Situated Pedagogical Authoring: Authoring Intelligent Tutors from a Student's Perspective

H. Chad Lane[1], Mark G. Core[2], Matthew J. Hays[2],
Daniel Auerbach[2], and Milton Rosenberg[2]

[1] University of Illinois, Urbana-Champaign
Department of Educational Psychology & Illinois Informatics Institute
Champaign, IL USA
hclane@illinois.edu

[2] University of Southern California
Institute for Creative Technologies
Playa Vista, CA USA
core, hays, auerbach, rosenberg@ict.usc.edu

Abstract. We describe the Situated Pedagogical Authoring (SitPed) system that seeks to allow non-technical authors to create ITS content for soft-skills training, such as counseling skills. SitPed is built on the assertion that authoring tools should use the learner's perspective to the greatest extent possible. SitPed provides tools for creating tasks lists, authoring assessment knowledge, and creating tutor messages. We present preliminary findings of a two-phase study comparing authoring in SitPed to an ablated version of the same system and a spreadsheet-based control. Findings suggest modest advantages for SitPed in terms of the quality of the authored content and student learning.

Keywords: authoring tools, intelligent tutoring systems, virtual humans.

1 Introduction

Despite decades of strong empirical evidence in their favor, the uptake of intelligent tutoring systems (ITSs) remains disappointing [1]. Although many factors have contributed to this lack of adoption [2], one widely agreed upon reason behind slow adoption and limited scalability of ITSs is that the engineering demands are simply too great. This is no surprise given that many attribute the effectiveness of ITSs to the use of rich knowledge representations [3, 4], which are inherently burdensome to build. Heavy reliance on software engineers has proven to be a significant hindrance for the widespread adoption of ITS technologies.

These challenges have led to decades of research aimed at reducing both the skills and time to build intelligent tutors. The resulting ITS authoring tools generally seek to enable creating, editing, revising, and configuring the content and interfaces of ITSs [5]. A significant challenge lies in the accurate capture of the domain and pedagogical expertise required by an ITS, and many authoring tools focus on eliciting this knowledge. In Murray's review of authoring tools [6], the top two goals identified are to decrease (1) the effort required to build an ITS (e.g., time, cost), and (2) the "skill threshold" for building ITSs. Systems addressing the first goal include those built for cognitive scientists and programmers, such as the cognitive modeling suite of tools in CTAT [7]. Murray's second goal, reducing the skill threshold of authors, is the focus of this paper. Systems in this category seek to leverage intuitively accessible tools that elicit the content and knowledge required by an ITS from non-technical users, such as instructors and subject-matter experts. Further, they share much in common with earlier efforts to address the knowledge elicitation problem [8], but with the additional burden of needing to address issues related to pedagogy.

A number of research efforts have directly sought to lower the skill threshold of ITS creation. For example, CTAT's second mode of authoring (distinct from the cognitive modeling components) allows authors to develop *example-tracing* tutors [9] that heavily leverage demonstration as a key knowledge elicitation technique. REDEEM, another extensive effort to reduce the technical expertise needed for building ITSs, provides intuitive interfaces and a well-defined workflow to produce adaptive, lightweight ITSs for the presentation and assessment of knowledge [10]. ASPIRE, also in the same category, asks users to design a basic domain ontology and solve problems while the system infers constraints for an ITS [11]. Evaluations of these tools typically focus on demonstrating efficiency [7] and completeness (to what degree do authored models align with hand-crafted models) [12]. Very little work has attempted to demonstrate the teaching efficacy of the ITSs that can be created, with REDEEM being a major exception [13]. The remaining sections of this paper summarize situated authoring (our approach), describe our authoring prototype that focuses on soft-skills training, and report initial results of an experiment intended to test the hypothesis that novice authors working in an environment that matches the learner's environment create higher quality and more effective tutoring content.

