Intelligent Tutoring for Ill-Defined Domains in Military Simulation-Based Training

Elizabeth Owen Bratt, Center for Study of Language and Information, Stanford University, Stanford, CA 94305, USA
ebratt@stanford.edu

Abstract. This paper describes the role of simulation-based training in the military. Interviews and observations of military instructors in the damage control and shiphandling domains provide examples of how the instructors extend the student’s training beyond the well-defined simulated world with qualitative reasoning about context, hypothetical variants, and critical factors of the scenario. An intelligent tutoring system for a simulator can have a well-defined core area of domain knowledge, but to replicate more of the human instruction typically given in simulation-based training, the ITS should include a capability to deal with the ill-defined periphery of domain knowledge. Natural language interaction, including asking students open-ended questions about their performance, can help support tutoring of the ill-defined material.

Keywords. Military simulators, ill-defined domains, open-ended questions, natural language interfaces

INTRODUCTION

Several workshops on ill-defined domains (Aleven, Ashley, Lynch, & Pinkwart, 2006, 2007, 2008) have raised the question of which domains are ill-defined for intelligent tutoring systems (ITS’s), what makes these domains ill-defined, and what approaches to developing ITS’s might be best suited to address which aspects of these domains. In this paper, I describe military simulation-based training as including some domains for which an ITS can address a well-defined core, but not an ill-defined periphery. Examples from human instructors illuminate both the nature of the ill-definedness, to allow comparison with other ill-defined domains, and provide a basis for understanding how an ITS might be developed to address the ill-defined areas. The large body of experience using simulations provides material for modeling how to fit the orderly, well-defined core of skills and knowledge into the larger ill-defined area necessary for students to master. Developers of natural language interfaces to simulators and simulation-based ITS’s have confronted similar issues of ill-definedness when moving beyond a well-defined set of menu items, and the experience of the Spoken Conversation Tutor for Damage Control (SCoT-DC) ITS with open-ended questions provides an example of approximating some of the ill-defined periphery within a system built around the well-defined core.

Military training often aims to give the student a sufficient appreciation of the principles and values involved in decision making, so that the student can make a reasonable choice in a difficult situation for which there is no clearly optimal response. Even experts will often differ in their priorities and strategies, so the goal of training is to develop the student’s knowledge and reasoning skills to the point that their chosen response will avoid the worst outcomes, and rank with the best possibilities envisioned by experts. Military training frequently consists of lectures and readings to build basic knowledge, followed by practice in a simulator while a human instructor coaches the
student and assesses the student’s performance in practical applications of the principles learned in the classroom. Simulators approximating the real world impose some definition on ill-defined training tasks, since various details which could affect outcomes in certain scenarios may not be simulated, and the data and programming of the simulation are likely more homogeneous and simple than the real world. Intelligent tutoring and coaching systems have been added to simulators to help the student focus on the critical aspects of performance and avoid practicing mistakes (e.g., Thomas & Milligan 2004, Peters, Bratt, Clark, Pon-Barry, & Schultz, 2004). If an ITS can replace the more routine parts of human instruction, it can allow human instructors to focus on challenging and complex aspects, and reduce the cost of training. It also permits the military to train students outside the schoolhouse at more convenient times, such as onboard ship when there is little action (e.g., Fletcher, 2009).

This paper uses data from human military instructors, as they train students and discuss their training methods and objectives with researchers, as the basis for specifying how an ITS could approach this kind of domain, in which the instructors use simulation as a springboard to convey to the student a deeper understanding of the principles involved for true competence in the area. Much of military training is well-defined and could be modeled by an ITS with a well-defined domain core, but to capture the ill-defined domain periphery, an ITS will need to use alternate techniques.

Natural language interaction and open-ended question capabilities relying on constraint-based models are a promising way to address the areas that human instructors do. Based on data from experiments and trial uses with the Spoken Conversational Tutor for (Navy) Damage Control (SCoT-DC) (Peters et al., 2004), these methods are practical and feasible. Implementing these methods in an ITS will also allow systematic evaluation of the effectiveness of the various techniques used by human instructors in the ill-defined domain periphery.

SIMULATOR-BASED TRAINING IN THE MILITARY

Simulators are frequently used in military training (e.g., Wampler et al., 2006), as a means of practicing skills and experiencing consequences of various actions, while avoiding the use of extensive support personnel, the risk of damaging costly equipment or endangering the lives or safety of personnel. Simulators enable practicing scenarios that are close to the actual military tasks, providing situated learning (Brown, Collins & Duguid, 1989). Simulators can present the same situations with the same conditions, revealing the effects of different student actions, and may have flexible temporal control, allowing the student to work through problems at an appropriately speeded, slowed, or paused rate (Helfesrieder & Shankararaman, 1999). Solving varied problems of increasing complexity with a simulator can support a constructivist training methodology, which can be more effective than lecture-based training (Childs, Schaab & Blankenbeckler, 2002), though simulators are generally used with substantial human instructional guidance, avoiding the problems raised by Kirschner, Sweller and Clark (2006) for minimally guided constructivist learning.

Military simulators are generally used for providing realistic experience with a task, but they are not used as the sole method of training for the task. A human instructor is generally used for coaching, mentoring and after-action review. This additional support is in line with the interactive qualities of a human tutor’s feedback and access to expository reference material that Pilkington and Grierson (1996) found useful to improve the performance of a simulation-based learning environment. Similarly, Thomas & Milligan (2004) describe a need for simulator support, guidance and focus from human experts through objectives, directions, context, corrective feedback and hints.
Anderson (1988) distinguishes three types of knowledge that need to be tutored: procedural, declarative (appropriately organized for reasoning), and causal knowledge. For simulator-based training, all three of these types of knowledge are likely to be involved. Students have studied the military doctrine of what procedures to use for certain situations, so practicing how to use those procedures in the simulator is a central reason for simulator use. The simulator also provides representations of many objects of the domain and their properties, which would be useful for the student to learn. Finally, the causal knowledge of what the possible outcomes of situations are depending on which actions are taken or omitted at what time, is a critical part of simulator-based training. All three of these types of knowledge can be addressed by an ITS with a well-defined domain core, but all three also can involve the ill-defined periphery of a domain, not easily captured in an automated system. Domain procedures and objects can have variants and details not captured by the simulator, and the whole space of possible causes and effects of actions within the domain as understood by an expert are not likely to be simulated.

