Adaptive Tutoring for Self-Regulated Learning: A Tutorial on Tutoring Systems

by Robert A Sottilare and Anne M Sinatra

Approved for public release; distribution is unlimited.
NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position 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.

Destroy this report when it is no longer needed. Do not return it to the originator.
Adaptive Tutoring for Self-Regulated Learning:  
A Tutorial on Tutoring Systems

Robert A Sottilare and Anne M Sinatra
Human Research and Engineering Directorate, ARL
Adaptive Tutoring for Self-Regulated Learning: A Tutorial on Tutoring Systems

The tutorial described in this report was conducted at the Human-Computer Interaction International Augmented Cognition Conference in Creta Maris, Crete, Greece, in June 2014. The purpose of this tutorial was 3-fold: 1) familiarize participants with the fundamental principles of adaptive Intelligent Tutoring System (ITS) design; 2) illustrate how ITS design influences/enables self-regulated learning; and 3) discuss the need for standards for authoring of ITS, modeling of learners and experts, automated instructional strategies, and methods of analysis for ITS technologies.
## Contents

1. Tutorial Objectives 1
2. Fundamentals of Tutoring 1
3. Generalized Intelligent Framework for Tutoring 6
4. Learner Modeling 26
5. Instructional Management 42
6. Domain Modeling 48
7. Authoring 50
8. Analysis of Effect 63

Bibliography 67
Distribution List 72
INTENTIONALLY LEFT BLANK.
1. **Tutorial Objectives**

The purpose of this tutorial is 3-fold:

- Familiarize participants with the fundamental principles of adaptive Intelligent Tutoring System (ITS) design.
- Illustrate how ITS design influences/enables self-regulated learning (SRL).
- Discuss the need for standards for authoring of ITS, modeling of learners and experts, automated instructional strategies, and methods of analysis for ITS technologies.

2. **Fundamentals of Tutoring**

This section reviews the fundamentals of ITSs. Slide 3 illustrates a typical tutor-user interface used by the ITS to deliver content and feedback to the learner, and receive learner input. This interface may include a tutor natural language feedback window, which is used by the ITS to provide verbal feedback or direction through a virtual human. It usually provides a text feedback window, a content presentation window, and a learner response window where the learner provides text input in response to ITS directions, questions, or feedback. A running log of the conversation or chat window may also be part of the tutor-user interface.
Slide 4 illustrates the “sweet spot” for ITSs, the area where they may be most useful. This is within tutoring for complex skills versus simpler tasks. To support SRL, tutors adapt their delivery and the challenge level of scenarios to match those of the learner. If there is no adaptive change to match the learner’s needs, then this is simple computer-based training where the training is the same for everyone and not adaptive tutoring.

What is SRL? SRL is learning (acquisition of knowledge or skills) that is guided by metacognition (thinking about one’s thinking), strategic action (planning, monitoring, and evaluating personal progress against a standard), and motivation to learn. It takes great awareness and discipline to guide your own learning. Expert human tutors are much more effective and efficient at guiding learning. Just as expert human tutors guide learners, adaptive ITSs may also augment SRL by shaping instructional content and scaffolding support to meet the learner’s needs. The computer-based ITSs must be situationally aware of the learner’s state and the instructional context to be effective guides and support efficient instruction.

As noted in Slide 4, complex skills may include cognitive tasks where challenging decision-making and strategic thinking are exercised or affective tasks where interpersonal skills and ethical conduct are tested or psychomotor skills where coordination and timing are critical to physical tasks (e.g., land navigation) or operating sophisticated systems. ITSs may be used to prepare for live training, enhance learning in conjunction with virtual training environments, or act as job or decision aids during actual operations.

The primary reasons that tutors are limited in their use in education and training (see Slide 5) are 1) they have insufficient ability to adapt to learner needs and 2) they are expensive to author. It takes approximately 200 h of interdisciplinary team labor to make 1 h of coursework for tutoring. More adaptive equals more expensive.
Limitations of Current ITS

Expensive to author and are insufficiently adaptive* to support highly effective and tailored training experiences across a broad spectrum of military tasks.

- Adaptiveness drives the need for additional authoring... more authoring, more development time, more cost
- Adaptiveness is largely based on knowledge of learner performance... other attributes influence learning (e.g., individual differences)
- ITSs have been primarily applied in limited, well-defined domains... with limited application to military tasks

*adaptive systems customize themselves automatically in response to users or changes in the environment.

Significant evidence exists that computer-based intelligent tutoring systems are just as effective as human tutors under certain conditions and generally within more well-defined domains (e.g., mathematics, physics) where there is generally one correct answer to a problem. The goal is for ITSs to be as effective as expert human tutors under all conditions and domains. The effect sizes shown in Slide 6 are baselined against traditional classroom training. The ultimate goal for ITSs is to impact learning with effect sizes equivalent to raising average (“C”) students to experts (“A” students) through tailored instruction and reinforcement of deep learning principles. In other words, the goal is to be as effective as or more effective than expert human tutors.
Many organizations are feeling the crunch to achieve more with less. While training and education remain important to maintaining organizational competency, it may not be practical or feasible to achieve and maintain competency in a traditional classroom. One-to-one training (Bloom 1984) has been shown (Slide 7) to be more effective than classroom training, but it is not practical to have one-to-one tutoring for every person in a large organization.

