Liu, C.-C. et al. (Eds.) (2014). Proceedings of the 22nd International Conference on Computers in Education. Japan: Asia-Pacific Society for Computers in Education

# From Tutoring to Cognitive Rehabilitation: Exploiting CBM to Support Memory Training

**Antonija MITROVIC[a*], Moffat MATHEWS[a], Stellan OHLSSON[b], Jay HOLLAND[a], Audrey MCKINLAY[c], Scott OGDEN[a], Anthony BRACEGIRDLE[a] & Sam DOPPING-HEPENSTAL[a]**
[a]*Intelligent Computer Tutoring Group, University of Canterbury, New Zealand*
[b]*Department of Psychology, University of Illinois at Chicago, USA*
[c]*Department of Psychology, University of Melbourne*
*tanja.mitrovic@canterbury.ac.nz

**Abstract:** Constraint-Based Modeling (CBM) is an effective student modeling approach which has been used successfully in a wide range of instructional domains. Within the Intelligent Computer Tutoring Group (ICTG), we have developed numerous constraint-based tutors and demonstrated their effectiveness in real courses. In this paper, however, we discuss how we use CBM in the area of cognitive rehabilitation after stroke. Our computer-based treatment is aimed at improving prospective memory. Participants are first trained on how to use visual imagery and then practice in a Virtual Reality (VR) environment. We present how we use constraints to track the participant's progress when performing tasks in the VR environment.

**Keywords:** constraint-based modeling, prospective memory, virtual reality environment

## 1. Introduction

Constraint-Based Modeling was originally proposed by Ohlsson (1992) as a way to overcome problems with student modeling. Since 1995, when the work on the first constraint-based tutor started, we have developed numerous tutors using CBM at ICTG (Mitrovic, 2012). Our early work focused on showing that CBM was an effective way of modeling domains and student knowledge. In order to develop SQL-Tutor, we proposed a number of extensions to CBM, including a way to develop long-term student models, the distinction between syntax and semantic constraints, and the use of ideal solutions in order to deal with multiple correct solutions (Mitrovic, 1998; Mitrovic & Ohlsson, 1999).

Many other constraint-based tutors followed after SQL-Tutor: some teach other design tasks, such as EER-Tutor (Suraweera & Mitrovic, 2004; Zakharov, Mitrovic & Ohlsson, 2005), UML class diagrams (Baghaei, Mitrovic & Irwin, 2007) and Java (Holland, Mitrovic & Martin, 2009). In addition to these design tasks which are all ill-defined (Mitrovic & Weerasinghe, 2009), we have also developed tutors that teach well-defined tasks, like NORMIT in the area of data normalization (Mitrovic, 2005), thermodynamics (Mitrovic et al., 2011), capital investment decision making (Mitrovic et al., 2009) and oil palm plantations management (Amalathas, Mitrovic & Ravan, 2012). We have investigated various types of long-term student models ranging from overlays to probabilistic ones (Mayo & Mitrovic, 2001), and investigated the effectiveness of various teaching strategies (Mitrovic, Ohlsson & Barrow, 2012; Mathews & Mitrovic, 2007; Weerasinghe et al., 2011; Najar & Mitrovic, 2013). Over the years, our research focus expanded to other research challenges. We extended CBM to represent and support meta-cognitive skills such as self-assessment (Mitrovic & Martin, 2007) and self-explanation (Mitrovic, 2005; Weerasinghe & Mitrovic, 2006). We also developed COLLECT-UML which supports pairs of students working together and provides feedback on both problem solving and collaboration (Baghaei, Mitrovic & Irwin, 2007). We investigated affective modeling (Zakharov, Mitrovic & Johnston, 2008), and the use of data mining and eye-tracking to improve our ITSs (Elmadani, Mathews & Mitrovic, 2012; Mathews et al., 2012; Elmadani, Mitrovic & Weerasinghe, 2013).

We have also developed ASPIRE[1], a general authoring tool and deployment system for constraint-based tutors (Mitrovic et al., 2009). Many tutors have been developed in ASPIRE, by

[1] http://aspire.cosc.canterbury.ac.nz/

members of ICTG and researchers all over the world. CBM is now a thoroughly tested and widely used methodology; is not used solely by ICTG, but also by various groups of researchers worldwide – see e.g. (Rosatelli & Self, 2004; Riccucci et al., 2005; Petry & Rosatelli, 2006; Mills & Dalgarno, 2007; Siddappa & Manjunath, 2008, Menzel, 2006; Oh et al., 2009, Galvez, Conejo & Guzman, 2013; Le & Menzel, 2009; Roll, Aleven & Koedinger, 2010).

In our current project, however, we use CBM in a completely different situation. Since 2011, we have been developing computer-based training for improving prospective memory in stroke patients. People with brain injury (including stroke) have severely impaired prospective memory in comparison to healthy people (Mathias & Mansfiled, 2005; Brooks et al., 2004). Prospective memory, or remembering to perform actions in the future, is of crucial importance for everyday life (Titov & Knight, 2000). Prospective memory failure can interfere with independent living, as it can result in forgetting to take medication, switch off the stove or missing doctor's appointments. It is a complex cognitive ability, which requires dynamic coordination of multiple cognitive abilities: spatial navigation, retrospective memory, attention and executive functioning (Knight & Titov, 2009).

