On Dynamically Presenting a Topology Course

Paul Cairns and Jeremy Gow

Interaction Design Centre
Middlesex University
The Burroughs, Hendon
London NW4 4BT, UK
{p.cairns, j.gow}@mdx.ac.uk
www.cs.mdx.ac.uk/imp

Abstract

Authors of traditional mathematical texts often have difficulty balancing the amount of contextual information and proof detail. We propose a simple hypermedia framework that can assist in the organisation and presentation of mathematical theorems and definitions. We describe the application of the framework to convert an existing course in general topology to a web-based set of materials. A pilot study of the materials indicated a high level of user satisfaction. We discuss further lines of investigation, in particular, the presentation of larger bodies of work.

1 Introduction

When considering the organisation and retrieval of mathematical knowledge, it is important to consider the needs of the human reader. Humans read mathematics not just for the satisfaction afforded by a complete, logical deduction but also for the new insights into the mathematical objects described. Indeed, it could be argued that it is the ideas of mathematics that are more important to humans than the bare logic. Using computers, it may be possible to provide interactive presentations that expose and explicate the central ideas of mathematical results without losing the necessary rigourous logic.

Proof is central to modern mathematics. We have developed a theoretical framework for interactively presenting proofs. Specifically, the framework sets out how to use hypertext as a means of balancing the context of results and the details of their proofs. The framework is supported by authoring tools and we have begun to evaluate its efficacy in user trials. However, as it is unlikely that a single presentational style could please all possible readers, we have chosen a user base for whom rigorous proof is a main feature of their work, namely, university and research level mathematicians

There is a natural progression from aiding understanding to educational support but education is far more than useful presentations[16]. Rather, the new framework should be considered as a supplement or even replacement for existing mathematical texts. In this sense, the framework is of educational value but it alone does not constitute a complete educational tool.

Having addressed some of the conflicts in context and detail when presenting mathematics, the paper considers a naive, interactive presentation of mathematical knowledge. This leads to the proposed new presentational framework based on the work of Polya[14] and Lamport[10]. The framework was applied to an undergraduate topology course, which was evaluated in a pilot study. We describe both the user and author experience of this course and the issue of implementing tools to support the authoring process. We also discuss ideas for future developments.

2 Context and Detail

On paper, much of modern mathematics is organised into a standard form: definitions and axioms come first then theorems that follow from them, then the proofs of those theorems, then further theorems and proofs, more definitions and so on. And even within a single proof, the statements of the proof follow in logical progression from the previous statements, theorems or axioms. However, there are well known problems with this format:

1. Without sufficient contextual content, the specific form of definitions and theorems may seem arbitrary[9]...
2. ...however such context is essentially irrelevant to the logical argument.
3. Too much detail in proof obscures the important insights [17] ...
4. ...but with insufficient detail a reader will not be able to follow the argument [13].

There is a clear tension between too much or too little content and furthermore between too much or too little contextualisation. When authors prepare printed mathematical text, they necessarily take a cut through all the possible material that they could present in the hope of providing a balanced quantity of context and detail. Almost inevitably, the balance chosen is not appropriate for all readers. Indeed, it may not be appropriate even for a single reader at different times. For instance, a reader may require a detailed proof initially but after digesting the proof would no longer need the details and would look rather for the context and motivations behind the proof.

With hypermedia, mathematical authors need not be constrained to choose specific levels of context and detail. Readers could interact with on-line materials to manage the presentation to their satisfaction, without unnecessary materials being forced upon them.

The context of a result is basically anything that might help a reader to understand what the result is and why the reader should need to know it. However, understanding the significance of mathematics is often distinct from the strict logical statements of definitions and the arguments followed in proofs. Instead, insights come from a variety of sources. Some concepts seem obscure without the rationale that led to their particular form. A typical example of an obscure definition is the concept of bounded variation. It in fact evolved from being a technical condition in theorems related to Riemann-Stieltjes integration[9]. The importance of a particular result often comes through its relationship to other results. Wiles' proof of Fermat's Last Theorem, though sensational for the actual result, is more important mathematically for the unification of two previously disparate branches of mathematics [18]. This also shows that sometimes it is the proof itself that is significant about a particular theorem.