So, if we wanted an ideal tutoring system that could adapt to our needs, what would it look like? Slide 8 shows a set of salient characteristics for a platinum-level tutor as described by Sottilare and Gilbert (2011).
Before we continue in our discussion of tutoring systems, we should examine their fundamental elements and processes (Slide 9). Nearly every tutoring system has 4 fundamental elements: a learner model, a pedagogical (instructional) model, a domain model, and a communication model. The green boxes in the slide below show these fundamental elements as *modules* versus models because they manage processes in addition to modeling the learner, the instruction, the domain, and the communication.

ITSs use the processes in Slide 9 to assess the learner and manage instruction. ITSs are defined as intelligent software-based agents that guide instruction by observing and interpreting learner data (behaviors, physiology, demographics) to classify learner states (e.g., engagement, competency, emotions). The ITS uses these states to adapt/tailor instructions to match the learner’s capabilities and needs to optimize learning. Slide 10 shows the interaction between tutoring agents, the learner, and the training environment (e.g., simulation, game, presentation).
Finally, the adaptive tutoring learning effect model shown in Slide 11 illustrates the interactive loops of the tutoring process. Learner module data/processes are shown in green boxes. Pedagogical module data/process are shown in light blue boxes and domain module processes in light orange.

### 3. Generalized Intelligent Framework for Tutoring

Traditionally, ITSs are expensive and time consuming to develop. Additionally, they tend to be linked to specific content and are not easily changed. Reuse in tutoring systems is virtually nonexistent. The Generalized Intelligent Framework for Tutoring (GIFT) is an open-source, domain-independent intelligent tutoring framework (Slide 13). It is intended to provide flexibility and the ability to create full tutors with content of the author’s choice. By being domain-independent, it allows for the reuse of materials and a reduction in both the time and monetary costs of developing tutoring systems.
An S&T effort to develop an open-source tutoring architecture to:

- capture best tutoring practices and support rapid authoring, reuse and interoperability of ITSs
- lower costs and entry skills needed to author ITSs
- enhance the adaptiveness of ITSs to support self-regulated learning (SRL) per the Army Learning Model

- ontology
- tools
- methods
- standards
- exemplars

Adaptive Tutoring Systems

- Adaptive
- Affordable
- Effective

- Gritty
- Flexible
- Collaborative
- Critical Thinkers

The fundamental tutoring processes within GIFT are consistent with the previously described components of the tutoring process, which are reviewed in Slide 14.

As with most ITSs, GIFT has 4 primary modular components: Domain Module, Learner Module, Pedagogical Module, and Gateway Module. Additionally, this diagram represents the process that occurs with user input to GIFT when the user is interacting with a training application. Additional information about these modules and their processes is provided further along in the “GIFT Tutors GIFT” portion of this tutorial. In addition to the 4 primary modules, GIFT also provides a sensor module that is used to capture behavioral and physiological data about the learner which can then be used to interpret/classify that data into learner states for use by the learner module and the pedagogical module as described in the “Adaptive Tutoring Learning Effect Chain”. The interaction of the modules is illustrated in Slide 15.
The modules in GIFT can be launched manually through the Module Monitor shown in Slide 16. The image on the left is the Module Monitor before the modules have been launched. The image on the right is the Module Monitor after the modules have been successfully launched. In GIFT 3.0 and above, it is also possible to launch the modules in one step, which includes the launching of the tutor web page.

Slide 17 provides a visual of the Admin Tools in the Module Monitor (left) and the GIFT login screen on the tutor web page (right). There is also a simple login page that can be used for experiments and does not require a password.
Within the Admin Tools Tab is a series of authoring tools. The GAT (GIFT Authoring Tool) is a new element of GIFT 2014-1-X, as is shown in Slide 18. It provides a user-friendly way for authors to create their courses in GIFT. The course selection dashboard is pictured in the current slide. Each line represents a current course within GIFT, and the green checkmark indicates that it is complete and validated. Courses can be opened by highlighting the name and then clicking “Edit”.

The functionality of the new GAT is demonstrated in Slide 19. The transitions and elements of the course are presented on the left side of the screen. Once one of transitions is selected, the associated fields to enter content into are displayed on the right side of the screen. Additional transitions can be added using the menu on the top left of the page.
Surveys are authored in GIFT through the Survey Authoring System. It provides the ability to create questions and surveys. Additionally, it can be programmed such that the surveys are automatically graded.

The next set of slides (pp. 11–15) provides an animated sequence that explains GIFT in simple terms. GIFT has 3 distinct functions. First, it is an authoring capability to develop new ITS components and whole tutoring systems. Second, it’s an instructional manager that integrates selected tutoring principles and strategies for use in ITSs. Third, it’s an experimental test bed to analyze the effectiveness and impact of computer-based tutoring systems components, tools, and methods. GIFT is intended to be a community platform for you to contribute to and to help you with your research and development.
GIFT is based on a learner-centric approach and is therefore designed to be consistent with the Adaptive Tutoring Learning Effect Chain. In this chain, the learner data informs learner states, which then informs instructional strategy. Selection of the appropriate strategy at the correct time is expected to lead to learning gains.

Each of these processes is captured in individual modules of GIFT. For this tutorial, we will go through the functions, inputs, and outputs of each GIFT module. Let’s talk about the first one, the Domain Module. Although the Domain Module is one module in reality, it is divided into 2 modules for you to better understand each of its 2 functions: performance assessment and strategy implementation.

The learning effect chain starts by monitoring learner performance and sensor data. Learner performance is assessed from data captured in a training application, which is fed into GIFT’s Domain Module via the Gateway Module. GIFT processes learner interaction data by comparing the Domain Module inputs against designated models of expert performance. In order to keep GIFT generic to all training applications, domains are represented as a hierarchical structure of “Concepts”.