We start by presenting related work in Section 2. Section 3 presents the treatment we devised: the visual imagery training, the VR environment and the constraints we developed to track the user's behavior in the VR environment. In Section 3, we also briefly present the evaluation study we are currently conducting. We conclude the paper with a discussion of future work.

## 2. Related Work

Stroke is the second leading cause of death[2] and a major contributor to disability. Cognitive impairment plays a crucial role in determining the broader outcomes of a stroke survivor (Barker-Collo et al., 2009). The extent of impairment affects aspects of daily functioning, and often necessitates constant care. Customised rehabilitation is required but is labor-intensive and expensive (Lee et al., 2010). Rehabilitation outcomes are disproportionate in many countries; for example, in New Zealand lower outcomes are achieved in low socio-economic, Māori, and Pasifika areas due to a lack of resources (Dyall et al., 2008; Ministry of Health, 2002).

Prospective memory is defined as the ability to remember future intentions (Ellis & Kvavilashvili, 2000). There are two critical aspects of PM: it is closely related to retrospective memory (remembering what was learnt and experienced previously), as it is necessary to know what the task is (e.g. taking medication at 3pm) in order to actually perform the task. The other aspect is the retrieval of the intention at the time appropriate for the action. There is a distinction between event- and time-based prospective tasks. In the case of a time-based task, a certain action needs to be performed at a certain time (e.g. having a doctor's appointment at 4pm). In event-based tasks, an action needs to be performed when a certain event happens (like asking a friend a question when we see them next time).

To be able to perform a task in the future, a person needs to know the task, a level of intention and a cue. Cues are prompts that help people remember the tasks to be performed in the future. When a person perceives a cue, it delivers the information that was previously associated with the cue to the consciousness, and the person remembers the task. Previous research indicates that cues help a person's memory as it reinforces the intention to execute a task (Gollwitzer, 1996).

Prospective memory is very difficult to assess using neuropsychological tests as conventional tests consist of simple, abstracted activities that are very different from real-world tasks. In order to assess prospective memory, it is necessary to obtain information about how a patient functions in everyday life, which is difficult to achieve in laboratory settings. Research shows that scores from neuropsychological tests often cannot be translated to conclusions about the level impairment and therefore rehabilitation goals because many conventional tests lack ecological validity (i.e. similarity with real life) (Knight, & Titov, 2009). It is therefore necessary to replace such tests with tasks that mirror real-word activities. However, assessing patients in real-world situations entails logistic problems and is not achievable in rehabilitation units (Brooks et al., 2004).

In the last decade, many research projects have used Virtual Reality (VR) in neuroscience research and therapy (Bohil, Alicea & Biocca, 2011), ranging from the use of VR for assessing cognitive abilities, over neuro- and motor rehabilitation to psychotherapy, such as treatment of phobias.

[2] http://who.int/mediacentre/factsheets/fs310/en/

VR environments are computer-generated environments that simulate real-life situations and allow users to interact with them. They provide rich, multisensory simulations with a high degree of control and rich interaction modalities. They can also have a high level of ecological validity. VR has been used for assessment of prospective memory in patients with traumatic brain injury (TBI) (Knight & Titov, 2009) and stroke patients (Brooks et al., 2004). VR is suited for prospective memory as it supports complex, dynamic environments that require coordination of many cognitive abilities. VR environments are convenient and safe for patients. Non-immersive, PC-based environments have been used more than the immersive ones, which require special hardware (such as head-mounted displays) and therefore are more expensive and induce more anxiety in patients than PC-based ones.

Although there has been some research done on how to assess PM, there is very little available on rehabilitation strategies for PM. Yip and Man (2013) involved 37 participants in 12 sessions (held twice a week) of prospective memory training using non-immersive VR. The participants were asked to perform a set of event- and time-based prospective memory tasks in parallel with an ongoing task, all performed in a virtual convenience store. The prospective memory training was based on remedial and process approaches. The remedial approach provides repetitive exercise within the VR environment. The process approach, on the other hand, aims to support multiple facets of prospective memory, and supports encoding of intention, retention and performance interval and recognition of cues. Participants were given a list of four shopping items they needed to memorize, and their recall was tested before entering the VR environment, where they needed to perform the tasks. The VR training showed significant improvement in participants' immediate recall of PM tasks, performance on both time- and event-based tasks as well as ongoing tasks, and also a significant improvement in self-efficacy.

## 3. Our Approach to Prospective Memory Training

The primary goal of our project is to develop an effective PM treatment that could be used by the stroke survivors without the input of clinicians. Our approach combines the use of visual imagery and practice in VR environments. We developed a treatment based on visual imagery, and a VR environment in which the patient will be able to improve their prospective memory. We start by describing the visual imagery training, and then present the VR environment, constraints used in the VR environment, and the evaluation study.

### *3.1 Visual Imagery Training*

Visual imagery is a technique in which the participant forms a visualization of a given word. The same strategy can also be used to make a visualization of a pair of words, by linking the words and making the visualization as unusual as possible to make it more memorable. Previous work (Lewinsohn, Danaher & Kikel, 1977) has shown that visual imagery improves retrospective memory. McDaniel and Einstein (1992) showed that PM performance improved when participants were given pictures of targets, or when participants formed mental images of cues.

