Cognitive Media and Hypermedia Learning Environment Design: A GOMS Model Analysis

Terry Shikano, School of Psychology, Georgia Institute of Technology, Atlanta, Georgia 30332, U.S.A.

Mimi M. Recker, Box 600, Victoria University of Wellington, Wellington, New Zealand

Ashwin Ram, College of Computing, Georgia Institute of Technology, Atlanta, Georgia 30332, U.S.A.

Abstract. In our research, we have been developing a design framework for educational multimedia, based on the cognitive aspects of the users of that information. Design based on "cognitive media" appeals to the particular cognitive aspects of learners, whereas design based on types of "physical media" appeals to particular sensory modalities. This framework informed the design of AlgoNet, a computer science educational hypermedia system that used cognitive media as its basic building blocks. In this paper, we describe a model of student usage and learning with AlgoNet. This model, using the GOMS methodology, provided a useful description of the procedural knowledge required to interact with the AlgoNet system. In addition, our implemented simulations provided estimates of learning and execution times for several instances of the model. Together, the parameters in the simulations and their resulting estimates help clarify the impact of system design, and hence our design framework, on students’ browsing and learning strategies.

INTRODUCTION

In recent years, the role of networked, interactive multimedia has become increasingly important in educational and training settings. Coupled with the recent growth of the Internet, educational multimedia is becoming a desktop reality. And, as argued by many throughout the educational community, it clearly affords educational benefits. However, it would be naive to assume that simply providing greater access to information leads to better learning and, hence, education. Instead, designers of educational multimedia require strong theories to help guide their design and implementation. These design theories need to be informed by an understanding of learning and reasoning within computational environments. In particular, designers need advice on the best ways to organize, present, and index multimedia information to maximize effective learning and knowledge construction by students.

In earlier work, we presented a design framework for organizing information and activities within educational hypermedia systems (Recker et al., 1995). As we will describe, this framework posits that such systems should not be characterised primarily in terms of the types of physical media (for example, text, video, sound) supported by computing systems. Instead, the important aspect is the content that can be represented within a physical medium. Methods of information access and usage should be based on cognitive aspects of the users of that information, in ways that support effective learning and reasoning strategies. These we characterise in terms of "cognitive media."

The evaluation of new educational materials, digital or otherwise, is a complex and multi-faceted process. To evaluate our framework of cognitive media, we have taken a 3-pronged approach, involving (1) design, (2) empirical evaluation, and (3) user model analysis. Specifically, we have developed a system called AlgoNet, which contains instructional material targeted at undergraduate computer science and engineering students. AlgoNet provides to students definitions, examples, exercises, case studies, etc., organized as cognitive media. In
this way, AlgoNet serves as a test of the validity of the cognitive principles upon which it is based.

Secondly, we conducted empirical studies with the system, involving over 100 university students (Recker et al., 1995). Our study involved analyzing the learning and usage patterns of students using AlgoNet, and comparing these to students who learned from a contrasting system that used physical media as its basic building blocks.

In this paper, we focus on the third component of our approach: a model analysis of the data collected in our empirical study. In particular, we used GOMS (Card, Moran, and Newell, 1983) to model the usage patterns of students in our study. Typically, AI models focus on simulating normative or one class of behavior. This model then provides a level of analysis and explanation of the observed phenomena. In our approach, however, we focused on simulating several classes of students interacting and learning from the two versions (cognitive and physical) of AlgoNet. As we will show, the GOMS model provided a useful task analysis and description of the procedural knowledge required to interact with AlgoNet. Moreover, the simulations provided estimates of learning and execution times for several instances of the model. Together, the parameters in the simulations and their resulting estimates help clarify the impact of cognitive versus physical media on students' browsing and learning strategies.

The design of AlgoNet provides an instantiation of our framework based on cognitive media. The subsequent empirical studies serve to evaluate its impact on student interactions and learning. In turn, the GOMS models and their analyses provide an evaluation of the design and usability of the AlgoNet system, and how it may support and hinder student learning. More importantly, our user model provides feedback on our theory of cognitive media, and provides an evaluation of the notion of cognitive media as a basic category in the design of hypermedia learning environments.

We begin by describing our system and results from the empirical studies. The remainder of the paper focuses on describing our GOMS model, and analysis of simulations of students' patterns of interactions.

COGNITIVE MEDIA: DESIGN CONSIDERATIONS FOR HYPERMEDIA

When building interactive systems targeted for educational settings, designers must address important issues in how novices can gain access into a potentially large database of interconnected types of media. In particular, designers need to address the kinds of indices that support access and learning in media-rich environments. In the short history of educational multimedia systems, some research has been conducted into how people actually use such systems, what kinds of usage improves learning and under what conditions, and what types of educational materials such systems should provide.

