PROBING STUDENT PROBLEM SOLVING SKILLS IN MATHEMATICAL INDUCTION USING A SCENARIO BASED THINK ALOUD PROTOCOL

Benjamin Yu
Department of Computer Systems Technology
British Columbia Institute of Technology
3700 Willingdon Ave
Burnaby, BC, Canada V5G 3H2
byu@bcit.ca

Kimberly Voll
Department of Computer Science
University of British Columbia
2366 Main Mall
Vancouver, BC, Canada V6T 1Z4
kvoll@cs.ubc.ca

ABSTRACT

Think aloud protocol has traditionally been used to probe thinking and problem solving skills. In a series of student interview sessions where think aloud protocol was used to study student problem solving skills in mathematical induction at the University of British Columbia, a significant percentage of the students had difficulty verbalizing their thoughts while solving a problem. At other times, students felt pressured in coming up with a solution, or they were afraid of making incorrect steps and wanted to be certain before they actually articulated their thought process. As a result, a number of interviews resulted in almost complete silence and little was gained. To this end, we developed an enhanced, scenario based think-aloud protocol, designed to engage subjects during the interviews without having them feeling pressured or having to perform in front of an interviewer. In the process, the subject becomes less and less conscious of being studied and becomes more immersed in the actual problem solving with her own thoughts and ideas. In this paper we will describe this augmented think aloud protocol, and discuss our experiences of using this tool to assess mathematical induction reasoning in computer science students. A number of essential skills are identified through these interviews that may help instructors identify improved pedagogical methods for teaching the subject.

Categories and Subject Descriptors

K.3.2 [**Computers and Information Science Education**]: Pedagogy, education research, *think aloud, mathematical induction, in-class activities, problem-based learning, analysis*

General Terms

Measurement, Experimentation

Keywords

Think aloud, protocol, interviews, thinking, problem solving

WCCCE '11, May 6-7, 2011, Price George, BC, Canada.

1. INTRODUCTION

One of the greatest challenges facing teachers is to be able to understand how students think and process information. Without knowing how the students relate what they have learned in the past with the way they process new information, it is difficult to identify the source of misconceptions. Ericsson and Simon have shown that it is possible to objectively study the form of thinking that occurs covertly during many typical activities in everyday life using "think aloud" or "talk aloud" protocols (or, protocol analysis) [5]. Subjects are essentially asked to verbalize or talk through their thoughts as they work through a problem. This protocol is often used in usability testing, and has been adapted in Human Computer Interface (HCI) design to understand and observe how users work with different interfaces [6, 9], from software applications to hardware, or any task that requires interfacing with some form of media. The same protocol has also been used in education research to understand how novice and expert differ in their approaches to problem solving in different areas. In [1], students learn how to program by observing how experienced programmers work through a problem as they perform think aloud through the process.

Our initial inspiration for the use of think aloud protocol came from our studies at the University of British Columbia in methods for teaching mathematical induction. When considered separately, the core competencies required in the production of a successful mathematical induction proof are not complex, yet many students seemingly fail to grasp how these concepts combine to form a proof over an infinite series. A number of different approaches in teaching this topic has been proposed and used in the classroom [3, 4, 7, 8] but students often fail to generalize to new problems and instead resort to rote memorization of problems they have seen before. As such, final grades can be deceptively misleading when it comes to actual, deep conceptual learning [2]. Crucially, we are interested in observing what thought processes the students engage in when attempting mathematical induction problems and how we might adjust our teaching to create a more cohesive picture for the student.

Although we have found that the think aloud protocol is quite useful in some situations, our initial studies failed to provide much insight into student thinking during mathematical induction problem solving as hoped. We have followed the proper procedure, setting up an appropriate environment for these interviews to be conducted, but yet, most students did not offer much articulation of their internal thinking process from these interview sessions.

In this paper, we first identity the traditional think aloud protocol and why this protocol sometimes fails to illuminate such thought processes. We then propose an enhanced think-aloud protocol, demonstrating how this can be applied to education research in computer science. We will describe the interviews conducted and the results collected.