GIFT is... an Intelligent Tutoring System
...an experimental testbed
...an architecture
...a research nexus
...a community platform

... with tools for authoring
... with data extraction/logging
... with interchangeable parts
... as an exit vector for research projects
... informed by Advisory Boards
• This tutorial will show you GIFT’s Main Modules and Processes.
• The implementation of GIFT is centered around the Adaptive Tutoring Learning Effect Chain (ATLEC). Each component is intended to inform strategies to influence learning gains.

For all identified Concepts and Subconcepts, a learner can perform “at standard”, “above standard”, or “below standard”, as determined by comparison of performance to the expert model. Outputs from the Domain Module are fed into the Learner Module and contain performance states associated with specific concepts represented in the domain representation.

GIFT also uses information collected from sensors to monitor a learner’s reactive states while interacting with a system. The Sensor Module takes in raw sensor data streams as inputs and applies filters to convert the data into metrics correlated with cognitive and affective states. This filtered data is passed to the Learner Module for inferring affective and cognitive state determinations. You wouldn’t use the same instruction for a bored student as an energetic one, would you? Sensors may be used to collect data about the learner and determine the learner’s states. Data from the learner may be filtered or unfiltered.
So far we’ve learned that performance information from the Domain Module and sensor information from the Sensor Module serve as inputs to the next function of the learning effect chain, the Learner Module. The Learner Module consumes this information and performs processes to designate a “Learner State”, which may be cognitive (e.g., engagement), affective (e.g., emotions), or physical (e.g., fatigue or arousal). Once a state is classified, the results are then fed to the Pedagogical Module. In addition, the Learner Module maintains information about the learner’s stable traits, experiences, and competencies, which are also used to customize strategy selections for the individual learner.

The Pedagogical Module uses the “Picture of the Learner” state information and learner trait data, and then recommends generic strategies to accommodate individual learners. If the student is bored and passing your course, then you probably want to adjust the course. For instance, the Pedagogical Module may provide guidance and feedback to aid in performance, adjust course difficulty, present more interactive material, or request an assessment of the learner’s ability.
Once the Pedagogical Module uses learner state information to select a strategy, the strategy selected is then sent to the Domain Module. The Domain Module then chooses an appropriate method to implement the request as a defined tactic or action. The selection of the tactic is based on the learner’s ability levels. For example, it may be more appropriate for one learner to receive a hint while a metacognitive prompt may be more suitable for another.

The Gateway Module receives the tactic that was output by the Domain Module and presents it to the learner. Depending on the type of request, this can involve actions to execute tactics within the training application environment (e.g., serious game) or by the Tutor User Interface, a window to present information to and receive information from the learner.
The Gateway Module translates this interface into GIFT-assessable messages, allowing for the whole learning effect to operate. If you already have a training application, the Gateway Module is where you can build an interface to it and add individualized and research-based instructional strategies.

All of the modules communicate to each other via defined protocols. This means that you can easily replace one version of a module with another, which assists in setting up experiments and training with a variety of applications. You can make one module and know with confidence that the rest of the modules will do their duty. Research done for GIFT can make its way back to the community.

You can download GIFT for free (for life) and get support for your development at www.gifttutoring.org.
Slides 30–48 discuss the various releases of GIFT and their affiliated capabilities.

GIFT has been continuously developing and growing since its initial release in May 2012. The initial release included the fundamental modules and elements that are still part of the current releases. An initial version of the Module Monitor is pictured in Slide 31.

The initial release also included the ability to record information from sensors, such as EEG (electroencephalography), a temperature/humidity-sensitive mouse, and a self-assessment sensor.
• Sensors
  – EEG – Emotiv Epoch
  – Temperature and humidity-sensitive mouse
  – Software-based surrogate sensor
    • sensitivity testing

Research has been performed with the various releases of GIFT. An initial experiment was performed to assess low-cost sensors and their impact on learning cognition and affect. This work helped to inform the future direction of sensors within GIFT. Work is consistently being done with GIFT to improve and add to its capabilities.

The initial release included support for Virtual BattleSpace 2, a serious game, as an external training application.
Additional features are being consistently added to GIFT based on both research objectives and user input. GIFT 2.0 was upgraded by including a number of different tools that are of use to course designers and researchers.

Additional sensors were added to GIFT 2.0. Once an interface for a commercial sensor is added to GIFT, the GIFT community benefits by never having to integrate that sensor again.
GIFT 2.0 also included the addition of Microsoft PowerPoint as a training application. This addition is extremely useful, as it makes it easier to harness reusable content in the form of PowerPoint presentations. Additionally, PowerPoint is a program that individuals of many skill levels are familiar with, and it increases the flexibility that GIFT course designers have with their included material.

The photo in Slide 38 demonstrates an experiment in action. GIFT is visible on the left side of the screen, with the training application on the right. Additionally, the participant is wearing the Q-sensor, which measures electrodermal activity.
GIFT 3.0 included tools such as SIMILE, which made the ability to link real-time interactions to learning objectives more straightforward for GIFT course authors. In its present form, SIMILE works for TC3Sim/Vmedic; however, future work is being done that will allow it to work with additional training applications. Further, the 3.0 release included the engine for Macro and Micro Adaptive Pedagogy (eM2AP) for managing instruction and elements of AutoTutor through dialogues and tutoring scenarios.

