



# Evolving Training Scenarios with Measurable Variance in Learning Effects

Brandt Dargue<sup>1(✉)</sup>, Jeremiah T. Folsom-Kovarik<sup>2(✉)</sup>,  
and John Sanders<sup>3(✉)</sup>

<sup>1</sup> The Boeing Company, St. Louis, MO 63166, USA  
Brandt.W.Dargue@Boeing.com

<sup>2</sup> Soar Technology, Inc., Orlando, FL, USA  
Jeremiah@SoarTech.com

<sup>3</sup> US Army, Ft. Knox, KY, USA  
sandersj@holf.org

**Abstract.** One major cost driver in simulation-based training (SBT) and Intelligent Tutoring System (ITS) development is authoring scenario content. Effort is compounded when creating a catalog of scenarios to enable optimal scenario selection for the individual learner. Automated Scenario Generation (ASG) methods have been successful at creating variants of scenarios using procedural generation, evolution, or event templates with variable parameters. These methods typically describe boundaries within which content may vary, but do not describe how the variation will affect learning.

The authors developed an evolutionary algorithm approach to generate variations in training scenario content that can integrate with the Generalized Intelligent Framework for Tutoring (GIFT) being developed by the Army Research Lab (ARL). One of the primary goals of the approach is to greatly increase not just surface variations that do not affect learning, but variations on scenario content and instructional feedback that make the scenarios measurably different as described by multidimensional learning effect metrics. The approach leverages a structured cognitive task analysis (CTA) which outputs a set of scenarios in a schema that directly drives the ITS. A small change to the CTA identifies dimensions and constraints of variation and the effect on learning from the variations. Both manual and machine learning approaches are being explored.

The key contributions of our research are (a) developing domain-general computational measures of effect on learning for creative variation, (b) making evolution effective for many teaching and tutoring domains by evolving textual content in addition to event sequences, and (c) leveraging human expert input and usage data to influence and feed back into evolution.

This paper summarizes the need for ASG, describes the novel approach, and shares the key contributions of our research.

**Keywords:** Adaptive instructional system · Automated Scenario Generation · Scenario-based learning

---

J. Sanders—US Army (ret), Ft. Knox, KY, USA.

## 1 Introduction

Intelligent Tutoring Systems (ITS's), scenario-based training, and simulation-based training can be very effective [e.g. (NTSA 2018; Kulik and Fletcher 2016; VanLehn 2011)]. They are also time-consuming to develop, test, and modify. The extra time and cost results in reduced use of training using these methods, relatively few choices being created, and existing scenarios stagnating. In nature, systems improve and adapt to changes such as the environment they are in through methods such as evolution. Likewise, a training system should evolve to keep it up to date, to adapt to new situations, and to enable the system to provide a better learning experience. In nature, the variant that survives evolved in a way that enables it to overcome the problem. For training, where the goal is effective learning, it should be the instructional value of the variant scenario that determines its worth. We are prototyping two methods for evolving. One is by design, and one by random mutations. The scenarios generated by both methods are assessed by their instructional value, so determining the instructional value is a primary focus of this paper.

