Intelligent Tutoring Systems for Commercial Games: The Virtual Combat Training Center Tutor and Simulation

William R. Murray

Boeing Phantom Works
P.O.Box 3707, MC 7L-66
Seattle, Washington 98124-2207
william.r.murray@boeing.com

Abstract

Game manuals and tutorial scenarios are insufficient for new players to learn games of deep complexity such as highly realistic tactical simulations of modern battlefields. Adding post-game after-action reviews improves the situation, but these typically do not provide guidance during the mission and tend to focus on quantitative feedback, rather than specifics about what the player did wrong and how to improve. Intelligent tutoring system (ITS) technology provides a higher level of interactivity and a more specific *qualitative* analysis to guide players *during* game play. This use of an AI technology is demonstrated with the integration of an ITS component with the tactical simulation *Armored Task Force* (ATF) resulting in a combined system called the the Virtual Combat Training Center (V-CTC).

V-CTC simulates the Army's combat training center at Fort Irwin and its instructors, called observer / controllers. The ATF game itself was modified to send an event stream over TCP-IP sockets to the ITS component, which interprets the events and acts accordingly. V-CTC was originally intended for a military context: either classroom use, field instruction, or embedded deployment. However, in non-military games, tutors (or non-player characters acting in that role) may well enhance the gaming experience of players. Such players might otherwise become frustrated with learning very challenging games, or simply fail to appreciate the tactical possibilities and depth of strategy possible in a well-designed game.

The Need And Opportunity

Armored Task Force is an example of a game that might at first appear simple but turns out to be staggeringly complex. Its graphically simple two-dimensional interface has none of the breathtaking graphics that commercial players are used to seeing; indeed, the game appeals primarily to hardcore war gamers who want the most realistic modern war game possible today. The bare-bones interface hides a simulation of fire support and combined-arms warfare at the battalion and company level for armored task forces. Tactical exchanges are modeled using the same pH (probability of hit) and pK (probability of kill) parameters that the Army's National Training Center uses. The game uses the same digital terrain elevation contour maps that the Army uses. It displays military units and vehicles either with NATO standard icons or markers showing vehicle depictions. Actually mastering the game can take a lifetime as it requires mastering the same tactical skills that the Army teaches in its field manuals. Consider, for example, that in the tutorial mission a player needs to know the missile range capabilities of the following enemy force vehicles: a BRDM-2 ATGM, a T-80U, and a BMP-2. Not knowing what these vehicles are, or their capabilities, or how to suppress or destroy them with artillery, will most likely result in mission failure.

With this much complexity, most users will never appreciate the depth and scope of the game, unless, they, too, are military professionals. One way to lessen this gap and reach a wider audience is for a tutor to provide tactical assistance. Even military professionals will be challenged in learning to coordinate artillery, armor, infantry, and air to arrive at the same time and the same place for a successful assault (the "combined-arms" part of "combined-arms warfare").

This kind of tutor is different from the use of a tutorial scenario of a game manual intended to teach the basics of game play. Those techniques provide enough to make the user "dangerous." This tutor instead takes the user much farther along the road toward seeing the full vistas of what the game offers, e.g., how fire support is used to suppress targets in ATF, how smoke is used in breaching a minefield, and by showing what expert solutions look like for difficult missions that the user has attempted, but only after the player has tried them.

The tutor also tracks the kinds of errors the player makes, notices patterns, and suggests FM (field manual) references that address them, or presents mini-tutorials (small collections of web pages) that directly address those mistakes.

An Example

We will show a running example to illustrate how the tutor assists the user and the kinds of help it is currently implemented to give. The tutor is available to government workers at no charge as an overlay (patch) to *Armored Task Force*, which may be purchased over the Internet at www.prosimco.com.

