SEMINAR REPORT
Seminar: Programming Languages in Winter term 2024/2025

Liquid Haskell

Mehran Shahidi, Saba Safarnezhad
Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau, Department of Computer Science

This report gives a brief overview of Liquid Haskell, a tool that extends Haskell with refinement types. Refinement types are types that extends expressiveness of Haskell types systems by providing predicates that can verify invarients of the program. This report explains briefly how SMT solvers leveraged by Liquid Haskell and how to use Liquid Haskell by providing some examples. Finally, we also discuss the limitations of Liquid Haskell and compare it with other tools.

1 Introduction

2 Background

3 Working with LiquidHaskell

Whatever the addressed problem is: now the author can start to explain her/his work. After the explanation, experiences, results or just a discussion of the presented work has to be given in Section 5.

4 Example Application

state the problem

Pick a thing I did in Agda

Or find a problem with

5 Conclusions, Results, Discussion

This section will usually be the last section in a paper. Authors may split results and conclusions into two separate sections.

6 More on LaTeX

There are also some particular features we consider noteworthy.

- (Pseudo-)Code can be typeset by a number of packages, e. g. `listings`. See Listing 1 for an example.
- With the `todonotes` package it is easy to leave little reminders in your document.

Should we explain more?

```
1 def sort[T](list: List[T])(implicit ord: Ordering[T]): List[T] = {
2   list match {
3     case Nil => Nil
4     case x :: xs =>
5       // partition list based on pivot element x
6       val (lo, hi) = xs.partition(ord.lt(_, x))
7       sort(lo) ++ (x :: sort(hi))
8   }
9 }
```

Figure 1: An example algorithm. Do you recognize it? Line 6 is crucial!

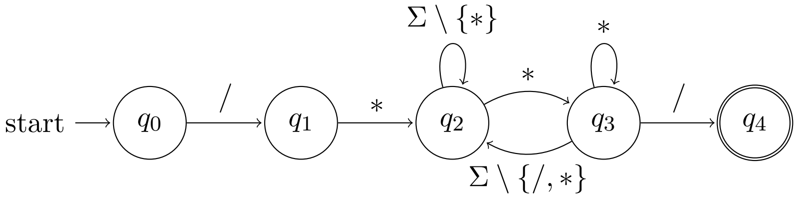

Figure 2: Example for a TikZ image; here using the `automata` library.

- TikZ is a great package for creating all sorts of graphics, see e. g. Figure 2. Find an extensive list of examples (with sources) on texample.net/tikz/examples. Your first graphics in TikZ *will* be time-eaters, but with some practice simple sketches are done in minutes. With a little bit of extra tuning, the same sources produce figures of unsurpassed quality.
- You can use document class `beamer` for creating presentation slides. It introduces some special syntax; best check the documentation and/or search the web.

7 General research tools

Google Scholar & backwards search

Google Scholar can be used to find papers and books of all kinds and often has links to PDF versions that are freely available, e. g., on authors' websites. Moreover, Google Scholar can be used for finding other relevant articles in a field by backward-searching for papers that reference a given paper; see screenshot in Figure **??**. Google Scholar has some basic support for exporting entries to BibTeX; the results often need some manual polishing, though.

dblp is a searchable index of computer science bibliography with BibTex entries for all articles.

Mendeley is a (free) web service for storing references (including PDFs).

The collected references can be exported to BibTeX.

aspell or some other spell checker.

As a general rule, typos that can be found by running a spell checker are unacceptable.

Git or some other version control system (like Subversion)

Very useful for keeping old versions around and highly recommended. It works best on text files, but can also handle binary files. A comprehensive introduction is available as free ebook [3].

Git repositories can easily be cloned, e. g., onto an external drive for backup. That way, you get incremental backups of your work basically for free.

As Git (and all other versioning systems) typically operate in a *line-based* way, it is very advisable to break your text into short lines (≈ 100 character). For documents that are edited collaboratively, it has been good practice to start a new line for each sentence (or even sub-clause)[1].

8 Bibliography

[1] Recall that a newline is treated by LaTeX like an ordinary space. Only empty lines indicate a new paragraph.

References

[1] Adams et al. (1986): *The Seville Statement on Violence. 6th International Colloquium* on *Brain and Aggression*.

[2] L. Barroca (1995): *A style option for rotated objects in LaTeX/*, 2.10 edition.

[3] Scott Chacon (2009): *Pro Git (Expert's Voice in Software Development)*. Apress. Available at `http://www.git-scm.com/book`.

[4] Deutschsprachige Anwendervereinigung TeX e. V. (2014): *Erste Schritte mit TeX, LaTeX und Co.* Available at `http://www.dante.de/tex/TeXAnfaenger.html`.

[5] A. E. Gill (1980): *Some Simple Solutions for Heat-Induced Tropical Circulation. Quart. J. R. Met. Soc.* 106, pp. 447–462.

[6] M. Goossens, F. Mittelbach & A. Samarin (1994): *The LaTeX/ Companion*. Addison-Wesley.

[7] Ranjit Jhala & Niki Vazou (2021): *Refinement Types: A Tutorial. Foundations and Trends® in Programming Languages* 6(3–4), pp. 159–317, doi:10.1561/2500000032. Available at `http://dx.doi.org/10.1561/2500000032`.

[8] David S Johnson (2001): *A Theoretician's Guide to the Experimental Analysis of Algorithms. American Mathematical Society* 220(5–6), pp. 1–36. Available at `http://www2.research.att.com/~dsj/papers/experguide.pdf`.

[9] D. E. Knuth (1984): *The TeX/book*. Addison-Wesley.

[10] L. Lamport (1985): *LaTeX/ – A Document Preparation System*, first edition. Addison-Wesley.

[11] T. Oetiker (1996): *The Not So Short Introduction to LaTeX/*, 2.2 edition.