The ability to evolve scenarios provides a method for what is known as Automated Scenario Generation (ASG). Busch et al. (1995) investigated and validated the ability to auto-generate instructional tutoring for maintenance training based on a summary of the state-of-the-problem solution in different ways, suggesting a strategy, suggesting a system or component that still needs to be tested, or suggesting a specific test. Auto-generating these multiple levels of mentoring enables scenarios that are automatically generated to be used instructionally without any further human effort. A second method of evolving new scenarios is by creating novel sequences of scenario segments or a specific sequencing of events within a scenario (Zook et al. 2012). Dividing the scenarios into segments that are natural for the domain make the segments effective learning experiences when used standalone, when assembled into complete stories, or when assembled with short transitional segments that keep the learner engaged. Using natural scenario segments also provides for an intuitive framework to lower the effort of manual authoring and provides entry points for more focused practice in the specific segment. This method was applied when developing scenarios for the SUAS COMPETE (Small Unmanned Aviation Systems Company Employment Training Exercises) adaptive training prototype (Durlach and Dargue 2010) which was selected for use in this current project. Selecting scenario individual phases is a typical method used for skill practice. Both commercial pilot and tactical pilot students practice the more difficult landing and take-off phases of flight more than cruise phases. One of the great benefits of flight simulators is that students can repeatedly practice landings without having to taxi, take off, and perform approach phases. Musicians greatly benefit from repeating specific sections of a musical score without having to repeat the entire score. Similarly, athletes such as baseball players hone their skills by practicing specific portions of their sport such as batting. These methods of ASG are effective at providing unique, tailored learning experiences for individuals or teams. However, they require substantial authoring effort to create the individual building blocks that are assembled into the unique scenarios. This project set out to alleviate the authoring burden by automatically generating the individual segments or the entire scenario. We also set out to explore and

validate domain-general methods to create novel scenario variants in less predictable ways. These methods are still instructionally sound, and better prepare learners to be more adaptive, resourceful, and innovative.

### 1.1 The Theory of (Scenario) Evolution

Training scenarios are authored to intentionally provide a learning experience for specific learning objectives. If not done correctly, a variant could very easily reduce the value of that learning experience. A primary objective of this project is to increase, or at least maintain, the instructional value of the training exercise. To avoid superficial or counterproductive variants, ASG systems need to understand the domain. Novelty Search helps alleviate the domain dependency by rewarding how unique a variant is rather than rewarding progress toward a goal (Stanley and Lehman 2015b). The solution is still measured against the goal, but the *distance* from the goal is not the primary measure of *value/worth*. If the *strain/branch/variant* is unique and still within bounds, then it is worth keeping and worth exploring (evolving) further. Novelty Search then needs two litmus tests for each variant: a determination of whether it is within bounds and a measure of novelty or uniqueness from other variants. Both of these measures are domain-specific.

The visual demonstration (Stanley and Lehman 2015a) on the Novelty Search Users Page is in the domain of solving a maze. The goal is to reach the end of the maze, and the demonstration shows how the typical approach of focusing on proximity to the goal can lead to failure. For Novelty Search, the demonstration uses the maze walls as bounds and distance from previous mutations rather than distance to goal. In a domain where, for example, a student needs to learn how to create a smoke screen to hide from an enemy, the measures of novelty might be position of the student, position of the enemy, and wind direction. Boundaries for this exercise might be that the wind direction can only be the eight primary compass directions and another measure is minimum and maximum distance between student and enemy. For this project, we want a measure of novelty to include instructional value. An instructional value of novelty for the smoke exercise should be something like the angle between the vectors of wind and line-of-sight to the enemy, and distance to the goal along that line-of-sight, as these are the factors that the student must use to determine the correct placement of smoke.

### 1.2 The Goal of This Paper

Mager (1962), emphasized that a learning objective needs to be written as a clear description of competent performance so that it is clear how to generate instruction for, and assessments of, those learning objectives. For the smoke lesson described above, a learning objective written in this format might be “Given a terrain map showing position of the learner’s unit, positions of the enemy, position of the goal, and weather information including wind direction and speed, the learner will be able to select the target location for smoke grenades and number of grenades within a given tolerance to sufficiently keep the enemy from seeing the learner’s approach toward the goal.” With this information, a developer can determine how to evolve the scenarios to provide practice and assessment of that objective. The goal of this project is to enable software

to automatically generate a set of scenarios. In addition to the mechanism/algorithm that performs the automation, we need a method to describe to the ASG system how to automate the generation and how to assess the learner's attempt. We also need to make this domain-neutral. This paper focuses on this aspect of defining the learning objectives in such a way that the scenario meets the learning objectives (the instructional value can be determined). For more information about how the authors are implementing Novelty Search for this project, see Folsom-Kovarik and Brawner (2018).