HUMAN COACHING AND TUTORING ON SIMULATOR PERFORMANCE

The particular perspectives, concerns, and strategies of military instructors are illustrated in examples from interviews researchers conducted with them, and videotaped coaching, pre-briefing, and review the instructors conducted in conjunction with simulator practice. The simulators used were the research prototype DC-Train simulator for damage control (Bulitko & Wilkins, 1999), the later prototype voice-enabled DC-Train (Peters et al., 2004, Bratt, Schultz, Peters, Chen, & Pon-Barry 2005) and the COVE simulator for shiphandling (Roberts, 2001), originally a research prototype, now transitioned to a commercially available part of the official training curriculum (Lundquist, 2007). Interviews and videotaping were conducted by Stanford University researchers and research team members from the University of Illinois and the Naval Undersea Warfare Center, mainly at SWOS, the Surface Warfare Officers’ School, with Naval officers as students, but also at Stanford University with Stanford students as coached students. The examples are coded by source in the format domain-location-year, as seen in Table 1. Debriefs are videotaped interactions after the student uses a simulator; coaching is videotaped interaction during simulator use.

The corpora were reviewed for examples of areas that fell outside the coverage of the current or immediately achievable ITS’s. The goal was to enumerate distinct areas, to lay out specific areas that might form part of future experimental ITS’S, to establish areas in military simulation-based training domains to compare more readily to other ill-defined domains, and to facilitate typological comparisons. The examples are given to illustrate the existence and relevance of these types of human coaching and instruction, as a concrete foundation for describing characteristics of the military simulation-based training domain.

---

1 Because the corpora vary tremendously in the type and length of interaction, level of mastery of the students, and the domain covered, quantitative data comparing counts and frequencies of the various types of examples in the various corpora were not collected.
Table 1
Corpus identifiers and properties

<table>
<thead>
<tr>
<th>Example Code</th>
<th>Simulator</th>
<th>Location</th>
<th>Year</th>
<th># Instructors</th>
<th># Students</th>
<th>Interaction</th>
</tr>
</thead>
<tbody>
<tr>
<td>DC-SWOS-2000</td>
<td>DC-Train</td>
<td>SWOS</td>
<td>2000</td>
<td>4</td>
<td>15</td>
<td>Debrief</td>
</tr>
<tr>
<td>DC-Stanford-2002</td>
<td>DC-Train</td>
<td>Stanford</td>
<td>2002</td>
<td>1</td>
<td>0</td>
<td>Interview</td>
</tr>
<tr>
<td>DC-Stanford-2005</td>
<td>DC-Train</td>
<td>Stanford</td>
<td>2005</td>
<td>2</td>
<td>4</td>
<td>Coaching</td>
</tr>
<tr>
<td>DC-SWOS-2006a</td>
<td>DC-Train; SCoT-DC</td>
<td>SWOS</td>
<td>2006</td>
<td>2</td>
<td>10</td>
<td>Debrief</td>
</tr>
<tr>
<td>DC-SWOS-2006b</td>
<td>DC-Train</td>
<td>SWOS</td>
<td>2006</td>
<td>2</td>
<td>0</td>
<td>Interview</td>
</tr>
<tr>
<td>SH-SWOS-2005</td>
<td>COVE</td>
<td>SWOS</td>
<td>2005</td>
<td>9</td>
<td>4</td>
<td>Coaching</td>
</tr>
<tr>
<td>SH-SWOS-2007</td>
<td>COVE</td>
<td>SWOS</td>
<td>2007</td>
<td>2</td>
<td>0</td>
<td>Interview</td>
</tr>
</tbody>
</table>

EVIDENCE FROM HUMAN CORPORA ON ILL-DEFINED AREAS

The human instructors made many types of contributions to simulation-based training, beyond the well-defined material used in the simulator. This paper divides the areas beyond the core material into two groups: first, areas that directly reinforce the simulator’s world model and learning objectives, and second, areas that step outside the simulator’s world model in ways that would be difficult to include as an extension of the existing automated system. The areas belonging to each group are listed in Table 2, and will be presented in more detail in the following sections.

Table 2
Ill-defined areas supported by human instructors

<table>
<thead>
<tr>
<th>Extension of Well Defined Core</th>
<th>Ill-Defined Periphery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Situational Awareness</td>
<td>Fidelity and Scope</td>
</tr>
<tr>
<td>Imagery</td>
<td>Hypothetical Variants</td>
</tr>
<tr>
<td>Actions and Effects</td>
<td>Adapting to Unexpected</td>
</tr>
<tr>
<td></td>
<td>Doctrine Change/Commander Variation</td>
</tr>
</tbody>
</table>

Extending the Well-Defined Core of Simulator Coverage

Various types of human instruction go beyond the scope of the training provided by the simulator, but they support the simulator’s world model and learning objectives by giving additional perspectives and considerations relevant to the simulated scenario. Because these types of instruction augment the core of the automated training by the simulator, they could possibly be incorporated to some extent into a rich, detailed ITS, without necessarily treating the domain as ill-defined. Because these areas might be difficult or time-consuming to implement without fundamentally altering the conception of a well-defined domain for the ITS to model, they may be better handled within a system already poised to handle ill-defined material. These examples are included in this paper as evidence of what may lie just within the boundary of the well-defined core domain for the ITS, in contrast to the later examples
more securely established beyond that boundary. The specific areas extending the well-defined core of these domains are situational awareness, imagery, and effects of actions.

