. Posner, ed.). pp. 527-579. Cambridge, MA: MIT Press 1989
24. Wenger, E.: Artificial intelligence and tutoring systems. Los Altos, CA: Morgan Kaufmann 1987
25. White, B.Y., & Frederiksen, J.R.: Causal model progressions as a foundation for intelligent learning environments. Artificial Intelligence. *42*, 99-157 (1990)
26. Wilke, H.A.M., Messick, D.M., & Rutte, C.G. (eds.): Experimental social dilemmas. Frankfurt/M.: Peter Lang 1986

Regian, J. W. and Shute, V. J. (1993). Basic research on the pedagogy of automated instruction In D. M. Towne, T. de Jong, and H. Spada (Eds.), Simulation-based experiential learning (pp. 121-132). Berlin: Springer-Verlag, Series F, Vol. 122.

Basic Research on the Pedagogy of Automated Instruction

J. Wesley Regian and Valerie J. Shute

Armstrong Laboratory, Human Resources Directorate, Brooks Air Force Base, TX 78235, U.S.A.

Abstract: In this chapter we argue for the importance of basic research on the pedagogy of automated instruction, and outline a relevant research project currently underway at the Armstrong Laboratory. The goal of the project is to delineate general principles as well as specific guidelines for developers of automated instruction. The project seeks to contribute to knowledge of automated instruction in the following four ways. First, we are supporting the development and use of a task-decomposition taxonomy to promote and support synthesis of results across studies. Second, we are developing a set of standard criterion tasks that will allow benchmarked comparisons of instructional approaches. Third, we are developing automated instructional systems, including Intelligent Tutoring Systems (ITS), based on formal theories of knowledge and skill acquisition. Fourth, we are rigorously evaluating these instructional systems in a controlled laboratory setting. In this chapter, we briefly describe this approach to conducting basic research on automated instruction, and then present three examples of research findings from our laboratory: (a) An alternative approach to using computer resources in training environments that, for certain classes of tasks, *quadruples* the number of trainees that may be trained with fixed training resources; (b) An inexpensive instructional intervention that significantly reduces or eliminates the post-training gender gap in performance of a highly spatial, complex dynamic control task; and (c) A simple, inexpensive intervention that reduces post-training error rates by 50% and performance latency by 33% in a procedural console operation task. We conclude by briefly discussing the implications of these findings for developers of automated instruction.

Keywords: automated instruction, basic research, computer-based training, intelligent tutoring systems, pedagogy

Introduction

Automated instruction is now commonplace. Most major software packages today include "tutorials" as a means of teaching new users about the software. Business and industry routinely use automated instruction, as evidenced by the wide array of Computer-Based Training (CBT) authoring systems on the market today. There are also several Intelligent Tutoring System (ITS) authoring packages under development [1, 8, 21]. This trend suggests that automated instruction will be even more common in the future than it is now. As powerful microprocessors become increasingly inexpensive, and as authoring systems reduce development time and expense, we are likely to see more automated instruction developed, delivered, and sometimes even used. But what are the chances that it will be high quality (i.e., pedagogically sound) automated instruction?

There is good evidence that carefully developed automated instruction can be instructionally effective. For example, in the past ten years, many ITSs have been built incorporating various approaches to instruction, diagnostic student modeling, and remediation, and a subset of those systems have been formally evaluated. Shute and Regian [19] reviewed four ITSs that have undergone empirical evaluation: (a) The LISP tutor, which teaches programming in LISP [2]; (b) Smithtown, which teaches scientific inquiry skills in the context of microeconomics [17, 18]; (c) Sherlock, which teaches avionics troubleshooting [10]; and (d) Bridge, which teaches programming in Pascal [16]. The results of these evaluations were impressive. Learning efficiency (rate) with ITSs was accelerated in comparison to control conditions. Students acquired the subject matter faster from ITSs than from more traditional environments. Subjects working with the LISP tutor learned the knowledge and skills in 1/3 to 2/3 the time it took a control group to learn the same material. Subjects working with Smithtown learned the same material in 1/2 the time it took a classroom-instructed group. Subjects working with Sherlock learned (in 20 hours) skills which were comparable[1] to some of the key skills possessed by technicians having almost 4 years experience. Subjects learning from the Pascal ITS acquired in 1/3 the time, equivalent knowledge and skills as learned through traditional instruction. With regard to learning outcome, ITSs again performed well in comparison to control conditions. The LISP tutored group attained the same (or in one study, 43% better) criterion scores as a control group not using the tutor. Results from the Smithtown analysis showed that subjects learned the same material as a classroom group, despite the fact that the tutor focused on the instruction of scientific inquiry skills, not the subject matter. The outcome data from subjects using Sherlock showed increases in scores comparable to an advanced group of subjects and significantly better than a control group. In all cases, individuals learned faster, and performed at least as well, with the ITSs as subjects learning from traditional environments.