The idea behind the visual imagery training is to teach participants to remember a list of tasks with their associated cues using visual imagery as a mnemonic strategy. The training is presented on a computer in the form of a set of sequential pages. Pages contain recorded voice messages that participants can hear, images, video, written text, buttons to navigate to the previous and next pages, and a replay button. Sometimes participants are asked to interact with the page (e.g. during testing). On such pages, buttons are provided for the user to record the answers.

In the first phase of training, participants are introduced to visual mnemonics by being shown how to form mental images in order to remember a list of paired words. The user is presented with pairs of words and taken through creating the image for each pair. For example, for the pair (*rabbit*, *pipe*), the participant is first shown pictures of a rabbit and a pipe, and they listen to the recording of the following text: *Look at the image displayed of a rabbit. Imagine its bristly fur and its long ears wriggling. Really focus on it, like it's right there in front of you. Now look at the picture of a pipe. Imagine this in your mind. Smoke is coming out of the pipe, giving off a smoky smell. Imagine grasping the pipe, and feeling it. The pipe feels round and smooth in your hands. The more senses you use, the more memorable the image will be.*

The following training page (illustrated in Figure 1) shows the two previously shown pictures of a rabbit and a pipe, and also the combined picture, and plays the recording of this text: *Now that you have imagined the two images individually, we are going to visually link them together, which will help you to remember them. This technique of visually linking them together will allow you to recall the individual words in the future. So, what I want you to do right now is to imagine the rabbit smoking the pipe, like it is in the third image. Close your eyes and really think about it. The rabbit is puffing away and more and more smoke is coming out. In your mind, imagine the rabbit taking the pipe out and blowing a smoke ring and then putting it back in its mouth. What a silly rabbit! Ok, now open your eyes. Now that you've done this, the image of the rabbit smoking the pipe should be firmly in your memory, so that if we gave you the image of a rabbit, you would immediately think of it smoking a pipe, which will lead you to the second word: pipe!*

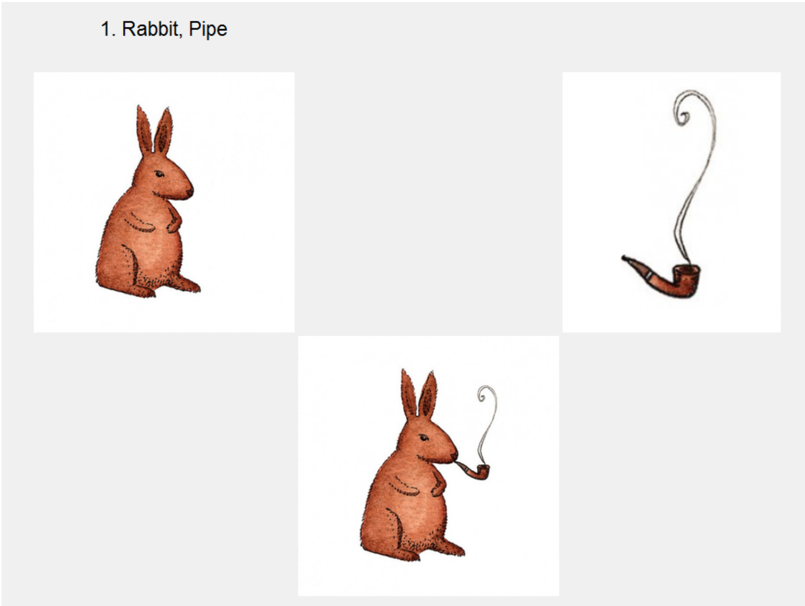


Figure 1: The screenshot of the training for the paired words *rabbit* and *pipe*

The audio instruction is designed to be as descriptive as possible in order to better aid the user's visualisation. The user is encouraged to mentally add to the presented images, personalizing them and making them more concrete. After presenting the initial three pairs of words, the user is tested by presenting one word from the pair and asking them to record the other paired word. Next, the user is presented five pairs at once, which he/she needs to visualize, and then the user is quizzed on them.

In the next phase, the training is modified by removing the combined picture, and asking the user to generate his/her own combined image. In the third phase, the training becomes even more demanding, as the user needs to visualize both given words as well as to generate the combination picture. The later phases provide the user with example tasks and show how, for each example task, the context could be related to the items to be done. The user is taught that each task consists of either object-action pairs or time-action pairs. An object-action pair is where some action is required when a certain object is encountered. A time-action pair is when some action is required at a particular time. An example of an object-action pair was then given: "Tell Laura about tomorrow's weather when you meet her next". The object in the pair is Laura and the action is to tell her about tomorrow's weather. Using this pair, a corresponding visual mnemonic could be made; for example, seeing Laura being swept up and carried by a tornado or the sun shining brightly out of Laura's head. Participants are taught that the more concrete (using real places, real people, real things, or real time) and more detailed a visual mnemonic was, the more personal and real it would be, and thus more memorable. The user is taught that the more silly or humorous a visual mnemonic was, the more memorable it would be.

In the last phase, the user is given lists of tasks to memorize, and then completes four problems. Each problem consists of watching a video containing the given tasks. During the video, when the user identifies the cue for a task, he/she needs to stop the video and record the action that corresponds to the cue, before carrying on with the video. They cannot rewind the video to enter in missed tasks.