2. TRADIONAL THINK ALOUD PROTOCOL

Think aloud sessions are usually conducted in a one-on-one interview format [10]. In education research, a subject is usually presented with a problem to be solved and asked to verbalize their thoughts as they attempt to solve it. Some guidance may be provided by the researcher initially so the subject will know what is expected during the process. For example, the researcher may illustrate the process of adding two three-digit numbers together and verbalizing how she would add the unit digits first, then the ten's digits next with possible carry, etc. Additionally, asking the subject about their background, major, and any other relationship building questions will help establish the rapport and ease the subject into a mode of expressing their thoughts later on the session.

The key to a think aloud session that produces genuine result of the thought process is to have the subject verbalize their immediate thoughts rather than their meta-level explanation of how they go about solving the problem. The latter obscures the actual thought process as the subject explains their strategy. For example, the subject may switch strategies mid-solution if they suspect an error, but simply trying to explain why one solution may work will hide many details of that thought process that govern that switch.

Furthermore, in the case for student subjects, original problems should be provided to prevent them from spending time trying to recall a past solution or formula. While there is an obvious influence of past experiences in our ability to solve almost any problem, the objective in think aloud sessions is to uncover how the interactions of so many factors, such as independent thinking, past experiences, specific problem domains, problem context, reasoning skills, communication skills, et cetera, affect the subject's problem solving process and not solely their memory recall.

Finally, the interviewer should resist the temptation to prompt or probe the subject, even when the subject may be deep in their thinking and silent for awhile. Giving in to this temptation risks disrupting the thought process and may bias any answer given.

3. PROBLEMS WITH TRADITIONAL THINK ALOUD PROTOCOL

While think aloud sessions can be effective in eliciting a subject thought processes, the success rate of obtaining insightful data depends on a number of factors. As mentioned before, the choice of a suitable problem is critical. However, even with a well selected problem, the subjects may be either so involved with the problem, or so unused to verbalizing their thoughts, that most of the sessions are but silence. Even when probed to solicit some response from the subject, though not recommended as mentioned above, the subject may only minimally describe what they were thinking. Other times, the subjects may have little or no idea how or where to start the problem, and may feel uncomfortable expressing this. For some students, they may feel embarrassed in their lack of confidence to solve the problem, not recalling how a similar problem had been taught in class and shown how to be solved, or having thought of some ideas to solve the problem but they only lead to dead ends. Many students often expressed during the interview sessions that they "should" know how to solve the problem but have forgotten how to do it, trying to recall instead what they have learned rather than approach the problem afresh and risk failing in their attempts.

4. SCENARIO BASED THINK ALOUD PROTOCOL

We now describe an augmented form of the think aloud protocol, which appears to draw the direct attention away from the subject, helping them become more comfortable in solving a problem in front of the interviewers, and encouraging verbalization. It goes directly against the notion of unassisted think aloud interviews by guiding the subjects deliberately in the process. In this form of think aloud protocol, the subjects are asked to observe how two other individuals would solve a problem. While the setting can vary, we have found it effective to place the subject in a position to make a decision on hiring one of two interns in a fictitious company. To do so, the student must observe how these two interns work together to solve a problem and finally evaluate how each approaches the problem. This relieves some of the "performance anxiety" subjects face as the focus is instead drawn to the two individuals solving the problem. The subject now becomes a third-party observer, while they are actually being observed by the interviewer. The subjects also become more at ease and provide commentaries as they involve themselves in working through the problem, alongside the two problem solvers in the scenario.

To conduct this think aloud session, the subjects are given a script that presents the scenario and the "recorded" conversation of the two collaborating individuals. Interspersed in between the scripts at various points are questions directed to the subjects on whether they agree on the steps taken, the arguments that the two problem solvers in the scenario have constructed / defended / opposed as they work through the problem. From time to time, the subjects are also asked to pause from reading the script and to offer their ideas on how they would solve the problem themselves.

This form of scenario based think aloud session confines the subjects in providing commentary on their observations of the two individuals solving a problem in a specific way, rather than giving them the freedom to solve the problem anyway they want. Even though the subjects are asked on a number of occasions how they would have solved the problem on their own, it is quite different from the traditional think aloud protocol. Having applied this form of think-aloud session in both computer science and physics, our initial results show that the students are much more engaged than the traditional free form think aloud sessions.