**Situational Awareness (Extension of Well-Defined Core)**

An important aspect of military simulator exercises is maintaining situational awareness, that is being aware of all the relevant factors, events, and entities that might have an effect on your goals, even if they are not immediate or likely problems. Simulator practice can involve some of these factors in each scenario, but situational awareness should encompass all the items that did not have an effect, as well as those that did. Thus, in terms of the training domain, the simulator provides these factors to some extent, in terms of providing displays or queries that can give information on them, but it does not directly ensure that the student is aware of all of them. Instructors coaching students and reviewing their performance often try to gauge the student’s situational awareness, and encourage further development of it in future practice. For example, instructors asked questions such as *What fire pumps were on-line? Was pump five ever on-line? Did everyone report in manned and ready?* (DC-SWOS-2000). A coach will draw the student’s attention to relevant reports and problems (DC-Stanford-2005). Instructors also guide the students for how to check that events are proceeding as intended, as in the following dialogue (SH-SWOS-2005):

**Instructor:** What do you want to see as you are closing?  
**Student:** As I'm closing?  
**Instructor:** As you are closing the destroyer, what do you want to start see happening? It’s kind of a silly question, but if you think about it on these terms, it will really keep you out of trouble. What do we see right now, on that DDG?  
**Student:** What do I see?  
**Instructor:** Yeah.  
**Student:** I see the gun mount on stern right now.  
**Instructor:** The stern and a little bit of the port. Uh are you getting the port view coming in on the port side?  
**Student:** Yeah, yes.  
**Instructor:** So you want the stern view to start going away, right?  
**Student:** Yeah  
**Instructor:** That is the sort of thing you should think about.

![Fig. 1. Situational awareness: looking for the stern view to recede.](image)

The instructors attempt to show how situational awareness can lead to different preferred strategies. If there are multiple fires on the ship, you may choose to set boundaries on all of them to prevent their spread, rather than going through all the steps to extinguish each fire one by one (DC-SWOS-2000). If a repair team responsible for one area of the ship is overtasked dealing with several fires while other repair teams are free, the student should assign a free repair team to some of the tasks, rather than let them pile up waiting for the usual team (DC-SWOS-2000).

Situational awareness in terms of which personnel are aware of which facts and which communications have been acknowledged is also an important factor stressed by the instructors. An
instructor said that while DC-Train does not give a way to query repair teams as to how many tasks they currently have, in real life you should ask them, and they should also report to you on their own initiative (DC-SWOS-2000). A coach guided a student confused about whether a simulated officer was asking permission to perform an action when that officer was in fact giving the student permission to do the action (DC-SWOS-2006a). The instructors also emphasized that it is important to keep the entire ship updated on how events are proceeding, though commands to the parties that take actions are more important. While there was never a difference in simulated outcome based on whether the ship’s crew had been updated on events, this was a factor the instructors wanted the students to keep in mind (DC-SWOS-2006b).

For situational awareness where the relevant objects and properties are already represented in the simulator, awareness of these objects and properties could form part of a well-defined domain for an ITS. Awareness is a step away from core simulator usage, which deals with concrete actions and their effects, but an ITS could extend its domain representation to include situational awareness, and strategies for assessing and teaching it. For example, knowing the on-line status of fire pumps at critical points or how many tasks a repair team has at any moment are directly based on simulation facts, and could be incorporated into the well-defined core of the damage control domain. To the extent that the important objects and properties for situational awareness are not captured within the simulator, and are not readily incorporated, this area may provide material for approaches to ill-defined domains. As an example, teaching the student to notice important aspects of the simulated view during shiphandling may involve spatial relationships and named objects that are not represented in the simulator, and may be difficult to specify precisely.

**Vivid Imagery (Extension of Well-Defined Core)**

One way in which the instructors dealt with bridging the gap between the simulated tasks and the real-world tasks they were training the students for was to add vivid imagery and descriptions of other personnel’s emotional reactions due to the effects of the student’s actions. One instructor, debating contrasting strategies tells a student that while the student’s approach to dealing with the fires in the scenario might work, *Are you wrong by attacking the ones down below first?* No. OK. But the captain might think you are. Because CIC and things like that are definitely vital spaces for him, especially since he’s probably getting pretty hot up there on the bridge right, you know, between fires and everything like that going on up there, he’s probably getting a little concerned. He’s just lost his nav - he’s lost his nav picture, he’s lost his radar, he’s lost all that stuff (DC-SWOS-2000). A shiphandling instructor conveyed the importance of steering the ship carefully during a man overboard simulation exercise by saying, “*Help me! Help me!*” Get a little closer to him and you can hear him saying that, "help me!" (SH-SWOS-2005), though this is not part of the simulation. For another student traveling at higher than desired speed close to the simulated man overboard, the instructor says, It's going to be close. He's pretty scared right now. (SH-SWOS-2005). A shiphandling instructor also urged a student to make more use of a tugboat in maneuvering the ship by discussing the tug personnel’s earning the wages paid to them, rather than anything more directly simulated: *I would’ve used your tug though earlier too. You got it; they're getting paid an hourly wage. Why not? Let them do the work* (SH-SWOS-2005).

Human instructors appear to find imagery a valuable tool for drawing students’ attention to important facts, and adding to the realism of the simulated experience. Imagery provides an indirect way to guide a student to realize the desired action, by emphasizing contributing factors. Adding vivid
imagery into the well-defined core domain would take skill and attention to detail of appropriate vocabulary and phrasing, and ensuring the right correspondence between the imagery and the objects and properties of the domain. Using imagery effectively in tutoring interactions would involve navigating potential pitfalls, such as repeating the same description and losing the power of its novelty. It would also require being able to account for student responses to the imagery, particularly in less constrained interfaces such as natural language or free drawing in graphical input. This area of human instruction could with effort be added into the well-defined core domain, but it may again benefit from techniques designed for ill-defined domains.

Reasoning about Actions and Effects (Extension of Well-Defined Core)

A third area in which human instructors extend the well-defined core of the domain involves leading the students to reason about the effects of their actions, to make the implicit rules and connections within the simulator more visible and explicit, as in the dialogue shown in Figure 2 (SH-SWOS-2005). This discussion of underlying forces directly relates to the functioning of the simulator, although the instructors may venture beyond the simulated causes and effects that a student could possibly experience.