As well as external influences, there may be internal pressures that lead to the particular form of a theorem. For instance, why is a certain condition important to a particular theorem? And how does the theorem apply in specific situations? These questions can be answered through examples illustrating the theorem, possibly with diagrams, and also examples that illustrate what happens when certain conditions are omitted from the statement of the theorem. All such extra-logical material, both external and intrinsic to the theorem, we refer to as context. It is this context together with the logical details of a proof that we aim to manage with a good hypertext structure.

3 A Naive Approach

The simplest approach to producing a hypertext version of a mathematical text would be to mark it up with HTML, converting references to definitions and theorems to hyperlinks through to the referred text. However, it is clear that this does not solve any of the issues of providing context and detail — that could only be done by altering the content. Additionally, as mathematics is usually dense with specific concepts and results, it turns out that much of the text becomes hyperlinked. After a point, it is hard to work out what a user should expect to see on following a link and if they do follow a series of links, they can rapidly get 'lost in hyperspace' [20].

Adding more context and detail to the pages, even if 'hidden' on pages of their own, would only proliferate the number of links and increase the likelihood of getting lost. The naive approach then is certainly not going to be sufficient to organise content and help the user navigate through the content. Instead, the issue of managing context of the mathematical ideas and the details of the proofs is partitioned.

4 The Polya-Lamport Framework

4.1 Polya's Four Steps

Contextual material can be useful when trying to solve problems. This was carefully discussed and demonstrated by Polya [14, 15]. He proposed an approach of problem solving that actively applied heuristic thinking. Rather than considering a problem in isolation, the problem-solver is encouraged to make links with similar problems, address the problem in concrete or limiting cases and to reflect on any progress. Polya set out four steps that the solver should follow:

Understand	the problem with examples, diagrams and a careful examination of each of the terms and unknowns in the problem
Plan	how you intend to solve the problem
Execute	the planned solution with care
Reflect	on the result, how it relates to other results and how it might be proved differently

In other words, Polya promotes thinking about the context of a problem in order to solve it (**understand**) and having solved it to bring it back into the wider theory from which it arose (**reflect**). The solution is also explored at different levels of detail (**plan** and **execute**).

Just as Polya's steps are appropriate in understanding a problem, they are helpful in understanding a theorem too. After all, for a particular person every theorem is initially an unproven challenge. We propose to use these steps as a *framework for the presentation of theorems*. The **understand** step is used to clarify the statement of the theorem. It also provides the point at which to link to definitions used in the theorem. The **plan** step can be used to provide an overview of the key steps in the proof and also provide links to other related results that are essential to the proof. The **reflect** step can provide links to future theorems or further definitions and so position the theorem within the theory. This leaves the **execute** step which must contain the detailed proof of the theorem.

4.2 Lamport's Structured Proof

It is the **execute** step that contains the challenge of providing a sufficient but not overwhelming amount of logical detail. Lamport has proposed structured proof, an approach to managing proof detail which he has already employed with good effect in organising proofs on paper[10].

A proof written in this format consists of recursive lists of key proof steps, each justified by a subproof. In this way, the high level steps give the outline of the proof and the low level steps the details. The obvious adaptation to hypertext is to hide or reveal the subproofs under the reader's control. This was first done by Grundy[6] for a calculational style of proof (i.e. a linear derivation). We propose to implement the **execute** step of our framework with an expandable/collapsible structured proof based on Lamport's original design for general proofs.

The framework for theorems, then, is to use the Polya steps to provide the links to the context and to use the Lamport proof presentation as the **execute** step. In this way, the reader can fully control the amount of context or detail that they wish to see. The theorem presentation framework is shown in the top half of Table 1.

4.3 Presenting Definitions

The framework can also be extended to cover definitions. Definitions are an integral part of mathematics and, indeed, developments in definitions seem to drive and be driven by developments in theory [9]. As there is not generally a proof associated with a definition, the amount logical detail is not a problem but it would still be useful to provide a context for a definition. We therefore looked at adapting the Polya steps to definitions. Clearly, the **understand** step, that is, providing links with existing definitions, giving examples and diagrams carries over to definitions easily. As too does the **reflect** step. The **plan** step clearly has no relevance to a definition but as the place where the general justification of the theorem was given, this has been replaced by the **plausibility** step that gives a justification of the definition in terms of its evolution and significance. The **plausibility** is then backed up by the **employment** step, replacing the **execute** step. **Employment** links to key results in which the definition plays an important role or which exemplify the need for the definition.