A review of this research literature reveals a common design approach in which access and structure of a hypermedia system are designed in terms of the various "physical" properties of information supported by computational systems. Indeed, the language of multimedia and CD-ROM development is defined by terms such as digital video, digital audio, graphics, animations, etc. Similarly, the basic objects of the World-Wide Web are specified as MIME types. These include HTML, text, graphics (gif and jpeg), digital video (Quicktime), etc. These physical media specify the building blocks of Web-based systems.

At the same time, much research surrounding multimedia and learning has sought to identify what types of physical media enhance learning (e.g., Clark, 1983; Najjar, 1996; Wu and Martin, 1997). Taken together, it is not surprising to find educational and training systems designed in terms of physical media contained in the system (cf, Jonassen and Mandl, 1990). For example, Stemler (1997) recommends designing educational multimedia systems in terms of elements such as graphics, animations, video and audio. The Demarco system, an art history archive, provides access to digital video without any other contextual information (MacKenzie, 1996). Lastly, consider the common educational Web page which offers its users raw access to simple video or sound clips.

In fact, studies that compare learning from different types of media are inconclusive in showing advantages of one physical medium over another. In general, research on what types
of digital media and materials facilitate learning is confusing and contradictory, and appears largely dependent on the specific learning context (Kozma, 1991; Najjar, 1996).

The lack of conclusive evidence reflects the fact that many factors, above and beyond simple media, affect a student's learning process. These factors include, for example, students' background knowledge, their motivation and interests, their learning strategies and goals, and the overall learning context (Chi et al., 1989; Ng and Bereiter, 1991; Ram and Leake, 1995). Therefore, rather than base the design of a hypermedia system on the physical properties of the information contained in the system, we propose that access to and structure of the system should be based on cognitive aspects of the users of that information. By this, we mean that the access methods in a hypermedia systems should be "cognitively relevant" to the learning, reasoning, and information seeking goals of the user.

There have been some previous attempts to utilize "cognitively meaningful" information of this kind. For example, TextNet (Trigg and Weiser, 1986) uses "link types" to link nodes in structured texts. TextNet is a text-based document composition environment that support two types of nodes: chunks and tocs. Chunks are the basic data objects that contain text; these include data entries such as author, date, and text, as well as links to other chunks. Tocs, similar to chunks but lacking the text field, are hierarchically organized as a table of contents connecting multiple chunks. Links between these nodes are explicitly specified by "link types," such as summary, example, continuation, and criticism. Link types are "cognitively relevant" to the information-processing goals of the users; for example, a link might correspond to expanding an idea or criticizing that idea. One difference between this approach and ours is that link types in TextNet are used mainly to organize thoughts, not as primary means to navigate through the multimedia system. In manual navigation mode, users must rely on the table of contents or keywords to jump to a specific text entry.

In our work, we have been developing a theoretical framework for designing indices for educational hypermedia systems. In this framework, we argue that their design are best thought of terms of the cognitive roles that media play in reasoning and learning, in what we call "cognitive media." Intuitively, we would not expect a student in a learning situation to ask for, say, a piece of text or a voice-over (physical characteristics of information content); instead, he or she might want to see a worked example or a concept definition (cognitive characteristics of information content) which then may be presented using text, voice-overs, pictures, or other physical media.

Text is an example of a physical medium. It can be used to represent several types of cognitive media. For example, text can be used to present abstract, general instructions; it can also be used to define concepts, or to provide explanations and annotations. However, instructions, worked examples, definitions, and explanations encapsulate fundamentally different types of knowledge, and support different types of reasoning and learning processes. Similarly, animations and pictures are examples of physical media. Pictures can be used to display graphical relations among concepts. Although often distinguished based on perceptual modalities (for example, visual vs. auditory), they may be also characterized by the types of inferences that they facilitate. For example, figures (or diagrammatic representations) can facilitate spatial inferences (Larkin and Simon, 1987).

Thus, media types can be classified based on "cognitive" roles that depend not on physical characteristics of media but on the reasoning processes of users. In general, cognitive media are characterized in terms of the inferential processes of the human user rather than physical properties of the computer representation. Cognitive media encapsulate different kinds of problem solving information which might, in turn, be composed of many different physical media.

In general, there is a many-to-many relationship between cognitive and physical media. Figure 1 illustrates the relationships between several kinds of cognitive media (definitions, case studies, principles) and physical media (text, animations, graphics). Similarly, a single physical medium might be used as part of representing several cognitive media.
We believe that it is essential to focus on cognitive media in order to understand how best to design multimedia systems that can support novices in learning or training situations, as well as aiding experts in on-the-job situations (Minsk et al., 1995). We are developing a design framework and taxonomy of cognitive media that we believe are useful in learning situations. In addition, we are attempting to specify at which points during learning they may prove to be more advantageous. For example, is a general principle more useful at the beginning of a learning session, or after the student has gained some experience in a domain? Are interactive simulations of a phenomenon more useful before or after a learner has had exposure to basic definitions and concepts in a domain? Finally, we are interested in determining the types of physical media that can best represent the different types of cognitive media.