What can we make of these findings? More importantly, what can the instructional development community make of these findings? It is important to note that, as always, there is a selection bias for publication of *successful* instructional interventions. Thus, while these published evaluations are good reason for optimism, they must be weighed against the total number of automated instructional systems that have been developed. Do these findings imply that instructional developers in business, industry, defense, and education are now prepared to build excellent automated instruction? In our experience, the answer is NO. These instructional developers are typically not aware of the literature cited here, nor would it be particularly useful to them if they were. Some of the reasons for this are:

> (1) The authoring packages commonly available today do not support "intelligent" pedagogy at run-time. Although a few ITS authoring packages are emerging, they are not yet ready for general use. The expense involved in building intelligent instruction is still prohibitive for most organizations.
>
> (2) The jargon and prior knowledge required to understand the literature prevents access by instructional developers in applied settings. Even to the

[1]This is not to say that 20 hours of Sherlock instruction is equivalent to 4 years of experience. Rather, 20 hours of Sherlock instruction produced students who performed certain relevant tasks as well as technicians with 4 years of experience.

> initiated, it is typically not clear how to generalize from the results of published studies to new instructional domains. Many real-world instructional domains differ on several dimensions from the instructional domains discussed in the literature.
>
> (3) The available literature is not intended to be prescriptive, and is too dispersed to be useful to the applied community. It is also incomplete in that the selection bias for publication of effective interventions limits the availability of negative information. If one cannot easily learn from the literature how to build ITSs, it is even more difficult to learn what not to do.

Invariably, successful ITSs have been built by teams of researchers with years of experience in the fields of education, psychology, computer science and whatever discipline is the target of instruction. The tacit knowledge brought to bear by these teams is both broad and deep. With regard to building instruction, that which seems obvious and self-evident to John Anderson, William Clancey, Alan Lesgold, or John Self is not obvious to the average instructional developer. Applied instructional developers cannot be guided by their intuitions on key issues of pedagogy. In applied settings, instructional developers tend to be guided by broad frameworks such as Gagné's *nine events of instruction*, or the Instructional Systems Design (ISD) process [7]. These frameworks provide little more than procedural checklists for instructional developers. Although they provide useful information about *what* needs to be done in developing instruction, they tell very little about *how* to do it.

In our work with instructional developers in applied settings, we are often asked to provide practical "how-to" information. In the long run, we believe that one of the key roles of Artificial Intelligence in education and training should be to provide instructional developers with ready access to general principles and specific guidance on the pedagogy of automated instruction. Regardless of whether the delivered system is to be a non-intelligent CBT system or a sophisticated ITS, expert systems should be able to guide the developer during the development process. After an extensive review of the literature on automated instruction [11], we concluded that clear and consistent guidance on the pedagogy of automated instruction is not available. Rather, studies on instruction are often plagued by noisy data, methodological flaws, small samples, and various unpleasant constraints arising from the realities of educational environments. Donchin described some of the problems as follows:

> As my colleagues and I examined the literature on training and practice, we became increasingly, and painfully, conscious of the fact that it is very difficult to integrate the studies we were reviewing. The theoretical acumen and the ingenuity of previous investigators was beyond reproach. A vast number of papers had been published within such domains as 'learning theory', 'training', 'motor behavior' and similar areas. However, it was quite evident that the diversity of paradigms and theoretical approaches within which the phenomena were studied, and the models tested, made it very difficult to compare result across studies. The many contradictions which are frequent in any body of literature were difficult to resolve because much of the conflict could be attributed to the different settings, and paradigms, in which the phenomena were studied. [4]