Instructor: Where's the wind going to affect more, your bow or your stern?
Student: Bow.

Instructor: Sure about that?
Student: It should

Instructor: Where's more of your sail area with relation to your pivot point?
Student: In relation to the pivot point, I guess the stern.
Instructor: Yep. So, what's fighting against you right now?
Student: The wind on my bow

Fig. 2. Reasoning about actions: wind affecting bow vs. stern.

The instructors guided the students to use a mental model, which may not be obvious from interacting with the simulator, specifically, separating the ship into two halves, the stern and the bow, and then controlling each one separately, the bow with a tug, and the stern with the rudder and the engine (SH-SWOS-2005), explaining how attempting to control the stern with the tug will run into problems with wind and current. The instructors gave general guidance about outer limits for settings, such as at critical locations maintaining a speed at least two knots because of the effects of wind and current, but no more than three knots in order to work with the tug (SH-SWOS-2005). The instructors pointed out the effect of momentum on ships of about ninety-seven hundred tons, and how engines cannot stop motion instantly (SH-SWOS-2005).

The instructors quiz the student on actions and consequences, such as engine settings and the different effects on the rudder of forward and backward engines (SH-SWOS-2005). The instructor makes connections for the student, of how certain facts, such as a too high heading meaning that the ship’s bow is moving too fast (SH-SWOS-2005). The instructors queried the students about what preconditions are necessary for actions, such as reports from other personnel, and had them refine the answers to greater precision, such as what degree of electrical and mechanical isolation is necessary before you can fight a fire (DC-SWOS-2000).
Throughout the reasoning, instructors frequently emphasize that there are different possible right decisions, with different consequences and different relative merits (DC-SWOS-2006b, SH-SWOS-2005), and that the most important training consideration is that the student understands the reasons for their actions and can act confidently (DC-SWOS-2006b). Again, this could be encapsulated in ITS rules; however, the complexity and patterns described by human instructors pose a problem of scale for an ITS.

**Instruction and Coaching in Ill-Defined Areas**

The human instructors also guide the students to step outside the world model of the simulator in ways that are less easily captured in a well-defined core for an ITS. This kind of instruction makes the student aware of the simulator’s limitations and encourages the student to abstract away from the particular scenarios and procedures that are the core material of the simulation. While some of these particular examples may be addressed by a different well-defined world model, the main point of these areas is to emphasize reaching beyond the core, into the ill-defined periphery.

**Simulator Fidelity and Scope (Ill-Defined Periphery)**

Setting the appropriate level of fidelity and scope for the simulator is not a simple task. The sheer expense and time needed for highly detailed representations (Yardley, Thie, Schank, Galegher, & Riposo, 2003) can limit what systems can be built for training, and the ideal fidelity level for training purposes may also vary depending on the population to be trained, because novices may not have sufficient memory and cognitive ability to work with complexities entailed by high simulation fidelity (Alessi, 1988). Partly as a consequence of the expense of implementation, simulators are generally developed for a core common situation, such as for a single ship in the Navy (the DDG-51 is the most common). Training policies often only give credit for exercises performed on the actual systems of the student’s own ship (Yardley et al., 2003, DC-SWOS-2006b), identifying simulation-based training as insufficient due to its different fidelity and scope.

All military training simulators will have limits on how closely they replicate the real world. Even within the area targeted by the simulator, the models of objects and their functional capabilities will likely be more homogeneous and simple than the real world. Objects and actions less central to the training goals will be simulated with less fidelity. When a student is practicing interactions with simulated personnel, the student’s own possible actions will likely have a larger range of simulated possibilities and realistic effects than the actions and reactions from the simulated support personnel. Complex human characteristics, such as combat performance degradation under stress, for which little data is available, are difficult to model accurately (Page & Smith, 1998). When several different simulators interoperate to create a complex training situation, the problem of multiresolution modeling arises, in which one of the simulators may need to aggregate or disaggregate entities to function with the other (Page & Smith, 1998), thus changing its level of fidelity.

Given these competing factors on selecting the appropriate fidelity and scope for simulation-based training, particularly the possibility that different levels may be appropriate for different students or different learning objectives, human instruction often sought to address the mismatches between the fidelity and scope of the simulator and what they saw as important training issues. Some of the examples here may be taken as directions to improve the simulation, rather than as a fundamental challenge to the definition of its domain, but the underlying tension between abstraction
and elaboration likely will persist. When the decision of which objects and properties to simulate is not a stable foundation for the ITS domain, the basis for tutoring becomes much more ill-defined.

The human instructors expressed concerns about the simulator not including certain responsibilities for the student, and about it including other inappropriate responsibilities. Instructors felt that an important part of damage control was omitted from DC-Train, namely giving routes for personnel to use as they moved through the ship to combat the fires, floods, and smoke (DC-SWOS-2000). Instructors advised that the early version of DC-Train ought to track manned-and-ready-zebra status for the student, since having the student track it imposed an additional cognitive burden (DC-SWOS-2000).