2 Situated Pedagogical Authoring

Like REDEEM, ASPIRE, and example-tracing tutors, the Situated Pedagogical Authoring system (SitPed) is designed as an easy-to-use authoring tool for eliciting ITS content from subject-matter experts. The current implementation focuses on problem-solving through conversation, such as how to address personal problems in the workplace or motivational interviewing for therapists and social workers. Our research builds on a substantial history of using virtual humans in support of learning [14], and specifically to act as role players that provide practice opportunities for soft skills [15]. In all previous cases, ITS technologies included in these systems were implemented by programmers based on expert interviews and cognitive task analyses.

SitPed was created to overcome this limitation by allowing non-technical authors to provide ITS content without programming. The aim is to place authors in an environment that is maximally similar to the one learners see, in part to constantly remind authors of the learner's experience, but also because it is the context in which their expertise is most beneficial. *We want authors to explicitly tell the system what* *learners should, and should not, be doing in a way that is familiar to them already*. For the purposes of this paper, therefore, we define "situated" authoring to be authoring that is completed in the same learning environment that learners will be using. Our primary hypothesis is that novice authors will create pedagogical content of higher quality when authoring is situated, and thus produce a more effective resulting product. We return to this hypothesis in section 3.

The implementation of SitPed described here is designed to support practice in the ELITE learning environment for leadership training [16]. Scenarios involve interacting with a virtual human via menus and according to an instructional model derived from a cognitive task analysis. Tutoring in this context involves the assessment of actions that are taken (i.e., how well they align with the prescriptions of the cognitive task analysis) and provision of guidance (i.e., hints and feedback). The ELITE team worked with the USC's Center for Innovation and Research on Veterans and Military Families to create a variation of the system designed for motivational interviewing, MILES, and we specifically used this content while developing and testing the system. In the rest of this section, we describe the current implementation of SitPed and discuss our approach to make authoring of this content more intuitive.

2.1 SitPed workflow

SitPed includes several connected supporting tools and typically involves many iterations over scenario data. The primary activities, shown in figure 1, include 1) defining tasks that will be practiced, 2) connecting those tasks to scenario data to enable assessment, 3) authoring feedback messages that learners will see, and 4) adding support for post-practice reflection. In this paper, we focus on the provision of coaching during practice (i.e., 1-3). In addition, we assume that scenarios are created by scenario writers separately, leaving SitPed authors the tasks identified above. In the case of ELITE, a separate tool is used for the creation of scenarios,[1] and so tighter integration of the complete authoring process is something we will consider in future work. For the purposes of this paper and the study below, authors focus only on ITS content and use pre-defined scenario files.

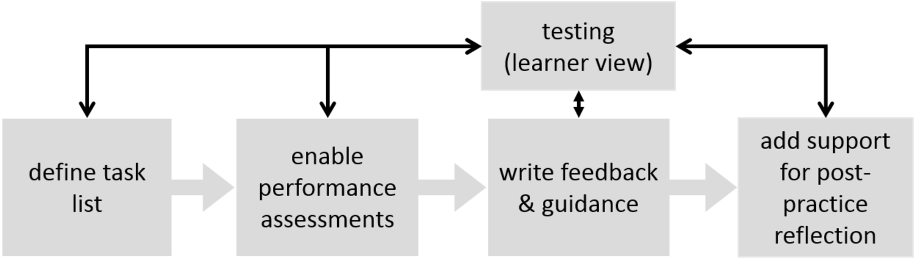

testing
(learner view)
define task list
enable performance assessments
write feedback & guidance
add support for post-practice reflection

Figure 1. The SitPed workflow.

Testing one's work is critical in SitPed (as it is with all authoring systems) so the author can see the results of their work in context. The loops present in figure 1 show

[1] http://www.chatmapper.com/

how an author might need to return to either edit or create tasks, adjust the assessment links, or update feedback content. The idea of being situated is most apparent when providing assessment knowledge and creating feedback in that the author must:

- specify paths through the problem space by simultaneously *solving* problems (either correctly or incorrectly) and indicating the relevant skills
- pause during problem solving to create hints and feedback messages associated with the current situation.

Since these activities take place in the same learning environment that learners use, SitPed falls roughly into the category of WYSIWYG authoring tools [6] because authors are constantly reminded of what the learner sees and does. With SitPed, demonstration is not simply a technique to hide technical details, but a way of organizing the tasks of authoring. It can be difficult for authors to visualize a learner's perspective when working in environments that are simply believed to be intuitive.

