SYSTEM DEMONSTRATION CONTENT PLANNING AS THE BASIS FOR AN INTELLIGENT TUTORING SYSTEM

Reva Freedman[1†], Stefan Brandle[2], Michael Glass[2], Jung Hee Kim[2], Yujian Zhou[2] and Martha W. Evens[2]

[1]LRDC #819
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15260

[2]Department of CSAM
Illinois Institute of Technology
10 W. 31st Street 236–SB
Chicago, IL 60616

freedrk+@pitt.edu, {brandle, glass}@charlie.cns.iit.edu, janice@steve.csam.iit.edu, zhouyuj@charlie.cns.iit.edu, csevens@minna.cns.iit.edu

http://www.csam.iit.edu/~circsim

1. INTRODUCTION

The negative feedback loop which maintains a steady blood pressure in the human body is one of the more difficult topics for first-year medical students to master. CIRCSIM-Tutor v. 3 is the latest in a series of dialogue-based intelligent tutoring systems intended to help students master the concepts involved.

CIRCSIM-Tutor v. 3 differs from many other ITSs in that text planning is an integral part of the system rather than part of a natural-language front-end. It contains a global planner whose fundamental goal is "generate a conversation resulting in the student knowing <concepts>" rather than "teach <concepts>." Constraints on the plan operators can be used to take a variety of knowledge sources into account, including the tutorial history, the domain knowledge base and a student model.

To ensure that CIRCSIM-Tutor is useful in the classroom, we have paid particular attention to broad coverage of the domain, maintenance of a coherent conversation and fast response time. We often need to model what human tutors do without a deep model of why they say what they say. As a result our content planner uses a schema-based representation which allows us to control the decomposition and sequence of goals. Through the use of a reactive planning architecture, we can update our plan based on the student's answers. Text realization is accomplished via a high-level template mechanism based on a mini-syntax of potential answers. Both the content schemata and the text realization templates are based on detailed modeling of the dialogue patterns of expert human tutors.

2. DESCRIPTION OF DOMAIN AND USER INTERFACE

CIRCSIM-Tutor helps students practice the reasoning they have learned in Introduction to Physiology. Students are given a simplified qualitative model of the heart, followed by a series of problems which utilize the model. In each problem, an incident such as the administration of a drug affects the processing

This work was supported by the Cognitive Science Program, Office of Naval Research under Grant No. N00014–94–1–0338 to Illinois Institute of Technology. The content does not reflect the position or policy of the government and no official endorsement should be inferred.

†This work was performed while Reva Freedman was at the Illinois Institute of Technology.

of the heart. The student is then asked to predict the direction of change of seven core variables:

HR:	Heart rate (beats/min)	MAP:	Mean arterial pressure
IS:	Inotropic state, a measure of the heart's contractile ability	SV:	Stroke volume (vol/beat)
		CVP:	Central venous pressure
TPR:	Total peripheral resistance	CO:	Cardiac output (vol/min)

The qualitative causal relationships between the core variables (i.e. an increase in X causes an increase/decrease in Y) are shown on the left-hand side of Figure 1. In this diagram, NS = nervous system and Baro = the baroreceptors in the neck which recognize a change in blood pressure. A finer-grained knowledge representation is also available for the tutor to use when needed. A section of this knowledge base, the leg from the nervous system to TPR, is shown on the right-hand side of Figure 1.

Each variable must be predicted at three points: the DR or *direct response* phase immediately after the incident, the RR or *reflex response* phase, which shows the effect of the nervous system, and the SS or *steady state* phase after a new steady state has emerged. After the predictions are made, the tutor engages the student in a dialogue to help the student learn the correct answers and the concepts underlying them.

The basic user interface is the screen shown in Figure 2. A description of the current problem is shown at the top of the screen. The left-hand side of the screen contains a table where the student can fill in predictions for the three stages, and the right-hand side contains a window where the dialogue evolves.