GIFT 3.0 included a very useful feature: the ability to export a created tutor. This allows more flexibility in the way that an individual student will interact with GIFT. Additionally, a full installation of GIFT no longer needs to be completed on each computer that will be running the developed tutor.
The Human Affect Recording Tool (HART) was included with the release and can be used by researchers. HART is an app for the Android platform that implements the Baker-Rodrigo Observation Method Protocol (BROMP) 1.0, a method for assessing human affect and engagement in field settings, allowing for synchronization between field observations and log files of student-software interaction.

HART is described in Baker et al. (2012) and implements the protocol described in Ocumpaugh et al. (2012).

Additionally, the ability to track facial expressions, posture, and head pose was added with the integration of the Microsoft Kinect as a sensor.

GIFT 3.0 also included additional example courses that used TC3Sim/Vmedic. Additional research was conducted using GIFT as a test bed to examine the self-reference effect in context.
of cognitive tasks in a computerized learning environment. The materials that were used in this study were released as an additional course titled “Logic Puzzle Tutorial” in GIFT 4.0.

GIFT 4.0 also included an enhanced version of eM2AP, additional sensor filters, and integration with new training domains (Marksmanship and Logic Grid Puzzles) based on experimentation with GIFT 3.0.

As mentioned previously, a psychomotor training domain, Marksmanship, was facilitated through capabilities in GIFT 4.0. This allows for the development of expert marksmanship models. Slide 44 displays the interaction between the different GIFT modules and the marksman (learner).
Further, GIFT 4.0 is being used to assess future capabilities. As it continues to develop, GIFT will be adapted to support use by multiple learners in a server-based environment. Further, GIFT is being integrated into a mobile learning environment (The US Army Research Laboratory’s [ARL’s] Soldier Centered Army Learning Environment [SCALE]) as a learning engine.

Research efforts are ongoing to mature architecture, authoring tools, and modeling of the learner. As they mature, these capabilities are integrated within GIFT.
GIFT 2014-2 and future versions are moving toward separate interfaces and tools for different users (learners, power users, and domain experts). In addition, there will be different configurations for different situations/environments (e.g., classrooms, experiments, mobile learning).

ARL has the goal for GIFT to be a learning engine for several types of training environments. Targets of opportunity include the US Department of Energy’s National Training & Education Resource, the US Advanced Distributed Learning Initiative’s Personalized Assistant for Learning (PAL), and ARL’s SCALE.
• Continue to leverage AutoTutor and AutorTutor Lite functionality
  – Speech Act Analysis, Complex Dialog Management, Artificial Intelligence Markup Language, Learner Characteristics Curves

• Enhanced cognition and affect detectors
• Extend learning environments to military training domains (complex and ill-defined domains)
• GIFT as learning engine for PAL

Demo of GIFT bootup
4. Learner Modeling

There are 2 key steps in learner modeling: data acquisition and state classification. Learner data, as noted in the adaptive tutoring learning effect model, may be acquired in real-time from sensors or learner input (e.g., surveys, interaction with training environment) or in pretutoring from a long-term learner model (LTLM) sometimes called a persistent learner model. Slide 51 examines low-cost sensors for acquiring learner data.

Slide 52 reviews approaches for using learner data to classify learner states (e.g., emotions, engagement, performance, competence). Of particular note are the preliminary findings on this slide.
The learner states of interest during tutoring are primarily centered in 3 areas that moderate learning (acquisition of knowledge and skills): cognition, motivation, and affect (see Slide 53).

The results of an experiment that sampled low-cost sensors and their ability to reliably detect learner states is shown in Slide 54.

Various machine learning techniques have been evaluated for use in classifying learner states. Slide 55 illustrates a clustering technique called “growing neural gas”. Learner data may be labeled (supervised), unlabeled (unsupervised) or semi-supervised. If you are interested in machine learning techniques, check out WEKA, an open-source software tool for machine learning: http://www.cs.waikato.ac.nz/ml/weka/.
In an effort to develop ITS best practices for learner modeling, ARL convened a group of experts in 2012 to examine the state of practice, emerging concepts, and future directions. This information was captured in the first volume of the Design Recommendations for ITSs, which is available for free at: https://gifttutoring.org/documents/42.

If you were to build a learner model, what kinds of data would you want to capture and what states would you want to classify/detect?
What should be in a learner model?

Don’t look ahead...
Make your own list

Learner data, states, traits, and demographics may be useful in the ITS decision processes, but data are generally expensive to collect and maintain. So, we want to be selective about which data we choose for our model.

LTLMs may span careers or lifetimes of learners and are used to store enduring, variable, and transient characteristics of the learner, including traits, states, and demographics.
Slide 60 shows some student (learner) data that is separated into general information and recorded behaviors.

Learner characteristics may be enduring, but even enduring characteristics may change over a long period of time. For this reason, the label of long-term learner model may be more appropriate than persistent learner model.
Enduring Characteristics... maybe?

- Gender
- Culture First language
- Physical constraints (e.g., color blind/deaf)
- Qualifications/certifications
- Work History
- Education achievements and history
- Transcripts
- Affiliations
- Values
- Personality

Slide 62 lists learner characteristics that are variable but fairly persistent.

Variable, but fairly persistent long term...

- Second language(s)
- Domain Competencies
- Security Clearance level
- Inductive Reasoning Capacity
- Working memory Capacity
- Divergent Associative Learning Capacity (ability to connect new to prior knowledge)
- Learner contact info; identifier
- Learner relations (e.g., teacher/class id)
- Learner security credentials (e.g., password)
- Learner preference information
- Learner portfolio information
Slide 63 lists transient characteristics that may be worth tracking/detecting.