2.2 Defining tasks

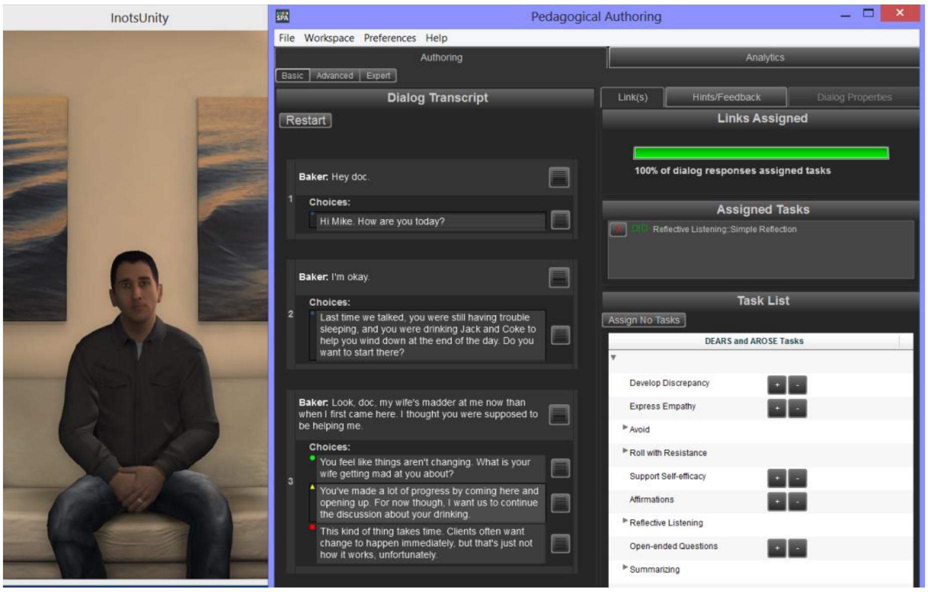
SitPed provides a simple tool to create simple, hierarchal task models, which define correct and incorrect behavior in scenarios (an example task list can be seen on the right of figure 2, which shows it being used in the assessment phase of authoring). Task lists in SitPed are roughly equivalent to multi-level numbered lists available in many word processors. Such tasks should be derived from a cognitive task analysis or some definitive resource, but we currently impose no such requirement (it is not an automated cognitive task analysis system).

The resulting list, which can be updated as needed throughout the workflow, acts as the functional glue holding the system together. It is not only a description of correct and incorrect behavior, but also a lightweight knowledge representation allowing the linking of instructional elements (e.g., a choice in a scenario) to behavior descriptions at other stages in the authoring workflow. Task lists form the basis for assessment and communication of that assessment to instructors and students. Higher levels of the hierarchy act as general categories while branches and leaves are more concrete, often corresponding to actions that can be taken in a scenario. Leaves of the hierarchy can even contain common misconceptions/mistakes associated with a task.

2.3 Assessment and situated linking of tasks to scenario data

The current version of SitPed targets branching conversations. At each step in the conversation, learners are selecting utterances from a menu and the virtual role player consults a tree to lookup its response and the next set of menu items. This conversation tree simply contains the lines of the conversation as well as the associated animations corresponding to performance of the role player lines. In branching conversations, it is necessary for the author to play through all branches of the tree and link each possible learner choice to the skills and misconceptions of the domain. This process is illustrated in figure 2. Although the goal is to recreate the learner experience as much as possible, authors need to be able to see relevant context (e.g., the dialog history in the middle) and make annotations corresponding to the skills and common mistakes of the domain.

To avoid overwhelming novice authors, they are first presented with just dialogue choices and the character, but once they choose to annotate an utterance, a list of tasks is opened and they are allowed to indicate any links that are relevant. For example, if an utterance is an example of reflective listening, the author will click the "+" button next to reflective listening in the task list (see figure 2). This action updates the screen to show the task has been assigned, and this assignment will re-appear on the authoring screen any time this utterance is revisited. SitPed also provides a progress bar which tracks coverage of the problem space.



