

CSC2/455 Software Analysis and Improvement

Analyzing FP Computations

Sreepathi Pai

April 15, 2025

URCS

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

So far

Homework 3 asked you to write the concrete semantics for floating point operations. We'll discuss some of the design decisions in this class.

Problem

- We want to analyze code for the presence of exceptional values
- Will this expression ever yield $\pm\text{inf}$ or NaN?
 - can also be stated as “Will this variable ...”

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

How are floating point numbers represented?

- The homework asked you to consider (g, e, s)
 - with no limits on the ranges
- How should we represent exceptional values $\pm \text{inf}$ and NaN?
 - as special encodings in (g, e, s) ?
 - as separate values?
 - what about ± 0 ?
- How should we handle rounding and normalization that is present in FP?

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

Semantics of Addition

- Normal + Normal
- Subnormal + Subnormal
- Normal + Subnormal
- Exceptional + Normal
- Exceptional + Subnormal
- Exceptional + Exceptional

Which of these operations produce exceptional values?

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

Semantics of Multiplication

- Normal * Normal
- Subnormal * Subnormal
- Normal * Subnormal
- Exceptional * Normal
- Exceptional * Subnormal
- Exceptional * Exceptional

Which of these operations produce exceptional values?

Some issues I noted

- Not acceptable
 - Plain English semantics
 - Non-mathematical semantics – e.g., using programming language notation

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

- Floating point multiply and add (FMA)
 - $a * b + c$
 - Common in matrix multiply implementations
- How are exceptional values handled?
- What are the combinations?
- Aside: Tensor cores

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

How to check your semantics

- Some of your semantics are very detailed, how do you know they are correct?
 - Incorrect semantics
 - Missing semantics
- Good news:
 - FP is implemented by hardware
 - FP semantics is formalized in SMT
- How can we use these to check?

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

Finding the dynamic range of floating point array

```
float a[2] = {...}

vmax = a[0];
vmin = a[0];

for(int i = 1 ; i < 2; i++) {
    if(a[i] > vmax) {
        vmax = a[i];
    }

    if(a[i] < vmin) {
        vmin = a[i];
    }
}

float drange = vmax - vmin;
```

Adapted from code suggested by Prof. Ganesh Gopalakrishnan of the University of Utah.

Outline

Introduction

Representation

Addition

Multiplication

FMA

Checking your semantics

A realistic problem

Abstraction

Abstracting the concrete semantics

- Intervals?
- Soundness conditions?
- How should some of the operations in the concrete semantics be implemented on intervals?
 - Addition, Subtraction
 - Shifts
 - Multiplication
 - Division

Implementation

- Input: C files
- Analyses:
 - Abstract Interpretation
 - SMT (possibly hand translated)
 - CBMC
 - KLEE