



Introduction to Floating-Point Analysis and Reproducibility




Ignacio Laguna, Harshitha Menon
Lawrence Livermore National Laboratory

Michael Bentley, Ian Briggs, Pavel Pancheckha, Ganesh Gopalakrishnan
University of Utah

Hui Guo, Cindy Rubio González
University of California at Davis

Michael O. Lam
James Madison University



FPCore

Standard Format for FP Analysis Tools

Pavel Panchekha
University of Utah

The Numeric Design Phase

$$b - \sqrt{b^2 - 1}$$

Herbie

Find accurate
formula

FPTaylor

Verify accuracy
bounds

C/Fortran/...

Tuning, testing,
production

CHALLENGE:

Common format
for numerical tools



The FPBench Project

Benchmarks

For tool authors

118 examples

Browsable online

Formats

Common format

Standard metrics

Reference impl's

Tools

Compilers

Transformers

Search tools



Demo

Compose a FPCore program

Apply Herbie

Transform and export to FPTaylor

Export to C code & compile it



The FPCore Format

Easy to parse **common** format

Support for **loops, branches, standard functions**

Support for complex **mixed-precision** operations

The FPCore Format

Arguments

```
(FPCore (b)
```

```
:pre (< 1e5 b 1e6)
```

Precondition

```
(let ([discriminant (- (* b b) 1)])
```

Let, while, if, ...

```
(- b (sqrt discriminant))))
```

Standard functions



Tools - Exporter

Export to languages (C) & tools (FPTaylor)

```
racket export.rkt in.fpcore out.c
```

Customizable through metadata, flags

```
:precision binary32
```



Tools - Transformer

Common code **transformations** for analysis

```
racket transform.rkt --unroll 3 in.fpcore -
```

Automate inter-tool **communication**

```
tool1 | racket transform.rkt --cse | tool2
```



You Can Help!

HPC authors:

Submit examples

File bugs + test exporters

Looking for tensor kernels

Tool authors:

Accept FPCore input/output

Contribute transformations

Web-accessible tools



Using FPBench

Examples: <http://fpbench.org>

Github: [fpbench/fpbench](https://github.com/fpbench/fpbench)

```
racket export.rkt in.fpcore out.c
```

```
racket transform.rkt --unroll 3 in.fpcore -
```