Understanding the Role of Computers in Mathematical Problem Solving

Andrea Bunt, Edward Lank, and Michael Terry
David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada
<abunt, lank, mterry>@cs.uwaterloo.ca

ABSTRACT

Computer Algebra Systems (CAS) have been designed to aid users by automating a class of mathematical computations. However, despite widespread availability, little is known about how usable, useful, and used computers in general and CAS in particular are in mathematicians' work. Recently we presented a first examination of the overall work process of expert mathematicians and of the drawbacks of CAS in supporting this work process [4]. In this paper, we expand our discussion of mathematicians' work practices by highlighting two primary tasks -- problem solving and project management -- and discussing the different artifacts used during these primary tasks. We also show how a better understanding of mathematicians work practices can lead to the design of computational tools that better support mathematical problem solving. The study results reported here and in our recent work are the first analyses of how mathematicians work and of why current computational tools have only narrow utility in their work.

Keywords

Guides, instructions, author's kit, conference publications

INTRODUCTION

Mathematical problem solving is an expert task involving many different tools. For example, mathematicians use pen and paper to depict steps in their problem solving process. They use whiteboards or blackboards to describe problems and explore potential solutions with collaborators. They use Computer Algebra Systems (CAS) (e.g. Maple, Mathematica) or Numerical Computing Packages (e.g. MATLAB) to perform difficult or lengthy calculations. Finally, they use mathematical equation generation tools such as LaTeX to publish and disseminate novel mathematical proofs they create.

Because of the need for tools in mathematical problem solving, many computer science research projects seek to create improved tools to support mathematical problem solving. Researchers in Symbolic Computation or in Numerical Analysis seek to create more effective algorithms to perform difficult calculations. As well, one research thrust of one of this paper's authors [4, 8, 22] has centered on the design of Pen Math Systems, computational systems where mathematicians can use an electronic stylus on either bibliographic (i.e. small, personal devices such as Tablet PCs) or epigraphic (i.e. large, shared screens such as electronic whiteboards) surfaces to draw mathematical equations [8, 9, 10, 13]. Equations, drawn by hand with an electronic stylus, are recognized at the semantic level, and the electronic representation of the mathematical data can then be used as input to, for example, Computer Algebra Systems. One specific Pen Math System, MathBrush [8], is pictured below. Other similar systems can be found both in the research literature [9] and as commercial software applications [13, 14].

While many researchers have explored the design of pen-math systems, as Human-Computer Interaction (HCI)

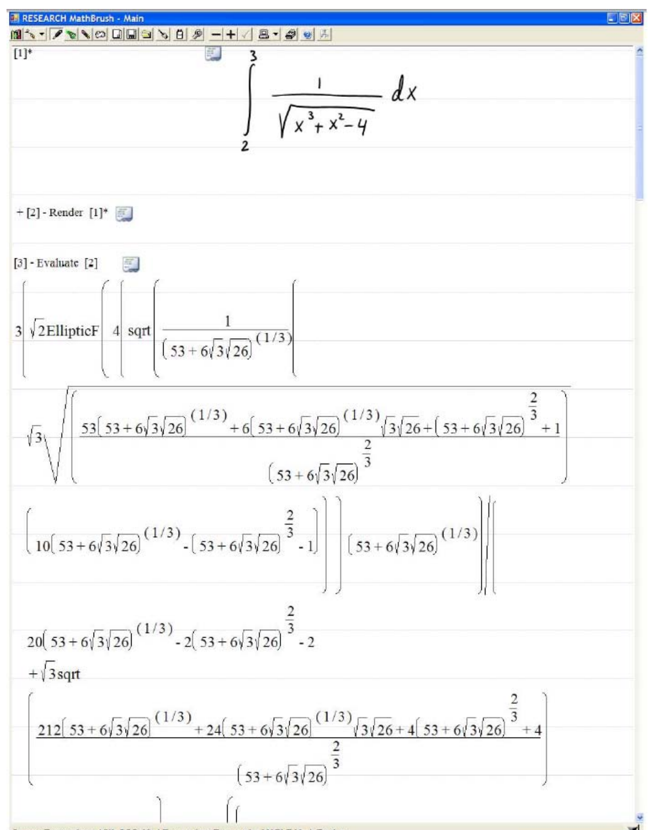

Figure 1: MathBrush, depicted here, is one popular Pen-Math system.