Cronbach and Snow [3], Donchin [4], Kyllonen and Shute [10] and others have argued that instructional research would greatly benefit from a more systematic approach. Researchers at the Armstrong Laboratory have a unique opportunity to facilitate such a systematic approach because we conduct in-house programs of research and simultaneously provide funding to other researchers through our extramural program. In the next section, we describe the basic tenets of our program of research on automated instruction.

A Program of Research on Automated Instruction

Our approach to basic research on the pedagogy of automated instructional systems may be depicted as involving four basic tenets, characterized as follows.

(1) Adopt a standardized taxonomic characterization of human performance to guide research and support generalization of principles. In order to generate guidelines for developers of automated instruction, we feel it is important to establish a clear and concise language for describing task characteristics. In order to do so, we are beginning to develop a standard set of primitives for describing task performance. Using a taxonomy of such primitives, one should be able to generalize across research findings, as well as from research findings to new tasks for which one seeks to build instruction. A team of researchers including: John Anderson, Arthur Fisk, Earl Hunt, Alan Lesgold, Jim Pellegrino, and ourselves have drafted an initial task performance taxonomy for this purpose. We are planning to publish the current form of the taxonomy in an appropriate forum, although we expect that the taxonomy will continue to evolve based on empirical results.

(2) Design and develop a set of criterion tasks that serve as benchmarks for pedagogical research. Our goal is to compare the instructional effectiveness of alternative instructional interventions derived from various theories of knowledge and skill acquisition. In order to do so, we have developed a set of criterion tasks, guided by our taxonomy, that are designed to be laboratory analogs of a broad spectrum of real-world instructional domains and tasks. We are currently making the first set of criterion tasks available to interested researchers. Two representative criterion tasks, *Space Fortress* and *Loader*, are discussed later in this chapter. The set of criterion tasks will continue to grow with time. In 1993, we will begin to provide funding for interested researchers to develop additional criterion tasks.

(3) Develop theory-based, automated instructional systems for these criterion tasks. We have an in-house research program that is already studying pedagogy, using performance on our criterion tasks as the outcome measures. We are developing automated instruction for the criterion tasks, and providing extramural funding to qualified researchers who are willing to focus their instructional interventions on one or more of our criterion tasks.

(4) Empirically compare performance on the criterion tasks after various instructional interventions. We have built a research laboratory (Co-Lab, for Cooperative Laboratory) with funding, personnel, hardware, and software to support 50,000 hours of subject-data collection per year. For each of the studies we conduct in the Co-Lab, subjects are hired, year-round, through temporary personnel services. Our subjects are retained for from three hours to three months, with sample sizes ranging from 30 to 500. Because of this unique situation, we are

able to specify target-population characteristics for the studies, and then hire subjects who conform to the specified characteristics.

Representative Studies

In this section, we briefly describe a few of our studies (and some preliminary results) from the first year of this research program. To achieve brevity, we focus on only two of our criterion tasks (currently there are 14), *Space Fortress*, and *Loader*. Our goal in outlining these studies is to give a flavor of the types of research we are conducting.

***Space Fortress* Research on Small-Group Pedagogy.** *Space Fortress* is a video game-like laboratory task with a long record as a research tool for studying the training and acquisition of high-performance skill (see Volume 71 of *Acta Psychologica*, 1989). This extremely complex dynamic control task was originally developed at the University of Illinois under funding from DARPA (the Defense Advanced Research Project Agency) as part of the Learning Strategies project [4]. We have developed an updated version of the program to run in our laboratory, with more flexibility than the original for research purposes. Shebilske and Regian [13], and Shebilske, Regian, Arthur and Jordan [14] report some of the research conducted in our laboratory using *Space Fortress*. We have learned that for this task, it is possible to train up to four trainees on a single computer while achieving individual performance levels equivalent to those attained by trainees trained in the same amount of time on four separate computers (see Figure 1). This is achieved by virtue of a training protocol involving a combination of whole-task practice, shared part-task practice, and observational learning among trainees.