Transient
- Current task
- Current time constraints
- Current affect
- Current goals
- Current location (geographical)
- Current time of day/week
- Current competency model (within learning experience)
- Current context (e.g., meeting/car, work, home, etc.)
- Physiological state

We just discussed what might be useful to include in an individual learner model. What should be in a model of a team of learners?

What should be in a team model?

Don’t look ahead...
Make your own list
Adaptive Tutoring Learning Effect Chain for Teams

Team Models:
- Competency
- Performance
- Cognitive State
- Selection State
- Test
- Communication


Team Model Development Process

- Structured Literature Review
  - Individual Tutoring
  - Team Performance
- Model Constructs
  - What models are needed?
  - What variables influence each model?
  - What variables are observable/unobservable?
  - What is the effect size of each variable?
  - How do we measure critical variables?
- Structural Equation Models
- Model – Test – Model
In 2013, an extensive review of the literature was undertaken relative to the target team models identified by Sottilare et al. (2011) and later described by Sottilare (2013). A sample of the outcome for the performance model is shown in Slide 68.

The results of the team performance model literature review revealed the following antecedents (influencers) of team performance (see Slide 69).
Examining antecedents to effective team performance

<table>
<thead>
<tr>
<th>Outcome</th>
<th>K</th>
<th>N²</th>
<th>Ro</th>
<th>Rho</th>
<th>% Variance accounted for</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overall</td>
<td>183</td>
<td>12910</td>
<td>.18</td>
<td>0.19*</td>
<td>30.0%</td>
</tr>
<tr>
<td>Communication</td>
<td>48</td>
<td>3367</td>
<td>.23</td>
<td>0.26*</td>
<td>26.2%</td>
</tr>
<tr>
<td>Coordination</td>
<td>25</td>
<td>1798</td>
<td>.21</td>
<td>0.23*</td>
<td>28.8%</td>
</tr>
<tr>
<td>Conflict</td>
<td>32</td>
<td>2061</td>
<td>-0.08</td>
<td>-0.09*</td>
<td>23.3%</td>
</tr>
<tr>
<td>Coaching/Leadership</td>
<td>50</td>
<td>3863</td>
<td>.22</td>
<td>0.24*</td>
<td>33.3%</td>
</tr>
</tbody>
</table>

1. K is the number of effect studies analyzed for this outcome
2. N is the number of individuals evaluated in this outcome
3. Rho is the corrected effect size, Ro is the uncorrected effect size; both weighted for sample size

*Statistically significant; confidence interval excludes zero

The process of identifying antecedents for each of the primary team models continued, and these are being used to construct initial team models for GIFT. Moving forward, we will examine the relationship between individual and team modeling. Slide 70 shows the interaction between the tutoring agents, the training environment, and the learner. Tutoring agents for teams may have similar functions but will track progress toward team objectives.
You have seen this diagram before—it is the individual tutor loop.

The tutor loop in Slide 72 shows a distributed or server-based interaction between an individual learner and the tutor. How will this need to be modified for teams?

Going back to the agent-environment-learner interaction for a moment, Slide 73 shows interaction between a team of 5 learners and the tutoring agents, a team of 5 learners and training environment, and finally, what is different in this diagram, interaction between learners who must be accounted for in the team models of the tutor.
Slide 74 shows individual learners interacting with their own tutor loops (something must moderate/manage the team interaction), interaction between the learners (communication, trust, performance, shared mental models [cognition, affect]), and how competency influences the other models.

Team modeling has evolved into 6 state models with consideration for 2 additional models for context and culture pending the determination of their influence on team performance.
Each of the team models is reviewed in Slides 76–82.
Team Competency State Model

- **Surface Level Composition**
  Individual team member characteristics that have the potential to impact or influence the accomplishment of team goals.

- **Deep Level Composition**
  Individual knowledge, skills, abilities, and experience in the domain are moderators of success.

Interdependence of tasks and matching of skills and roles may also influence the accomplishment of team goals.

Compatible values (e.g., culture) and personality may also be moderators.

This model provides an index of team competency based on a composite of the competence levels of individual team members. Successful/unsuccessful performance influences individual competence and may influence team competence. Any significant changes in individual performance of team tasks are assessed by the individual competency state model to determine if the threshold has been met to change individual competency (e.g., beginner, journeyman or expert). Changes in individual competency may or may not be of sufficient significance to affect a change in the team competency state model, but if a change in the team competency state model occurs, a message is generated to update the team competency state models of the other team members.

See Fletcher and Sottilare (2013) for additional insight on Shared Mental Models.

Team Cognitive State Models

- **Team (Shared) Mental Models**
  An evaluation of the shared cognitive state of all team members. This evaluation can include the degree to which team members:
  
  a) structure knowledge in a similar manner
  b) understand the roles, expertise, and expectations of fellow team members
  c) have a shared impression of several aspects of the team’s status (e.g., slack resources, progression towards team objectives)

This state model is a compound model of the Cognitive State of all team members. Cognitive State models already exist as part of the individual ITS, but synchronization of this information with all the team member’s ITS is critical in assessing the function of the team. Depending on the collaborative task and the roles of team members in accomplishing that task, the cognitive model may be key in determining instructional strategies. For example, for some tasks the weakest understanding of the task among team members may indicate the risk of completing the task successfully. For other tasks, only key team members may need to have higher understanding of the task to reach a successful outcome.
Team Affective State Models

- Attitudes (not observable)
  - Collective Efficacy
  - Cohesion
  - Justice
  - Effort
  - Cooperation
  - Trust (next slide)

Values-Attitudes-Behaviors

This state model is a compound model of the Affective State of all team members. Affective State models already exist as part of the individual ITS, but distribution of this information to all the team members ITS is critical in assessing the function of the team. For example, if team performance is below expectations and the affective state of one or more team members is negative, knowledge of their state by other individual ITS provides the opportunity to prompt their associated team members to take action (e.g., communicate – support or direct).