researchers, we found ourselves asking a number of questions about this overall research thrust. First, how do mathematicians work? In HCI, before examining different options for technology design, best practice would dictate that researchers and designers first develop a solid understanding of the work practice of target users. However, in the case of mathematicians, we found little work describing the tasks, phases, or artifacts used in mathematical problem solving. Second, how are computational tools (e.g. CAS) incorporated into mathematical problem solving? Computational systems to solve mathematical problems are undoubtedly necessary aids to mathematicians. For example, in a two month period after the release of an updated version of Maple in 2006, approximately 800 000 licenses were activated. However, while we know that these tools are used, we do not know how frequently they are used, and for what style of tasks. Finally, what do mathematicians think of current computational tools? Do they trust the tools? Do they use them frequently? Do they offload all by-rote computation to the tools?

Understanding these questions has implications in the creation of tools to help mathematicians, both at the design level and at the user interface level. For example, understanding how mathematicians work allows us to develop insight into potential new interactive tools for mathematicians. Understanding how computers are currently used suggests ways that current computational interfaces can be improved. Finally, understanding attitudes allows us to better gauge whether tools we design will be used by mathematicians and, hopefully, how they will be used.

In recent work [4], we presented a first analysis of the role of computation in mathematical problem solving. The goal of this paper is two-fold. First, we seek to disseminate our past analysis to researchers working in mathematical user interfaces. Second, we expand on our analysis of work practice of mathematicians, specifically highlighting approaches to project management. Finally, we explore some design possibilities that arise from our research on the work practice of mathematicians. It is our hope that presenting some of our extended results in a workshop venue will encourage discussion amongst researchers interested in the design of interactive tools for mathematical problem solving.

It should also be noted that there are limitations to our past work, and we welcome discussion with the MathUI community about these limitations. One limitation is in the selection of research subjects that we studied. We specifically chose to examine the work practices of theoretical mathematicians (as opposed to applied mathematicians, scientists, engineers, or students who study mathematics). The goal of theoretical mathematicians is to develop new mathematical knowledge, not to present numerical answers to specific problems. Other groups – including applied mathematicians, scientists, engineers and students – are undoubtedly more interested in concrete answers, so some themes may not fully transfer. We will discuss these issues in additional detail when we present future work. Second, the data from which we develop insight into mathematical problem solving processes and the attitudes of mathematicians towards computational tools was based on a set of contextual interviews and an examination of work artifacts (papers, whiteboards, etc.) in our subjects' offices. Self-reported work practices may differ from actual work practice. To guard against this, we were careful to ask for walkthroughs of recent problem solving tasks, rather than for generalizations of the problem solving process. We also asked to see the recent work we were discussing so that our data is based around reconstructions, not recollections, of past work.

This paper is organized as follows. In Related Work, we highlight past work studying the role of computers in mathematical problem solving. We also give a detailed overview of the results of our recent SIGCHI work on the role of CAS in Mathematical Problem Solving. We then expand on the mathematical problem solving process by describing differing approaches to project management by mathematicians we studied. We then present some design issues that arise from our work, focusing specifically on what tools should be designed for mathematicians, and how those tools might be used.

RELATED WORK

Computers in Mathematical Problem Solving

Past research, including HCI research, on the use of computers as it intersects mathematics has largely focused around laboratory studies assessing general features, or around the use of computers in high school mathematics classes.

In a laboratory evaluation of mathematical problem solving with high-school students as participants, Oviatt et al. studied the impact of different media on problem-solving performance. The experiment included four conditions: pen and paper, an Anoto pen, a pen-based Tablet PC, and a graphical equation editor [16]. The authors found that problem solving performance was better with pen and paper or the Anoto pen compared to the other two conditions. Drawing on Cognitive Load Theory (e.g., [23]), the authors attribute the results to the familiarity students have with entering and manipulating expressions with physical media, leading to comparatively higher cognitive loads when using digital media for these tasks.