One of the ATF scenarios is called Crash Hill Defense. In this scenario, you, the commander, are defending the Brown-Debnam passes at the National Training Center (NTC) from two waves of enemy forces, attacking first with two battalions of armor and then with one follow-on battalion.[1]

Before the first wave of enemy battalions descends, a combat reconnaissance patrol (CRP) is sent out. Let us suppose our player, who takes the role of the lieutenant-colonel (LTC) of the defending battalion, targets the CRP lead tank with BBDPICM (base-burn dual-purpose improved conventional munitions) from the 3 M109A6 howitzer batteries at his disposal.

There are several problems with this action that most players would never find out about without tutor feedback, or another player's helpful guidance. In this case, the tutor pauses the simulation and pops up the guidance shown in Figure 1.

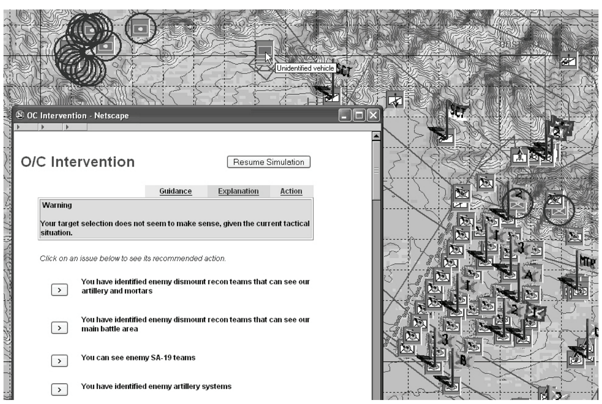

Figure 1. Tutor advice with suggested targets highlighted by circles

The tutor points out that the most important targets to dispatch now are the enemy observer targets within his own backfield. Why? These observers are calling in enemy infantry and the friendly units are dug in, i.e., they have no place to go, or if they do displace, they will lose their defensive advantages. First the enemy will take out the friendly fire control units, disabling friendly artillery, and then other high value targets such as Bradley Linebackers (air defense units).

Next, the player takes the advice and dispatches the observers with 1 BN HE (1 battalion high-explosive, or 18 M106A9 rounds each).

Let's look a little later, when he sees enemy tanks coming down an inlet to the side and he decides he would like to block them by firing 3 BN FASCAM (3 volleys of 1 battalion, i.e., 18 guns, of FASCAM[2]). Again, the tutor intervenes, but this time presenting a mini-tutorial on when it is and is not appropriate to use FASCAM. Now is not a good time, as it will take too long to fire the field of mines and the tanks will have slipped by. Figure 2 shows just the first part of this mini-tutorial.

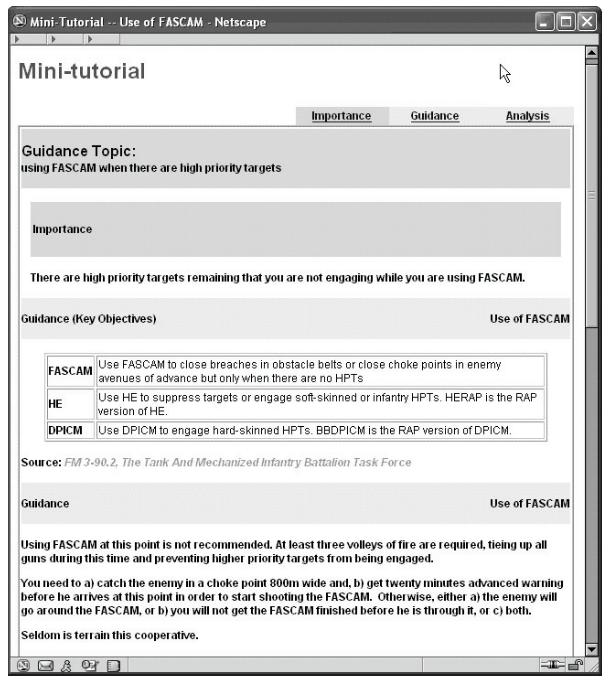

Figure 2. A mini-tutorial on using FASCAM

From these examples, it might appear that all assistance pop-ups during the game and much of it is concerned with fire support. What we do not have space to show here is the AARs (after-action reviews), which include detailed solutions for each mission from the subject matter expert, along with an analysis of the pattern of

[1] Most scenarios at NTC before Iraq were armor-heavy. Now many have been reconfigured for low to medium-intensity-warfare.