Team Trust State Models

- Trust
- Credibility
- Psychological Safety (Openness)

This state model is a compound model of the Affective State of all team members. Affective State models already exist as part of the individual ITS, but distribution of this information to all the team members ITS is critical in assessing the function of the team. For example, if team performance is below expectations and the affective state of one or more team members is negative, knowledge of their state by other individual ITS provides the opportunity to prompt their associated team members to take action (e.g., communicate – support or direct).

The trust relationships are bi-directional in that Team Member “A” may trust Team Member “B” more, the same or less than Team Member “B” trusts Team Member “A”. Trust is influenced by several factors including perceived competency, perceived integrity, perceived benevolence, knowledge of the other team members (Hung, Dennis and Robert, 2004) and perceived benefits of the relationship (Gyulay, DeAngela, Fulsem and Barber, 2006). Since teams work toward common goals where roles and responsibilities are distributed, perceived competency is an essential element of team performance. The perception that other team members may be unable to perform their tasks is detrimental to trust and team performance. Personality may also play a part in trust. Individuals with low openness and/or high neuroticism scores in the Five Factor Model of Personality (McCrae and Costa, 1994) may have developed habits unfavorable to the development of trust. Low openness scores might indicate an unwillingness to disclose information while high neuroticism scores might result in more frequent perception of events/interactions as negative. Positive or negative emotions can also influence the assimilation of information (Linnenbrink and Pintrich, 2002) and thereby communications, understanding and trust.
Team Communications
State Models

- Coordination
  - Mutual Support
  - Reflexivity
  - Monitoring
- Conflict
  - Task Conflict
  - Relationship Conflict
- Leadership
  - Leadership Styles
  - Shared Leadership
  - Leadership Behaviors
- Communication (frequency, source, receiver(s))
- Transition Processes
- Action Processes
- Interpersonal Processes
- Conflict Management
- Organizational Citizenship Behaviors (OCBs)

Observable behaviors between group members which either directly impact progression towards task completion or indirectly facilitate synchronization between team members.

This model is composed of interaction data between team members for the purpose of observing team cohesion and task execution. Providing accurate information in accordance with operating procedures, providing communications when asked, repeating communications to ensure delivery, sharing information and acknowledging receipt of information are all vital actions observed in teams with effective communication skills (U.S. Coast Guard, 1998). In team settings communication among members builds holistic situational awareness and coordinates future actions to be carried out. Based on events and interactions in a scenario, team members are responsible for updating one another in real-time.

Other Influencing Factors

- Context
  - Task or Project Characteristics
  - Autonomy
  - Interdependence
  - Familiarity
  - Team Tenure
  - Organization Type
  - Leader Characteristics (Age/Tenure)
  - Organizational Resources
- Culture
  - Individual Culture
  - Team Culture
  - Organizational Culture
  - National Culture
  - Team Climate
  - Organizational Climate
  - Diversity

- Coordination
  - Mutual Support
  - Reflexivity
  - Monitoring
- Conflict
  - Task Conflict
  - Relationship Conflict
- Leadership
  - Leadership Styles
  - Shared Leadership
  - Leadership Behaviors
The team model development process is shown in Slide 83.

- **Structured Literature Review**
  - Individual Tutoring
  - Team Performance
- **Model Constructs**
  - What models are needed?
  - What variables influence each model?
  - What variables are observable/unobservable?
  - What is the effect size of each variable?
  - How do we measure critical variables?
- **Structural Equation Models**
- **Model – Test – Model**

5. **Instructional Management**

Good instructional practices are hard to replicate. Modeling expert human tutors is a good place to start.
Dialogue-based tutors take turns interacting with the learner (see Slide 86) to improve the quality of their answers without giving them the answer. Dialogue-based tutors guide the learner and assess their understanding of concepts. GIFT has incorporated many of the web-based services used in AutoTutor and AutoTutor Lite to support dialogue-based tutoring.

SIMILE is a tool to link learner actions in a simulation or game to performance assessments and ultimately to tutor decisions about instructional options (e.g., feedback, support, change in challenge level).

The relationship between the tutor and the learner is critical to the learner’s engagement and motivation. The learner should perceive the tutor as credible and supportive for significant learning to occur. While this might not be important for single exposures of the learner to the tutor, over the long term, it is critical for the learner to develop a relationship with the technology.
Another approach to managing instruction is to categorize instructional differences in different learning domains. Research is ongoing to determine generalized instructional management methods across and within learning domains.

Each of the learning categories (cognitive = thinking, affective = feeling, and psychomotor = doing) is reviewed in Slides 90–93.
Cognitive Learning

- Cognitive learning (thinking) – Anderson & Krathwohl, 2001
  - *behaviors indicating increasingly complex and abstract mental capabilities*
  - Creating (high): ability to put parts together to form a new whole
  - Evaluating: ability to judge the value of learned material
  - Analyzing: ability to break down material into its component parts
  - Applying: ability to use learned material in new situations
  - Understanding: ability to grasp the meaning of material
  - Remembering (low): ability to recall previously learned material


Affective Learning

- Affective learning (feeling) – Krathwohl, Bloom and Masia, 1964
  - *behaviors indicating emotional growth; the manner in which we handle emotions, such as feelings, values, appreciation, enthusiasms, motivations, and attitude*
  - Characterizing (commitment - high): has a value system that controls their behavior
  - Organizing (responsibility): organizes values into priorities; comparing, relating and synthesizing values
  - Valuing (appreciation): the worth or value a person attaches to a particular object, phenomenon, or behavior
  - Responding (interest): active participation on the part of the learner
  - Receiving (awareness - low): awareness, willingness to hear, selected attention

Psychomotor Learning

- **Psychomotor learning (doing)** – Simpson, 1972
  - Includes physical movement, coordination, and the use of the motor-skill areas; development of these skills requires practice and is measured in terms of speed, precision, distance, procedures, or techniques in execution.