Our taxonomy of cognitive media and how they can be implemented within physical media is still under development. In addition, we are examining additional critical facets of learner-computer interaction. In one line of research we are examining the interaction between cognitive media and the kinds of learning strategies students may bring to learning tasks (Shippey et al., 1996). In another approach, we are investigating the relationship between the kinds of goals that students bring to learning tasks (e.g., Ng and Bereiter, 1991; Ram and Leake, 1995) and the way cognitive media may help or hinder these goals (Byrne et al., 1996). In the main, the learning context, learning goals, media types, and interaction strategies all serve to influence the complex nature of student learning in networked, multimedia information repositories.

ALGONET

Overview of AlgoNet

The cognitive media design framework was used as the basis for the implementation of a computer-based, self-paced educational environment. For the implementation, we used the "MultiMedia Education Delivery System" (MMEDS) as our multimedia authoring tool. MMEDS runs on Unix workstations under X-Windows and Motif, and provides a set of tools for authoring and presenting multimedia-based courseware. These tools enable the author to create, organize, and synchronize educational information within an open and extensible client-server architecture. Authored course modules utilize a hypermedia, networked organizational model and support the presentation of text, audio, still graphics, visualizations, and other arbitrary programs (Li et al., 1994).

Using the MMEDS authoring environment, we designed an educational module, called AlgoNet, targeted for introductory computer science and engineering classes. The module...
allowed students with minimal background to learn about basic algorithmic concepts, such as graphs. The subject material for the module came from introductory computer science courses in Georgia Tech’s College of Computing that are taken by computer science and engineering majors in their first or second year.

AlgoNet consists of a collection of information nodes, which are linked together to create one large, educational document. These nodes are analogous to sections of a book in that they each cover one specific subsection of the entire module. However, unlike a book, the nodes do not have to be viewed in any particular sequence. Moreover, the nodes within the module can contain a variety of different media such as text, graphics, animations, buttons, and sound. Each node can be activated (or played) many times. In addition, AlgoNet is an active document. Many nodes are interactive, requiring active student input, construction, and involvement. Specifically, some nodes use constructive, interactive visualizations, a kind of dynamic interactive representation important in computer science.

The constructive visualizations and exercises were implemented in Polka (Stasko and Kraemer, 1993), a software environment for developing dynamic interactive visualizations of the execution of computer programs and algorithms. It also included an interactive component so that users may directly control the visualization. MMEDS directly supports Polka visualizations, so they can be included in nodes just like other presentation media.

The AlgoNet module is comprised of 57 nodes or screens of information. The materials are hierarchically organized into three modules: (a) a glossary, (b) applications, and (c) case studies, with an overview node that shows the overall structure of AlgoNet. As described in more detail below, AlgoNet provides three sets of navigational buttons and one set of control buttons. The navigation buttons are (1) topic buttons, (2) cognitive media buttons, and (3) system buttons. The control buttons are used to direct the flow of a document.

Figure 2 shows the "Example" node in the "Edge Weights" node in the glossary module. For each topic in this module, we provided two types of cognitive media: a definition and an
example. In addition, some nodes, such as "Edge Weights," include an additional interactive exercise, implemented using Polka. The goal of learners in this glossary module is to understand the concepts used in graphs and graph algorithms and to identify specific components of graphs using examples.

In "Applications," we provide two widely-used algorithms in engineering and computing fields, namely the minimum spanning tree (MST) and the shortest path algorithms. Each algorithm is explained with the following cognitive media: "Definition," "Example," "Algorithm," and "Exercise." The algorithm node shows a particular implementation of the algorithms.

Figure 3 shows the "Visualization" node for Kruskal’s algorithm. In visualization nodes, learners can play, stop, and repeat the animations that run specific algorithms. In exercises, learners actively construct graphs and run the specific algorithms to see how the algorithms function.

![Visualization of Kruskal's algorithm](image)

**Figure 3.** Visualization of Kruskal's algorithm.

In addition to buttons for topics and cognitive media, we provided system navigation buttons, at the upper-left corner of the AlgoNet window, which allow users to go back to the top-level node or to the previous screen. They are "Origin," and "Back," respectively. Finally, flow control buttons are provided below the cognitive media buttons for controlling the current screens. The flow control buttons include "Play," "Pause," "Rewind," and "Quit," represented using familiar, VCR-like icons.

Information nodes can be started in two different ways. One way is to click on the "Play" button in the flow control panel. The other way is to use the middle mouse button when clicking on a topic or a media type. Although we explicitly described this use of the middle mouse button in the manual, many students never used this functionality. Instead, they used the left mouse button to visit a node and then pressed the "Play" button to start that node.
Empirical evaluation of AlgoNet

To evaluate our design framework of cognitive media, we implemented two versions of the AlgoNet courseware. In the first system, access into the courseware was via indices that specify particular types of cognitive media. As described above, students could choose to view definitions of concepts, examples of concepts, or case studies that use the concepts, etc. In the second, contrasting system, access was provided via types of physical media. For example, students could click on buttons to view animations, pictures, text, etc. In other words, the "local navigation" buttons were labeled with types of cognitive media in the first system and by physical media types in the second. The only difference between these two systems was the labeling of the media buttons; topics and contents of the two systems were identical.