A number of studies have focused on the problem of entering mathematical expressions. Anthony et al. compared pen-based entry to keyboard-and-mouse, speech, and pen plus speech. The authors found that expression entry with keyboard-and-mouse was significantly slower than the other three conditions and that pen-based entry was the most preferred [2]. A pair of studies has also considered expression entry in the context of pen-math systems: systems use a tablet PC as an interface to CAS software (or some type of mathematical backend) with the goal of creating a more natural input interface. LaViola [9] and Labahn et al. [8] both assessed the user's ability to correctly enter expressions and solve a number of small problems. The evaluations showed that while expression recognition can be challenging in such systems, users were able to complete their tasks effectively once their expressions had been recognized. These studies again suggest that pen-based input has particular advantages, but that digital systems do not offer clear-cut advantages with current interfaces and recognition engines.

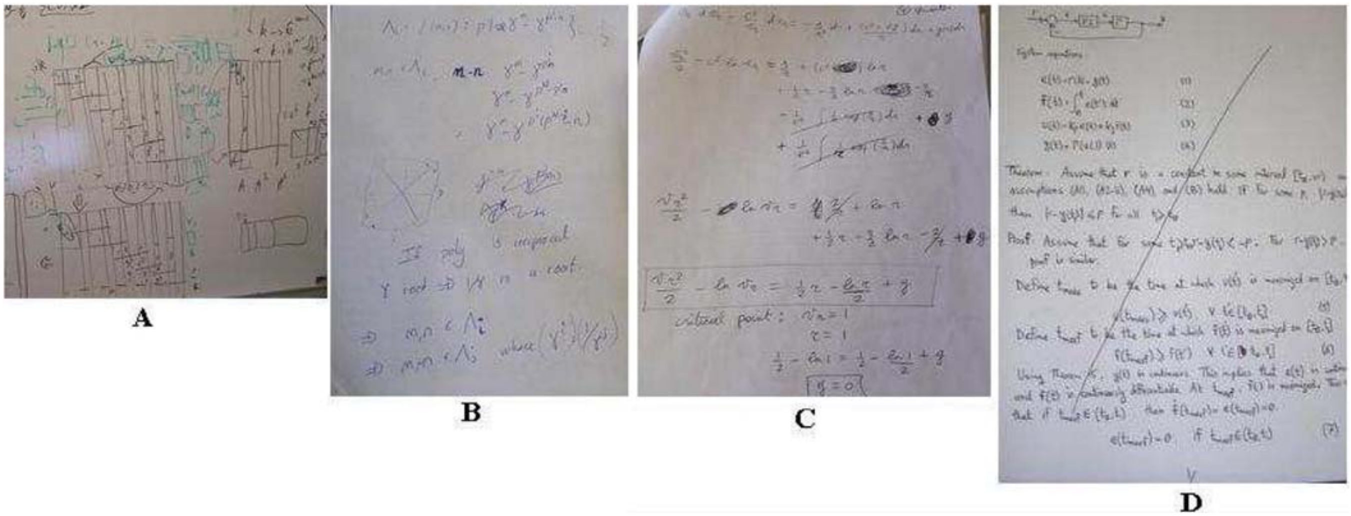

Figure 2: Photos of artifacts mathematicians work with. From left to right, we see increasing levels of formality, echoing P6's description of the work process.

Moving outside of the laboratory setting, various research efforts have investigated how CAS integrate with high-school and undergraduate education (e.g., [3], [11], [17], [18], [20]). These studies have uncovered a number of advantages to using CAS in the classroom. For example, one study found that students are able to experiment with different expressions more easily, which can promote a higher-level understanding of relevant concepts [3]. In addition, by delegating some of the work to the CAS, students are more able to focus on problem-solving processes as opposed to focusing solely on calculation details [11]. This line of research, however, has also shown that integrating CAS into the classroom requires careful lesson planning [11] and teacher support [17], and that some students have difficulty translating CAS output into representations that they understand [3]. Finally, some students feel that they learn more when solving problems by hand or that "real mathematics" is done by hand, not by computers [18].

This initial set of studies, both in the laboratory and in educational settings, provides important insights into the potential benefits and limitations of current CAS software. For example, expression input can be singled out as one area in need of further improvements. However, these studies characterize only short-term use of such software, typically in fairly well-defined, well-directed ways (e.g., in an experimental study or in the classroom with well-defined task).