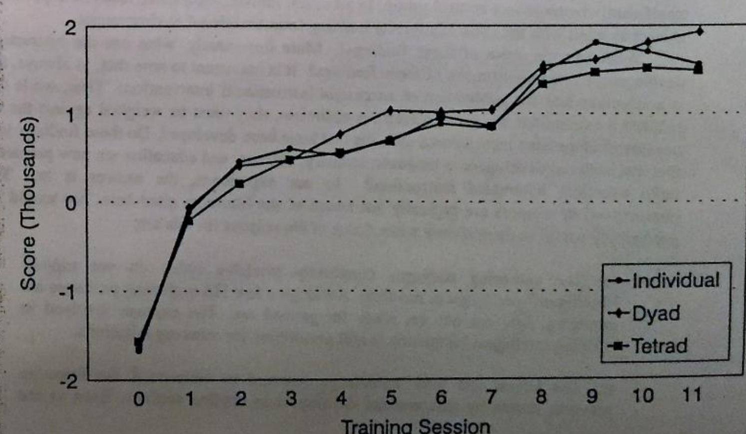

Figure 1: Comparable performance by students who trained alone, in pairs, or in groups of four on a single computer. Scores are based on test trials where students performed alone, with training time held constant across groups.

Based on these findings, we are beginning to design prototype instructional systems which operate from the perspective of small-group pedagogy rather than individual pedagogy. It is not our intention to argue that student modeling for individualized instruction is not feasible. Rather, we argue that, in some instructional settings with some kinds of tasks (or at certain stages of skill acquisition), automated instruction may be more cost-effective or even more instructionally-effective when designed for small groups rather than for individuals.

We are observing clear and specific benefits of small-group targeted instruction. For example, by focusing on subcomponents of the target task, students are able to master these components more completely. During observation periods, students are able to learn about task complexities and alternate strategies by observing other students performing the task. Requiring students to perform the entire task regularly enables them to integrate task components. The common desire for better group performance leads students to teach one another, benefiting both the provider and the recipient of the instruction. Since students diagnose and remediate each other's performance, the difficult problems of automated diagnosis and natural language processing are avoided. By teaching multiple students simultaneously, the requisite hardware investment is reduced, as are demands on human instructor time.

Space Fortress **Research on Gender-related Performance Differences.** In *Space Fortress*, as with many tasks involving significant spatial components, there is a robust gender effect (see Figure 2). Men typically outperform women on spatial tasks, including static spatial tasks (e.g., mental rotation, mental paper folding, form boards) as well as dynamic spatial tasks (e.g., collision estimation, dynamic control of moving figures). There has been a long-running argument over whether this gender difference in spatial performance is due to nature (hard-wired attributes) or nurture (differential experiences growing up). There is clear evidence for both --some of the difference has been shown to be related to differences in testosterone levels [9, 20] and some due to experiential differences.

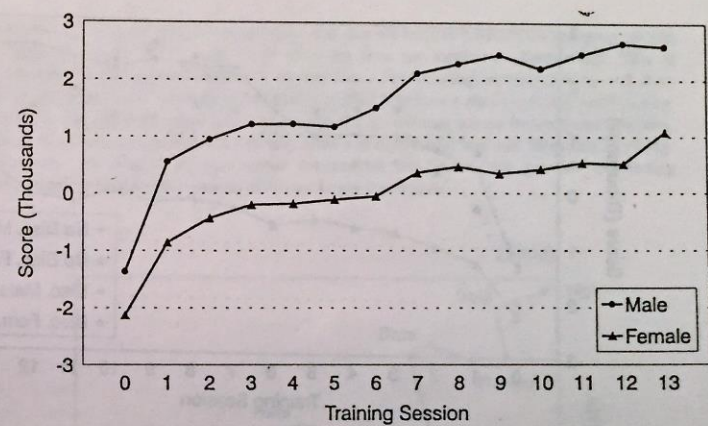

Figure 2: Initial performance differences between males and females on Space Fortress tend to increase after extensive practice at the task.