Students involved in the empirical study were 111 volunteers from 2 computer science courses. Of these, 81 were drawn from a first-year course, and 30 from a second-year, follow-on course. Thus, students in the second-year course possessed more relevant background knowledge. Students used AlgoNet as part of an extra laboratory session in their course. Prior to using AlgoNet, students read a manual describing the system. They were then asked to navigate through AlgoNet and to learn algorithm concepts. The students ended the session by answering a post-test questionnaire.

Recall that AlgoNet is comprised of 57 nodes of information. On average, students in the first-year course visited 87 nodes (indicating that they made repeated visits to nodes) and interacted with the system for an average of slightly over 38 minutes (2292 seconds). Students in the second-year course visited an average of 79 nodes in just over 39 minutes (2362 seconds).

Results

The AlgoNet system was designed to keep a log of all student interface actions. Analyses of user logs has been shown to be a useful methodology for inferring patterns and strategies of usage in computer-based systems (Recker and Pitkow, 1996). These patterns were also compared to analyses of the structure of the two systems. In addition, students’ performance was analyzed in terms of their scores on the post-test.

Detailed analyses of results from this study are presented elsewhere (Recker et al, 1995). In this section, we briefly describe two key aspects of our results: system usage and learning outcomes.

Overall, the results showed that the navigation strategies of students using the cognitive media version of AlgoNet were significantly influenced by the way the modules were organized. This was not true in the physical media condition. Thus, design based on cognitive media appeared to act as a stronger guide for student learning than did design based on the physical characteristics of multimedia.

Learning outcomes were analyzed by comparing students’ performance on the post-test. For students in the first-year course, there was no significant difference in performance between students using the two versions of AlgoNet. In contrast, students in the second-year course using cognitive media as a method for information access performed better than students in the physical media condition. This suggests that students with more background knowledge were better able to set learning goals, form information-seeking strategies, and take advantage of the cognitively-based access methods than students with less-relevant background knowledge.

GOMS USER MODELS OF ALGONET

GOMS analysis

GOMS is a cognitive modeling approach for the analysis of user-system interactions (Card, Moran, and Newell, 1983). The basic premise behind GOMS is that the knowledge involved in performing a task is usefully analyzed in terms of Goals, Operators, Methods and Selection
rules. In fact, GOMS analysis is one of the most widely used cognitive modelling approaches in Human-Computer Interaction (John and Kieras, 1996; 1996a).

A GOMS model consists of four elements: Goals, Operators, Methods and Selection rules. Goals specify what the user has to accomplish. Goals are often broken down into sub-goals, which must also be achieved to reach the goal.

Operators describe the basic actions that the user may perform in pursuit of a goal. Operators can be perceptual, cognitive, motor actions, or a composition of these. Within HCI, operators often represent basic interface actions.

Methods describe sequences of operators that accomplish a goal or sub-goal. The composition of methods necessarily depends on the nature of the task and the set of possible operators.

Lastly, selection rules pick among available methods to accomplish a sub-goal or goal. If several methods are available to accomplish a goal, selection rules are necessary to represent the user's knowledge of which method is applicable. This knowledge typically results from prior experiences.

Hence, a GOMS model describes the procedures required for accomplishing a general class of tasks. The model thus serves as an explicit representation of the important features of a task environment and the knowledge required by the user to accomplish the task goal. Specific instances of a general model can be constructed by representing a set of values for the task parameters.

**NGOMSL.**

Analyses of GOMS models can inform several aspects of a user interface design. For example, the goal structures in GOMS models can be used to estimate measures of performance (Kieras, 1988). GOMS models can also be used as a tool for evaluating the complexity of user interfaces, in an approach called cognitive complexity theory (CCT) (Kieras and Polson, 1985; Bovair, Kieras, and Polson, 1990). The number of GOMS statements that is required to operate the target system is considered a measure of its cognitive complexity.

Kieras and Bovair (1986) applied this approach to analyzing users learning how to use an interface. In their GOMS model analysis, they relied upon production rules to represent task knowledge. In particular, they showed that methods comprised of production rules that the user already knows are easier to learn than methods containing many new production rules. Therefore, to make a system easier to learn, designers must employ methods that employ common production rules. When a system is designed in this way, learners can learn the interface quickly and, as a result, they can focus on the semantics of the system.

A variant of GOMS, called NGOMSL, refines the basic GOMS approach by representing methods using the CCT cognitive architecture (Kieras, 1996). NGOMSL is a structured natural-language for constructing and representing the content of a CCT model. Therefore, the model provides predictions about operator sequence, execution time, and the time required to learn the methods.