Understanding Mathematical Work Practices

To the best of our knowledge, the recent study of Bunt et al. [4] is the first study of how professional mathematicians make use of computer applications in authentic work situations. An understanding of how these tools are adopted and applied in professional environments is valuable, both in guiding improvements to these tools and in identifying new tools which can aid mathematicians.

To analyze mathematicians' work processes and their use of computers, Bunt et al. interviewed nine different mathematicians from their institution. Subjects included three faculty members, three post-doctoral fellows, two Ph.D. students, and one Masters Student. Bunt et al. analyzed their results using established techniques from qualitative research. The primary results of their research can be grouped into two main themes: First, they develop an understanding of how mathematicians work; Second, they identify a series of factors that inhibit the use of current computational tools in mathematical problem solving.

How Mathematicians Work

Mathematical problem solving is an expert-level task involving a number of discrete steps. One mathematician described solving a new mathematical problem as follows:

Okay, this is how I work. First of all, I think about the problem. I draw some meaningless figures like this [artifact] and then I translate what I see to some equations. Then I write my equations down [in a way] that is readable by someone else, like this [artifact]. [...] And then I type it and then I submit it. (P6)

Artifacts also provide good clues into mathematical work practice. Artifacts, drawn from a number of participants, align well with this work practice, as shown in Figure 2. The first phase of P6's work, where meaningless figures are used as an aid to insight, coincides with Figure 2A. Figures 2B and 2C represent the intermediate phase, when insights are translated into mathematical equations. Finally, Figure 2D is the third phase, where the equations are written in a way that is readable by someone else. It should be noted that the artifacts in Figure 2 do not belong to P6 specifically; instead, they are artifacts drawn from random participants in our study. These artifacts demonstrate the commonality across work practices of different mathematicians.

Based on an analysis of the comments of all of the participants plus an analysis of photos of artifacts of the participants, Bunt et al. identify four discrete phases in mathematical problem solving:

1. *Ideation*: A brainstorming phase where ideas are generated.
2. *Execution*: Ideas are carried out by solving, deriving, and constructing mathematical proofs.
3. *Formalization*: The results of the previous two phases are refined such that the work becomes a more complete mathematical narrative.
4. *Dissemination*: The work is prepared such that it can be presented to others, either via publication or a more formal presentation to a supervisor.

While mathematical problem solving progresses through these phases, Bunt et al. are careful to note that, as in any creative endeavor (software engineering, architecture, etc.), the phases may be interleaved in various ways. For example, during the process of formalization, participants may need to revisit the ideation phase to refine ideas, or during the process of dissemination, additional steps must be performed at the execution phase to refine ideas. However, the above phases of work represent an overview of the essential stages involved in developing and refining new mathematical knowledge.

Usefulness of Current Computational Tools

Prior to our work on the use of computational tools in mathematics, we believed that mathematicians valued and used computers in their work practice. It seems obvious that computer applications cannot solve all of the problems mathematicians work with. However, what caught us by surprise was how assiduously mathematicians avoided using computational tools. Here, we first overview our findings, reported in Bunt et al. on how mathematicians use computers in their work. We then explore their attitudes toward these tools.

There are two primary tools used by mathematicians in their work practice. The first is LaTeX, considered the gold standard for formatting technical documents for dissemination. Mathematicians invest significant time into mastering LaTeX, and the system is considered powerful and efficient for document formatting and typesetting expressions.

The second tool for our participants is Computer Algebra Systems, specifically Maple for the mathematicians we interviewed. Mathematicians used Maple for two purposes during problem solving. The first is to solve difficult expressions that are either challenging or tedious. For example, as two participants note:

Usually if it is a complicated expression that I can't resolve myself. [...] the kind of tedious work that is sort of boring and uninteresting but where it is easy to make mistakes. (P1)

If I have some horrible expression that I don't like, some large amount of tedious computation, integratethis or reduce this giant mess to something useful, then sometimes I'll stick it in Maple to see if it can solve the problem for me. (P3)

The other common task for computer algebra systems is in rapid experimentation, essentially testing a large set of alternatives to determine how an expressions responds to various inputs:

It's a matter of just testing all possible solutions to see if they are solutions or not. And the algorithms are really the fastest way I can test that. (P2)

While Maple was considered an aid by our participants in their problem solving, they were frequently reluctant to use Maple. The discomfort with Maple was a result of a four distinct issues with current computer algebra systems:

- A lack of transparency in the problem-solving process
- A lack of support for free-form 2D representations
- Difficulties transcribing representations from physical to digital representations
- No support for collaboration

We briefly touch on each of these issues here.