We have been using *Space Fortress* to look at instructional interventions that might overcome the experiential deficits. We were surprised to find a very simple intervention that seems to go a long way in this direction; namely, placing women in discussion groups with men. We had female subjects participate in brief but regularly-occurring (and specifically structured) discussion groups with male subjects, to talk about *Space Fortress* strategy and tactics. This simple intervention dramatically increased women's *Space Fortress* performance. Figure 3 shows that, on average, short discussion groups which follow practice sessions produce a small positive effect on men's skill acquisition, but a significant positive effect on women's. The women under this treatment perform nearly as well as the men.

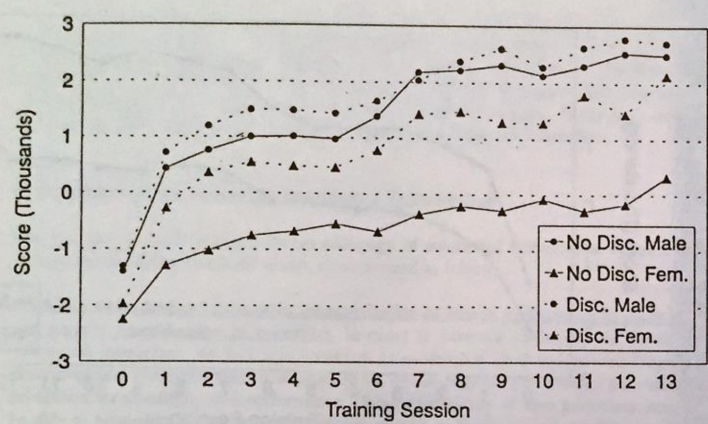

Figure 3: Initial performance differences between males and females on *Space Fortress* are significantly reduced by a simple intervention. Short group discussion periods after each training session have a large positive effect on female performance.

Loader **Research on Mental Models during Training.** *Loader* is the name of a complex procedural task designed and developed in our laboratory [5]. It requires subjects to execute long sequences of console-operation actions (e.g., button presses, switch actuations, dial rotations) to accomplish specific goals. The task is based on a simulated console which controls railroad cars, tracks, and cranes in a fictitious railroad yard. *Loader* is operated through a graphical user interface developed in Asymetrix Toolbook, running under Microsoft Windows. The task is designed to be a laboratory analog of procedural operations and process control tasks, which are common in industrial and defense settings.

We hypothesized that acquisition of *Loader* performance skill would be accompanied by the development of a dynamic mental model linking console actions to events in the "railroad yard." That is, in the process of learning to carry out a specific sequence of actions to accomplish a goal, one would come to imagine the corresponding events in the railroad yard, even if the operator could not actually see the yard while operating the console. We therefore conducted the following experiment [6]. Forty subjects were shown a simple, static, bird's-eye view of the railroad yard indicating the layout of tracks, the initial location of cars, and the locations of bins and the crane. Subjects were told that they would learn to operate a console that would enable them to move the cars around on the tracks, and that moving the cars was necessary to access the storage bins. Furthermore, they were informed that the crane was needed to move canisters between bins and cars. Subjects were randomly divided into two groups (i.e., dynamic model vs. no model). During training, both groups received identical text-based instruction in an instructional-window above the *Loader* interface. One group, however, additionally saw a *dynamic* version of the bird's-eye view of the railroad yard. After training, both groups were tested under identical conditions. They were asked to perform the complete procedure without guidance and without access to either type of railroad yard

representation. The results were striking. Rather than becoming dependent on the animated rail-yard model, subjects in the dynamic model condition apparently internalized the model, as evidenced by their performance after the model was removed. Post-training performance was significantly faster (Figure 4) and less error-prone (Figure 5) for subjects trained with a dynamic graphical model compared to the no-model condition.