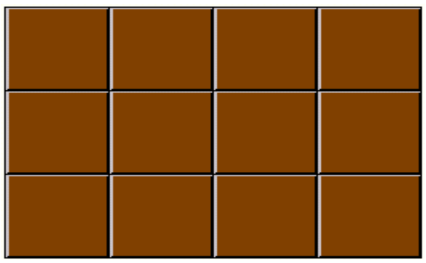
This form of guided think aloud interview allows us to probe how students think under a specific scenario. It is unclear how the constraints imposed on the subjects through the approaches taken by the two problem solvers in the scenario affects the subjects' actual thought process in their problem solving. In any case, when compared to the traditional think aloud sessions where there is minimal verbalization of their thought process, we believe that the scenario based think aloud protocol produces greater insights. Once we have collected some data from the interviews based on an initial draft of the scripts, we can then modify the scripts to probe deeper on what the subjects may be thinking at different points in future studies.

5. AN EXAMPLE SESSION

The following is a scenario based think aloud script used in computer science student interview to probe student thought process in solving a mathematical induction problem. The script consists of a dialogue between two interns, Jasmine and Sara, in a company who collaborated in solving a problem. The subject is asked to evaluate the conversation and decide which of the interns she would hire.

The problem is stated as follows:

Assume you have a chocolate bar consisting, as usual, of a number of squares arranged in a rectangular pattern. Your task is to split the bar into small squares (always breaking along the



lines between the squares) with a minimum number of breaks. How many will it take? Here is a sample of a chocolate bar to get you started:

The entire script is available at http://tinyurl.com/sbthinkaloud but it is too long to be reprinted here. Here is the beginning portion of the script which should give our readers a good idea what is involved during the interview process. The subject is asked to read the script and answer the questions in bold interspersed in the script.

After reading the chocolate splitting problem, Jasmine and Sara sit down and start thinking about the problem.

Jasmine thinks for awhile and says:

<Jasmine> It is difficult to think where to start. The question asks for a minimal break, so there is probably an efficient way to splitting the chocolate bar than just randomly splitting it.

1. **Do you have any ideas about what is the fastest way to split the chocolate bar? What would you try to figure out first? (No need to work out the whole problem right now.)**
2. **How do you feel about this problem?**
   a. **Could solve it on my own right now.**
   b. **Would rather solve it in a group setting.**
   c. **Need more information before I could say.**
   d. **It's completely over my head.**

   Please explain further:

<Sara> I'm sure they expect a fast method, that's sort of the whole point of the problem. Don't you think? But they also ask for "how many", so they are probably expecting a formula based on the size of the original chocolate bar.

<Jasmine> Sure. I guess you are right. How do we go from a chocolate bar to a formula then?

3. **What do you think?**

After a little discussion they agree on a plan.

<Sara> Ok, let's try to figure out how many splits we need for the sample chocolate bar presented in the problem.

4. **Is this a reasonable plan? Is there any reason that their plan may be a waste of time?**
5. **Is there something you want to do next or would you rather see what the girls are planning?**

<Jasmine> I am losing count on how many splits I need for the sample 3 x 4 chocolate bar, but I think the best I can do is 11 splits.

<Sara> Yeah, I have the same number too. Are there faster ways to split it? Does it matter if I split a big chunk or a small chunk early?

<Jasmine> I am not sure. Let's try some other chocolate bars of different sizes and see if there is a pattern!

6. **What do you think is the answer to the number of splits required for any *m x n* chocolate bars?**
7. **How confident are you about your answer?**
   a. **Positive**
   b. **Pretty sure**
   c. **Think it's close**
   d. **Not sure at all**

<Jasmine> This is interesting. For a 2 x 1 bar, we need one split. For a 3 x 1 bar, we need two splits. For a 3 x 2 bar, we need five splits. Can we conclude that for a m x n bar, we need (m x n) – 1 splits?

<Sara> Fascinating! It will be nice to do a few more samples, but the bigger the chocolate bars, the more time consuming it is to figure out how many splits. My head is starting to hurt already! Why don't we find a way to show that your theory is indeed correct without having to work through each chocolate bar size?