1 **Baker:** Hey doc.
Choices:
Hi Mike. How are you today?

2 **Baker:** I'm okay.
Choices:
Last time we talked, you were still having trouble sleeping, and you were drinking Jack and Coke to help you wind down at the end of the day. Do you want to start there?

3 **Baker:** Look, doc, my wife's madder at me now than when I first came here. I thought you were supposed to be helping me.
Choices:
🟩 You feel like things aren't changing. What is your wife getting mad at you about?
◆ You've made a lot of progress by coming here and opening up. For now though, I want us to continue the discussion about your drinking.
● This kind of thing takes time. Clients often want change to happen immediately, but that's just not how it works, unfortunately.

100% of dialog responses assigned tasks

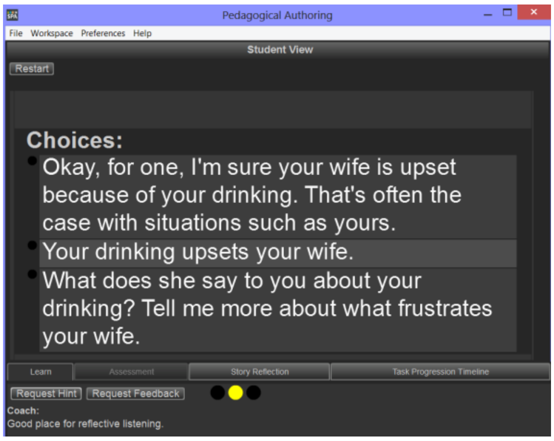
Reflective Listening::Simple Reflection

Figure 2. SitPed authoring screen used for linking scenario content to tasks. The virtual role player is animated and speaks according to author choices in the center column (which advances as the interaction proceeds).

This exhaustive exploration of the possibilities is necessary because of the difficulty of automatically understanding the dialogue well enough to identify skills such as reflective listening. As an author works through a scenario, s/he will frequently restart the dialogue to explore new branches and establish links along all or most of the branches in the space. It is acceptable to not tag every action (essentially saying they are not associated directly with any task) and to link an action to multiple tasks. In task domains like counseling it is common to have actions that have both pros and cons – this can be captured by creating a positive link (e.g., clicking the "+" sign next to reflective listening) and creating a negative link (e.g., linking to a mistake such arguing with the client). SitPed displays a colored shape next to each utterance as tags are added: a red circle means "incorrect" (all links are negative), a green square means "correct" (all links are positive), and a yellow diamond means a "mixed" set of links.

2.4 Authoring hints and feedback

When an ITS gives a hint or explains why something is wrong, it is a critical moment in learning. In SitPed, it is simple to create either hints (that are delivered when a learner is stuck or unsure about what is best) or feedback (that explain why an action had a specific impact on the character). Authors can choose to author tutor messages simultaneously with assessment tagging, or do it separately in a second pass through the scenario. To do so, when an action is selected (i.e., an utterance is clicked in the center column of figure 2), the author can select the “Hints/Feedback” tab in the authoring environment and enter the text they want to be delivered. To see a message delivered, an author can use “testing” mode, which is described next.



Choices:

- Okay, for one, I'm sure your wife is upset because of your drinking. That's often the case with situations such as yours.
- Your drinking upsets your wife.
- What does she say to you about your drinking? Tell me more about what frustrates your wife.

Coach:
Good place for reflective listening.

Figure 3. SitPed testing screen.

2.5 Testing and iterative development

Although the main authoring screen is situated, it was still necessary to provide a special testing screen. One advantage of the testing screen is that all editing controls and displays can be removed. Furthermore, the testing screen can replicate the user interface that delivers the authored content. Figure 3 shows the current testing screen. The virtual human is also displayed but we omit this from figure 3 for space reasons. The choices for how to respond to the virtual human appear on the testing screen. The correct, incorrect and mixed color codes are shown to learners in a sideways traffic light display which currently shows a mixed assessment of the previous choice. The lights provide immediate flag feedback and come from the links authors have made to the task list (section 2.3). Hints and feedback are solicited and learners click the appropriate button to request guidance when it is available. In this case, the user has clicked “Request Hint” and we see the hint in the bottom left corner.