## 2 Training Scenario Variability

The US Army scaffolds a Soldier's training using a methodology known as crawl-walk-run (Headquarters, United States (U.S.) Army Combined Arms Center 2016), which conducts training events sequenced in progressive levels of difficulty. Intelligent tutoring systems such as the US Army Research Lab's Generalized Intelligent Framework for Tutoring (GIFT) (Sottolare et al. 2012) feature the ability to scaffold the learner by sequencing content in such a progressively more difficult order. These systems can automatically select a training exercise (scenario) from a pool of exercises based on the learning complexity of the exercise. A single dimension of complexity is often sufficient within a specific learning objective. For example, the difficulty factor for adding two positive integers might be based on whether or not the sum of any of the digits is greater than 9 so that the learner has to perform the concept of carrying. However, within the larger learning objective of adding two numbers, there are multiple additional dimensions of complexity, such as whether or not there are negative numbers and whether there are fractions. To properly challenge the learner, the complexity should increase along the appropriate dimension.

Simulation scenarios are much more complex than the simple math example given. For example, the scenarios used for this project involved 49 enabling learning objectives that address nine higher-level concepts. Additionally, there are three distinct phases to the scenarios where the learner must plan for the mission, prepare for the mission, and execute the mission. Each of the SUAS COMPETE scenarios use over 300 multi-dimensional learner performance measures at approximately 45 discrete decision points to precisely determine the learner's mastery of the learning objectives. Evolving the scenario within specific learning objective dimension will enable the system to generate variants that provide learning experiences and practice tailored to a learner's particular needs. However, these learning objectives are domain-specific, so we need more generic dimensions of variability.

### 2.1 Scenario Complexity

Dunne et al. (2015) developed a tool to measure a simulation scenario complexity based on three characteristics of the scenario: Task Complexity (TC), Task Framework (TF), and Cognitive Context Moderators (CCM). Each characteristic is based on factors about a given scenario, such as the number of cues, actions, subtasks across actions, interdependent subtasks, possible paths, criteria to satisfy, conflicting paths, and distractions (see Table 1, Scenario Characteristics).

**Table 1.** Scenario characteristics

|  |
|--|
| <ul style="list-style-type: none"> <li>• <b>Task Complexity</b> <ul style="list-style-type: none"> <li>– Component Complexity           <ul style="list-style-type: none"> <li>• Number of <i>Tasks/Actions</i> required</li> <li>• Number of <i>Information Cues</i> that must be monitored and assessed</li> </ul> </li> <li>– Coordinative Complexity           <ul style="list-style-type: none"> <li>• Number of <i>Sub-Tasks</i></li> <li>• Number of <i>Inter-Dependent Tasks</i></li> </ul> </li> </ul> </li> <li>• <b>Task Framework</b> (How well defined is the domain)           <ul style="list-style-type: none"> <li>– Task Paths – Presence of multiple potential ways to accomplish outcome(s)</li> <li>– Task Criteria – Number of desired outcomes</li> <li>– Unknown/Conflicting Paths – Degree of uncertainty</li> </ul> </li> <li>• <b>Cognitive Context Moderators</b> <ul style="list-style-type: none"> <li>– Distraction Factor</li> </ul> </li> </ul> |
|--|

Dunne et al. summed these dimensions into a single complexity factor for the scenario and validated with domain experts. The single value attribute of complexity allows the instructor to select the scenario with the proper level of difficulty for individual learners. It also allows analysts to verify that they have authored enough scenarios with the proper coverage of complexity levels to enable the US Army’s crawl-walk-run methodology. Providing the right level of challenge enables the learner to “get into the flow of optimal learning” described by the psychologist Csikszentmihályi (1990). His research discovered three conditions required for getting into the state of optimal performance and learning. First, the task must be a challenging activity with a clear set of goals and a measure of progress toward those goals. Second, there must be immediate feedback to the person about the progress in those measures. Third, the person must perceive the level of challenge of the task is aligned to one’s own perceived abilities. In other words, the learner must believe that the task is difficult and requires effort, the learner must be able to access and interpret the cues that are required to measure progress toward the goal, and the learner must believe that achieving the goal is possible. The ITS used for this project—like the Sherlock ITS it was based on (Lesgold et al. 1988; Lesgold and Nahemow 2013)—prepares the learner for the task through instruction, then immerses the learner into the task-based scenario. Sherlock further encouraged the learner that she/he had the skills, and that the system would provide help if needed.