### 3.2 VR Environment

We have developed a VR environment using the Unity game engine, in which the participants need to perform a given set of time- or event-based tasks. The VR environment represents a house with common household objects, and a garden. Figure 2 shows two scenes from the environment. The user is given a problem, which consists of several PM tasks they need to visualize first, and then perform in the VR environment. As discussed earlier, some tasks are time-based and the user is able to view a clock whenever they choose. The user is able to carry out a number of actions on a variety of objects such as a radio or a washing machine.

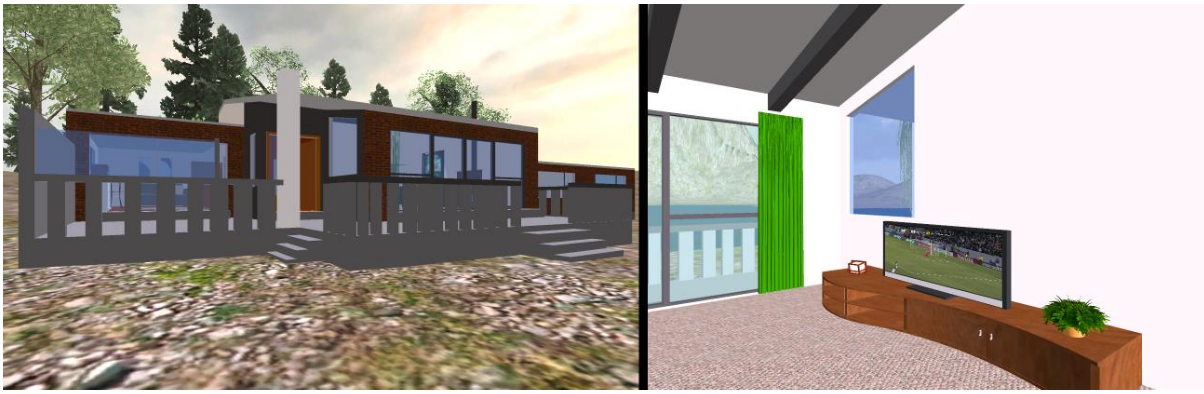

Figure 2: Two scenes from the VR environment

The tasks vary in complexity: the ones in early sessions consist of a cue and a single action, such as *Turn on the TV at 6pm*. In later sessions, the user will be given more complex tasks, such as *When the oven timer beeps, take the cake out of the oven and put it on the dining table.* To perform an action, the user needs to select the object first, and then to specify the action from a menu. The screenshot in Figure 3 shows a situation when the user is interacting with the oven. The selected object (oven in this case) is highlighted in red, and the user is given a menu listing the actions that can be performed on the object. In this particular situation, the user wants to take the cake out. Some tasks, such as taking the cake out of the oven, involve other objects, which are added to the inventory. Other tasks require inventory items to be collected beforehand. Consider the task *Take the white dress and iron it*. The first step involves collecting the inventory item *white dress*, while the second step involves operating the iron. The user is able to view their inventory at any time. The problems range in complexity: the initial ones contain only three simple tasks, and they become more complex as the user practices in the environment.


Figure 3: Two scenes from the kitchen

In order to be able to track the progress of the user, the system maintains the list of active tasks. Tasks should only be attempted from a point known as 'cue discovery'. Time-based tasks become active several minutes before the stated time. For example, if the task is *Turn on the radio at 6.00pm*, the user can start to move towards the radio a few minutes earlier in preparation. Event-based cues only begin when the stated event occurs. Consider the task: *Bring in the washing when it starts raining.* For this task, the user has no way of knowing when it is going to rain, and so they should not begin the task before the cue is discovered.

For every task, there is a finite amount of time for which the task can be completed before it becomes obsolete or impossible. However, this alone is not the only factor in determining which tasks are more important. Some tasks, such as turning off the stove, have worse outcomes for failing to complete than other tasks do, such as turning on the TV. Each task therefore has a priority level, which is an integer from 0 to 5, with 5 representing the highest priority. Tasks with a level 5 priority are tasks with a very real chance of injury or household damage if they are not completed on time. A typical priority 5 task is *When the timer beeps, turn off the stove top*. By contrast, a priority 0 task may be: *When you are finished all other tasks, watch television.* From cue discovery, the user has a fixed time to complete the task before it becomes obsolete.

### *3.3 Using CBM for PM Training*

We have developed a set of constraints that enable us to evaluate the participant's actions and provide feedback. As originally proposed by Ohlsson (1992), each constraint has two components: a relevance condition and a satisfaction condition. The relevance condition specifies features of situations for which the constraint is relevant, while the satisfaction condition details what must be true for the constraint to be satisfied. A constraint can be described as: *If <relevance condition> is true, then <satisfaction condition> had better also be true, otherwise something has gone wrong*. If a constraint is violated, the user needs some means of knowing that he/she has made a mistake, and they need to know what needs to be done differently next time. This is the role of feedback: it informs the user on what tasks need to be performed, and what objects need to be interacted with.