3 Preliminary two-phase study

The hypothesis driving the design of SitPed was that an authoring environment that maximizes similarity to the actual learning environment will be more accessible to novice authors and support them in creating more pedagogically effective and higher quality ITS content. The study summarized in this section focuses on both properties of the authored content *and* on how well students learn from it. Thus, a two-phase study of SitPed was conducted in 2014 with subject-matter experts (phase 1, in the spring) and with college students (phase 2, in the fall) who had no experience with motivational interviewing (MI), our selected task domain.

3.1 Experimental design and procedure

In the first phase, a set of 11 domain experts from the USC School of Social Work with academic training and practical experience in MI were paid $50 to author ITS content for one scenario. They were split across three authoring conditions with the authoring interface acting as the lone independent variable:

1. **Full SitPed (N=4)**: the system as described in this paper.
2. **SitPed Lite (N=4)**: scaled-down version with hypertext-only (no graphics or sound, or supporting tools, like the progress bar)
3. **Spreadsheet (N=3)**: a specialized spreadsheet containing fields corresponding to data populated by SitPed, such as assessment links and tutor messages.

The Spreadsheet group was designed to intentionally be *non-situated* and those authors did not have the opportunity to test their resulting system at any time (i.e., they only filled in a spreadsheet and were given none of the SitPed tools). The spreadsheet was carefully created by an Excel expert (the third author) and designed to be as supportive as possible by restricting values in certain places, fixing the title rows, and so on. As a way to learn about why they were authoring, participants in phase 1 (experts) were asked to interact with a character from a different scenario and see tutoring in action. All participants were told that the data they were providing would be used for novice MI students at a later time. The same scenario data and task lists were given to all authors who were asked to link actions in the scenario to tasks and craft tutor messages (both hints and feedback). The predefined task list was a simplified version of the actual task list used in the MILES system, and contained 12 entries. The design of the three conditions is intended to capture three varying degrees of "situatedness", with a spreadsheet being entirely divorced from the learning environment and full SitPed being an almost full match. SitPed lite ablates many of the features of full SitPed and was designed to provide interactive authoring without many of the immersive features (animation, sound, etc.).

In the second phase, the data sets generated from each condition were used to create three separate tutoring systems, randomly using one of the data sets from each corresponding group. 71 college students from the University of Southern California participated in phase 2 of the study and were either compensated with course credits or paid. To measure knowledge, we used the Motivational Interviewing Knowledge and Attitudes Test (MIKAT) [17], which consists of 15 true/false questions followed by a selection task that gauges understanding of MI principles. Participants began by taking the MIKAT, watching a video about MI and how to use the testing screen of SitPed, and then interacted with the test scenario (from one of the three conditions) 3 times in a row. Participants then interacted with a new scenario without tutoring, to act as a performance-based post-test. Finally, participants took the MIKAT again and completed a post-test survey. A summary of the full experiment is shown in figure 4.

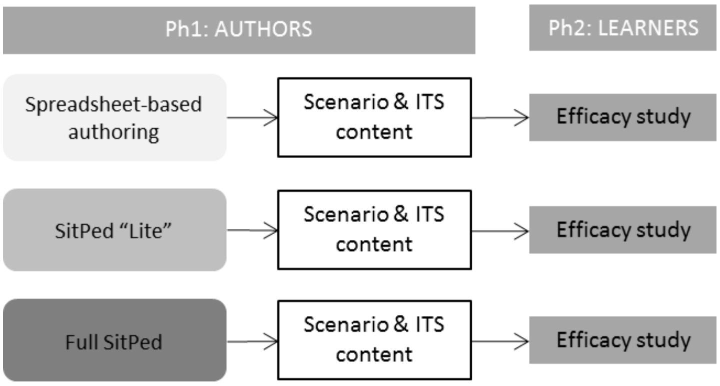


Figure 4. SitPed two-phase experimental design.