Other concerns involved how simulator fidelity and scope trained the student to make different choices from real life, to prioritize differently, or to weigh costs and benefits differently than in real life. The instructors wanted the student to choose actions based on prioritizing which types of compartment would be affected by a fire or flood, and the early DC-Train did not present the student information about compartment types, and did not simulate distinctions between compartment types appropriately, allowing a void space in the ship to catch on fire, which would be impossible (DC-SWOS-2000). Instructors also emphasized how in real life, the mission of the ship and the source of the damage would influence what actions were correct, and these elements were entirely unsimulated in DC-Train, leaving the student to treat all fires, floods, and smoke the same. Many times, the instructor explains how the simulator fidelity and scope mean that certain actions or strategies may be appropriate for the simulator, but the real world would require more details in the actions, or more context to determine a choice. Instructors also pointed out that while DC-Train simulated the systems that repair lockers would handle, it did not take into account the ship-wide consequences of taking down any other ship systems in order to handle a fire or flooding. Integration with all other systems outside damage control is lacking (DC-SWOS-2006b). The instructors critiqued DC-Train for requiring the student to set watertight boundaries in the superstructure of the ship, following the same procedure as for lower parts, since if the ship’s superstructure is not above water, then the ship will already have sunk far enough to make the flood boundaries irrelevant (DC-SWOS-2000). Another factor relevant for real-world decisions about flooding in particular, as well as fire at times, is the stability of the ship, and how large water-filled compartments affect it, which is not simulated within DC-Train (DC-SWOS-2006b). Instructors also critiqued DC-Train for its simulation of fire spread, saying that fires ought to be simulated as spreading more quickly in the vertical direction and more slowly in the horizontal direction than DC-Train’s model, since this could impact the student’s prioritization of which fires to combat first, and they wanted to avoid negative training (DC-SWOS-2000). Instructors also discussed how DC-Train forced students to put actions into a sequence when they would really happen concurrently in real life, and might be issued as a single command (DC-SWOS-2000, DC-Stanford-2002).

Human instructors will talk about how different outside factors make different courses of action appropriate. For example, in damage control, if the Commanding Officer needs to preserve weapons for battle, it would not be appropriate to douse a magazine compartment with water to prevent fire from spreading to that compartment, even though in most situations this would be the prudent move to save the ship. The damage control assistant needs to recognize the tradeoffs between preserving the weapons and removing the possibility of the weapons exploding. These outside factors are not part of the simulator; rather, they are motivating factors for action, while the simulator provides a means of seeing how events unfold toward the desired conclusion.
In all of these cases, the instructor would note how DC-Train diverged from reality, but discuss how the student handled the situation, given those circumstances, and emphasize the importance of performing the actual procedures in real life, even if the simulator made them awkward or hard to execute (DC-SWOS-2000). Instructors would help students become familiar with the level of fidelity of the simulator: When a student asked if a flooding event required a dewatering, piping and patching team, the instructor responded, *It doesn’t really go that into depth. You want to dewater that space* (DC-SWOS-2006a). When a student wanted to investigate whether a hole caused flooding on the ship, the instructor advised that DC-Train does not support this (DC-SWOS-2006a). The instructor also advised the student that DC-Train does not simulate the required actions after extinguishing a fire, namely, setting a reflash watch, overhauling and performing post-atmospheric testing (DC-SWOS-2006a). The instructor also notes that some fires can be fought and overhauled at the same time, but others cannot (DC-SWOS-2006a). The instructor confirmed the student’s belief that after a flood, there would be similar required actions, which DC-Train does not simulate: going through a checklist about the status of electrical items, dumping flooding water into the deck drains and pumping it overboard or into the grey water tank (DC-SWOS-2006a). Other similar levels of detail that a knowledgeable student might expect to use, but which DC-Train did not simulate, were the method of desmoking, that is, using a ram fan, a box fan, natural ventilation or installed ventilation, and the method of dewatering, that is, using a submersible pump, installed drainage or the main drainage system (DC-Stanford-2002).

Even if the student’s own particular ship happened to be the one used in the simulator, the extensive complex programming in the simulators means that it is difficult to keep a simulator up to date with all small changes to individual ships, which could affect the consequences of student actions. Instructors recommend that students examine the physical layout of their own actual ships after conducting simulations, to check if there were particular aspects of the ship that would have affected events, and changed their choices (DC-SWOS-2006b).

Stepping outside the simulation to examine possible different levels of fidelity and scope is not something that can be handled easily as an extension of a well-defined core. This area of instruction falls in the ill-defined periphery.

**Hypothetical Variants of Simulated Events (Ill-Defined Periphery)**

Another way instructors extend the simulation-based training is to examine hypothetical variants of the simulated events experienced by the student (DC-SWOS-2000, DC-SWOS-2006a, DC-SWOS-2006b, SH-SWOS-2005). These events often are not simply events that did not happen in this particular session or scenario, but instead may be events that are not simulated at all. For many of the examples, sufficient time and funding could allow the hypothetical variants to be included in a simulator, but this form of instruction involves stretching beyond the core areas, and seeing the relations between the covered material and the larger domain.

Sometimes these hypothetical variants have several right answers, but also at least one definite wrong answer, such as if a space is flooded to within 1 foot of the ceiling, good choices are either topping it off with an extra foot of water to make the space more stable, or repairing and dewatering it, but not simply leaving the space incompletely flooded (DC-SWOS-2006b). Another example of multiple right answers, but a definite wrong answer, is a case where the firemain has lost pressure in one of the loops and the location of the leak is not clear. It would be okay to announce to the ship’s crew that they should help look for the leak or to stop the pumps, but it would be definitely wrong to...
start additional pumps (DC-SWOS-2006b). When experiencing a misrecognition of a command, which led to unanticipated flooding of a compartment, an action normally used to prevent explosives from catching fire, a student asked the instructor about when this action would in fact be warranted, what kind of permission would be needed, and what method would be used to accomplish it, such as a sprinkler or hose team (DC-SWOS-2006a). The instructors emphasize that damage control often involves an 80% solution, not a 100% solution, so if the student can name the relevant factors to consider and their effects, that is more important than which action they choose for this particular situation (DC-SWOS-2006b).

Particularly in emphasizing multiple imperfect solutions, the instructors use the simulation as a starting point for reflection on hypothetical variants, rather than as a self-contained world. These steps outside the structure of the simulation, in ways that do not readily extend the existing model, form part of the ill-defined periphery of human-coached simulation-based training.