The most frequently cited problem with Maple involved a lack of transparency in how numbers were generated. This lack of transparency seemed to limit both the insight provided by Maple-generated solutions:

Computers are great for running through large amounts of examples, but you don't get the same insights. Whereas if you did something by hand, sometimes you just get more insight and can figure out the general pattern. (P2)

As well, the limited transparency meant that mathematicians were reluctant to trust the results:

I tend to not trust the results from the symbolic toolbox [...] Although it is very infrequent that the results are incorrect. (P6)

In part, these observations may also arise from a stated desire of mathematicians to stay sharp by exercising "certain parts of my grade 12 calculus class." (P1).

However, it seems that if participants had a better understanding of the steps taken by a computer application to generate a solution, then they would be more likely to trust that solution.

The second area where computers short-change mathematicians in problem solving is in flexibility of layout. As one participant noted:

And I don't even necessarily work down the page. [...] I just sort of have everything all in one spot. Obviously it's not very neat or easy to deal with, but just having everything on one page kind of makes a big difference [...] I think it's easy having everything all in one spot. It just stops me from forgetting anything. (P7)

While some computer algebra systems (e.g. Mathematica) allow some annotation of the problem solving process through free-form text, our mathematicians still desired more freedom in the placement of text and graphics than traditional CAS allow.

Third, there were a set of difficulties associated with transcription. These difficulties were related to the need to master a command set for expression input into computer algebra systems. Any new command set has a learning curve, and when the representation is different from the common paper-based representation, there is always a risk of failing to note errors. As one participant stated:

I'll type in an expression, I'll have spent an hour trying to figure out what it means and what the results are, and then I realize I've made an error typing. (P1)

Finally, our mathematicians frequently make use of blackboard and whiteboards because they often collaborate with other researchers in problem solving. Computational tools currently provide only limited support for collaboration.

Summary

Bunt et al.'s recent work on how mathematicians use computers provides useful data as input to the design of new computational tools. It also suggests ways that existing tools can be improved. While we have spent significant time reviewing this past work here, we feel that this extensive review is a useful starting point for discussion at the MathUI workshop. We now turn to some additional results, not previously reported, which can also provide guidance for design and refinement of interactive mathematical tools.

PROJECT MANAGEMENT IN PROBLEM SOLVING

In our past work [4], we presented a detailed analysis of the problem solving process mathematicians use to create new mathematical knowledge. However, working through a solution using the four problem-solving phases is only one aspect of mathematical problem solving.

In addition to the four problem-solving phases, many participants also discussed a meta-mode, which we refer to as Project Management. Project Management pertains to tracking the progress of one or more ongoing projects. Several participants mentioned creating and maintaining archives of work completed to date, allowing them to keep track of their current progress and more easily re-orient themselves after time away from the work. While we did not see a strong correlation between research experience and the tendency to engage in this meta-mode, one participant mentioned that this was something he did earlier in his career, but now found himself (as a postdoc) concentrating on only one project at a time

I think after the PhD, you work on a topic and either you give up or you make a draft and submit it. (P5).

Participants indicated using either LaTeX- or paper-based project management schemes. As an example of a LaTeX-based scheme, P1 provided us with a sample document consisting of a collection of work related to a particular problem. He indicated that the document represents a rough collection of material that he has worked on, that begins to resemble a paper as the project progresses. Analysis of the document revealed that in addition to resembling a draft of a paper, it also contains notes on things that need further exploration, and ideas and formulations that did not pan out (the latter stored in an Appendix). Thus, for this participant, the latex document appeared to aid in both formalization and project management.

Paper-based project management schemes included folders for grouping related projects and special notebooks with removable and re-configurable pages.