[2] FASCAM stands for family of scatterable munitions. Basically, cluster munitions. Yes, they can be fired from a gun, too.

errors the user made, and recommendations for improvement.

As an example of the kind of sophistication such a tutor can provide, V-CTC provides four scores (e.g., see Figure 3) for each fire mission in the AAR:

1. *Target selection*—how appropriate the user's target selection is, compared to that required for the current phase of the maneuver plan, which V-CTC tracks.
2. *Munition selection*—given a choice of target, was the type of munition and amount selected sufficient for the intended goal (destroy or suppress target)?
3. *Observation planning*—given the enemy's high value targets, how appropriate is the placement of the player's scouts, COLTs (combat observation and lasing teams), and FIST-Vs (fire support team vehicles)?
4. *Observer survivability*—have the observer assets been placed in locations where they can survive long enough to be useful?

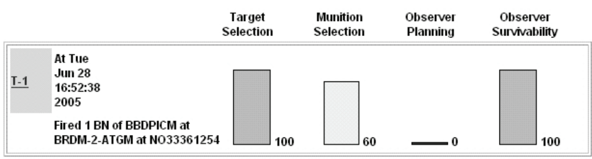

Figure 3. Feedback for one fire mission in the AAR

A mini-tutorial (see Figure 4) may be triggered by poor observation planning or by unacceptably low observer survivability, e.g., if too many scouts are destroyed.

These examples illustrate how the tutor assists the player tactically in the way a more expert guide would. The advice is, in fact, modeled after the advice that an observer / controller would give at the Army's National Training Center. Similarly, for non-military games, one can also imagine simulating the advice of more experienced or expert players.

Integration

Integration proceeded by modifying the game *Armored Task Force* to send out an event stream corresponding first to its 'spot reports' and then to include unseen tactical events, too. The simulated instructor needs access to additional information that the simulation itself may not normally need, such as the set of vehicles that one side or the other can see at any time.

The tutor component and the simulation are connected via TCP-IP sockets. A simple protocol was defined for communication. We did not use XML but would in a reimplementation.

The tutoring system is separate from the simulation and designed to be reusable for other PC-based tactical simulations although that was never tested. The main components (student model, domain knowledge, and tutor strategies) are modular. The implementation is in SWI-Prolog.