**Adapting to the Unexpected (Ill-Defined Periphery)**

Another way the instructors place value on students taking an extra step beyond the simulated material is in stressing the importance of being able to adapt to the unexpected. As one instructor put it, *What makes a good ship driver? Is a good ship driver the one that comes in all perfectly every single time, or the person who, faced with adversity, can overcome? Thinking on your feet, right?* (SH-SWOS-2005). Coming in to port perfectly each time would be the kind of skill directly trained by the simulator, but the instructor specifically ranks this skill below understanding shiphandling well enough to overcome unexpected problems. The dynamic nature of the simulator means that the student can make mistakes, then recover from them well (or poorly), so each student may have different sets of challenges from the same initial scenario, so some variability can be part of a well-defined system. The instructors would also like to see simulators where simulated ship personnel sometimes give helpful suggestions, and sometimes give wrong suggestions or start to take wrong actions, so that the student has to rein them in, and actively manage the situation with good attention, rather than trusting other personnel to be correct (DC-SWOS-2006b). However, the instructors encourage the student to consider factors beyond these regular, ordinary variations. In choosing strategies and actions, the instructors emphasize planning for unexpected events, such as positioning a ship so that you can easily move the ship away if there were trouble or if the wind changed (SH-SWOS-2005), even though these events are not likely to be part of the simulated scenario.

**Doctrine Change/Commander Variation (Ill-Defined Periphery)**

A different dimension along which the instructors encourage the students to be aware of the limits of the simulation is in the specific military doctrine, that is, the official recommended procedures, determining which actions are correct. The instructors make the students aware that doctrine can change over time (DC-Stanford-2002, DC-SWOS-2006a, SH-SWOS-2005), such as the Navy’s change from a triangle-based symbology for representing damage control events to an circle and slash icon-based symbology (Klinkenberger, 2003), with some concomitant changes in procedures. The instructors point out that the simulator may require the student to make decisions for simulated personnel, whereas normally those personnel would be allowed some discretion as they handle the problem, such as boundarymen choosing different boundaries from the default ones that might be ordered (DC-SWOS-2006b). The instructors also emphasize that the captain of a ship can establish
certain rules and procedures, so they will need to follow the captain of their particular ship (SH-SWOS-2005). This variation can be a point of strong differences, as illustrated below in a discussion about turning a rudder as far as it can go, that is, hard rudder (SH-SWOS-2005):

**Instructor:** There you go. That was something—Hard left rudder, aye. Did your captain let you use a hard rudder?

**Student:** We had to ask for it.

**Instructor:** Did you really? Okay, I couldn't use it and I never thought about using it because they would have freaked out.

Fig. 3. Individual Variants in Doctrine: Hard Rudder.

When the instructional point is to be cautious about rules and that flexibility is necessary, it is hard for a simulation and an ITS to encode the relevant examples directly as a well-defined area of study. Treating this as another aspect of the human instruction reaching into the ill-defined periphery may capture its spirit best.

**Help With the Simulator (Ill-Defined Periphery)**

The final area of examples where the instructors step beyond the limits of the simulated world is in giving help to the student, which may relate to the scope of the domain or to unexpected simulator behavior. Certainly, much help falls in the well-defined core for automated instruction, but some examples necessarily break out of the limits of the simulated world.

In the category of helping the student understand how to interact with the simulator and understand its behavior, coaches give guidance on how to word spoken commands (DC-Stanford-2005), what the windows on the display represent (DC-SWOS-2006a) and check that the student knows how to determine information needed for decisions (DC-Stanford-2005). The instructor explains the spoken interaction protocol of acknowledging reports: The command ‘very well’, he'll stop repeating that command. Say ‘very well’, say it again, say ‘very well’ he'll stop saying it over and over (SH-SWOS-2005). The coach gives guidance on how to work out irregular cases, for example, unusually large ship compartments which span multiple decks (DC-Stanford-2005). The instructor will also need to step in when the simulator has bugs or unexpected behavior. This includes ensuring commands are carried out: Your port engine didn't stop. It says it stopped, but it didn't stop. I'll stop it for you (SH-SWOS-2005), telling the student where a pier missing from the display ought to be (SH-SWOS-2005), and reassuring the student that the instructor will maintain progress in the scenario despite any glitches: Okay, try to keep driving and I'll step in when I need to (SH-SWOS-2005).

All of these examples illustrate the point that the instructors find the simulator practice beneficial and central to their training task, but they find many areas for training beyond the simulator practice, and guide the student to reflect on how to situate their simulator learning in a larger, less defined context.
Model Tracing for the Well-Defined Core

As discussed by Lynch, Ashley, Aleven and Pinkwart (2006), a model-based tutor can operate in a well-defined subset of an ill-defined domain. Instructors working on a trial basis with the model-tracing SCoT-DC ITS liked seeing the comparison of students’ actions to a correct sequence of actions, and asked for more details in the comparison, such as exactly how many minutes late an order was, and the location of incorrect boundary frames compared to correct ones (DC-SWOS-2006b). In other simulators, expert performance cannot easily be defined by the particular actions taken, but rather by the achievement of goals and the avoidance of undesirable events, requiring a combination of performance metrics to automatically assess the student (e.g., Sheldon et al., 2002, Roberts, Pioch, & Ferguson, 2000).

Constraint-based Approaches for the Ill-Defined Periphery

For the parts of an ill-defined domain that resist definition, a model-based tutor does not match the nature of the domain. Rather than setting the student’s actions against a specific, desired set of expert actions, an ITS may choose to concentrate on different kinds of domain knowledge. Making sure the student is aware of constraints on actions, and possible positive and negative effects, can be very useful, even if these constraints and causal links do not provide a ranking of optimal actions.

