Contents

Preface . . . xi

List of Algorithms . . . xiii

Introduction . . . xv

Chapter 1. Floating-Point Arithmetic . . . 1

1.1. FP numbers as real numbers . . . 1
1.1.1. FP formats . . . 2
1.1.2. Unit in the last place . . . 3
1.1.3. FP rounding . . . 4
1.1.4. FP operations . . . 6
1.2. Error analysis . . . 7
1.2.1. Absolute and relative errors of a single operation . . . 7
1.2.2. Direct and inverse errors . . . 8
1.3. Exceptional values . . . 9
1.3.1. IEEE-754 levels . . . 10
1.3.2. Binary encoding . . . 11
1.3.3. Rounding and exceptional values . . . 12
1.4. Additional definitions and properties . . . 13
1.4.1. Faithful rounding . . . 13
1.4.2. Double rounding . . . 14
1.4.3. Rounding to odd . . . 15

Chapter 2. The Coq System . . . 17

2.1. A brief overview of the system . . . 17
2.1.1. Propositions as types . . . 18
2.1.2. Dependent types . . . 19

2.1.3. Propositions and types, predicates and sets 20
2.1.4. Inductive types 21
2.1.5. Extraction 24
2.1.6. Computation 24
2.2. Tactics 25
2.2.1. Discharging a goal 26
2.2.2. Applying a theorem 26
2.2.3. Dealing with equalities 28
2.2.4. Handling logical connectives and quantifiers in the conclusion . . . 28
2.2.5. Handling logical connectives and quantifiers in the context 29
2.2.6. Forward reasoning 30
2.2.7. The SSReflect tactic language 31
2.3. Standard library 31
2.3.1. Integers 32
2.3.2. Real numbers 33

Chapter 3. Formalization of Formats and Basic Operators 35

3.1. FP formats and their properties 37
3.1.1. Real numbers and FP numbers 38
3.1.2. Main families of FP formats 39
3.1.2.1. The FLT formats 40
3.1.2.2. The FLX formats 41
3.1.2.3. The FIX formats 41
3.1.2.4. Some other formats 42
3.1.3. Generic formats 43
3.1.3.1. Definition 43
3.1.3.2. Basic properties 44
3.1.3.3. Requirements for having a useful exponent function 45
3.1.3.4. Usual FP formats 47
3.1.4. Main properties 49
3.1.4.1. Unit in the last place 50
3.1.4.2. Predecessor and successor 51
3.2. Rounding operators and their properties 53
3.2.1. Rounding relations 53
3.2.1.1. Directed rounding 54
3.2.1.2. Rounding to nearest 56
3.2.1.3. Rounding to nearest, tie breaking to even 57
3.2.2. Rounding functions 58
3.2.2.1. Definition and basic properties 58
3.2.2.2. Rounding errors 61
3.2.2.3. Double rounding 63
3.3. How to perform basic FP operations 64
3.3.1. Rounding, addition, and multiplication 65

3.3.1.1. Locating real numbers and rounding them 65
3.3.1.2. Effective rounding 69
3.3.1.3. Addition and multiplication 73
3.3.2. Division and square root 74
3.3.2.1. Division 74
3.3.2.2. Square root 76
3.4. IEEE-754 binary formats and operators 77
3.4.1. Formalizing IEEE-754 binary formats 78
3.4.2. Bit-level representation 81
3.4.3. IEEE-754 arithmetic operators 82
3.4.3.1. Exceptional inputs 83
3.4.3.2. Specifications 84
3.4.3.3. Implementation 88

Chapter 4. Automated Methods 91

4.1. Algebraic manipulations 92
4.1.1. Polynomial equalities 92
4.1.2. Linear and polynomial inequalities over the real numbers 94
4.1.3. Linear inequalities over the integers 95
4.2. Interval arithmetic 96
4.2.1. Naive interval arithmetic 97
4.2.1.1. Computing with bounds and outward rounding 98
4.2.1.2. Implementation of interval arithmetic 100
4.2.1.3. Elementary functions 103
4.2.2. Fighting the dependency effect 105
4.2.2.1. Bisection 106
4.2.2.2. Automatic differentiation 107
4.2.2.3. Polynomial approximations using Taylor models 110
4.2.3. Enclosing integrals 113
4.3. Bounds on round-off error 116
4.3.1. The Gappa tool 117
4.3.2. Dependency effect and forward error analysis 121
4.3.2.1. Example: relative error of the product 122
4.3.2.2. Forward error analysis 124
4.3.2.3. Relative error of the sum 125
4.3.3. Variations around enclosures 127
4.3.3.1. Enclosures of relative errors 127
4.3.3.2. Zero and subnormal numbers 129
4.3.4. Discreteness of FP numbers 130
4.3.5. Rounding operators 132
4.3.6. Backward propagation 134
4.3.7. User hints 136