I'm very excited about the notebook. This is why: so these ones, they're from France, and the pages are removable. So they are like a binder, so that you can take the pages out and back in. And one thing that's nice is I do different things and I can group together stuff and if I finish a notebook and I haven't quite finished the project then I can take the stuff and put it back in. (P9).

The elaborate nature of many of these project management schemes suggests an opportunity for design. Mathematicians need to archive paths that were not fruitful, and to indicate avenues which may prove useful but have not yet been fully explored (perhaps because multiple alternative paths forward exist). As well, because the process of developing new insight may require sporadic intervals of work over long periods of time, they need some mechanism for tracking their on-going progress. The heavy-weight nature of electronic solutions (partial documents typed in LaTeX) and the ad-hoc nature of paper-based project management schemes represent work-arounds developed to perform this necessary task.

This identification of one design opportunity leads naturally to potential tools that may arise from our work. In the next section, we present some thoughts on potential tools of use to mathematicians in their work.

DESIGNING TOOLS FOR MATHEMATICIANS

Given an understanding of the work practices of mathematicians, mathematics user interface work can address two needs. First, systems can be designed which aid mathematicians in additional tasks they perform during their work. Second, tools can be designed (or redesigned) to address those aspects of mathematical problem solving that are poorly served. Alongside the design of new or improved tools, an understanding of work practices also allows us to evaluate current research thrusts (such as pen-math systems) in light of the data we have collected on how mathematicians work and their attitudes toward technology.

Designing New Computer Applications for Math

In this paper, we highlight several drawbacks of current computer algebra systems. Many of the drawbacks of CAS can be addressed by improvements to current CAS systems. For example, if transparency is a problem, make computations more transparent to expert users. If collaboration is a problem, introduce tools for collaboration. If additional support is needed for free-form placement of expressions, modify the current workspace inside computer algebra systems to support this free-form placement.

There are problems with this narrow approach to improving tools for pen-math systems. Consider first transparency. While transparency might make CAS more trusted, there is still a desire to "exercise grade 12 calculus" skills and a belief that doing math by hand guides insight. This belief was expressed by participants in our study, and it was also observed by researchers studying high school students [16, 18]. It then becomes an open question as to whether computer tools for math should always assume the computational burden, or should they instead sometimes strive to support verification of manual tasks. There is clearly a role for traditional CAS systems in solving difficult or tedious problems. In these situations, a CAS that presented some transparency to the user might be more trusted. However, there is also a role for systems which verify work done by the user.

One way to implement systems which verify rather than replace the mathematician is to explore intelligent paper systems. The Anoto pen [6] is one example of an intelligent paper application. The Anoto pen has a camera, a power source, memory, and possibly a Bluetooth transmitter embedded in the barrel of a standard ballpoint pen. When writing on special paper, the pen captures and stores the strokes drawn. The strokes can then be uploaded to a computer via Bluetooth, if present, or via a USB connection. Anoto pen technology has been used to design intelligent mark-up systems for paper documents, systems that use specialized gestures to interact with both the content on the paper and with computer systems that monitor the content. One could imagine extending these designs to the world of mathematics. Researchers and students could work on problems using Anoto pen and paper. Specialized gestures, monitored using Bluetooth, could be used to highlight work that should be verified. If errors exist, a computer located nearby could indicate the error, or, if no suitable computer is nearby, a computer which receives the Bluetooth data could text message a user's cell phone with potential errors.

Collaboration presents similar alternatives. While collaboration within CAS is one alternative, it seems that restricting collaboration to the small screens typical of PCs running CAS is not an optimal solution. Instead, electronic whiteboard based solutions might be more suited to collaborative math. Systems such as Flatlands [15] and IBM's Blueboard [19] have explored novel ways of supporting the serendipitous collaborative creation of information on a malleable display located on an electronic whiteboard. In Flatlands, different regions of the board are defined dynamically as people work, and other regions are pushed aside and resized to make room for the new content. Within the region, the work being done is analyzed, and specialized environments for different work (e.g. a set of math tools, etc.) are applied. With Blueboard, collaboration is simplified by easily supporting the dissemination of shared drawings. Tools such as these might be much more useful in supporting shared discussion than would a shared canvas in a CAS.

