Infinite Precision Document Formats

Author: Samuel Moore[1]
Partners: David Gow[2]
Supervisor: Prof Tim French

THE UNIVERSITY OF
WESTERN AUSTRALIA
Achieve International Excellence

April 11, 2014

What we will be doing and why.

See also the Introduction. See also the Conclusion.

- Research into document formats is mostly focusing on embedding dynamic content in the document
- TODO: Desperately try to find research into increasing precision in documents
  - I suspect the reason it is hard to find references saying "We increased the precision you can get in this graphics format" is because the real research is into "how to get more precision out of a number" and increasing the precision in a graphics format is just an application.
  - Talk about why precision in itself is a good thing?
    - Reduced error in computations
  - Somehow manage to link this to graphics formats.
    - Increased precision means you can have more zoom, you can have more range
    - This is useful because the view of a document as a sheet of paper is increasingly outdated
  - Precision in itself is necessary in order to address the document format issue.
- In particular we are looking into increasing precision vs IEEE floats because:
  - Such research is important in its own right; eg: Doing numerical calculations requires high accuracy results
-

I guess what I am trying to say very badly is:

We want to be able to do more things with documents. We are currently limited by the precision of floating point operations. This is a fundamental and physical limit on what can be done.

Even things like postscript are already turing complete. All the fancy papers about putting dynamic content and javascript and things in documents are not addressing a physical limitation, merely one of convenience for the document creator.

So it is worth looking into how we can reduce or eliminate the physical limitations on what can be done with a document due to floating point precision.

Contents

Chapter 1

Introduction

The introduction will be witty and engaging and a joy to read.

I hope.

Points to address:

1. What this is.
2. Why we should care
3. Most recent development
4. Do we really mean “infinite”? (No. Sorry. We lied)
5. Foundational papers. Historical notes.

See also: The Abstract. See also: The Conclusion.

The remainder of this report will be organised as follows: Chapters 2 and 3 will discuss approaches to representation of numbers in both Hardware and Software respectively; Chapter **??** will outline the approach [reword] considering ¡insert fuzzy list of faculty dependent requirements here (Safety, Ethics, Group Work Process, etc)¿. Chapter/Appendices 5 will provide implementation notes. Chapter **??** will describe our results. Chapter 6 will be the conclusion.

NOTE: Given that I don’t actually need a full thesis, change from Chapters to Sections? NOTE2: Will treat as a full thesis but if it gets too big, submit a condensed version as the MCTX ”report”.

NOTE3: In final version, put the Approach at the end and/or in appendices before the Implementation notes rather than the start (?)

Chapter 2

Physical Limitations on Precision

This chapter will focus on the physical limitations on precision due to hardware. It will describe the traditional approaches to working with real numbers and the current state of the art.

NOTE: Depending on how much I get into the VHDL stuff, this chapter may seem less relevant to the actual research we are doing and it might be very short. However I think even if I do not actually do any hardware designs, a literature review about precision of numbers would not be complete if it only mentioned software algorithms.

2.1 Real Number Representations

This section will discuss representations of real numbers in hardware, including the trivial: fixed point, IEEE floating points, and anything else I find that is interesting.

2.2 Floating Point Units

This section will give an overview of FPUs, focusing on IEEE.

2.3 Graphical Processing Units

This section will discuss anything relevant we can find on GPUs, but will probably be very short.

Chapter 3

Virtual Representations of Real Numbers

SoftFloat libraries (fixed precision)

boost library for increased (but still fixed) precision.

Algorithms for arbitrary precision? Mathematica uses arbitrary precision numbers. Will they have a reference or report on this anywhere?

Chapter 4

Approach

Repeat the plans as outlined in the project proposals[1, 2][1].

Explain changes from the original proposal.

- No mention of hardware considerations in proposal, but Sam Moore's topic "how can we increase the precision of real number representations" was in the original scope, and can't be answered without considering hardware.

[1] Sam Moore's proposal has not been assessed but was submitted jointly with David Gow's to CSSE

Chapter 5

Implementation Notes

This will be the easiest and also least important chapter to write.

Chapter 6

Conclusion

We will have hopefully done the following:

- Demonstrated precision issues due to traditional (Select one or more of the following!)
  - Physical representation of real numbers
  - Algorithms
  - Document formats
- Demonstrated something better
- Written a less shitty report than this one.

References

[1] Sam Moore. Infinite precision document formats (project proposal). `http://szmoore.net/ipdf/documents/ProjectProposalSam.pdf`, 2014.

[2] David Gow. Infinite-precision document formats (project proposal). `http://davidgow.net/stuff/ProjectProposal.pdf`, 2014.