Keeping the measure of complexity for each characteristic or even each base variable separate will help tailor the experience more precisely for the individual learner. For example, a particular learner may not have any problem with the number of cues or distractions, but needs more practice in scenarios that have a high number of interdependent subtasks. Individual measures of each dimension for the scenarios and for the learner’s mastery will enable the scenario selection mechanism to select the appropriate scenario for the learner. It may not be practical to directly measure the learner’s level of proficiency in each of these complexity dimensions independently in the same way as measuring proficiency in each learning objective. Rather than

measuring directly, analytics of the learner's performance in the scenarios can provide insight to the learner's competency in the individual factors.

A training scenario is a series of situations, events, decisions, actions, and tasks. Some dimensions of complexity are only valid for specific phases or tasks within the scenario. For example, factors that vary the complexity of scenarios for piloting an airplane include length and direction of runway, cross-wind, and wind shear. However, those factors only influence the complexity of the tasks performed during takeoff and landing, whereas route changes only add complexity during cruise or approach. As shown by experiments by Biddle et al. (2006), allowing distinct phases to be selected independently from a pool of variants will allow for more focused practice for the individual needs of each learner. The example of smoke above is a distinct phase of an overall scenario breaching a minefield. For that scenario, new elements might be wind speed, multiple enemy positions, uncertainty of enemy position, and the position for the goal to which the student/unit must move. Therefore, the complexity factors and other instructional value measures should be individually computed and associated with each distinct phase of the scenarios.

Any LMS that is compliant to SCORM 2004 or AICC can select scenarios based on measured gaps in a learner's expertise of explicit learning objectives (Perrin et al. 2004; Biddle et al. 2006). However, the scenario selection algorithms need to be able to determine which scenarios focus on which learning objectives. Typically, scenarios are intentionally authored for explicit learning objectives (LOs). For ASG, either the system similarly needs to generate scenarios explicitly for the specific LOs or there needs to be a method to determine which LOs a student will encounter in each of the machine-generated scenarios.

### 3 The Approach

To address the needs of generating scenarios for specific instructional value or generating using Novelty Search then calculating the instructional value, we are leveraging the SUAS COMPETE scenario XML files. The structure of those scenarios comprise optional paths through a series of situations, events, decisions, actions, and tasks. Therefore, the scenario complexity can be autonomously determined by software. Additionally, for each action along the paths in the scenario XML there are individual measures for each of the 49 LOs. Although the LOs are domain-specific, the format of the LO measures in the XML is domain-independent.

#### 3.1 Dimensions of Evolution and Instructional Value

We defined specific ways in which the scenario can evolve as dimensions. We looked at specific dimensions for the scenarios we considered and dimensions that were envisioned to be generic and thus applicable across domains. As one method to validate the dimension as generic, we considered domains such as corporate leadership, factory workers, engineering, and compliance training.

The domain-generic dimensions of complexity were validated with the subject matter expert (SME) who authored the original scenarios. LTC John Sanders (US Army, retired), whose specialty areas include combined arms, authoring tactical operations doctrine, and advanced military instruction, helped define the roles of SUAS. He agreed that those factors measure the complexity of learning, and, in general, also measure the complexity of the tasks within the scenarios. The relevant effect of each measure varies based on the phase of the scenario. Additionally, the effect is dependent on the specific task or decision within the phase.