The student part of the dialogue is free text. Instead of restricting what students can say, we attempt to guide them toward understandable responses through the way the questions are framed, e.g. by asking short-answer questions instead of open-ended ones. Input processing is based on finite-state transducers. Each question the system can ask is associated with a transducer which can categorize and extract a variety of student answers. Often if the student just uses some of the right words, the sentence is accepted since the input processor uses very little syntax. Spelling correction, an essential in any system which allows free-text typing, is based on algorithms developed by Elmi [1998]. The transducers were coded by hand using as input both dialogues with human tutors and logs from earlier versions of CIRCSIM-Tutor.

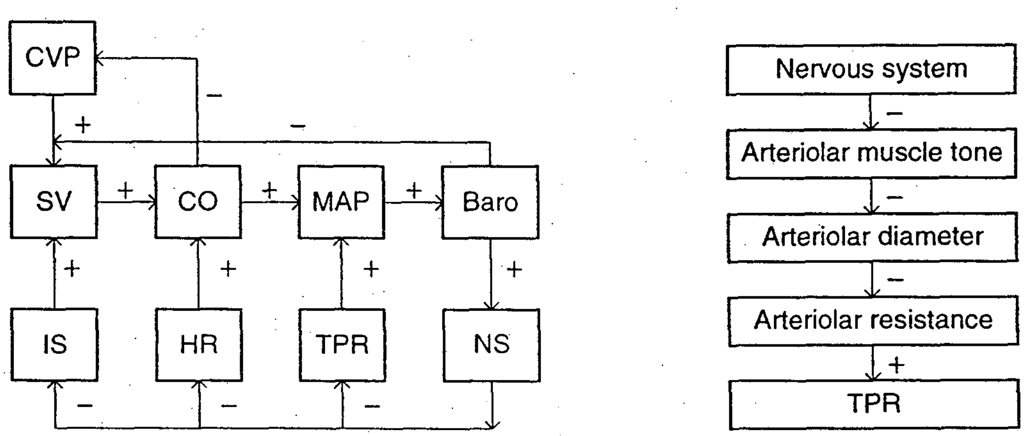

Figure 1: Two aspects of the CIRCSIM-Tutor domain model

3. DIALOGUES GENERATED BY THE SYSTEM

At the top levels, the conversation generated by the system is hierarchical. Within each stage, the text is divided into segments, one for each incorrect core variable. The variables are discussed in a partially ordered sequence which corresponds to the solution trace of the problem.

Each variable is tutored using one of a number of *tutoring methods* which we have isolated from studies of human tutoring transcripts. The tutoring methods are implemented using an extended form of schema which allows full unification, static and dynamic preconditions, and recursion. The following schema is typical (schemata are implemented in Lisp):

> To correct student's ideas about any variable ?v controlled by the nervous system
> > Teach about mechanism of control of ?v
> > Teach about when this mechanism is activated
> > Check to find out whether student knows the correct answer now

Circsim-Tutor v3.0   Help Quit Debug

Problem: Pacemaker malfunctions, increasing to 120 beats/min.

	DR	RR	SS
Central Venous Pressure	-		
Inotropic State	0		
Stroke Volume	-		
Heart Rate	+		
Cardiac Output	+		
Total Peripheral Resistance	+		
Mean Arterial Pressure	+		

T> Can you tell me what controls TPR?

S> Maybe vasoconstriction?

T> And what causes that?

S> The nervous system.

T> Right. And what stage are we in now?

Student notes go here.

Figure 2: User interface screen

Each tutoring method is composed of a number of topics. Unless it includes a recursive call to another schema, each topic is instantiated using standard text generation primitives like *elicit* and *inform*. In addition to arguments specifying the content, the primitives can be modified with arguments specifying where the primitive falls on Halliday's interpersonal and narrative axes. Thus, for example, a sentence like *Remember that we're in the pre-neural period* could be generated from a form like <T-informs info=DR-info attitude=remind>. Optional arguments are also provided for generating several kinds of discourse markers and temporal clauses.