Finally, freeform placement on a 2D canvas is a desirable feature mentioned by our subjects in our earlier study [4]. However, even with freeform typed text placement, CAS still place a significant burden on users to either type free-form content or to use specialized commands to enter expressions from the keyboard. In contrast, some research in pen-math systems has focused on integrating free-form notes with expressions to be recognized [9, 10]. In some versions of the MathPad^2 system, users can switch modes between equation entry and note taking. While the system was conceived of as a tool for students taking notes in a classroom, it seems that this ability to switch back and forth between note taking and expression entry might be a desirable feature. Whether such systems are implemented on tablet computers or with Anoto pens, the ability to drop the constraint for formally correct expressions and to sketch out ambiguous intermediate solutions for later refinement seems a desirable feature for mathematicians of all levels.

Beyond the problems identified with current tools, significant aspects of mathematicians' work practice are currently unsupported. As we noted earlier, the Project Management task that many mathematicians engage in while doing research over longer periods of time could be supported by novel interactive tools. Here, again, systems like the Anoto pen [6] or Wellner's Digital Desk [24] might monitor a mathematician's current work. Through the use of specialized gestures or placement of documents (for example if a document is tossed in the recycling bin) the mathematician might be able to off-load some project management and tracking onto intelligent computer systems.

Evaluating Current Research in Pen-Math

Many pen-math systems [8, 9, 25] are premised on the assumption that equation entry with LaTeX [25] or in CAS [8, 9] is exceedingly difficult. For LaTeX, this assumption may be mistaken, as mathematicians express little hesitation about using LaTeX and seem to have little desire to change typesetting tools. For CAS, the problem seems to have less to do with entry and more to do with verification, as subjects note that they can't detect typing errors. Even with pen-math systems, it is very possible to miss recognition errors which change the semantics of your expression.

Beyond a possibly mistaken assumption that expression entry is the most pressing need for mathematicians, pen-math systems are also designed to speed problem solving, similar to current CAS. The goal is to allow mathematicians to enter equations, to have the systems recognize the equations, and then to solve the mathematical problems represented by the equations rather than have the mathematician solve the problem by hand. As such, while they may replace CAS and solve some of the problems with CAS (such as 2D input), they still suffer from the drawbacks of CAS, namely a lack of transparency and an off-loading onto the computer of calculations that our participants like to do by hand to "stay sharp". Whether pen-math systems are the most appropriate avenue of near-term research effort is something that is open to debate.

FUTURE WORK

While our on-going research represents the first study of mathematicians work practices in professional settings and their attitudes toward the computer tools they use in these settings, it, like all work, has potential weaknesses. For example, it may be the case that a study of applied mathematicians would yield markedly different results. To address this issue, we have begun conducting interviews of applied mathematicians, engineers, and first and second year undergraduate students. While our preliminary results indicate that generating numerical answers using packages like MATLAB may be more important than symbolic manipulations for some subjects, many similar themes (transparency, collaboration, free-form input, narrow utility) are still present in our data. We continue to analyze qualitative data with the goal of providing an even richer view of how mathematicians work.

As well, our work to date has focused on qualitative interviews to capture work practice, with the attendant risk of inaccurate description of actual practice. This risk is always present in self-reports of work practice. In the future, we plan to perform both experience sampling and diary studies to capture more accurately how mathematicians work with computers.

Finally, while we have proposed possible technology to design solutions for mathematicians, we have yet to prototype any systems for this space. Prototyping of possible applications is a natural next step that we plan to take as our understanding of mathematicians' work practice continues to evolve.

CONCLUSION

In this paper, we present an overview of our work studying mathematical problem solving as practiced by theoretical mathematicians. We find themes in their work and their attitudes toward computation that echo findings by researchers studying high school students. We highlight these findings and speculate on possible avenues of UI design for mathematicians. It is our hope that the work we present here might encourage a discussion on the role computers can best serve in mathematical problem solving.

ACKNOWLEDGMENTS

The research described here was supported by the Natural Science and Engineering Research Council of Canada (NSERC) through the Discovery Grant Program and by the Ontario Ministry of Innovation through a Post-Doctoral Fellowship. Ethics review of this research was performed by the Office of Research Ethics, University of Waterloo.