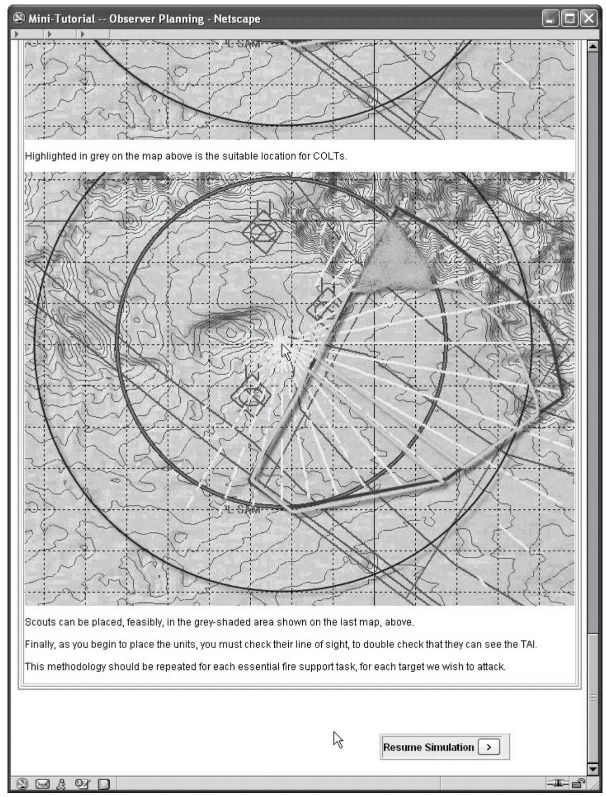

Figure 4. Last page of mini-tutorial on observer planning

Other architectural features built in to promote reusability were the use of an ontology; the use of GUI templates for mini-tutorials, and tutorial pop-ups; the use of re-usable tutorial strategies common to multiple tactical applications, such as common AAR and pop-up strategies; and most of all, the explicit separation of the tutor component and the simulation itself.

Unforeseen benefits of leveraging an existing simulation were the ability to use the simulation's scenario editor to tailor scenarios, the ability to ask questions of an online community, and the ability to incorporate patches to the simulation as it developed in parallel to the tutor. A ready-made scenario library, as ATF provided, is a big savings in curriculum development, too. Typically, most intelligent tutoring systems take a team of individuals working over several years to develop. This ITS, while currently only at a beta-test level, has only taken approximately 18 man-months of work, i.e., about 1.5 FTE of resources to complete. Most of the savings have come from leveraging the several years of prior work that have gone into the existing *Armored Task Force* simulation, including its existing NTC scenarios.

Technical Approach

Most computer-based tutoring systems build student models based on exercises such as multiple-choice, drag-and-drop, and fill-in-the-blank exercises. While these types of measures are easily collected, they emphasize recognition over performance and do not provide a full assessment of student state. Not only is it important to have *knowledge about* a domain, but one must also be able to *apply those skills* necessary to perform the tasks and do so with confidence. In contrast, a typical intelligent tutoring system differs from these more traditional computer-based tutoring systems by being able to analyze the student's solution, either as it is being developed, or after it has been composed.

The classic ITS is a knowledge-based system that models the domain, the user, and the teaching process. For V-CTC the domain is the tactical domain, the user is the player, and the teaching process is the advice and AARs provided by an observer / controller. Since we are monitoring the player's war fighting in a simulated tactical domain, we are essentially layering one simulation (of an O/C) over another (of the battlefield).

This paper will not provide the level of technical detail about the operation of the intelligent tutoring system that is discussed in depth in the project's final report (Murray, 2005) or the prior I/ITSEC report on V-CTC (Murray and Sams, 2004). Instead, we will provide a brief overview of the most important aspects.

ATF is a real-time simulation (1X, 2X, 4X, or 8X of battle real-time). This real-time aspect is very important in helping trainees acquire an intuitive feel of how fast the battlefield changes and in learning how to synchronize different battle operating systems such as artillery and armor.

As mentioned, the simulation has been modified to provide an event stream to the V-CTC tutor via a TCP-IP connection. The tactical events that the tutor receives are encoded in the simulation event stream as ASCII strings. The tutor needs to be able to understand what this data means. First, words and meanings are identified using natural language parsing techniques to recover syntactic and semantic structure. With this technique, the tutor can identify simulation tactical events (e.g., fire orders, enemy destroyed events). Once identified, the events are time-stamped, placed on a time-line, and tied into the ontology to trigger any related knowledge sources.

A knowledge source is a very general kind of rule. It uses terms from the ontology to explicitly represent constraint violation rules, tactical inference rules, and tutorial meta-rules.

Architecture

General control for V-CTC is provided by a blackboard architecture. This software architecture allows meta-level reasoning, real-time reasoning, an integration of multiple kinds of knowledge sources, and evolving solutions to be constructed from lower-level solutions (Hayes-Roth, 1985).

The blackboard also provides a data structure that allows building a larger picture of the tactical and tutorial situation from lower-level events. Finally, the blackboard architecture can also be used to control filtering in the communications link (a DLL) between the tutor and simulation. The filtering determines the kind of events the simulation sends to the tutor.