Thus, the complete Polya-Lamport framework is able to organise the presentation of both definitions and theorems as summarised in Table 1. There are natural places at which to interlink concepts both with earlier theory and future ideas. Moreover, the user can be

Theorem Steps	**Purpose**
Statement	Statement of theorem
Understanding	Links to relevant examples, diagrams and definitions
Plan of Proof	High level insights into the proof
Execute Proof	Interactive, hierarchical proof of theorem
Reflect	Developments or variations of the basic statement

Definition Steps	**Purpose**
Statement	Statement of definition
Understanding	Links to relevant examples, diagrams and definitions
Plausibility	Justification for the definition
Employment	Key theorems employing the definition
Reflect	Developments or variations of the basic statement

Table 1: A summary of the Polya-Lamport framework

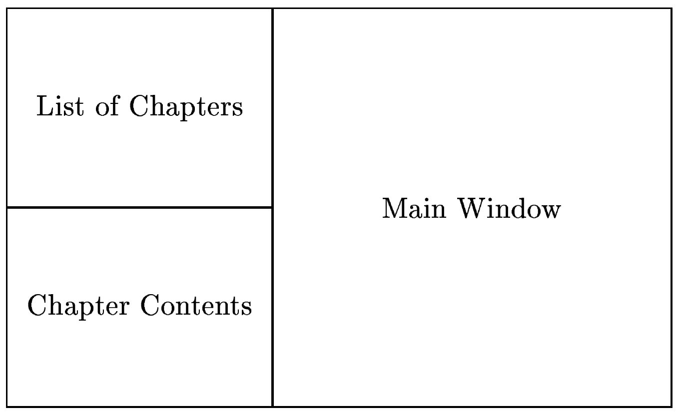

Figure 1: Layout of Windows in the Topology Course

certain of what to expect on following a link. There are also clear, consistent points at which to discuss the context of theorems or definitions.

5 Applying the Framework

In order to evaluate the usability and usefulness of the framework, it was applied to converting an undergraduate course in general topology. This course, "Elements of Euclidean and Metric Topology" by Dr. Peter Collins of Oxford University, is taught to all first year mathematics undergraduates at Oxford.

The course consists of five chapters (with a zero-th chapter containing necessary background material) covering basic notions of continuity, connection and compactness in metric spaces[19] as well as touching on more abstract topological ideas. As the framework is concerned with the presentation of individual theorems or definitions, some additional components were needed to show the structure of the course and help users navigate around.

We opted for a simple, but commonly found design, with a top left window displaying chapter titles and a bottom left window displaying actual theorems and definitions in that chapter as in Figure 1. Clicking on links in the lower left window changed the main content of the page. There were also links in the top left window that listed all theorems or all definitions in the lower left window. The evaluation materials used can be found at `www.cs.mdx.ac.uk/imp/topology`.

Within each of the theorem or definition pages, the content was presented using the Polya-Lamport framework described above. The pages were written in HTML, with sections corresponding to the steps given in Table 1, except for the **execute** proof steps for theorems, described below.

5.1 Expandable Proofs

Recall that in our framework, the theorem pages contain an the **execute** step that gives an interactive expandable proof based on Lamport's structured proof. This was implemented via a Java applet which allows the user to expand or collapse subproofs by clicking buttons located next to the subresult they prove. By default all subproofs are hidden, and the user can expand subproofs down to whatever depth they require.

Rather than hand-craft such a Java applet for each theorem page in the course, the applet accepted as input the text of the structured proof, with the structure marked up via XML[1]. For this purpose, we defined a simple XML grammar, or DTD, which formally captured Lamport's notion of structured proof. Lamport originally described this approach by example[11], but it was relatively straightforward to define a grammar which captured the spirit of the original. We omit a full description here, but the XML DTD (Document Type Definition) is given in Figure 2 to illustrate the approach. An alternative would have been to use OMDoc[7] to provide a structured input, and we intend to investigate this approach.