- **Origination (high)**: creating new movement patterns to fit a particular situation
- **Adaptation**: skills well developed and can be modified to fit special requirements
- **Complex Overt Response**: skillful performance of complex movements
- **Mechanisms**: learned responses have become habitual
- **Response**: early stages in learning complex skill; imitation; trial & error
- **Set**: readiness to act
- **Perception (low)**: ability to use sensory cues to guide motor activity

Another approach to enhanced adaptive instruction is to implement best practices based on learning theories. Two theories are Component Display Theory (Merrill et al. 1992) and the Zone of Proximal Development (Vygotsky 1978), which are described in Slides 94 and 95, respectively.
Component Display Theory
(Merrill, et al, 1992)

- Strategies that account for the learner's state:
  - Gain attention and motivate
  - Adapt to prior knowledge
  - Adapt to type of knowledge being presented
  - Adapt to learner attributes
  - Adapt to the learner's ability (IQ, EQ, adaptability…)


Automated Instruction
Zone of Proximal Development

What Is It?
- Tools and methods to adapt instruction based on the learner's state (e.g., cognitive and affective) and level of domain competence

What Does It Offer?
- The ability to select real-time instructional strategies which automatically lower/raise the training scenario challenge level and regulate the type and frequency of scaffolding (support) provided to the learner

Technical Approach:
- Literature Review to identify negative learning states (e.g., anxiety, boredom, frustration) and associated effective strategies
- Develop classification rules for incorporation within GIFT (*eM2AP*)
- Develop a Markov Decision Processes to address uncertainty associated with learner state classification

*eM2AP* = engine for Macro & Micro-Adaptive Pedagogy
6. Domain Modeling

While domain modeling is primarily associated with presentation of content to the learner, it also includes the following aspects: modeling of experts (also known as an ideal student model), which is used to assess the progress of the learner and identify errors and misconceptions; assessment of performance; and presentation of tactics (actions by the tutor to present content, change content, provide feedback, or provide support).

Domain models for typical ITSs today are in well-defined domains (math or physics are popular) and are generally procedural in nature (simple). Very few tutors cover psychomotor domains and are generally desktop and cognitive. ARL is exploring methods to expand domains to allow them to support tutoring in a wider range of domains and a broader range of dimensions, as shown in Slide 98.
Dynamic interaction modes and their associated characteristics are explored in Slides 99 and 100.
Domain Modeling Interaction Modes

What Is It?
- Research, tools and methods to facilitate interaction with learners and delivery of instruction during a broad spectrum of military tasks ranging from static to highly dynamic

What Does It Offer?
- The ability to support effective tutoring during Soldier training in increasingly dynamic training modes:
  - static (desktop training)
  - limited dynamic (adaptive marksmanship training)
  - enhanced dynamic (multi-learner tasks in instrumented spaces)
  - in-the-wild (instrumented learners)

Technical Approach:
- Evaluate interaction required to support each training mode and discover methods to facilitate:
  - capture of learner and environment data
  - assessment of learner state
  - selection of optimal instructional strategies and tactics
  - presentation of tactics (e.g., feedback, direction)

Evaluation of interaction is a necessary precursor to support tutoring in dynamic military training domains (e.g., embedded training, mixed and augmented reality training, and live training)

An example of a more dynamic psychomotor domain is presented in Slide 101.

Exemplar: Adaptive Marksmanship

STRI

Expanding tutoring capabilities to support complex and ill-defined Army training domains

7. Authoring

Authoring may be divided into 2 primary areas: reuse and automation. Improving methods to be able to use existing training environments as adaptive tutors is a major goal of our research. Automated authoring methods are the most critical need in ITS development today. Tutors are expensive and laborious to author, and they require sets of specialized skills to develop them. A long-term goal is to have any person with expert domain knowledge be able to author an effective tutoring system.
Authoring Goals

Authoring Goals for GIFT
(adapted from Murray, 1999; Murray, 2003)

• Decrease the effort (time, cost, and/or other resources) for authoring and assessing CBTS;
• Decrease the skill threshold by tailoring tools for specific disciplines to author, assess and employ CBTS;
• Provide tools to aid the designer/author/trainer/researcher organize their knowledge;
• Support (i.e., structure, recommend, or enforce) good design principles (in pedagogy, user interface, etc.);
• Enable rapid prototyping of CBTS to allow for rapid design/evaluation cycles of prototype capabilities.
• Employ standards to support rapid integration of external training/tutoring environments (e.g., games) (Sottilare & Gilbert, 2011)


Authoring Approaches

• Approach: Investigate methods and develop tools for rapid automated development of expert models
• Expert model development is an intensive process
• Exploring automated methods to author via text analysis
• Integrate with tutoring system(s) which make instructionally relevant decisions
• **Approach: functional user modeling**

  - standard structures and graphical user interfaces for a variety of users
  - GUIs based on function (e.g., researcher) and functional competency
    - learners
    - subject matter experts
    - instructional system designers
    - system developers
    - trainers
    - researchers
  