We have developed 15 constraints to track the user's progress within the VR environment. The constraints deal with navigation, prioritization of tasks, selection of objects to perform actions on, remembering/selecting actions to be performed and general skills of interacting with VR (such as selecting objects, selecting items from the menu or crouching). In order to be able to specify relevance and satisfaction conditions, we have defined a set of functions and predicates. For example, the *OnRouteTo* predicate takes the current position of the user (i.e. the room the user is currently in), the target position needed in order to perform the current task, and returns *True* if the current position is on a path to the target position. An example of a constraint where one object is required to perform action on another is:

*If the user has selected an action for Object X which requires Object Y,*
*Then Object Y should be in the inventory.*

Each constraint contains three feedback messages. When a constraint is violated for the first time, the user will be given a general message, in order to remind them that they have missed something. For example, if the user is going in the opposite direction from the target destination, he/she will be given feedback "*You're going the wrong way*!" If the user continues down the wrong path, the feedback for to the second violation of the same constraint becomes more specific: "*Perhaps you should be going to the [goalRoom]*" (*[goalRoom]* is a function which returns the position for the current task). This culminates on their third violation of the constraint with "*You should be going to the [goalRoom] and use the [goalObjects]*". This is the bottom-out feedback which instructs the user what to do. Figure 4 illustrates a situation when the user tried to put a white dress on the clothes line, but forgot to collect the dress beforehand.

Three constraints check whether the user is working on the correct task. Tasks with only one minute left should be done before tasks with more than one minute left, even if that task with more than one minute left is of higher priority. In this way the user can still complete all the tasks. It is also important to bear in mind that higher priority tasks will reach the point of only having one minute left a lot sooner than a lower priority task. If there are multiple tasks with less than one minute left, the user should choose the highest priority one. The next threshold is at five minutes. Users must do tasks with less than five minutes left before they attempt tasks with more than five minutes left. As discussed in the previous section, tasks are first stratified according to time left into less than one minute, less than five minutes, more than five minutes. From there they are ranked according to priority. If any tasks have equal time strata and priority, they can be done in any order, otherwise the user must pick the top one.

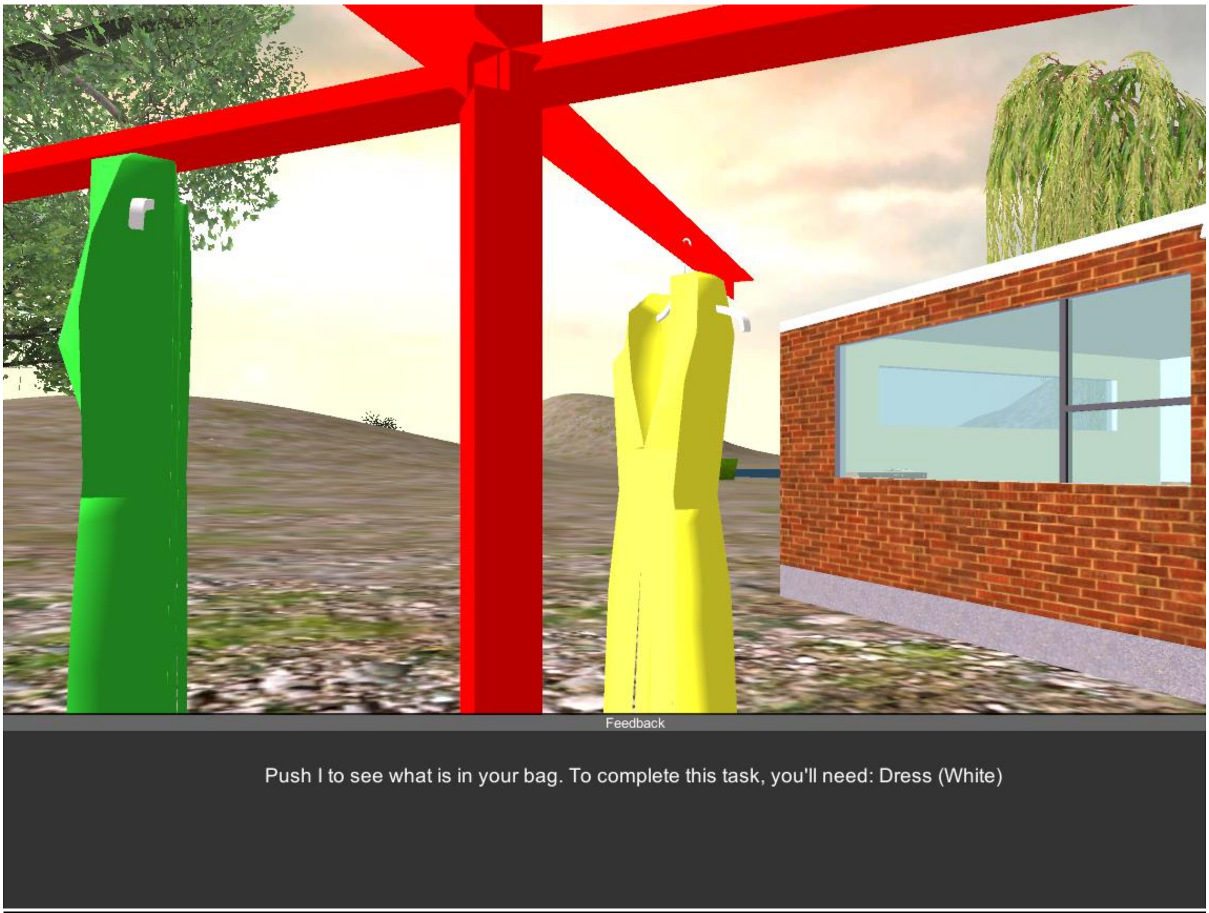


Figure 4: A screenshot showing feedback from a violated constraint

In addition to the feedback being displayed during the session, the user can also press the H key for more help. If the user has had a message displayed in the last 30 seconds, this message is displayed again to remind them of what they were doing wrong. Otherwise the default message is displayed. If there are no tasks left to do, the default feedback informs them of this. Otherwise it gives them increasingly specific hints as to what they should be doing.