5.2 Technical Issues

In order to make any materials produced with the framework widely and easily available, the implementation principle was that it should be compatible with existing browsers, namely Microsoft Internet Explorer and Netscape Navigator, without requiring plug-ins or fast server connections. This is reflected in our choice of a combination of HTML and Java. One of the main limitations of this approach, though, is that there is no direct way to display mathematical notation that is native to existing browsers. Many of the implementation decisions are a result of trying to work round this.

As noted, except for the Java proof applet, the framework could be easily implemented in HTML. However, in order to get mathematical symbols, these pages are written in LaTeX[11] and converted to HTML by TtH package, which uses e.g. HTML font tags. An alternative would be Tex4ht[5], which uses images.

Within the XML structured proof, mathematical symbols are written in LaTeX notation. These parts of the XML proof are preprocessed using Tex4ht, producing images that are displayed along with the proof in the Java applet. For simplicity, we decided for LaTeX rather than MathML, because the latter is very long-winded, still only has limited character sets and does not have open source display systems. Also, the tools for converting LaTeX to

```
<!ENTITY % opt   "CDATA #IMPLIED" >
<!ENTITY % label "id %opt; name %opt; child %opt; depth %opt; path %opt;" >

<!ELEMENT assume  (#PCDATA) >
<!ATTLIST assume  %label; item %opt;>
<!ELEMENT prove   (#PCDATA) >

<!ELEMENT theorem (assume*, prove+, proof?) >
<!ATTLIST theorem %label; >
<!ELEMENT let     (prove+, proof?) >
<!ATTLIST let     %label; >
<!ELEMENT case    (assume+, proof?) >
<!ATTLIST case    %label; >
<!ENTITY  % step  "(theorem | let | case)" >

<!ELEMENT ref     EMPTY >
<!ATTLIST ref     %label; >
<!ELEMENT qed     (ref+) >
<!ELEMENT term    (#PCDATA) >
<!ELEMENT calc    (term, ref*) >

<!ELEMENT sketch  (#PCDATA) >
<!ELEMENT proof   (sketch?, (theorem | ((%step;)+, qed) | (term, calc+) | ref+)) >

<!ELEMENT theory  (#PCDATA) >
<!ATTLIST theory  id %opt; name %opt; item %opt; >
<!ELEMENT title   (#PCDATA) >
<!ELEMENT imp     (title?, theory*, theorem) >
```

Figure 2: XML DTD for Lamport's structured proofs

HTML are well established and robust, which is sadly not the case for those that we tried for MathML to HTML.

This architecture may seem unnecessarily baroque, but without good, native browser support for mathematics, it is actually the simplest and most controllable technique that was clearly feasible at the outset. Already, we are developing a new implementation using Javascript[4] and XSLT[3]. It simplifies writing in the framework though it still requires substantial machinery in order to produce the final pages. We anticipate the technical issues will become less complex and the solutions more robust once technologies for displaying mathematics on the web have matured.

6 The User Perspective

For a pilot study, seven undergraduates were given tasks to perform with the web-based course notes, whilst being observed. Afterwards they answered a SUS questionnaire [2] and were interviewed about their experiences of the interactive materials. Though seven undergraduates is hardly a large sample set, we have followed Nielsen's view that five or six users are sufficient to find all the major problems in a user interface[12].

As a quick and dirty measure of usability, the SUS questionnaire formed the basis of the evaluation. It consists of a set of fairly general questions about the users' subjective satisfaction with the interface. The pilot study gave us an SUS score of 78% which we understand to mean that the users are well satisfied with the system though there is some room for improvement. Unfortunately SUS does not give specific feedback on what those improvements should be!

From the observations of users working with the materials, it seemed that they were able to find materials reasonably quickly. However, the proof applet seemed to cause some navigational problems as it had its own set of scroll bars within the browser window and expanding and collapsing parts of the proof tended to move material off the screen. This was borne out by dislikes mentioned in the interviews.