CONTRASTING LANGUAGE AND SIMULATOR COVERAGE

Many of the issues faced by human instructors using a simulator and designers of an ITS to work with a simulator are also faced by designers of spoken natural language interfaces for simulators. Spoken interfaces to military training simulators include COVE (Roberts, 2001), CommandTalk (Stent, Dowding, Gawron, Bratt, & Moore, 1999), Eucalyptus (Wauchope, 1994), InterLACE (Wauchope, 1996), and QuickSet (Cohen et al., 1996). The spoken interface must determine what level of coverage is appropriate for each of the system components, likely to be speech recognition, natural language interpretation, dialogue management, natural language generation and speech synthesis. This question is not simply a matter of what items are available to refer to and what actions may be taken. Everett, Wauchope, and Pérez Quiñones (1999) list object representation as one of the key challenges in constructing a spoken language interface to the virtual reality systems they worked with (a 3D model of the ex-USS Shadwell decommissioned Navy ship and VIEWER, a visual representation of the electronic battlefield as covered by radar and other devices. Everett et al. (1999) found various objects which were natural to refer to in spoken language that had never needed to be represented as objects in the virtual reality system database, such as compartments, so their team constructed a supplemental database. Similarly, events were not represented as objects in these systems, so when users asked What was that? about a passing object, the speech interface could not query the database for the event.

Beyond representing basic objects, spoken interfaces often naturally provide methods of organization, abstraction and generalization of related objects and events, which the simulator may not represent. The CommandTalk interface to the ModSAF simulator provided users with the ability to group objects by plural descriptions (M1 tank platoons), quantified descriptions (all friendly forces), conjunctions (A11 and B11) and referring to all objects in a circled area (Dowding, Bratt, &
Goldwater, 1999), which translated into iterated commands to the simulator. Damage control
discussion similarly might use parts of complex compartment names with numerical or directional
identifiers to conjoin or quantify related compartments, such as *crew’s compartments numbers three
and four, forward and aft chief’s berthing, and all crew’s compartments* (Bratt, Schultz, & Peters,
2007). Damage control instructors taught in terms of generalizing events in ways not supported by the
original DC-Train, such as treating all fires which occurred as a consequence of fires spreading as a
single event to be discussed (DC-Stanford-2002), and grouping fires by location or characteristic and
addressing them together (DC-SWOS-2000). Everett et al. (1999) list particular strengths of spoken
interfaces as the following, along with their Navy examples:

- identifying objects or sets of objects by description (*all the friendly ships*);
- selecting sets of objects that meet certain criteria (*all the platforms that don't have missiles*);
- moving the viewpoint directly to a named place (*put me in the Communications Centre*); and
- quickly selecting one item from a large set of alternatives (*show me the Thunderbird*).

Spoken interfaces may also make explicit certain distinctions that the simulator was able to leave
unclear. For example, the damage control assistant communicates in DC-Train only with the repair
lockers, and does not treat the constituent teams within it, such as boundary team or investigators, as
distinct units. Spoken commands might tell the repair locker head to send investigators to a
compartment, while the simulator treats all actions by subordinates as actions by the repair locker
itself. Conversely, it may be natural for spoken commands to rely on context or dialogue clarification
for a command like *Set boundaries*, rather than the more explicit, simulated commands *Set fire
boundaries, Set smoke boundaries, or Set flooding boundaries*.

These differences in treatment of the domain by the simulator and the spoken interface reflect
each part of the overall system performing its role in the most effective, natural, suitable way.
Similarly, an ITS should cover the subject area in a way that is close to the simulator, but it is likely to
have a slightly different treatment of the domain in order to be most effective and most natural (e.g.,
Fried et al. 2003).

**Answers to Open-Ended Questions**

In order to develop deep comprehension of causal mechanisms and complex systems, rather than
merely shallow knowledge of definitions, properties and steps to take (Graesser & King, 2008), an ITS
can extend its coverage beyond the specifics of the simulated scenario, in the manner of human
instructors reviewing the student’s performance. Aleven, Popescu and Koedinger (2001) discuss the
importance of intelligent tutoring systems asking students to explain their reasoning, rather than
simply asking students to solve problems and evaluate their solutions. If the student is asked to
explain what principles and reasoning guided the choices, the ITS can evaluate whether the student has
noticed relevant concerns and weighed them appropriately. If the ITS can reason about the constraints
involved in the answer, it can recommend future simulator scenarios to test the student’s
understanding and application of these constraints. In the Geometry Explanation Tutor (Aleven et al.,
2001), student explanations of geometry rules are handled by a knowledge-based NLU component if
possible, that is, matching the well-defined core knowledge, and by a statistical classifier as a fallback,
when the answers go beyond expectations, to determine whether the answer appears to address the
correct geometry rule. Open-ended questions provide more opportunities for the student to say something wrong, and thus receive targeted instruction (Rosé et al. 2003).

As a step toward deeper discussion, the SCoT-DC tutorial dialogue system asks why questions, and results from three experiments indicate that the student answers are generally tractable for automated speech recognition and natural language understanding. Collins (1975) states that even though understanding the natural language of student answers is the major difficulty for a Socratic tutorial dialogue system, the problem is not insoluble, “because the program does not have to understand the student very well,” and detecting relevant factors in the student answer can be sufficient for an effective tutorial dialogue. This is the approach taken in SCoT-DC, using the Nuance speech recognition system’s natural language interpretation component to produce a series of slots and fillers for relevant factors in answers to why questions (Bratt et al., 2007).

Table 3
Open-Ended Questions asked by SCoT-DC in 2004, 2005 Experiments

<table>
<thead>
<tr>
<th>Question Type</th>
<th>Question</th>
<th>Examples of Students’ Correct Answers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Definition</td>
<td>First of all can you tell me what primary boundaries are?</td>
<td>First two bulkheads around the crisis</td>
</tr>
<tr>
<td></td>
<td>And now can you tell me what secondary boundaries are?</td>
<td>Second pair of bulkheads around the crisis</td>
</tr>
<tr>
<td>Why-Question</td>
<td>Why is it necessary to investigate after the alarm sounds?</td>
<td>To make sure it is not a false alarm</td>
</tr>
<tr>
<td></td>
<td>Why is it necessary to isolate when you have a report of fire?</td>
<td>To protect crucial ship systems</td>
</tr>
<tr>
<td></td>
<td>Why is it necessary to set fire boundaries when you have a report of fire?</td>
<td>Contain the fire</td>
</tr>
</tbody>
</table>