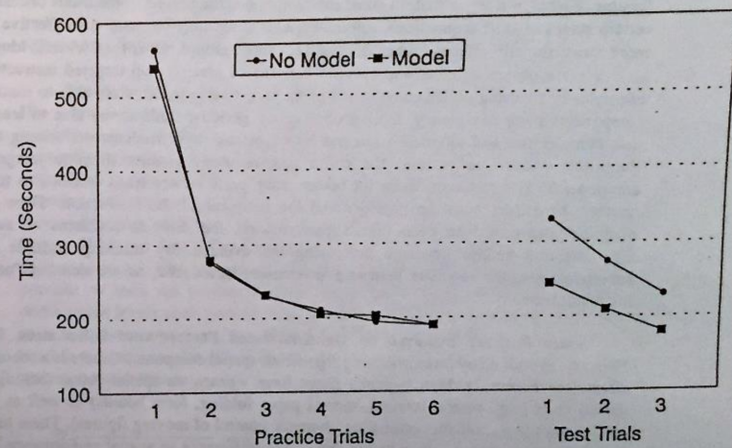

Figure 4: Providing students with a simple graphical model of a complex procedural task has no effect on latency during practice, but significantly reduces latency after the model is no longer available.

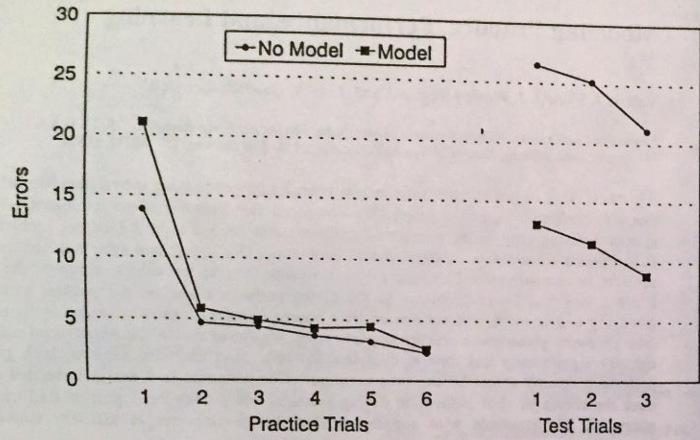

Figure 5: Providing students with a simple graphical model of a complex procedural task has a temporary negative effect on accuracy during early practice, but significantly reduces errors after the model is no longer available.

Discussion

In this chapter, we have outlined our approach to conducting laboratory research on the pedagogy of automated instruction. We believe that such an approach is an important first step in achieving the goal of providing instructional developers with both general principles and specific guidance on the pedagogy of automated instruction. It is important to note, however, that laboratory research alone (as illustrated in this chapter) cannot answer all questions of pedagogy. Field research is also important (e.g., conducting studies on instructional effectiveness directly in high school classrooms or industrial training areas). It is actually the *interplay* between laboratory and field research that will ultimately provide answers to pedagogical research questions.

Consider the relationships depicted in Figure 6. First, an inverse relationship exists between internal validity (which can be optimized in laboratory-based experiments) and external validity (which can be optimized in field-based experiments). In pedagogical research, increases in external validity tend to produce a decrease in internal validity (see Shute & Regian, in press, for more on this topic). Thus, as you increase your ability to generalize pedagogical findings to applied settings, you lose the level of experimental control afforded within the laboratory. We believe that the solution to this conundrum is to initially develop and test pedagogical principles in a laboratory setting with careful attention to experimental control. Promising approaches should then be tested in increasingly field-like settings, and ultimately in applied settings with careful attention to external validity. The

second point we wish to highlight in Figure 6 is that we believe research on pedagogy should be both driven by theory and constrained by data (or empirical observation). This is represented on the graph in Figure 6 by the sine wave iterating between theory and data. Theory is important in generating potentially fruitful hypotheses about teaching and learning, and in driving generalizations about pedagogical effectiveness across instructional domains. Empirical observation is important to test, often and rigorously, how our ideas fare in reality. Only empirical data, with appropriate comparison conditions, can provide convincing documentation on the effectiveness of theory-based instruction.