3.2 Phase 1 results: Differences in authored content

Because of the low number of authors (a total of 11 spread across the 3 conditions), we report only the raw, descriptive data here and consider them as formative. In general, the 11 experts were observed to work diligently in the three hours allocated and revise their work frequently. Further, they were also told that completeness was not a requirement, but to focus on areas they believed would be the most difficult for students. So, for example, it was not necessary to create a hint for each and every choice point or link multiple tasks to every action.

First, some important differences in terms of the number of tutoring messages created (see the "count" columns in Table 1). Authors using the spreadsheet created, on average, 40.7 hints out of a maximum of 72 and 80 feedback messages out of a maximum of 113 in the given scenario while those in the two SitPed conditions created far fewer. These large differences may be due to the spreadsheet doing a better job of helping the author see the scope of the work in front of them – i.e., they are able to see two columns of a spreadsheet that they are being asked to fill. However, when we look at the length of the messages authored ("length" columns in Table 1, which show the character counts of the messages), the reverse pattern is seen. Authors in the SitPed conditions created longer messages (113, 68, 105, and 98 characters) than those in the Spreadsheet group.

Table 1. Differences in tutor messages authored and authored links between groups.

Condition	Fbk count	Fbk length	Hint count	Hint length	1st links	2nd links
SitPed full	22.5	105	4.50	113	106	67.5
SitPed lite	6.25	98.0	11.8	68.0	98.3	59
Spreadsheet	80.0	42.0	40.7	17.0	111	37

Second, with respect to task links established, authors are being asked to identify tasks that are relevant to actions available at any given choice point. Table 1 (last two columns) show very few differences in this dimension, with the possible exception of creating second links, which imply that the author feels a particular action is related to more than one task.

3.3 Phase 2 results: Impacts of SitPed on student learning

In phase 2, participants were randomly assigned to one of three groups. Due to technical problems and participant errors (some independently chose to work through scenarios more times than requested), we ended up with 18, 20, and 16 in the three conditions (SitPed, SitPed lite, and the spreadsheet); thus, we only used data from 54 participants.

The MIKAT provided two different measures of learning: responses to the true/false questions and score on the concept selection task. In terms of T/F responses, we found a main effect of condition between participants favoring SitPed over the spreadsheet group (mean gains of .135 to .054, F(2,52)=3.635, p=.033). No other significant differences exist between the other groups, although a main effect overall was found (F(1,52)=20.511, p<.001). On the concept selection task, no significant differences emerged between conditions, although again an overall effect was found (F(1,52)=132.734, p<.001). Thus, the lower quantity of feedback and hint messages created in SitPed authoring did not hurt performance of learners. It may be the case that the SitPed condition had higher quality of links which drive the flag feedback seen by learners. Alternatively, messages in the spreadsheet condition may have actually hindered learning.

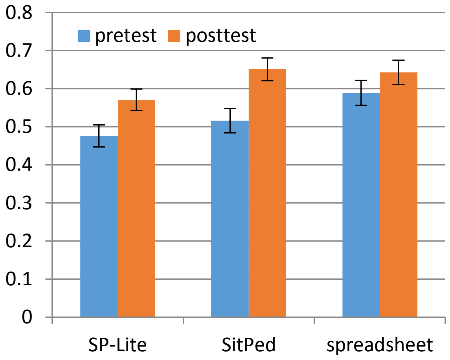


Figure 5. MIKAT scores across three authoring conditions.

4 Conclusion

We have presented Situated Pedagogical Authoring (SitPed), an approach to authoring built on the assertion that authoring tools should use the same learning environment that students use, to the greatest extent possible. Leveraging proven techniques such as programming by demonstration, SitPed authors are able to define positive and negative learner behaviors and create tutor messages in the context of the same environment that students use. Our preliminary study shows modest advantages for SitPed in terms of the quality of authored content and learning gains from the resulting tutors. In future work, we hope to deepen the integration of scenario authoring with ITS authoring and better understand the qualitative differences between tutoring content created in SitPed versus that created in less immersive systems, such as a spreadsheet or other non-contextualized approach.