Handling the interpretation of spoken natural language for free, more variable speech in response to open-ended questions has been a particular concern. For example, Freedman (1997) discusses how the CIRCSIM-tutor interactions were designed to avoid difficulties in interpreting student responses to open-ended questions, guiding the student toward short answers that the ITS could better interpret and relate to its agenda. In this section, we provide data from the SCoT-DC experience about how challenging the answers to open-ended questions typically were for system development. Table 4 lists some general characteristics of our corpus of responses to open-ended why questions, separating out the students who were not familiar with damage control before encountering our system and its brief introduction to the domain from the Navy officers with shipboard experience who were training to be damage control assistants. The USNA students used the system in a classroom environment, with about 20 students at once using noise-canceling headsets while talking with the system. This environment gave them more temptation to choose answers to amuse nearby classmates (hence the “uncooperative” out-of-vocabulary (OOV) words), or to say I don’t know to make the system move through the topics quicker. Still, the mean utterance length for both groups was similar, around 5 words per utterance, which is longer than a brief phrase, but still on the manageable side for parsing and interpretation. There was considerable variation in the exact wording of the answers, and a fair variety of words used in those answers, both in-vocabulary (IV) and OOV.
There were 134 Stanford/USNA students who were asked open-ended questions, and the corpus includes 725 answers, of which 443 had distinct wording. The SWOS Navy officer corpus includes 9 SWOS officers who were asked open-ended questions, and 18 answers to those questions, with all 18 answers being distinct. 205 out of the 725 Stanford/USNA answers were variants of I don’t know. The SWOS officers never answered I don’t know.

Table 4
Why-question answer characteristics

<table>
<thead>
<tr>
<th></th>
<th>Stanford/USNA students</th>
<th>SWOS Navy officers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of subjects with open-end ques.</td>
<td>134</td>
<td>9</td>
</tr>
<tr>
<td>Number of answers</td>
<td>725</td>
<td>18</td>
</tr>
<tr>
<td>Number of distinct answers</td>
<td>443</td>
<td>18</td>
</tr>
<tr>
<td>Number of “don’t know”</td>
<td>205</td>
<td>0</td>
</tr>
<tr>
<td>Mean utt. length for answers</td>
<td>5.17</td>
<td>4.83</td>
</tr>
<tr>
<td>Standard deviation of utt. length</td>
<td>2.77</td>
<td>1.84</td>
</tr>
<tr>
<td>Number of distinct IV words</td>
<td>313</td>
<td>37</td>
</tr>
<tr>
<td>Number of distinct OOV words</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>Number of total OOV words</td>
<td>59</td>
<td>2</td>
</tr>
<tr>
<td>Number of uncooperative OOV words</td>
<td>26</td>
<td>0</td>
</tr>
</tbody>
</table>

Most of the answers involved verbs with noun phrase or prepositional phrase objects, and many of them involved complete sentences, often subordinate sentences beginning with because. This shows that the students constructed answers of medium complexity, as a start toward a complex tutorial dialogue.

Currently, the knowledge of what answers to why-questions are correct is encoded as production rules along with the other domain knowledge, but if the system develops and the dialogues increase in complexity and sophistication, it may become necessary to treat the answers with a constraint-based approach (Weerasinghe & Mitrovic, 2006; Mitrovic, & Ohlsson, 1999). For example, SCoT-DC might ask a student, with graphical highlighting on the ship display, Why did you fight the fire in this compartment before the fire over in this compartment? The student would be expected to describe concerns such as which important nearby compartments might be affected in each case, or the level that the repair team was tasked at, and so on. A constraint-based system could identify which of those concerns were valid in this particular simulation exercise, and whether the concerns given in this answer conflict with each other or with other actions taken in the exercise. Another possible question for a constraint-based tutor might be Do you see how you could have better preserved stability on the ship?, where the instructor’s example of the mostly filled compartment with water dangerously sloshing around could be addressed by topping it off to fill it, or by dewatering it. As a final example, the system could also ask the student How might you have assigned fires to repair teams differently

---

2 With more complex answers, more sophisticated natural language processing may become necessary as well, such as the robust parsing in Gemini, which can detect and interpret meaningful sentence fragments, if a complete logical form cannot be constructed (Dowding, Moore, Andry, & Moran, 1994, Niekrasz, Purver, Dowding, & Peters, 2005). An alternate approach for classifying long, complex answers would be Latent Semantic Analysis, as used by Graesser, Penumatsa, Ventura, Cai, and Hu (2007) in AutoTutor.
The student would then be expected to describe the tasking levels of the repair teams and their locations compared to the fires.

CONCLUSION

Simulators are integral to military training, and the way human instructors fit the simulation-based exercises into a larger context provides a model for intelligent tutoring systems. It is important that the student learn the precise control and reasoning about the effects of actions illustrated in the simulator, as is possible in a model-tracing tutor of a well-defined domain. However, it is also important for the students to be aware of the larger context and hypothetical variations, which do not need to be part of the same detailed knowledge system. Thus, even though the larger context involves freer language and wider possible student input, an intelligent tutoring system can have a reasonable chance of success looking for relevant factors in student answers, and applying constraints to ensure that the student’s reasoning and knowledge falls within acceptable bounds.

ACKNOWLEDGEMENTS

This work has been supported by the Department of the Navy under research grant N000140010660, a multidisciplinary university research initiative on natural language interaction with intelligent tutoring systems, research grant N000140510144, on spoken language coaching during dynamic problem solving and UCLA research grant N00014-06-1-0711 subaward 0070 G HC532-1 on training models and tools for adaptive learning. I have benefited from the researchers who collected these corpora along with their research groups: David C. Wilkins, Susan Kirschenbaum and Stanley Peters. I am grateful for the valuable domain information provided to our team by Perry McDowell, Lt. Jonathan Hopkins, Lt. Tyson Young, Ryan Heflin, Captain Bruce Barnaby, and Captain Bud Weeks (ret.). I also appreciate anonymous SWOS instructors and SWOS, USNA, and Stanford students who allowed our team to observe their instruction and coaching. I am responsible for any errors in my use or interpretation of their information.

REFERENCES