REFERENCES

1. Adoption of Maple 10 into Academic Market Exceeds All Expectations. www.marketwire.com/press-release/Maplesoft-563405.html.
2. L. Anthony, J. Yang, and K. Koedinger. Evaluation of multimodal input for entering mathematical equations on the computer. In Proc. of CHI, pages 1184–1187, 2005.
3. M. Artigue. Learning mathematics in a CAS environment: The genesis of a reflection about instrumentation and the dialectics between technical and conceptual work. Int. J. of Computers for Mathematical Learning, 7:245–274, 2002.
4. A. Bunt, M. Terry and E. Lank, "Friend or foe?: examining CAS use in mathematics research." In *Proceedings of the 27th international Conference on Human Factors in Computing Systems,* CHI '09. 229-238.
5. Derive. www.chartwellyorke.com/derive.html.
6. F. Guimbretiere. Paper augmented digital documents. In Proc. of UIST, pages 51–60, 2003.
7. J. Hadamard. The psychology of invention in the mathematical field. Dover Publications, Inc., New York, N.Y., 1945.
8. G. Labahn, E. Lank, M. Marzouk, A. Bunt, S. MacLean, and D. Tausky. MathBrush: A case study for pen-based interactive mathematics. In Proc. Of SBIM, 2008.

9. J. LaViola. An initial evaluation of MathPad2: A tool for creating dynamic mathematical illustrations. Computers and Graphics, 31:540–553, 2007.
10. J. LaViola and R. R. Zeleznik. MathPad2: A system for the creation and exploration of mathematical sketches. In Proc. of SIGGRAPH, pages 432–440, 2004.
11. C. Leinback, D. Pountney, and T. Etchells. Appropriate use of a CAS in the teaching and learning of mathematics. Int. J. of Mathematical Education in Science and Technology, 33(1):1–14, 2002.
12. Mathematica for Students. www.wolfram.com/products/student/mathforstudents.
13. Mathjournal. www.xthink.com/MathJournal.html.
14. Microsoft Corporation, http://blogs.msdn.com/e7/archive/2009/04/23/ink-input-and-tablet.aspx.
15. Mynatt, E. D., Igarashi, T., Edwards, W. K., and LaMarca, A. 1999. Flatland: new dimensions in office whiteboards. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '99. 346-353.
16. S. Oviatt, A. Arther, and J. Cohen. Quiet interfaces that help students think. In Proc. of UIST, pages 191–200, 2006.
17. R. Pierce, S. Herbert, and J. Giri. CAS: Student engagement requires unambiguous advantages. In Proc. of the 27th Annual Conf. of the Mathematics Education Research Group of Australasia, pages 462–469, 2004.
18. R. Pierce and K. Stacey. Observations on students' responses to learning in a CAS environment. Mathematics Education Research Journal, 13(1):28–46, 2001.
19. Daniel M. Russell , Clemens Drews , Alison Sue, Social Aspects of Using Large Public Interactive Displays for Collaboration, Proceedings of the 4th international conference on Ubiquitous Computing, UBICOMP 2002, p.229-236.
20. K. Ruthven. Instrumenting mathematical activity: Reflections on key studies of the eductional use of computer algebra systems. Int. J. of Computers for Mathematical Learning, 7:275–291, 2002.
21. D. Schon. The reflective practitioner: How professional think in action. 1983.
22. D. Tausky, G. Labahn, E. Lank, and M. Marzouk. Managing ambiguity in mathematical matrices. In Proc. of SBIM, 2007.
23. J. vanMerrienboer and J. Sweller. Cognitive load theory and complex learning: Recent developments and future directions. Educational Psychology Review, 17(2):147–177, 2005.
24. Pierre Wellner, Interacting with paper on the DigitalDesk, Communications of the ACM, v.36 n.7, p.87-96, July 1993
25. Richard Zanibbi , Dorothea Blostein, James R. Cordy, Recognizing Mathematical Expressions Using Tree Transformation, IEEE Transactions on Pattern Analysis and Machine Intelligence, v.24 n.11, p.1455-1467.