LTC Sanders, the SUAS COMPETE SME, previously worked with the author to define three dimensions of complexity for military scenarios (see Fig. 1) (Sanders and Dargue 2012). These factors include the level of threat, complexity of the task or system used in the task, and environmental factors. The level of threat dimension could be expressed in a domain-generic way as level of risk or urgency. The eight dimensions defined by Dunn et al. discussed earlier are more specific and therefore can be classified in a hierarchy under those defined by Sanders and Dargue (see Table 2).

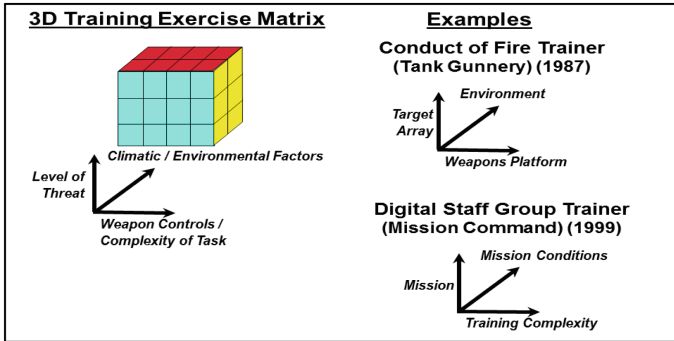


Fig. 1. Progressive training matrix in three dimensions from (Sanders and Dargue 2012)

Table 2. Three dimensions of scenario complexity

|   |
|---|
| • Level of risk or urgency                          |
| – Number of criteria to satisfy                     |
| – Number of conflicting paths                       |
| • Complexity of the task or system used in the task |
| – Number of actions                                 |
| – Number of subtasks across actions                 |
| – Number of interdependent subtasks                 |
| – Number of possible paths                          |
| • Environmental factors                             |
| – Number of distractions                            |
| – Number of cues                                    |

### 3.2 CTA and Authoring

The scenarios we are using were authored using a Cognitive Task Analysis (CTA) originally developed to capture knowledge of maintenance technicians for an ITS. The Precursor, Action, Results, Interpretation (PARI) (Hall et al. 1995; Means and Gott 1988). The PARI CTA is typically a structured interview process using two SMEs. Using a spreadsheet we enabled a single SME to perform PARI (Dargue and Biddle 2016). For each task, decision, or action to be performed in the scenario, the spreadsheet codifies the SME's mental model and four possible actions that can be performed.

The SME ranks each of the four possible actions and defines changes to the assessment of the learner's level of expertise for each learning objective. For each decision made in the scenario, the learner model is dynamically updated using these assessments. A unique, inherent capability is that for any given decision point, different learning objectives are scored based on which decision is made at that point. For example, if the situation requires the learner to select the proper tool to tighten a hex bolt, and the learner selects the proper wrench, she will get credit for understanding when to use a wrench. If she had selected a hammer, her measure of expertise in both wrenches and hammers will be reduced. Another feature of the PARI CTA is that the decisions are made in a specific context of other decisions. Each phase of the scenario comprises a series of tasks, actions, or decisions. The expert's path is

defined by the expert's decisions plus the outcomes or results of those decisions. To fully define the scenario, the tool is used to codify the results of each possible decision, how the mental model changes, and what new decisions can be made. This can be done by the same SME or multiple SMEs. In this manner, multiple paths with *continuity* in branching are defined for the scenario. One simple method to generate a scenario variant is to reduce the complexity of the scenario by removing optional or conflicting paths (Fig. 2).