Evaluation of user actions in the simulation is performed by deductive reasoning in SWI-Prolog over a domain knowledge representation. The knowledge representation is backed by a standard upper ontology to support reusable knowledge bases and domain concepts. A domain-specific knowledge base provides the tutor with rules for evaluating user fire missions along the four measures of performance (target selection, munition selection, observer placement, and observer survivability) shown earlier in Figure 3, and for detecting violations of Army doctrine (e.g., moving all artillery batteries simultaneously when there are high payoff targets that can be engaged, or attempting to engage moving targets with non-PGM ammunition).

Tutor Knowledge and Reasoning

Our knowledge representation is a tiered, modular structure built around an upper ontology representing basic (common) concepts, such as time and spatial relations. This common ontology has been developed by an international group and is in the public domain (Niles and Pease, 2001). This modular structure enables re-use of the ontology to new domains, while allowing more focused effort to develop the domain-specific knowledge at service-specific and task-specific levels.

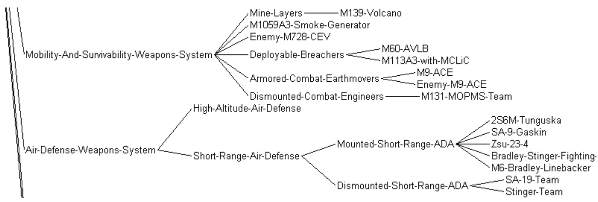

Figure 5. Part of an ontology of weapons categorized by battlefield operating system

V-CTC contains a model of the tactical domain, such as knowing about friendly and enemy vehicle types (a small part is shown in Figure 5). It uses this model to recognize, for example, that a 2S6M Tunguska is an enemy air defense artillery (ADA) piece, so if the user should be targeting enemy ADA the tutor will know that a 2S6M is a priority target.

Status

The initial selection of rules focused on providing a demonstration for 4 scenarios for a battalion fire support officer and 2 scenarios for instructing mech infantry or armor company team commanders. Thus the knowledge modeled is admittedly fragmentary and better illustrates the potential of the V-CTC concept, rather than providing a full-fledged tactical tutor, which would be a major undertaking even for one duty position.

The current system is installed as an overlay to the ATF game. 3 scenarios use maps from NTC, 2 from Death Valley, and 1 from Baghdad. Testing is still pending: We explored possibilities of informal testing with NTC but no venue could be found given the exigencies of the war situation at the time.

Related Work

SHAI has implemented a tutor (Stottler, et al., 2002) that communicates with a tactical simulation, but key differences from the current work include limited interactivity during the mission and no intended design for reusability from the start. The explainable AI work (Lane, et al., 2005) used with the oneSAF and Full Spectrum Leader simulations logs events and allows replays, but is not intended to simulate an expert instructor's advice *during* a mission, or an instructor's ability to provide targeted qualitative assessments of a trainee's strengths, weaknesses, and the areas that need to improve after the mission. Instead, the explainable AI systems provide an excellent way of reviewing what happened and querying the synthetic (friendly or enemy) AI components of the simulation. Still, it seems likely that players will more frequently want to know what went wrong rather than what went right, and more importantly, how to fix it. The work in branching storylines (Gordon, et al., 2004) provides useful practice in role-playing situations but (we hypothesize that) tactical expertise, like chess, will require hundreds to thousands of hours of practice, and only an engrossing simulation can provide this amount of practice. Consider that a tactical commander must learn to coordinate many different kinds of weapons for many different kinds of tactical situations in many different settings (different METT-TC[3] situations).

Summary

Intelligent tutoring systems (ITS) coupled with high fidelity simulations can provide highly interactive game tutors for commercial games and the state-of-the-art is such that they can be built and deployed for reasonable cost *now*. An ITS, such as V-CTC, can provide individualized instruction with the potential 2 sigma (standard deviation) improvement that good human tutors can accomplish. Such tutors can widen the appeal of games that would otherwise be too difficult for many to learn well enough to appreciate the depth of the game.