In our previous work with ITSs, constraints are evaluated when the student submits the solution, therefore requiring feedback from the system. The timing of constraint evaluation in the VR environment differs from this: there is no time when the user explicitly requires feedback. On the contrary, the system needs to be able to evaluate constraints when appropriate. The constraints that deal with task prioritization are evaluated at intervals of 0.5s. Other constraints are evaluated in the appropriate contexts: for example, navigation constraints are evaluated every time the user changes room, while constraints that deal with objects are evaluated when the user selects an object or an action.

We have conducted a case study with a stroke survivor, who used the VR environment for 30 minutes. The participant could interact with the environment, and the case study identified a few usability issues and further improvements to the timing and duration of feedback. We then had a domain expert interact with the system. The domain expert explored the virtual environment completing a number of tasks. The domain expert was able to compare the feedback generated by constraints with the feedback they expected from the system. All constraints were satisfied or violated as expected, and these results were recorded faithfully by the user model. At some points, the feedback actually led to the domain expert making more errors. In such situations, the user was alerted that they should be doing one of several tasks, and told all the tasks currently available. When the user completed the lowest priority of these tasks, they violated the constraint that they should be doing the most high priority tasks. This led to the recommendation that feedback messages should only suggest the single most important task at the current time. The findings were then used to improve the constraint set and the system.

### 3.4 Evaluation Study

We are currently conducting an evaluation study of our computerized treatment with stroke patients. The study consists of ten sessions. In session 1, the participants will be tested to determine the level of cognitive functioning by performing a battery of cognitive tests, including the digit span test, the CAMPROMPT test of PM and the visual imagery questionnaire. Throughout the study, the participants are asked to keep a diary of their activities, and note PM tasks they have managed or failed to complete in their everyday activities. The diary will be discussed at the beginning of each session.

The second session is scheduled 4 weeks after the first session, so that it is possible to track the PM skills after this initial period. Starting from the second session, there are two one-hour-long sessions per week, for four weeks. The participant undergoes visual imagery training (discussed in Section 3.1) in sessions 2-4, followed by videos in sessions 5 and 6. In sessions 4 and 5, the participant is also introduced to the VR environment so that they can become familiar with it and also get used to using the joystick. The next stage of the study (sessions 6-9) is practice in the VR environment. The participant will be given 3-4 problems per session. At the end of session 9, the participant will again undergo the PM assessment. This assessment will allow us to measure the effectiveness of the treatment. Finally, the last session will be held after 4 weeks, and will consist of repeated assessment of the participant's PM. We will then analyze the results to determine whether the expected improvement in PM skills holds over the 4 weeks since session 9.

## 4. Conclusions and Future Work

In previous research, constraint-based modeling has been used to develop domain and student models in intelligent tutoring systems. CBM has been proven to be an effective student modeling approach that is applicable in a wide range of instructional domains. In this paper, we describe how we have used CBM in order to track the user's prospective memory. We have presented a computer-based treatment for improving prospective memory for stroke survivors. Our treatment consists of training users on how to use visual imagery to improve their memorization skills. Later on, participants practice using visual imagery by interacting with videos and a VR environment.

The contribution of this research is in extending constraint-based modeling from modeling and supporting cognitive skills to modeling and supporting prospective memory skills. We have developed a constraint set that allows us to track the user's behavior in the VR environment. The constraints identify whether the user is prioritizing the tasks correctly, whether there are any problems with navigation, identifying cues (time or event ones), interacting with objects and specifying actions. The pilot study performed with one stroke survivor was promising. Three domain experts have interacted with the VR environment and expressed satisfaction with the feedback the system provides. We are currently conducting a full study, aiming to have a group of 20 stroke patients to undergo all the phases of our treatment.


## Acknowledgments

This research was supported by a grant UOC1004 from the Marsden Fund of the Royal Society of New Zealand. We would like to thank all members of ICTG, and the participants in our study.


## References


Amalathas, S., Mitrovic, A., & Ravan, S. (2012) Decision-Making Tutor: Providing on-the-job training for oil palm plantation managers. *Research and Practice in Technology-Enhanced Learning*, 7(3), 131-152, APSCE.

Baghaei, N., Mitrovic, A., & Irwin, W. (2007). Supporting collaborative learning and problem-solving in a constraint-based CSCL environment for UML class diagrams. *International Journal of Computer-Supported Collaborative Learning*, *2*(2-3), 159-190.

Barker-Collo, S. L., Feigin, V. L., Lawes, C. M., Parag, V., Senior, H., & Rodgers, A. (2009). Reducing Attention Deficits After Stroke Using Attention Process Training A Randomized Controlled Trial. *Stroke*, *40*(10), 3293-3298.

Bohil, C. J., Alicea, B., & Biocca, F. A. (2011). Virtual reality in neuroscience research and therapy. *Nature reviews neuroscience*, *12*(12), 752-762.