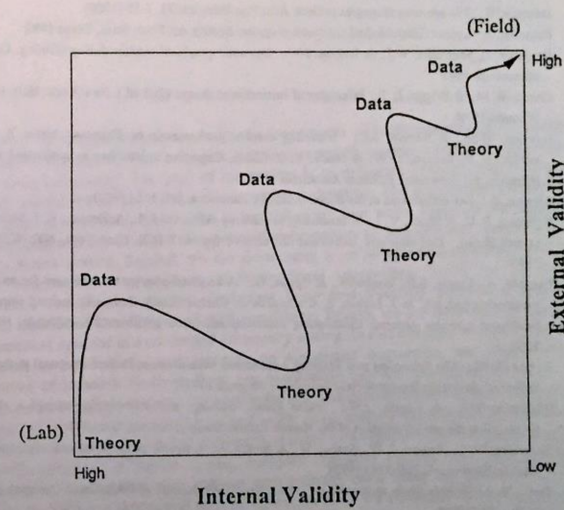

Figure 6: Inverse relationship between internal and external validity, and cyclical relationship between theory- and data-driven experimentation.

We have often been perplexed by the minimal effectiveness of theoretically-sound instructional interventions, and then surprised by the enormous power of relatively simple interventions. Theorizing about instruction without appropriate data is like studying astronomy without a good telescope. Your theories are likely to seem perfectly reasonable -- as far as you can see, that is.

References

1. Anderson, J.R., Corbett, A.T., Fincham, J.M., Hoffman, D., & Pelletier, R.: General principles for an intelligent tutoring architecture. In Regian, J. W. & Shute, V. J. (Eds.), Cognitive approaches to automated instruction. Hillsdale, NJ: Lawrence Erlbaum Associates 1992
2. Anderson, J.R., Farrell, R., & Sauers, R.: Learning to program in LISP. Cognitive Science, 8, 87-129 (1984)
3. Cronbach, L.J., & Snow, R.E.: Aptitudes and instructional methods: A handbook for research on interactions. New York: Irvington 1977
4. Donchin, E.: The learning strategies project, Acta Psychologica, 71, 1-15 (1989)
5. Farquhar, J.: Loader. Unpublished computer program. Brooks Air Force Base, Texas 1992
6. Farquhar, J., Shebilske, W.L. & Regian, J.W.: Dynamic graphical models during training. Unpublished manuscript 1992
7. Gagné, R. M., & Briggs, L. J.: Principles of instructional design (2nd ed.). New York: Holt, Rinehart, & Winston 1979
8. Johnson, W.B., & Norton, J.E.: Modeling student performance in diagnostic tasks: A decade of evolution. In Regian, J. W. & Shute, V. J. (Eds.), Cognitive approaches to automated instruction. Hillsdale, NJ: Lawrence Erlbaum Associates 1992
9. Kimura, D.: Sex differences in the Brain. Scientific American, 119-125 (1992)
10. Kyllonen, P. C. & Shute, V. J.: A taxonomy of learning skills. In P. L. Ackerman, R. J. Sternberg & R. Glaser (Eds.), Learning and Individual Differences (pp. 117-163). New York, NY: W.H. Freeman 1989
11. Lesgold, A., Lajoie, S.P., Bunzo M., & Eggan, G.: A coached practice environment for an electronics troubleshooting job. In J. Larkin, R. Chabey, & C. Cheftic (Eds.), Computer assisted instruction and intelligent tutoring systems: Establishing communication and collaboration. Hillsdale, NJ: Erlbaum 1992
12. Regian, J. W.: The Schooling and Training Approaches Measurement Project. Proposal to the Air Force Office of Scientific Research New Intitiatives Program 1992
13. Shebilske, W.L., & Regian, J.W.: Video games, training, and investigating complex skills. Paper presented at the annual meeting of the Human Factors Society, Atlanta, GA 1992
14. Shebilske, W. L., Regian, J. W., Arthur, W., & Jordan, J.: A dyadic protocol for training complex skills. Human Factors, 34, 369-374 (1992)
15. Shute, V. J.: Who is likely to acquire programming skills? Journal of Educational Computing Research, 7(1), 1-24 (1992)
16. Shute, V. J. & Glaser, R.: Large-scale evaluation of an intelligent tutoring system: Smithtown. Interactive Learning Environments, 1, 51-76 (1990)
17. Shute, V. J. & Glaser, R.: An intelligent tutoring system for exploring principles of economics. In R. E. Snow & D. Wiley (Eds.), Improving Inquiry in Social Science: A Volume in Honor of Lee J. Cronbach (pp. 333-366). Hillsdale, NJ: Lawrence Erlbaum Associates 1991
18. Shute, V.J., & Regian, J.W.: Rose garden promises of intelligent tutoring systems: Blossom or thorn? Paper presented at the Space Operations Automation and Robotics conference, Albuquerque, NM 1990
19. Shute, V. J. & Regian, J. W.: Principles for evaluating intelligent tutoring systems. To appear in a special issue of: Journal of Artificial Intelligence & Education, January, 1993.
20. Shute, V.J., Pellegrino, J.W., Hubert, L., & Reynolds, R.W.: The relationship between androgen levels and human spatial abilities. Bulletin of the Psychonomic Society, 26(6), 465-468 (1983)
21. Towne, D.M., & Munro, A.: Supporting diverse instructional strategies in a simulation-oriented training environment. In Regian, J. W. & Shute, V. J. (Eds.), Cognitive approaches to automated instruction. Hillsdale, NJ: Lawrence Erlbaum Associates 1992