**GLEAN.**

Our model of student interactions was built using the NGOMSL approach and implemented using the GLEAN simulation environment (Kieras, Wood, Abotel, and Hornof, 1995). GLEAN is a software development tool to assist in the development and analysis of GOMS models of interface usage.

GLEAN takes as input a GOMS specification of an interface and a task. It also accepts time estimates for basic interface actions. For example, the model builder would include time estimates for selecting a menu item, or clicking a mouse. As output, GLEAN produces a simulation of the usage of the interface, and reports an execution time estimate for each method executed, total execution time for the run, an estimate of the working memory load incurred by the user, and an estimate of the learning time required to learn the methods. Learning time estimates are computed by counting the number of steps that must be learned.
The modeling approach

In a hypermedia learning environment, the goal of a learner is to navigate through the system and learn or retrieve information necessary to understand and solve problems. Usage in our model was defined around four top-level tasks. These tasks corresponded to the major modules of AlgoNet, specifically browsing the 1) Overview, 2) Glossary, 3) Applications, and 4) Case Studies modules. Thus, a generic AlgoNet user had four tasks to accomplish. However, if an individual learner chose to visit the same module several times, each visit was treated as a distinct task.

As noted previously, our model of student interactions was built using the NGOMSL approach and implemented using the GLEAN simulation environment. Using a procedure similar to the one described in Bovair et al. (1990), the tasks were transformed into a set of production rules in the GLEAN language. These rules represented methods, which were information access procedures composed of interface operators. Operators, in turn, were basic actions such as clicking mouse buttons or moving the cursor. As the set of interface actions allowed within AlgoNet was small, this transformation was a straightforward process.

In addition, GLEAN requires time estimates for basic interface actions. Recall that the AlgoNet system kept a log file for each student in our study. The log file recorded all interface actions with a time-stamp. Using the time-stamp data, the mean time for executing each interface operators was calculated. For example, we computed the mean viewing time for each type of cognitive media. We assumed that these estimates include cognitive processes such as comprehension of the material. These average times were then used as fixed parameters by GLEAN when executing simulations of particular students.

As part of our evaluation of the design framework underlying AlgoNet, we modeled several kinds of users. First, we modeled the hypothetical generic user who simply navigated through the structure of AlgoNet. This model is described in the next section.

Second, we modeled actual student data, selected from the empirical study. In particular, our user models were constructed using estimates of viewing time and the log file data. The log files of individual student actions were used to construct selection rules that simulated individuals’ interactions within AlgoNet. Again, since the space of interactions with AlgoNet was constrained, adding and modifying selection rules was an easy process. For each running simulation, GLEAN reported estimated execution and learning times.

The model of the generic user

Prior to modeling student data, we built a GOMS model of a hypothetical generic user who is familiar with the structure and interface of AlgoNet. This generic user visited every single node once, except for the nodes that served as gateways to topics. These gateway nodes did not have much content, but served as navigational access to sub-topics. Thus, except for these gateway nodes, the generic student accessed every node only once.

Table 1 shows the beginning of a trace of the simulation of the generic model, as produced by the GLEAN environment. The trace shows how the simulation selects goals, methods for accomplishing a goal, and the various operators involved in each method.

Table 1. The beginning of a simulation trace of the generic model as produced by the GLEAN simulation environment.
The model of the generic student provided baseline data for the number of statements, and estimates of execution and learning times. For the generic user, 173 GOMS statements were required to model the AlgoNet task.

The training time estimated the time to learn the GOMS methods of a particular system, and was computed using the number of methods involved in accomplishing a goal. Since both the cognitive and the physical versions of AlgoNet had identical structure (the only difference was the labels for the types of media) the learning time for each condition was the same. A run of the generic model estimated 50 minutes (3000 seconds) to learn to use AlgoNet.

Lastly, the execution time indicated the amount of time needed to accomplish the specified tasks. For the generic model, this involved browsing four modules. Recall that estimated execution times were based on mean times for specific interface operators. These were derived from actual student data, and thus differed for each learning condition. As a result, the simulation gave different execution times for each interface condition, though the overall difference was very small. The execution time estimate of browsing through the cognitive media version of AlgoNet was just over 27 minutes (1620 seconds), and was used as the baseline estimate.