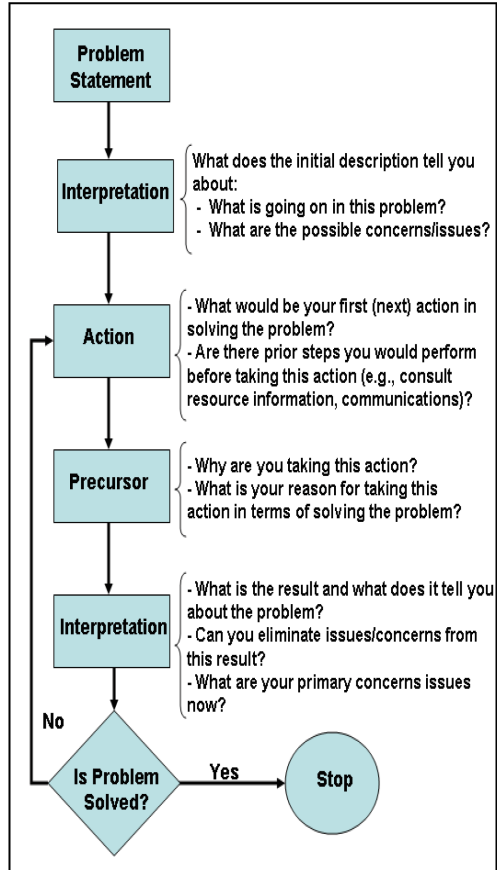


Fig. 2. PARI CTA process



A second method to generate a scenario variant for a specific learner is to modify the scenario to change the situations so that different decisions/actions should be made. A search through the decision points can determine which path will present the decision points that the learner needs to experience to target specific gaps in expertise. If the path is not on the optimal path or a path that the learner is likely to take, the software will need to know how to vary the situation at one or more decision points.

The problem statement and the precursor for each decision point captured by PARI contain the situation and the mental model of the expert. These are also the cues that are counted to determine scenario complexity. Currently, this information is free-form text in the XML describing the expert's thoughts for the precursor for the decision and the interpretation of the results of the decision. While this is effective for transferring the expert mental model to the learner, it is not easily understood by the software nor is it easily modified by the software. A relatively simple addition to the scenario authoring tools will capture the cues used by the expert for each decision in a machine understandable and variable way. Often the factor is an indication of cost or risk for that choice and is therefore weighed by the expert for making the decision. The addition to PARI for requesting the author to define the few factors of complexity that are not currently computable will make that critical information explicit for the software. With that information, the software can determine what level of challenge each decision point has and how that challenge level can be changed.

With the cues explicitly defined, the software can determine which factors need to be changed to make a less than optimal or even unacceptable path the optimal choice. The software will also be able to calculate the new complexity measure and be able to make changes to ensure the right level of challenge is given to the learner. Returning to the wrench/hammer decision, if the learner has already demonstrated mastery of selecting a wrench for hex bolts, we might want to change the number of conflicting paths complexity factor by adding a socket wrench, pliers, and a pipe wrench in addition to the original crescent wrench. We might change the number of actions complexity factor by stating there is a nut on the other side and that the bolt has to be torqued to a standard they have to calculate. If the learner has not demonstrated proficiency in hammers and nails, the software could change the fastener to a hammer.

### 3.3 Scenario Continuity

In many cases, such as our fastener example, the path from making the correct choice does not change: the path only depends on whether the learner selected the correct decision. For the fastener, when the correct tool is used, regardless of whether it was a nail or a bolt, the fastener is properly fastened. If the user selects the wrong tool, the fastener will work loose. In other cases, decision points further down the scenario path might need to reflect the change.

We also run the danger of making a variant that is nonsensical or impossible. For example, if the fastener we change to a nail is holding the wheel on a car. To avoid those cases, bounds need to be defined. A straightforward solution is to have the author specify the bounds or possible variants. In many cases, this is a natural extension to the

addition that defines the cues used by the expert to make the decision. For domains such as military tactical decision making, discontinuity is inherent. Uncertainty caused by the “fog of war” and unexpected changes from adaptive threats and deceptive techniques used by the enemy make discontinuity in the scenario more realistic and better prepares the learner for warfare.

Variants made by adjusting the number of cues, paths, subtasks, or outcomes do not present a risk of discontinuity or nonsense. The paths created by the original authors and encoded in the original scenario provide scenario continuity for all the possible paths, including the conflicting paths, as the scenario evolves or unfolds based on the decisions and actions performed by students in the simulation.