Several of the users commented on the expandable proofs as a good feature of the materials. This is a good sign of the value of the hierarchical proof structure though it must be taken with a pinch of salt: the users may not have had sufficient time to make a balanced assessment of the proofs. And positive comments notwithstanding, the users had plenty of suggestions of how they might like to see the applet improved including: more use of colour; links in and out of the proof applet to the materials; and the ability to annotate proofs like a paper proof. These are worth considering in future versions of the proof applet.

Another positive feature that users generally brought was the links to supporting material, in particular, the ease with which definitions could be found. This seems to support the breakdown of the materials into the Polya steps.

A significant comment made by a couple of the users was the unfamiliarity of the hierarchical proof structure. This comment has also been made in other presentations of the material. It may be that a hierarchical presentation ends up being unacceptably different from traditional proof styles despite initial enthusiasm for it. Or alternatively, if mathematicians really do want to see varying levels of detail, they may have to give up their familiar and well-worn styles of presentation!

7 An Author's Perspective

Although we have so far developed only basic tools to support authoring in the Polya-Lamport framework, there are one or two issues that arose whilst authoring that are worth highlighting.

As with any hypertext system, there is the problem of maintaining a consistent, reliable and memorable way of producing a large amount of inter-linked documents. This is particular relevant for the framework as there are not only links between pages but also many links within the parts of a proof. And if user comments are to be followed, more links between pages and parts of the proof are needed! Without good authoring tools, it is unlikely that linking will become any easier.

Secondly, though much of the topology course translated into the Polya steps, producing the hierarchical proofs was more difficult. The author still had to make a decision on what to present at each level of the proof. We have not found any hard and fast rules here, as it relies so much on the meaning and perceived importance of the proof steps.

Also, the author had to decide at what level the proof of a particular statement was complete. As humans naturally make logical leaps or skip trivial detail, the proofs were not logically complete, even after adding considerably more steps than appeared in the course book.

8 Discussion and Future Work

The pilot studies seem to show that the Polya-Lamport framework has at least solved some problems of balancing context and details of a proof. It was noticeable that no users commented on difficulties in using the Polya steps. We take this as a positive sign that the steps seemed natural and sensible or at the very least did not hinder users.

The hierarchical proof seems to be popular with users, at least initially, but there is definitely room for improvement. Certainly as an intuitive way of hiding or revealing detail, it is a central idea in the framework. The **plan** step is essential to the proof as without it, key insights, steps or proof techniques may be hidden inside a subproof. Even with the **plan**, the reader may have no way of knowing how to find these steps. A more sophisticated approach may be to summarize the subproofs by identifying these important elements allowing the proof to stand by itself without the **plan**. However, adding in improvements and new features may end up in it being too complex to be usable. We are therefore also considering other possible techiques for hiding and revealing the details of proofs.

For the authors, the difficulties with hierarchical proofs seem more about meaning than usability. Automated theorem provers could be used to complete proofs and so relieve the author of the burden of generating the lowest level logic, particularly if a complete logical proof is considered to be essential by the users. The current framework is a long way from integration with a theorem prover but we are very interested in trying to do it.

The framework so far only covers the presentation of particular statements, be they definitions or theorems. As the topology course was reasonably small and self-contained, navigating through it was not too taxing for the user. This may not be the case with larger bodies of work, be they large theories and/or large proofs. The user will need to be able to find appropriate content easily through a corpus of possibly several hundred pages. Moreover, when considering whole theories rather than a particular, focused topic, the context and justification of the theory could be as useful as the context of individual theorems.

The Polya-Lamport framework provides no place for theory-level context. It will be necessary to look elsewhere for inspiration. Lakatos suggests revealing the context and evolution of a theory using a dialogue with different characters representing different philosophies that affected the theory [9]. Indeed 'Proofs and Refutations' is a fine demonstratation of how the development of Euler's formula for polyhedra could be rationally reconstructed. But despite having been around for many years, there is not a stream of mathematicians repeating this process on other branches of mathematics. Our feeling was that the knowledge and literary skill required to develop such a presentation took an author so far away from mathematics that only exceptional mathematicians would be willing or even able to attempt it. We felt vindicated on finding only one other dialogue presentation of a theory and it was written by Knuth[8].