**Student simulations**

Based on the log files of student interactions recorded by the AlgoNet system, we build models of several students in our empirical study. We anticipated that constructing these models would yield a better understanding of the browsing strategies used by our students. In addition, we

```
METHOD FOR GOAL BROWSE-OVERVIEW
STEP 1 CLICK {PLAY} IN {FLOW}
STEP 2 ACCOMPLISH GOAL CLICK {OVERVIEW} IN {TOPIC}
METHOD FOR GOAL CLICK <BUTTON=OVERVIEW> IN <CONTROL=TOPIC>
STEP 1 LOCATE <BUTTON=OVERVIEW> IN <CONTROL=TOPIC>
STEP 2 MOVE CURSOR TO <BUTTON=OVERVIEW>
STEP 3 PRESS MIDDLE MOUSE BUTTON
STEP 4 RETURN WITH GOAL ACCOMPLISHED
RETURN WITH GOAL ACCOMPLISHED from (METHOD FOR GOAL CLICK <BUTTON=OVERVIEW> IN <CONTROL=TOPIC>)
Method time = 2.900 Total time = 5.200
STEP 3 READ
STEP 4 ACCOMPLISH GOAL CLICK {ORIGIN} IN {SYSTEM}
METHOD FOR GOAL CLICK <BUTTON=ORIGIN> IN <CONTROL=SYSTEM>
STEP 1 LOCATE <BUTTON=ORIGIN> IN <CONTROL=SYSTEM>
STEP 2 MOVE CURSOR TO <BUTTON=ORIGIN>
STEP 3 PRESS MIDDLE MOUSE BUTTON
STEP 4 RETURN WITH GOAL ACCOMPLISHED
RETURN WITH GOAL ACCOMPLISHED from (METHOD FOR GOAL CLICK <BUTTON=ORIGIN> IN <CONTROL=SYSTEM>)
Method time = 2.900 Total time = 28.300
STEP 5 RETURN WITH GOAL ACCOMPLISHED
RETURN WITH GOAL ACCOMPLISHED from (METHOD FOR GOAL BROWSE-OVERVIEW)
Method time = 20.800 Total time = 28.400
```
hoped the models would help to clarify the interaction between students’ actions and cognitive versus physical media.

Because of differences in the amount of relevant background knowledge, we conducted separate analyses for students in the first-year and second-year courses. We expected that since students in the first-year course knew little of the material, their navigational strategy would be the primary factor affecting their performance. In contrast, those in the second-year course had some prior experience in computer algorithms, and hence might skip the modules they were already familiar with without affecting their post-test performance.

Recall that students learned from either the cognitive or physical version of AlgoNet. In addition, based on students’ performance on the post-test, we classified students into a good learner group and a poor learner group. From each resulting group (good-cognitive, poor-cognitive, good-physical, and poor-physical) and for each course, we randomly selected two students. Their interaction patterns were then modeled in GLEAN. This enabled us to compare the interactions patterns of these four user groups, and for each course.

Models of students in the first-year course

Table 2 shows the results of building GOMS simulations of the selected students in the first-year course. On average, simulations of students in the various groups required more GOMS statements and had longer learning and execution times than the generic model. Only one simulation had a smaller number of GOMS statements than the model of the generic student. Inspection of the log file revealed that the student skipped the entire glossary module.

Table 2. Results of GOMS simulation of individual students in the first-year course

<table>
<thead>
<tr>
<th>Group</th>
<th>Number of statements</th>
<th>Training time (secs)</th>
<th>Execution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good-Cog</td>
<td>221</td>
<td>3757</td>
<td>4485.6</td>
</tr>
<tr>
<td>Good-Cog</td>
<td>194</td>
<td>3298</td>
<td>5256.8</td>
</tr>
<tr>
<td>Poor-Cog</td>
<td>240</td>
<td>4080</td>
<td>3970.5</td>
</tr>
<tr>
<td>Poor-Cog</td>
<td>164</td>
<td>4920</td>
<td>3854.0</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>179</td>
<td>3043</td>
<td>3099.3</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>252</td>
<td>4284</td>
<td>3843.3</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>231</td>
<td>3927</td>
<td>2944.4</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>290</td>
<td>4930</td>
<td>4444.0</td>
</tr>
<tr>
<td>Generic model</td>
<td>173</td>
<td>3000</td>
<td>1620.0</td>
</tr>
</tbody>
</table>

Table 3 shows the mean number of statements and execution and learning times for the simulations of students in the four groups. Results for the generic model are also shown. As can be seen, the simulations of students using the cognitive media version of AlgoNet required a smaller amount of statements than simulations of students using the physical version. This result appears to reflect the reported empirical finding that the cognitive media interface appeared to have a significantly greater influence on students’ overall browsing strategies than the physical media interface (Recker et al., 1995). In short, the impact on browsing arose more from interactions with the interface than from the contents of the individual simulations.

Table 3 shows the mean number of statements and execution and learning times for the simulations of students in the four groups. Results for the generic model are also shown. As can be seen, the simulations of students using the cognitive media version of AlgoNet required a smaller amount of statements than simulations of students using the physical version. This result appears to reflect the reported empirical finding that the cognitive media interface
appeared to have a significantly greater influence on students’ overall browsing strategies than the physical media interface (Recker et al., 1995). In short, the impact on browsing arose more from interactions with the interface than from the contents of the individual simulations.