Brooks, B. M., Rose, F. D., Potter, J., Jayawardena, S., & Morling, A. (2004). Assessing stroke patients' prospective memory using virtual reality. *Brain Injury*, *18*(4), 391-401.

Dyall, L., Feigin, V., Brown, P., & Roberts, M. (2008). Stroke: A picture of health disparities in New Zealand. *Social Policy Journal of New Zealand*, *33*, 178.

Ellis, J., & Kvavilashvili, L. (2000). Prospective memory in 2000: Past, present, and future directions. *Applied Cognitive Psychology*, *14*(7), S1-S9.

Elmadani, M., Mathews, M., & Mitrovic, A. (2012). Data-driven misconception discovery in constraint-based intelligent tutoring systems. In G. Biswas, L-H. Wong, T. Hirashima, W. Chen (Eds.) *Proc. 20th Int. Conf. on Computers in Education*, (pp. 9-16), APSCE.

Elmadani, M., Mitrovic, A., &Weerasinghe, A. (2013) Understanding student interactions with tutorial Dialogues in EER-Tutor. L. H. Wong, C-C Liu, T. Hirashima, P. Sumedi, M. Lukman (Eds.) *Proc. 21st Int. Conf. on Computers in Education*, Bali, (pp. 30-40). Uhamka Press.

Gálvez, J., Conejo, R., & Guzmán, E. (2013). Statistical Techniques to Explore the Quality of Constraints in Constraint-Based Modeling Environments. *International Journal of Artificial Intelligence in Education*, *23*(1-4), 22-49.

Gollwitzer, P. M. (1996). The volitional benefits of planning. *The psychology of action: Linking cognition and motivation to behavior*, *13*, 287-312.

Holland, J., Mitrovic, A., & Martin, B. (2009). J-LATTE: a Constraint-based Tutor for Java. In: Kong, S.C., Ogata, H., Arnseth, H.C., Chan, C.K.K., Hirashima, T., Klett, F., Lee, J.H.M., Liu, C.C., Looi, C.K., Milrad, M., Mitrovic, A., Nakabayashi, K., Wong, S.L., Yang, S.J.H. (Eds.) *Proc. 17th Int. Conf. Computers in Education* (pp.  pp. 142-146), Asia-Pacific Society for Computers in Education.

Knight, R. G., & Titov, N. (2009). Use of virtual reality tasks to assess prospective memory: Applicability and evidence. *Brain impairment*, *10*(01), 3-13.

Le, N. T., & Menzel, W. (2009). Using Weighted Constraints to Diagnose Errors in Logic Programming–The Case of an Ill-defined Domain. *International Journal of Artificial Intelligence in Education*, *19*(4), 381-400.

Lee, H. C., Chang, K. C., Huang, Y. C., Lan, C. F., Chen, J. J., & Wei, S. H. (2010). Inpatient rehabilitation utilization for acute stroke under a universal health insurance system. *Am J Manag Care*, *16*(3), e67-e74.

Lewinsohn, P. M., Danaher, B. G., & Kikel, S. (1977). Visual imagery as a mnemonic aid for brain-injured persons. *Journal of Consulting and Clinical Psychology*, *45*(5), 717.

Mathews, M., & Mitrovic, A. (2007). The Effect of Problem Templates on Learning in Intelligent Tutoring Systems. In R. Luckin, K. Koedinger, J. Greer (eds) *Proc. 13th Int. Conf. Artificial Intelligence in Education AIED 2007*, (pp. 611-613).

Mathews, M., & Mitrovic, A. (2009). Does framing a problem-solving scenario influence learning? In: Kong, S.C., Ogata, H., Arnseth, H.C., Chan, C.K.K., Hirashima, T., Klett, F., Lee, J.H.M., Liu, C.C., Looi, C.K., Milrad, M., Mitrovic, A., Nakabayashi, K., Wong, S.L., Yang, S.J.H. (Eds.) Proc. 17th Int. Conf. Computers in Education, (pp. 27-34) APSCE.

Mathews, M., Mitrovic, A., Lin, B., Holland, J., & Churcher, N. (2012). Do your eyes give it away? Using eye tracking data to understand students' attitudes towards open student model representations. In *Intelligent Tutoring Systems* (pp. 422-427). Springer Berlin Heidelberg.

Mathias, J. L., & Mansfield, K. M. (2005). Prospective and declarative memory problems following moderate and severe traumatic brain injury. *Brain Injury*, *19*(4), 271-282.

Mayo, M., & Mitrovic, A. (2001). Optimising ITS behaviour with Bayesian networks and decision theory. *Artificial Intelligence in Education*, 12(2), 124-153.

McDaniel, M. A., & Einstein, G. O. (1992). Aging and prospective memory: Basic findings and practical applications. *Advances in learning and behavioral disabilities*, *7*, 87-105.

Menzel, W. (2006). Constraint-based Modeling and Ambiguity. *Artificial Intelligence in Education*, 16(1), 29-63.

Mills, C., & Dalgarno, B. (2007). A conceptual model for game-based intelligent tutoring systems. *Proceedings of the 2007 Australasian Society for Computers in Learning in Tertiary Education*, 692-702.

Ministry of Health New Zealand (2002). Modelling Stroke - A multi-state life table model *Public Health Intelligence Occasional Bulletin* (Vol. 12).