Instead of planning the complete text as in a monologue, we interleave planning and execution, planning only as much as necessary to generate the next turn. When the student gives an unexpected response, which includes various kinds of "near-misses" as well as wrong answers, we can choose between retrying the current goal, adding a new goal at the top of the agenda, or dropping the current schema and replacing it by another one. In this way we can reply flexibly to the student while still maintaining a long-

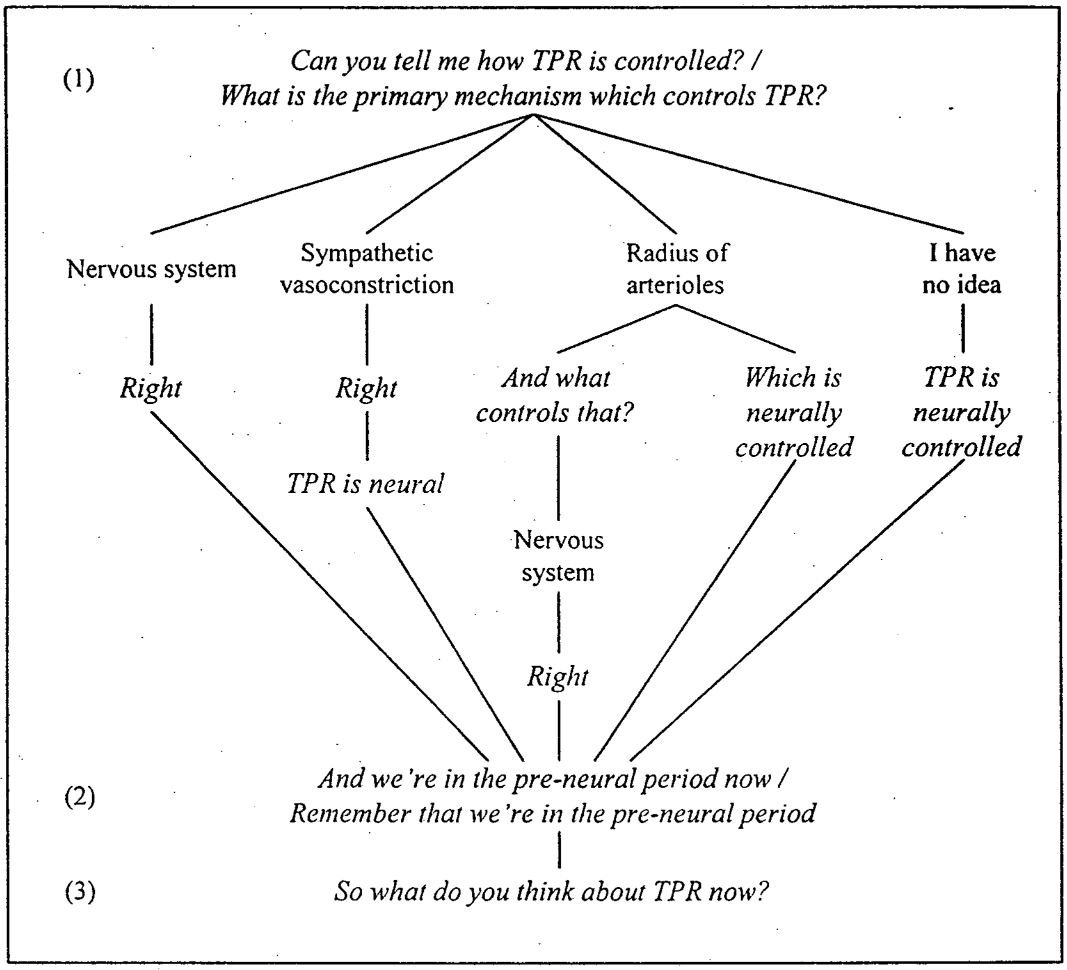

Figure 3: Sample dialogues

range plan. Each path through Figure 3 shows one piece of conversation which can occur as a result of the schema shown above. From left to right, the paths show a right answer, a couple of near-misses which require the use of the more detailed knowledge base, and a wrong answer.

4. REFERENCES

[Elmi 1998] Elmi, M. A. and M. W. Evens. 1998. Spelling Correction Using Context. *Proceedings of COLING–ACL '98*, Montreal.

5. ACKNOWLEDGMENTS

Professors Joel A. Michael and Allen A. Rovick of the Department of Physiology, Rush Medical College have been generous with both their time and their expertise.