Shute, V. J., Regian, J. W., and Gawlick-Grendell, L. A. (1993). Modeling practice, performance, and learning. In D. M. Towne, T. de Jong, and H. Spada (Eds.), Simulation-based experiential learning (pp. 133-148). Berlin: Springer-Verlag, Series F, Vol. 122.

Modeling Practice, Performance, and Learning

Valerie J. Shute,[1] J. Wesley Regian,[1] and Lisa A. Gawlick-Grendell[2]

[1] Armstrong Laboratory, Human Resources Directorate, Brooks Air Force Base, TX 78235, U.S.A.
[2] Computer Data Systems, Inc., 8626 Tesoro Drive, Suite 114, San Antonio, TX 78217, U.S.A.

Abstract: This chapter presents the results from a study examining the relationship between practice, performance, and learning. We compared two versions of an intelligent tutoring system differing only in the number of problems that needed to be solved per problem set (Abbreviated = 3 problems, Extended = 12 problems). Our hypotheses were that Abbreviated subjects, in comparison to Extended subjects, would: (a) take less time to complete the tutor because they had fewer problems to solve, (b) perform worse on the posttest measures (accuracy and latency), and (c) demonstrate poorer transfer of knowledge and skills across tutor problems given fewer practice opportunities. We found that, while Abbreviated subjects did take significantly less time to complete the tutor than Extended subjects, both groups performed *equally* across all outcome measures. Componential skill analyses enabled us to track the course of skill acquisition during practice, and predict the degree of skill transfer afterward. We conclude with suggestions for the development of efficient automated instruction.

Keywords: Cognitive abilities, componential skill analysis, intelligent tutoring systems, modeling, learning (outcome and efficiency), performance, practice effects, transfer (lateral and vertical)

Introduction

The purpose of this chapter is to examine the effects of practice on within-tutor performance and learning outcome. The relationship between practice and performance addresses the issue of how practice influences learning rates, errors, and the degree of successful transfer *during* the learning process [13]. And the relationship between practice and outcome addresses how practice affects what learners ultimately walk away with at the *end* of a learning task, including retention, application, and transfer to some novel task. Both of these relationships are believed to follow the well-documented tenet: "Practice makes perfect" [1-3, 5, 11, 15] as well as the related tenet: When the number or variety of example problems is restricted, skill acquisition tends to be rapid, but transfer tends to be weak [4, 7]. In this chapter, we submit both convictions to an experimental test to determine just how much practice makes perfect, what is weak transfer, and so on.

Another relationship we are interested in exploring, but which is not so clear, exists between performance and outcome. It seems reasonable to assume that acquisition performance is a good indicator of "learning." However, Schmidt and Bjork [10] have shown how, relative to a "standard" condition, practice environments that show little improvement, or even *decreased* performance during skill acquisition, may actually produce increased outcome performance. What is ultimately learned may therefore be poorly reflected by