## 4 Conclusions and Recommendations for Future Research

This project consists of researching, prototyping, and evaluating methods to autonomously evolve variants of simulation scenarios to be used in adaptive training systems. The primary purpose for generating a variety of variants is to provide a library of scenarios so that the optimal learning experiences can be selected for students. There are basically two approaches with slight alternatives being prototyped. The first approach is to use information within the scenario to purposely create variants. The second approach is to generate a large set of variants that are then analyzed for instructional value.

For the first approach, the existing scenario format in XML includes enough information to intelligently make a small set of variants for specific instructional outcomes. By adding information, such as cues an expert uses to make decisions to the scenarios, the software can generate a larger set of tailored scenario experiences that are focused on a greater number of different instructional outcomes. Both approaches require research and validation of methods to autonomously assess the instructional value of scenario variants. The first uses the variables of the assessment as guides in making the variants.

The second approach uses the assessment as a “litmus test” of the instructional worthiness of scenario variants. The variables used by Dunne et al. to determine scenario complexity provide what may be an ideal domain-independent method to determine the instructional value of a simulation scenario variant. Those variables also provide information that may be used to make specific variants to address an individual learner’s needs.

**Acknowledgements.** The research reported in this document/presentation was performed in connection with contract number W911NF-18-C-0005 with the U.S. Army Contracting Command - Aberdeen Proving Ground (ACC-APG). The views and conclusions contained in this document/presentation are those of the authors and should not be interpreted as presenting the official policies or position, either expressed or implied, of ACC-APG, U.S. Army Research Laboratory or the U.S. Government unless so designated by other authorized documents. Citation of manufacturer’s or trade names does not constitute an official endorsement or approval of the use thereof. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

## References

- USARMYCOMBATIVES.COM. 2-10 Crawl, Walk and Run. U.S. Army FM 3-25.150 Combatives (2018). <http://www.usarmycombatives.com/2-10-crawl-walk-and-run/>
- Biddle, E., Perrin, B., Dargue, B., Lunsford, J., Pike, W., Marvin, D.: Performance-based advancement using SCORM 2004. In: Proceedings of the 2006 Interservice/Industry Simulation, Training, & Education Conference (IITSEC), Orlando (2006)
- Busch, D., Dargue, B., Perrin, B.: ASTUTE: an architecture for intelligent tutor development. In: Interservice/Industry Training, Simulation, and Education Conference (IITSEC), Albuquerque, New Mexico, pp. 672–681 (1995)
- Csikszentmihályi, M.: Flow: The Psychology of Optimal Experience. Harper and Row, New York (1990)
- Dargue, B., Biddle, E.: Chapter 13 – Mining expertise: learning new tricks from an old dog. In: Sottolare, R., Graesser, A., Hu, X., Olney, A., Nye, B., Sinatra, A. (eds.) Design Recommendations for Intelligent Tutoring Systems: Volume 4 - Domain Modeling, pp. 147–157. U.S. Army Research Laboratory, Orlando (2016). [https://gifttutoring.org/attachments/download/1736/Design%20Recommendations%20for%20ITS\\_Volume%204%20-%20Domain%20Modeling%20Book\\_web%20version\\_final.pdf](https://gifttutoring.org/attachments/download/1736/Design%20Recommendations%20for%20ITS_Volume%204%20-%20Domain%20Modeling%20Book_web%20version_final.pdf)
- Dunne, R., Sivo, S.A., Jones, N.: Validating scenario-based training sequencing: the scenario complexity tool. In: Interservice/Industry Training, Simulation, and Education Conference (IITSEC), Orlando, pp. 2886–2898 (2015)
- Durlach, P., Dargue, B.: An adaptive training prototype for small unmanned aerial system employment. In: The 23rd Florida Artificial Intelligence Research Society Conference (FLAIRS-23). Daytona Beach, FL (2010)
- Folsom-Kovarik, J.T., Brawner, K.: Automating variation in training content for domain-general pedagogical tailoring. In: 2018\_05\_Proceedings of the Sixth Annual GIFT Users Symposium. U.S. Army Research Laboratory, Orlando (2018). [https://gifttutoring.org/attachments/download/2700/17\\_GIFTSym6\\_Authoring\\_paper\\_18.pdf](https://gifttutoring.org/attachments/download/2700/17_GIFTSym6_Authoring_paper_18.pdf)
- Hall, E.P., Gott, S.P., Pokorny, R.A.: The PARI methodology. Technical report 1995-0108. Brooks AFB TX: Armstrong Laboratory, Human Resources Directorate (1995)
- Headquarters, United States (U.S.) Army Combined Arms Center. FM 7-0, Train to Win in a Complex World. Fort Leavenworth, KS (2016). [https://rdl.train.army.mil/catalog/search?current=true&search\\_terms=%23TrainingManual](https://rdl.train.army.mil/catalog/search?current=true&search_terms=%23TrainingManual). Accessed 22 June 2018
- Kulik, J.A., Fletcher, J.D.: Effectiveness of intelligent tutoring systems. *Rev. Educ. Res.* **86**(1), 42–78 (2016). <https://doi.org/10.3102/0034654315581420>
- Lesgold, A., Nahemow, M.: Tools to assist learning by doing: achieving and assessing efficient technology for learning. In: Klahr, D., Carver, S.M. (eds.) *Cognition and Instruction*. Psychology Press (2013)
- Lesgold, A., Lajoie, S., Bunzo, M., Eggen, G.: SHERLOCK: A Coached Practice Environment for an Electronics Troubleshooting Job. University of Pittsburgh, Learning Research and Development Center, Pittsburgh (1988)
- Mager, R.F.: *Preparing Instructional Objectives* (Revised Second (1984) ed.). David S. Lake Publishers, Belmont (1962)
- Means, B., Gott, S.P.: Cognitive task analysis as a basis for tutor development: articulating abstract knowledge representations. In: Psozka, J., Massey, D., Mutter, S. (eds.) *Intelligent Tutoring Systems: Lessons Learned*. Erlbaum, Hillsdale (1988)
- NTSA. Why Use Simulation? Return on Investment. NTSA - National Training and Simulation Association (2018). [http://www.trainingsystems.org/publications/simulation/roi\\_effect.cfm](http://www.trainingsystems.org/publications/simulation/roi_effect.cfm). Accessed 21 June 2018