• **Approach: learner affect modeling**

  - what does the tutor need to know about the learner to classify their affect?
  - how does the tutor get that information?
  - which affective states are important to recognize?
  - how does classification of state influence instructional decisions?
**Authoring Approaches**

**Approach: learner configuration authoring tool**

- Simple interface for authoring learner models
- Tree structure driven by XML schema
- Prevents learner model authoring errors by validating against the learner model XML schema
- Provides ability to validate learner model using GIFT source w/o having to launch the entire GIFT architecture

**Approach: sensor configuration authoring tool**

- Implemented sensors
  - Affectiva DSensor
    - Electro-dermal activity (EDA)
    - Skin temperature and acceleration
  - Emotiv EEG
  - Temperature and humidity mouse (custom)
  - Surrogate sensors for temp, humidity and assessment

- Behavioral sensors
- Physiological sensors
- State classification models
- Sensors under consideration
  - NeuroSky and ABM EEGs
  - Webcam (1Hz)
  - Zephyr heart rate monitor
  - Sonar distance sensor
  - Pressure chair (custom)
  - Pupil diameter (custom)
  - Design Interactive EmoPro
Research question: what is the minimum set of sensors needed to assess engagement, workload, motivational level and emotional state?

- electro-dermal activity
- temperature
- X, Y, and Z acceleration

Approach: Domain Knowledge File (DKF) authoring tool (DAT)

- simple interface for authoring DKFs
- tree structure driven by XML schema
- prevents DKF authoring errors by validating against DKF XML schema
- provides ability to validate DKF content using GIFT source w/o having to launch the entire GIFT architecture
1. **Approach:** survey authoring tool
   - author questions
   - author surveys
   - assign surveys
   - present surveys

2. **Approach:** leverage elements of existing tutoring systems (e.g., AutoTutor & AutoTutor Lite)
**Authoring Approaches**

- **Approach: game-based tutoring**

  - Game World
  - Learner

  **Tutoring Agent(s)**

  - agent observes world
  - agent acts to change world
  - learner acts on world
  - learner observes world
  - agent acts to provide feedback or instruction
  - agent observes effect on game objectives
  - agent observes learner

  **Prototype Integration with VBS2**

  - real-time feedback
  - learner model influences challenge level within game
Game-based Tutoring

Automated Authoring Standards and Reuse

- Authoring & components
  - User models
  - Domain-specific knowledge
  - Instructional strategies
  - User-tutor interfaces
  - Tutor compiler

Standards
Define data structures and interface protocols for common tutoring components

Promote Reuse
Automate development of interfaces with external training & authoring capabilities to support interoperability and reuse

Existing Simulators
- Engagement Skills Trainer
- Virtual Humans

Existing Tutoring Systems
- AutoTutor Web Services
- Existing Content
- PowerPoint

Existing Content Authoring Tools
- VBS2
- SPA
**What Is It?**
- Tools, and methods to **automate** the development of expert models (modeling desired trainee behaviors) for use by adaptive tutors;
- Expert models, part of the GIFT domain module, are used to assess learner performance and the correctness of learner actions during tutoring

**What Does It Offer?**
- Reduced time, cost, and skill needed to develop expert models for training domains

**Technical Approach:**
- Investigate and develop methods to automate expert model authoring by extracting rules, principles, tasks, standards, conditions and hierarchical relationships from text in field manuals and other text-based data sources through data mining techniques

---

**Authoring Reuse: Interoperability with External Environments**

**DOE's National Training & Education Resource (NTER)**
Tools for Rapid Development of Expert Models (TRADEM)
Tools for Rapid Development of Expert Models (TRADEM)

GIFT-compatible Tutor or Chatbot Tutor

Situated Pedagogical Authoring

- What is it?
  - Providing ITS content in an environment that is as similar to the learner’s environment as possible
  - think WYSIWYG

- Hypotheses: Situated authoring will enable authors to...
  - Gain competence with the authoring tool more quickly
  - Produce more complete and pedagogically effective intelligent tutoring content

(than less situated authoring tools)
8. Analysis of Effect

An important part of ITS research is to determine the best (most effective) methods of learner modeling, instructional management (pedagogical strategies/recommendations), and domain modeling (e.g., expert modeling, assessment, and content and tactics presentation) to optimize outcomes (e.g., learning, performance, retention, competence).

How might this research and analysis be enabled?
The analysis test bed within GIFT allows for scientific evaluation of each element and subelements of a tutoring system design. The test-bed methodology shown in Slide 130 is based on Hanks et al.’s (1993) test-bed approach. Effect sizes analyzed using this test bed are based on Cohen’s $d$ (1992).
Demo of GIFT Tools
Thank you for your attention!

Questions?
Bibliography


Goldberg B. Explicit feedback within game-based training: examining the influence of source modality effects on interaction. [doctoral dissertation]. [Orlando (FL)]: University of Central Florida; 2013.


Holden H, Goldberg B. Student expectations and tutor acceptance: the impact on student affect (mood) and future usage intentions. international defense & homeland security simulation workshop. In: Proceedings of the I3M Conference. 2011 Sep; Rome, Italy.


Sottilare R, Gilbert S. Considerations for tutoring, cognitive modeling, authoring and interaction design in serious games. Authoring Simulation and Game-Based Intelligent Tutoring workshop at the Artificial Intelligence in Education Conference (AIED); 2011 Jun; Auckland, New Zealand.


INTENTIONALLY LEFT BLANK.