8. **How would you proceed from this point on? Do you think Sara's suggestion is even doable?**

<Jasmine> Great, where do we start? I am good at finding patterns in the cases we have worked through, but showing that the pattern will work for any case requires a leap of faith! Won't you agree?

<Sara> Well…. I remember learning about a proof method to generalize a statement or hypothesis for all numbers. I think it is called mathematical induction. We start with a base case, make sure the statement works for this base case. Then we assume it works for an arbitrary number, say n, and then somehow show that it will work for n+1.

Jasmine googles mathematical induction to refresh her memory.

<Jasmine> I remember this type of proof. I think I know the base case. If there is only a 1 x 1 chocolate bar, no split is required.

9. **What are you thinking right now? Is this a good base case?**

<Sara> Let me think… I think this will work. I was thinking of a 2 x 1 or 1 x 2 chocolate bar. In both cases, we need only 1 split.

At this point it is 5:00 so Jasmine and Sara pack up for the day.

10. **After the end of the first day, what are your initial impressions of Sara's and Jasmine's problem solving skills?**
11. **Do they both seem to know what they are talking about? Do you trust the facts and calculations that they've provided so far?**
12. **Who would you hire if you had to pick right now? Why?**

… continue to next day

6. INITIAL RESULTS

Thirteen students (two female) from the January session, 2010, of CPSC 121 (Models of Computation) at the University of British Columbia participated in the study which was conducted at the end of the term. None of the students have seen the exact problem before in their course. This course is the traditional, first introduction to mathematical induction in the standard CS program. Two faculty members were also interviewed to validate the script. Of the thirteen students, twelve of them worked through the script. All twelve of the students also attempted the open-ended questions when they were asked how they would proceed from certain points onward during the script. All of them took up the full hour allocated to each interview, and only two ran out of time to completely work through the entire script. This is an indication that the script is about the right length for a one hour interview.

All twelve students appeared very engaged in the problem, either working through the problem themselves or analyzing how the two interns were solving the problem. When they were asked how they would solve the problem at different points during the interview, all the students were able to make some attempts and provided some suggestions on how they would proceed most of the time. As a way of comparison, the chocolate bar problem itself, without the script of the interns working on the problem, was given to the thirteenth student to see if she could solve the problem in a traditional think aloud session. The student attempted the problem with almost no verbalization of her thoughts, and gave up after 5 minutes.

7. RESULTS AND DISCUSSIONS

Here are our initial results from the interviews:

Selection of Base Case: The base case selected among the twelve students ranges from one to four (for the chocolate bar size). This suggests that some of the students did not really understand the purpose of mathematical induction proof in generating a general solution over all natural numbers (i.e. 1 to infinity).

Choosing the right "n": In the scenario, the interns were distracted by the size of the chocolate bar being $m \times n$. The use of n is typical in the teaching of mathematical induction proofs in class, but it is also used in this particular problem to describe a dimension of the chocolate bar. All except one of the twelve students did not know how to apply the induction step from a chocolate bar of size $m \times n$ to $m \times n + 1$. Some tried $(m + 1) \times n$, others tried $m \times (n + 1)$, while some tried $(m + 1) \times (n + 1)$. Still some others were stuck in trying to "visualize" how a $m \times n + 1$ chocolate bar would look like! Most of them followed the wrong path like the interns initially and did not realize that they should have rephrased the problem in terms of a chocolate bar of total number of squares "n" rather than of dimension $m \times n$. Even after working out the solution with the interns, six of the twelve students did not fully grasp the significance of this renaming from $m \times n$ to simply just n. This seems to suggest that the ability to re-interpret the problem, or re-cast of the problem in terms of other variables need to be emphasized and practiced to enforce learning.

Writing out the proof: While some students may be able to apply a mechanical approach in working through the proof, they may not be able to reflect back and write out the entire proof even after all the steps have been completed. Often times, the students may wonder at the end of the proof whether they have actually proved anything. All of the students worked through the proof along with the interns, but when they were asked to summarize and write out the proof on their own, only two of the twelve students were able to do so. The ability to reflect and extract information from the proof activities and organize this in a concise presentation of the proof is a demonstration that the students have actually understood the problem fully, and articulate how the proof can be constructed. Failure to do so may be an indication that some parts of the proof may not entirely make sense to the students or they may be unsure of what are the essential parts necessary for this proof.