This does not mean that a rational reconstruction of the development of theory is impossible. Perhaps the dialectic element could be replaced with collections of hyper-linked materials. But this is clearly a matter requiring a great deal more thought and attention.

Throughout any future work, it is essential to ensure that what is produced is what users need. This can only be done with more in-depth and focused evaluations. We have plans to produce a larger topology course and also the proof of a large and complex theorem. These will be evaluated in user trials in the course of the project. We also have interest from other mathematicians in Britain and we are hoping that they may provide a variety of theories, approaches and styles with which to investigate the range of applicability of the Polya-Lamport framework.

9 Conclusion

Presenting mathematical proofs has conflicting concerns in providing a balanced amount of context and detail. Our overall aim is to develop a conceptual hypermedia framework for interactively presenting proofs that real mathematicians would like to use. The novel Polya-Lamport framework goes some way to this goal, organising context through a series of steps and managing detail using an interactive, hierarchical proof structure. Early user feedback indicates that this provides a usable environment that certainly improves upon naive approaches. There are possibilites to improve the framework but for us the more interesting question is how to organise entire mathematical theories and this will be a focus of future work.

Acknowledgments Many thanks to the undergraduate mathematicians of St. Edmund Hall, Oxford University and to Peter Collins for the use of his notes and his on-going support. This work was funded by EPSRC, grant GR/N29280.

References

[1] T. Bray, J. Paoli, C.M. Sperberg-McQueen & E. Maler (eds) , *Extensible Markup Language (XML) 1.0, 2nd edn*, W3C Recommendation October 2000
`http://www.w3.org/TR/REC-xml`

[2] J. Brooke, 'SUS — a quick and dirty usability scale',
`http://www.usability.serco.com/trump/methods/satisfaction.htm`

[3] J. Clark (ed), *XSL Transformations 1.0*, W3C Recommendation November 1999
`http://www.w3.org/TR/xslt`

[4] D. Flanagan, *JavaScript: The Definitive Guide, 3rd edn*, O'Reilly, 1998

[5] , M. Goossens & S. Rahtz, *The LaTeXWeb Companion*, Addison Wesley, 1999

[6] J. Grundy, 'A Browsable Format for Proof Presentation,' *Math. Universalis*, 2, 1996
`http://www.ant.pl/MathUniversalis/2/grundy/mu.html`

[7] M. Kohlhase, *OMDOC: An Open Markup Format for Mathematical Documents.* SEKI Report SR-SR-00-02, Universität Des Saarlandes, 2000.

[8] D. Knuth, *Surreal Numbers*, Addison Wesley, 1974

[9] I. Lakatos, *Proofs and Refutations: The Logic of Mathematical Discovery*, Cambridge University Press, 1976

[10] L. Lamport, *LaTeX: A Document Preparation System,2nd edn*, Addison Wesley, 1985

[11] L. Lamport, 'How to write a proof', *American Mathematical Monthly*, 102(7) p600-608, 1994

[12] J. Nielsen, 'Quick Ways to Improve Your Site's Usability', *ZD-Net ECommerce*, 2001
`http://www.zdnet.com/ecommerce/stories/main/0,10475,2679774,00.html`

[13] L. C. Paulson & K. Grabczewski, 'Mechanizing Set Theory,' *Journal of Automated Reasoning*, 17 p291-323, 1996

[14] G. Pólya, *How to Solve It*, Penguin Books, 1990

[15] G. Pólya, *Mathematics and Plausible Reasoning*, Princeton University Press, 1968

[16] P. Ramsden, *Learning to Teach in Higher Education*, Routledge, 1992

[17] B. Russell, 'The Study of Mathematics' in *Mysticism and Logic*, Penguin, 1953

[18] S. Singh, *Fermat's Last Theorem*, Fourth Estate, 1998

[19] W. A. Sutherland, *Introduction to Metric and Topological Spaces*, Oxford University Press, 1985

[20] Y. L. Theng & H. Thimbleby, 'Addressing Design and Usability Issues in Hypertext and on the World Wide Web by Re-Examining the "Lost in Hyperspace" Problem,' *Journal of Universal Computer Science*, 4(11), p839-855, 1998