- Perrin, B.M., Banks, F., Dargue, B.W.: Student vs. software pacing of instruction: an empirical comparison of effectiveness. In: The Proceedings of the 2004 Interservice/Industry Training, Simulation, and Education Conference, Orlando, FL (2004)
- Sanders, J., Dargue, B.: A design whose time has finally arrived: use of adaptive training technology for staff training. In: Presentation at the 23rd International Training Equipment Conference (ITEC), London (2012)
- Sottolare, R., Brawner, K., Goldberg, B., Holden, H.: The Generalized Intelligent Framework for Tutoring (GIFT). US Army Research Laboratory Human Research & Engineering Directorate, Orlando (2012)
- Stanley, K., Lehman, J.: Novelty search demonstration. The Novelty Search Users Page (2015a). <http://eplex.cs.ucf.edu/noveltysearch/userspage/demo.html>. Accessed 3 June 2018
- Stanley, K., Lehman, J.: The novelty search users page. The Evolutionary Complexity (Eplex) Research Group at the University of Central Florida (2015b). <http://eplex.cs.ucf.edu/noveltysearch/userspage/index.html>. Accessed 3 June 2018
- VanLehn, K.: The relative effectiveness of human tutoring, intelligent tutoring systems and other tutoring systems. *Educ. Psychol.* **46**(4), 197–221 (2011). <https://doi.org/10.1080/00461520.2011.611369>
- Zook, A., Lee-Urban, S., Riedl, M.O., Holden, H.K., Sottolare, R.A., Brawner, K.W.: Automated scenario generation: toward tailored and optimized military training in virtual environments. In: Proceedings of the International Conference on the Foundations of Digital Games, pp. 164–171. ACM (2012)