A traditional ITS has a user model that is inferred from user actions. V-CTC uses a Bayesian user model, but the user model was never fully developed. In fact, it seemed the biggest gains that V-CTC derived were from its solution analysis capabilities. We expect that an ITS approach using rule-based approaches alone for solution analysis and to select tutorial interventions will be sufficient to provide highly interactive and helpful tutoring behavior even without a user model. Users can be allowed to directly turn off or parameterize tutor behavior to their liking. Inferred user modeling can be added later, as a second step in complexity, to provide additional capability.

[3] METT-TC is a mnemonic standing for the most important contextual elements in a military situation: Mission, Enemy, Terrain, Troops available, Time, and Civilian considerations.

By implementing the tutor component itself as a reusable component and designing for reusability from the start, the ITS component cost may be further amortized across multiple games or a product line. This future application of AI technology promises more enjoyable games where users need not give up in frustration and where they can move more rapidly through initial learning curves, allowing them to appreciate deeper realms normally hidden from casual users.

Acknowledgements

The V-CTC project is a Phase II SBIR (Contract No. DAAH01-03-C-R302) funded by DARPA's Training Superiority (DARWARS) program. We would like to thank Ralph Chatham, Program Manager, for his visionary concepts and commitment to revolutionizing training. We would also like to thank Pat Proctor, President of ProSIM Company, for assisting us by sharing his expertise in the areas of tactics, combined arms warfare, Army training, and the operation of the ATF tactical simulation. Michelle Sams provided key help in getting this project going and in designing the HCI components. All work was performed while the author was at Teknowledge.

References

Chatham, R., and Braddock, J. (2001). *Training superiority and training surprise* (Defense Science Board Task Force Final Report). Avail on-line http://www.acq.osd.mil/dsb/trainingsuperiority.pdf

Gordon, A., Lent, Mike van, Velsen, Martin van, Carpenter, P., and Jhala, A. (2004) Branching Storylines in Virtual Reality Environments for Leadership Development. Proceedings of the Sixteenth Innovative Applications of Artificial Intelligence Conference (IAAI-04).

Hayes-Roth, B. (1985). A blackboard architecture for control. *Artificial Intelligence*, *26*, 251 – 321.

Lane, H. C., Core, M. G., Lent, M. V., Solomon, S, Gomboc, D. (2005) Explainable Artificial Intelligence for Training and Tutoring. *Proceedings of the 12th International Conference on Artificial Intelligence in Education.* Amersterdam, Holland: International Artificial Intelligence in Education Society.

Murray, W. R. and Sams, M. (2004). Virtual Combat Training Center (V-CTC): An *Intelligent Tutoring System + Tactical Simulation. In Interservice / Industry Training,* Simulation and Education Conference. 2004 Proceedings.

Murray, W. R. (2005). *Phase II Final Report for the Virtual Combat Training Center*. SBIR #SB022-030, Real-Time Assessment of Student State. Defense Advanced Research Projects Agency (DOD).

Murray, W.R., Pease, A., and Sams, M. (2003). Applying formal methods and representations in a natural language tutor to teach tactical reasoning. *Proceedings of the 11th International Conference on Artificial Intelligence in Education.* Brighton, UK: International Artificial Intelligence in Education Society.

Niles, I., and Pease, A. (2001, October). Towards a standard upper ontology. In Welty, C. and Smith, B. (Eds.), *Proceedings of the 2nd International Conference on Formal Ontology in Information Systems (FOIS-2001*). NY: Association for Computing Machinery.

Proctor, P. (2004). Armored Task Force. Available on-line at www.prosimco.com

Sams, M, and Murray, W. (2004). *Dialog agent system*. SBIR OSD-CR01 Phase II Final Report. Orlando, FL: Naval Air Warfare, Training Systems Division.

Stottler, R. H., Jensen, R., Pike, B., Bingham, R. (2002). Adding An Intelligent Tutoring System To An Existing Training Simulation. In Interservice / Industry Training, Simulation and Education Conference.