**Table 3. Mean results for simulation groups in the first-year course**

<table>
<thead>
<tr>
<th>Group</th>
<th>Mean number of statements</th>
<th>Mean training time (secs)</th>
<th>Mean execution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good-Cog</td>
<td>207.5</td>
<td>3527.5</td>
<td>4871.2</td>
</tr>
<tr>
<td>Poor-Cog</td>
<td>202.0</td>
<td>4500.0</td>
<td>3912.2</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>215.5</td>
<td>3663.5</td>
<td>3471.3</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>260.5</td>
<td>4428.0</td>
<td>3694.2</td>
</tr>
<tr>
<td>Generic model</td>
<td>173.0</td>
<td>3000.0</td>
<td>1620.0</td>
</tr>
</tbody>
</table>

In addition, the simulations of good students all produced faster training times. This suggests that good performers, regardless of media, learned to use AlgoNet faster. The poor performers, on the other hand, not only visited nodes in a more random fashion, but they also appeared to forget aspects of navigating AlgoNet despite having read the AlgoNet manual. Recall that a user could use either the left mouse button with the "Play" button or the middle mouse button to see a new node. However, poor students often clicked hyper-link buttons or system buttons several times before they realized that they had forgotten to press the "Play" button to bring up a current page. These unproductive actions contributed to the increased number of GOMS statements required in their user models.

Qualitative inspections of the simulations were also revealing. In particular, poor learners often came back to the same topic nodes to look at sub-nodes that they had not viewed in their previous visit to the topic. They therefore frequently returned to the "Overview node" to use as a gateway to previously-missed sub-nodes. This more random navigation strategy made additional demands on working memory, as it required learners to keep track of which nodes they had previously visited.

Finally, the good performers visited both Kruskal’s and Prim’s nodes in the Minimal Spanning Tree (MST) module, whereas none of the poor students visited either of these nodes. These two nodes consisted of interactive, constructive visualization in the application module. Since poor performers appeared to jump more randomly around the different modules, it may have been that they simply did not notice these sub-nodes. In the end, this may have contributed to their poor performance on the post-test.

**Models of students in the second-year course**

To model the learners in the second-year course, we used the same criterion of post-test performance to select good and bad learners. However, our criterion identified only one poor student in the cognitive media condition. Table 4 shows the results of building GOMS simulations of the selected students in the second-year course. Results for the generic model are also shown. Unlike the first-year group, simulations of students in the various groups did not necessarily have more GOMS statements and longer learning times than the generic model.

Table 5 shows the mean number of statements and execution and learning times for the simulations of students in the four groups, along with results for the generic model. As can be seen, simulations in the cognitive media group all had a greater number of GOMS statements and learning and execution times than simulations in the physical media group. This group also reported a significantly higher post-test performance. Students with more relevant background knowledge may have been able to use cognitive media to identify novel material. This resulted in higher levels of beneficial interaction with the system.
Table 4. Results of GOMS simulation of individual students in the second-year course

<table>
<thead>
<tr>
<th>Group</th>
<th>Number of statements</th>
<th>Training time (secs)</th>
<th>Execution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good-Cog</td>
<td>362</td>
<td>6154</td>
<td>5057.1</td>
</tr>
<tr>
<td>Good-Cog</td>
<td>141</td>
<td>2397</td>
<td>2451.5</td>
</tr>
<tr>
<td>Poor-Cog</td>
<td>178</td>
<td>5340</td>
<td>3104.3</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>117</td>
<td>1989</td>
<td>1342.2</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>205</td>
<td>3485</td>
<td>3313.0</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>136</td>
<td>2312</td>
<td>1152.2</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>135</td>
<td>2295</td>
<td>1851.6</td>
</tr>
<tr>
<td>Generic model</td>
<td>173</td>
<td>3000</td>
<td>1620.0</td>
</tr>
</tbody>
</table>

From a qualitative analysis of the simulations, it seemed that good learners had a tendency to skip the glossary module, suggesting that they felt they knew the basics. In contrast, poor learners tended to skip the application module. Finally, students in the second-year course did not appear to experience problems navigating through AlgoNet, except one who repeatedly clicked the same buttons like many poor learners in the first-year course. As a result, this simulation had over twice as many GOMS statements as the generic model.

Table 5. Means results for simulation groups in the second-year course

<table>
<thead>
<tr>
<th>Group</th>
<th>Mean number of statements</th>
<th>Mean training time (secs)</th>
<th>Mean execution time (secs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good-Cog</td>
<td>251.5</td>
<td>4275.5</td>
<td>3754.3</td>
</tr>
<tr>
<td>Poor-Cog</td>
<td>178.0</td>
<td>5340.0</td>
<td>3104.3</td>
</tr>
<tr>
<td>Good-Phy</td>
<td>161.0</td>
<td>2737.0</td>
<td>2327.6</td>
</tr>
<tr>
<td>Poor-Phy</td>
<td>135.5</td>
<td>2303.5</td>
<td>1501.9</td>
</tr>
<tr>
<td>Generic model</td>
<td>173.0</td>
<td>3000.0</td>
<td>1620.0</td>
</tr>
</tbody>
</table>

SUMMARY

First-year student simulations

Overall, simulations of students in the various groups required more GOMS statements and had longer learning and execution times than the generic model. This is not surprising, as the generic model was built to have a parsimonious interaction with AlgoNet, whereas students in the first-year course were encountering a great deal of novel material. In addition, the simulations of students using the cognitive media version of AlgoNet required a smaller amount of statements than simulations of students using the physical media version. This result suggests that students’ overall browsing strategies were strongly determined by interactions with the two interfaces. Lastly, the simulations of good students all produced faster training times. Good performers appeared to employ more effective and systematic navigation strategies.