Mitrovic, A. (1998). Experiences in implementing constraint-based modeling in SQL-tutor. In: B. Goettl, H. Halff, C. Redfield, V. Shute (Eds.) *Intelligent Tutoring Systems* (pp. 414-423). Springer Berlin Heidelberg.

Mitrovic, A. (2005). The Effect of Explaining on Learning: a Case Study with a Data Normalization Tutor. In C-K Looi, G. McCalla, B. Bredeweg, J. Breuker (Eds,) *Proc. Conf. Artificial Intelligence in Education* (pp. 499-506).

Mitrovic, A. (2012). Fifteen years of constraint-based tutors: what we have achieved and where we are going. *User Modeling and User-Adapted Interaction*, *22*(1-2), 39-72.

Mitrovic, A., & Martin, B. (2007). Evaluating the effect of open student models on self-assessment. *International Journal of Artificial Intelligence in Education*, *17*(2), 121-144.

Mitrovic, A., Martin, B., Suraweera, P., Zakharov, K., Milik, N., Holland, J., & McGuigan, N. (2009). ASPIRE: an authoring system and deployment environment for constraint-based tutors. *International Journal of Artificial Intelligence in Education*, *19*(2), 155-188.

Mitrovic, A., & Ohlsson, S. (1999). Evaluation of a constraint-based tutor for a database language. *Artificial Intelligence in Education*, 10(3-4), 238-256.

Mitrovic, A., Ohlsson, S., & Barrow, D. K. (2013). The effect of positive feedback in a constraint-based intelligent tutoring system. *Computers & Education*, *60*(1), 264-272.

Mitrovic, A., & Weerasinghe, A. (2009). Revisiting ill-definedness and the consequences for ITSs. In: V. Dimitrova, R. Mizoguchi, B. du Boulay, A. Graesser (Eds.) *Proc 14th Int. Conf. Artificial Intelligence in Education* (375-382).

Mitrovic, A., Williamson, C., Bebbington, A., Mathews, M., Suraweera, P., Martin, B., Thomson, D., & Holland, J. (2011). Thermo-Tutor: An intelligent tutoring system for thermodynamics. In *Global Engineering Education Conference (EDUCON), 2011 IEEE* (pp. 378-385). IEEE.

Najar, A. S., & Mitrovic, A. (2013, January). Examples and Tutored Problems: How Can Self-Explanation Make a Difference to Learning? In *Artificial Intelligence in Education* (pp. 339-348). Springer Berlin Heidelberg.

Ohlsson, S. (1992). Constraint-based student modeling. *Artificial Intelligence in Educatio*n, 3(4), 429-447.

Petry, P. G., & Rosatelli, M. C. (2006). AlgoLC: a learning companion system for teaching and learning algorithms. In: M. Ikeda, K. Ashley, T.-W. Chan (Eds.), *Intelligent Tutoring Systems* (pp. 775-777). Springer Berlin Heidelberg.

Riccucci, S., Carbonaro, A., & Casadei, G. (2005). An Architecture for Knowledge Management in Intelligent Tutoring System. In *Proc. IADIS Int. Cong. Cognition and Exploratory Learning in Digital Age* (pp. 473-476).

Roll, I., Aleven, V., & Koedinger, K. R. (2010). The invention lab: Using a hybrid of model tracing and constraint-based modeling to offer intelligent support in inquiry environments. In *Intelligent Tutoring Systems* (pp. 115-124). Springer Berlin Heidelberg.

Rosatelli, M. C., Self, J. A., & Thiry, M. (2000). LeCS: a collaborative case study system. In *Intelligent Tutoring Systems* (pp. 242-251). Springer Berlin Heidelberg.

Siddappa, M., & Manjunath, A. S. (2008). Intelligent tutor generator for intelligent tutoring systems. In *Proceedings of World Congress on Engineering and Computer Science* (pp. 578-583).

Suraweera, P., & Mitrovic, A. (2004). An intelligent tutoring system for entity relationship modelling. *International Journal of Artificial Intelligence in Education*, *14*(3), 375-417.

Titov, N., & Knight, R. G. (2000). A procedure for testing prospective remembering in persons with neurological impairments. *Brain Injury*, *14*(10), 877-886.

Weerasinghe, A., & Mitrovic, A. (2006). Facilitating deep learning through self-explanation in an open-ended domain. *International Journal of Knowledge-Based and Intelligent Engineering Systems*, *10*(1), 3-19.

Weerasinghe, A., Mitrovic, A., Thomson, D., Mogin, P., & Martin, B. (2011). Evaluating a general model of adaptive tutorial dialogues. In *Artificial Intelligence in Education* (pp. 394-402). Springer Berlin Heidelberg.

Yip, B. C., & Man, D. W. (2013). Virtual reality-based prospective memory training program for people with acquired brain injury. *NeuroRehabilitation*, *32*(1), 103-115.

Zakharov, K., Mitrovic, A., & Ohlsson, S. (2005). Feedback micro-engineering in EER-Tutor. In *Proc.Artificial Intelligence in Education*, (pp. 718-725). IOS Press.

Zakharov, K., Mitrovic, A., & Johnston, L. (2008). Towards emotionally-intelligent pedagogical agents. In *Intelligent Tutoring Systems* (pp. 19-28). Springer Berlin Heidelberg.