These initial results demonstrate the possibility for our approach in identifying key failings of student thought processes that can be used to direct teaching efforts. The script can be further tailored to probe student thought processes in problem solving. This will allow us to conduct research on specific areas where students may be failing in their ability to conduct the proof or synthesize their own complete proofs. More in depth probing in specific parts of the entire process can be added to the script to investigate how the students respond to other paths and directions to be taken by the interns in future studies.

Of interest is the impact the fictitious interns' mistakes have on the student thought process. It is not clear whether the students are in a position, having only been introduced to mathematical induction, to detect such failures, or whether they blindly follow the script. Continuing to apply this script to larger samples of students, not only at the beginner level, but those who have more experience with mathematical induction problem solving would produce useful results on their understanding as they progress in the study in computer science.

We are also interested in using this scenario based think aloud protocol in a class presentation and have students respond collectively to whether they agree or not on the interns' process of solving the problem. With the use of clickers (or personal response system), we can poll the students on whether they would follow what the interns are doing, or whether they would take on a different approach at different points in the process. Finally, we plan to apply our protocol in other aspects of CS education.

8. CONCLUSION

In response to the problem of frequent "silent" think aloud sessions, the scenario based think aloud protocol departs from the traditional approach by providing subjects with guided commentary on how others, possibly amateur problem solver(s) in our case, may go about solving a problem. The subjects are asked to evaluate the problem solvers and inject their own ideas, rather than having the focus placed on them as they attempt to solve the problem. The main benefit of our approach is that subjects are much more engaged, are better able to articulate their thought process, and they feel more at ease in working with the problem. Our initial results demonstrate that the method provides more details on the student thinking process than previously experienced with the traditional think aloud protocol. Finally, the use of scripted scenarios allows researchers to tailor their studies to examine certain aspects of student learning in the context of specific areas within a particular domain.

9. ACKNOWLEDGMENTS

Our thanks to Carl Wieman and Wendy Adams for their input on the scenario based think aloud protocol, and the support of CWSEI is greatly appreciated.

10. REFERENCES

[1] Arshad, Naveed. 2009. Teaching Programming and Problem Solving to CS2 Students Using Think-Alouds. SIGCSE '09, pp 372 – 376.

[2] BRACElet. 2006. BRACElet. Retrieved on February 19, 2009 from http://online.aut.ac.nz/Bracelet/repository2.nsf/HomePage?OpenPage.

[3] Dubinsky, E., and Lewin, P. (1986). Reflective Abstraction in Mathematical Education: The Genetic Decomposition of Induction and Compactness. Journal of Mathematical Behavior. 5, pp 55 – 92.

[4] Dubinsky, E. (1986). Teaching Mathematical Induction I. Journal of Mathematical Behavior. 5, pp 305 – 317.

[5] Ericsson, K.A. and Simon, H.A. (1998). How to Study Thinking in Everyday Life: Contrasting Think-Aloud Protocols with Descriptions and Explanations of Thinking. Mind, Culture, and Activity. 5(3), pp 178 - 186.

[6] Nielsen, J., Clemmensen, T., and Yssing, C. 2002. Getting access to what goes on in people's heads?: reflections on the think-aloud technique. In Proceedings of the Second Nordic Conference on Human-Computer interaction (Aarhus, Denmark, October 19 - 23, 2002). NordiCHI '02, vol. 31. ACM, New York, NY, 101-110. DOI= http://doi.acm.org/10.1145/572020.572033

[7] Polycarpou, I., (2006). Computer Science Students' Difficulties with Proofs by Induction: An Exploratory Study. Proceedings of the 44th Annual Southeast Regional Conference. pp 601 – 606.

[8] Segal, J., (March 1998). Learners' Difficulties With Induction Proofs. International Journal of Mathematical Education in Science and Technology. 29(2), pp 159 – 177.

[9] Stone, D., Jarrett, C., Woodroffe, M., Minocha, S. (2005) *User Interface Design and Evaluation*. Amsterdam: Morgan Kaufmann.

[10] van Someren, M, Barnard, Y, and Sandberg, J. (1994) The Think Aloud Method, *A Practical Guide to Modelling Cognitive Processes*, London: Academic Press.