Chapter 5. Error-Free Computations and Applications 139

5.1. Exact addition and EFT for addition 140
5.1.1. Exact subtraction 140
5.1.2. Exact addition when the result is close to zero 141
5.1.2.1. Exact addition when the result is zero 141
5.1.2.2. Exact addition when the result is subnormal 142
5.1.3. Fast2Sum and 2Sum algorithms 142
5.1.3.1. Fast2Sum algorithm 143
5.1.3.2. 2Sum algorithm 143
5.2. EFT for multiplication 144
5.2.1. Splitting of an FP number 145
5.2.1.1. Algorithm and idea of the proof of Veltkamp's splitting 145
5.2.1.2. Proof of Veltkamp's algorithm 146
5.2.1.3. About the tie-breaking rule 150
5.2.2. Dekker's product 151
5.2.2.1. Proof of Dekker's product 151
5.2.2.2. When underflow happens 154
5.3. Remainder of the integer division 155
5.4. Remainders of the FP division and square root 157
5.4.1. Remainder of FP division 157
5.4.2. Remainder of the square root 158
5.5. Taking the square root of the square 158
5.5.1. When the square root of the square is exact 159
5.5.2. When the square root of the square is not exact 160
5.6. Remainders for the fused-multiply-add (FMA) 161
5.6.1. EFT for the FMA 161
5.6.2. Approximate remainder of the FMA 162

Chapter 6. Example Proofs of Advanced Operators 165

6.1. Accurate computation of the area of a triangle 166
6.2. Argument reduction 169
6.2.1. Cody and Waite's argument reduction 170
6.2.2. Bounding the error using Gappa 171
6.2.3. Complete proof for the exponential function 172
6.3. Faithful rounding of Horner evaluation 174
6.3.1. How to guarantee faithfulness? 175
6.3.2. Faithfulness for the last Horner iteration (hypotheses on FP values) 177
6.3.3. Faithfulness for the last Horner iteration (hypotheses on mathematical values) 179
6.4. Integer division computed using FMA 180
6.4.1. Assembly code 180
6.4.2. Coq specification 181

6.4.3. Formal proof using Gappa 182
6.4.4. Specification of `frcpa` 184
6.5. Average of two FP numbers 185
6.5.1. The avg_naive function 187
6.5.2. The avg_half_sub function 187
6.5.3. The avg_sum_half function 189
6.5.4. Sterbenz' accurate algorithm 190
6.5.5. Correctly-rounded algorithm 191
6.6. Orientation of three points 191
6.6.1. Naive FP implementation 193
6.6.2. Homogeneous error bound 196
6.6.3. Writing a robust algorithm 200
6.6.4. Generalization to other predicates 202
6.7. Order-2 discriminant 202
6.7.1. Proof of the ideal algorithm 204
6.7.2. Proof when the test goes wrong 206
6.7.2.1. The execution should have taken the most accurate path . . . 206
6.7.2.2. The execution should have taken the short path 207

Chapter 7. Compilation of FP Programs 209

7.1. Semantics of languages and FP arithmetic 212
7.1.1. Java 213
7.1.2. C 213
7.1.3. Fortran 214
7.2. Verified compilation 214
7.2.1. A verified C compiler: CompCert 215
7.2.2. Formalization of FP arithmetic 216
7.2.3. Parsing and output of FP constants 219
7.2.4. Code generation 220
7.2.5. Code optimization 221
7.2.5.1. Constant propagation 221
7.2.5.2. Algebraic simplifications 222
7.3. Conversions between integers and FP numbers 224
7.3.1. From 32-bit integers to FP numbers 225
7.3.2. From 64-bit integers to FP numbers 226
7.3.3. From FP numbers to integers 230

Chapter 8. Deductive Program Verification 233

8.1. Introduction 233
8.2. Our method and tools for program verification 235
8.2.1. Specifications 235
8.2.2. Deductive verification 237
8.2.3. Our verification process 237

8.3. Examples of annotated programs 239
8.3.1. Exact subtraction 240
8.3.2. Veltkamp's and Dekker's algorithms 240
8.3.3. Accurate computation of the area of a triangle 241
8.3.4. Average computation 245
8.3.4.1. Absolute value 245
8.3.4.2. Accurate Sterbenz' average program 245
8.3.4.3. Correctly-rounded average program 248
8.3.5. Malcolm's algorithm 248
8.4. Robustness against compiler optimizations 251
8.4.1. Extended registers and FMA 252
8.4.2. Reorganization of additions 253
8.4.3. The KB3D example 253
8.4.3.1. Description of the KB3D example 254
8.4.3.2. Sign function 255
8.4.3.3. KB3D – verification with no optimization 256
8.4.3.4. KB3D – architecture-independent verification 257

Chapter 9. Real and Numerical Analysis 259

9.1. Running example: three-point scheme for the 1D wave equation 259
9.2. Advanced formalization of real analysis 262
9.2.1. Total functions 263
9.2.2. Tactics for automating reasoning about differentiability 265
9.2.3. Partial derivatives 267
9.2.4. Parametric integrals 268
9.2.5. Generalized limits 269
9.3. Method error of the 3-point scheme for the 1D wave equation 270
9.3.1. Description of the continuous problem 270
9.3.2. Description of the discretized problem 271
9.3.3. Description of the scheme properties 273
9.3.4. Domination (big O) 275
9.3.5. Differentiability and regularity 276
9.3.6. Consistency 277
9.3.7. Stability 278
9.3.8. Convergence 279
9.4. Round-off error 279
9.4.1. Local round-off errors 280
9.4.2. Convolution of round-off errors 281
9.4.3. Bound on the global round-off error 283
9.5. Program verification 284

Bibliography 289

Index 301