References

1. Nye, B.: ITS and the Digital Divide: Trends, Challenges, and Opportunities. In: Lane, H.C., Yacef, K., Mostow, J., Pavlik, P. (eds.) Artificial Intelligence in Education, vol. 7926, pp. 503-511. Springer Berlin Heidelberg (2013)
2. Nye, B.D.: Barriers to ITS Adoption: A Systematic Mapping Study. In: Trausan-Matu, S., Boyer, K., Crosby, M., Panourgia, K. (eds.) Intelligent Tutoring Systems, vol. 8474, pp. 583-590. Springer International Publishing (2014)
3. Mark, M.A., Greer, J.E.: The VCR Tutor: Effective Instruction for Device Operation. The Journal of the Learning Sciences 4, 209-246 (1995)
4. Shute, V.J., Psotka, J.: Intelligent tutoring systems: Past, present, and future. In: Jonassen, D.H. (ed.) Handbook for research for educational communications and technology, pp. 570-599. Macmillan, New York, NY (1996)
5. Murray, T., Blessing, S., Ainsworth, S.: Authoring Tools for Advanced Technology Learning Environments. Kluwer Academic Publishers, Dordrecht (2003)
6. Murray, T.: An Overview of Intelligent Tutoring System Authoring Tools: Updated Analysis of the State of the Art. In: Murray, T., Blessing, S., Ainsworth, S. (eds.) Authoring Tools for Advanced Technology Learning Environments, pp. 491-544. Springer (2003)
7. Aleven, V., McLaren, B., Sewall, J., Koedinger, K.: The Cognitive Tutor Authoring Tools (CTAT): Preliminary Evaluation of Efficiency Gains. In: Ikeda, M., Ashley, K., Chan, T.-W. (eds.) Intelligent Tutoring Systems, vol. 4053, pp. 61-70. Springer (2006)
8. Hoffman, R.R., Shadbolt, N.R., Burton, A.M., Klein, G.: Eliciting knowledge from experts: A methodological analysis. Organizational behavior and human decision processes 62, 129-158 (1995)
9. Aleven, V., McLaren, B.M., Sewall, J., Koedinger, K.R.: A New Paradigm for Intelligent Tutoring Systems: Example-Tracing Tutors. International Journal of Artificial Intelligence in Education 19, 105-154 (2009)
10. Ainsworth, S., Major, N., Grimshaw, S., Hays, M., Underwood, J., Williams, B.: REDEEM: Simple Intelligent Tutoring Systems from Usable Tools. In: Murray, T., Ainsworth, S., Blessing, S. (eds.) Authoring Tools for Advanced Technology Learning Environments, pp. 205-232 (2003)
11. Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., Mcguigan, N.: ASPIRE: An Authoring System and Deployment Environment for Constraint-Based Tutors. Int. J. Artif. Intell. Ed. 19, 155-188 (2009)
12. Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., McGuigan, N.: ASPIRE: an authoring system and deployment environment for constraint-based tutors. International Journal of Artificial Intelligence in Education 19, 155-188 (2009)
13. Ainsworth, S., Grimshaw, S.: Evaluating the REDEEM authoring tool: can teachers create effective learning environments? International Journal of Artificial Intelligence in Education 14, 279-312 (2004)
14. Swartout, W., Artstein, R., Forbell, E., Foutz, S., Lane, H.C., Lange, B., Morie, J.F., Rizzo, A.S., Traum, D.: Virtual humans for learning. AI Magazine 34, 13-30 (2013)
15. Kim, J.M., Hill, R.W., Durlach, P.J., Lane, H.C., Forbell, E., Core, M., Marsella, S., Pynadath, D.V., Hart, J.: BiLAT: A Game-based Environment for Practicing Negotiation in a Cultural Context. Int. Journal of Artificial Intelligence in Education 19, 289-308 (2009)
16. Campbell, J.E., Hays, M.J., Core, M., Birch, M., Bosack, M., Clark, R.E.: Interpersonal and leadership skills: Using virtual humans to teach new officers. Proc. of the 33rd Interservice/Industry Training, Simulation, and Education Conference, Orlando (2012)
17. Leffingwell, T.R.: Motivational Interviewing Knowledge and Attitudes Test (MIKAT) for evaluation of training outcomes. MINUET 13, 10-11 (2006)