Second-year student simulations

Unlike the first-year group, simulations of second-year students did not necessarily have more GOMS statements and longer learning times than the generic model. These students were encountering much less novel material, and thus may have been more focussed in the
navigational approach. However, simulations in the cognitive media group all had a greater number of GOMS statements and learning and execution times than simulations in the physical media group. Students with more relevant background knowledge seemed to better use cognitive media as a means to identify, learn, and beneficially interact with novel instructional material.

**DISCUSSION**

In our research, we are interested in understanding the contributions a design framework based on cognitive media can make in developing an interactive multimedia learning environment. Our methodologies involves designing educational systems using our framework, conducting empirical studies with students, and building user models of students in our studies. Results from empirical studies and user model analysis help clarify the effects of system design, and hence our design framework, on students’ browsing and learning strategies.

We used the GOMS modeling approach to analyze the navigational patterns of several classes of students in our empirical studies. Typically, cognitive science and AI models focus on modeling one class of behavior. However, in a learning context, students bring to bear a variety of knowledge, goals, and strategies. These can result in considerable variability in the nature of the activity that arises while interacting with computational artefacts. Therefore, as has been argued by others, we believe modeling efforts are more fruitful when focused on understanding the activities of individual students or classes of students (Recker and Pirolli, 1995; VanLehn and Jones, 1993).

Analyses of our GLEAN simulations yielded useful results from both quantitative and qualitative standpoints. For students in the first-year course, the good performers appeared to employ more effective and systematic navigation strategies, as reflected by shorter estimates in the time required to learn to use AlgoNet. This enabled the learners to focus their attention on the contents, rather than worrying about the next navigation step. In contrast, the browsing pattern of the poor learners seemed more random, and hence computationally expensive.

From a design point of view, our results support the notion of providing external memory aids within an interface. When using both topic buttons and media buttons, the interface needs to make obvious to users which nodes have already been visited. In short, the interface needs to act as a working memory navigational aid.

In addition, our simulations suggest that poor learners often forgot how to use AlgoNet. This, of course, is a severe disadvantage in an educational software system. Recall that in AlgoNet, learners were provided with two methods of starting a node: left mouse button click with the "Play" button, and the middle mouse button click. Although we provides these two methods for users’ convenience, it appears that these two different choices might have instead confused novice learners.

For students in the second-year course, simulations of students in the cognitive media group all exhibited higher levels of activity. Moreover, good learners appeared to identify and focus on materials that they did not know, particularly the application module, and spent time learning these materials. Together, these behaviors may have contributed to their higher post-test performance.

Textbooks are typically designed with a distinct order of topics, and the level of difficulty increases as students progress through material. Thus, students with some relevant background can tell how much they can skip to get to the unfamiliar subjects. In hypertext systems, however, the concept of order is reduced. Our intention is to provide different types of cognitive media so that individual learners can make appropriate choices. People who prefer a case-based reasoning approach may start from the case module or go to the application module to see what these algorithms are used for, and then come back to the theoretical aspects, whereas those who prefer a theoretical approach may learn basic graph concepts first by visiting the glossary module. This may lead some students to the illusion that they know the material covered in AlgoNet since the modules they first visit contain familiar concepts. This suggests that AlgoNet (and other educational systems) need a self-diagnosis node that students can use to reflect on their understanding and to help determine which nodes to visit next.
In summary, our GOMS model and resulting analyses of simulations provided a clearer understanding of the browsing strategies used by our students. In addition, they help clarify the interactions between students' actions and cognitive versus physical media. However, GOMS models are primarily designed to analyze and evaluate the *usability* of particular systems, but not the *learnability* of its content. At present, the GOMS cognitive architecture cannot adequately model learning, though efforts are underway in the direction (John and Kieras, 1996a). It does also not cope well with modeling changes in navigational strategies. These, of course, are a critical but difficult aspects in the design of effective educational computing environments. Future versions of GOMS or alternative modeling approaches are needed to better evaluate the learnability of content within educational hypermedia.

Acknowledgements

We thank George Li and John Stasko for their helpful comments on this paper and for their help in building the AlgoNet system. We are also grateful to Doug Merrill and two anonymous reviewers for insightful reviews of this paper. This work was supported by the National Science Foundation under the SUCCEED initiative, by the EduTech Institute, by the Army Research Institute under contract no. MDA-903-90-K-0112, and by the Georgia Institute of Technology. Portions of this paper appeared in the *Proceedings of the 1996 World Conference of Educational Multimedia and Hypermedia*, Boston, U.S.A.

References


