Numerical Algorithms and Digital Representation

Knut Mørken

September 3, 2011

Contents

1 Introduction **1**

1.1 A bit of history . . . 1

1.2 Computers and different types of information . . . 3

1.2.1 Text . . . 4

1.2.2 Sound . . . 4

1.2.3 Images . . . 4

1.2.4 Film . . . 4

1.2.5 Geometric form . . . 5

1.2.6 Laws of nature . . . 5

1.2.7 Virtual worlds . . . 5

1.2.8 Summary . . . 6

1.3 Computation by hand and by computer . . . 7

1.4 Algorithms . . . 8

1.4.1 Statements . . . 10

1.4.2 Variables and assignment . . . 11

1.4.3 For-loops . . . 12

1.4.4 If-tests . . . 13

1.4.5 While-loops . . . 14

1.4.6 Print statement . . . 14

1.5 Doing computations on a computer . . . 15

1.5.1 How can computers be used for calculations? . . . 15

1.5.2 What do you need to know? . . . 16

1.5.3 Different computing environments . . . 17

I Numbers **21**

2 0 and 1 **23**
2.1 Robust communication . . . 23
2.2 Why 0 and 1 in computers? . . . 24
2.3 True and False . . . 25
2.3.1 Logical variables and logical operators . . . 26
2.3.2 Combinations of logical operators . . . 29

3 Numbers and Numeral Systems **31**
3.1 Terminology and Notation . . . 31
3.2 Natural Numbers in Different Numeral Systems . . . 33
3.2.1 Alternative Numeral Systems . . . 33
3.2.2 Conversion to the Base-β Numeral System . . . 36
3.2.3 Conversion between base-2 and base-16 . . . 39
3.3 Representation of Fractional Numbers . . . 42
3.3.1 Rational and Irrational Numbers in Base-β . . . 42
3.3.2 An Algorithm for Converting Fractional Numbers . . . 46
3.3.3 Conversion between binary and hexadecimal . . . 47
3.3.4 Properties of Fractional Numbers in Base-β . . . 49
3.4 Arithmetic in Base β . . . 53
3.4.1 Addition . . . 53
3.4.2 Subtraction . . . 54
3.4.3 Multiplication . . . 54

4 Computers, Numbers, and Text **57**
4.1 Representation of Integers . . . 57
4.1.1 Bits, bytes and numbers . . . 58
4.1.2 Fixed size integers . . . 59
4.1.3 Two's complement . . . 60
4.1.4 Integers in Java . . . 61
4.1.5 Integers in Python . . . 62
4.1.6 Division by zero . . . 63
4.2 Computers and real numbers . . . 63
4.2.1 The challenge of real numbers . . . 63
4.2.2 The normal form of real numbers . . . 64
4.2.3 32-bit floating-point numbers . . . 66
4.2.4 Special bit combinations . . . 67
4.2.5 64-bit floating-point numbers . . . 67
4.2.6 Floating point numbers in Java . . . 68

4.2.7 Floating point numbers in Python . . . 68
4.3 Representation of letters and other characters . . . 69
4.3.1 The ASCII table . . . 70
4.3.2 ISO latin character sets . . . 73
4.3.3 Unicode . . . 73
4.3.4 UTF-8 . . . 75
4.3.5 UTF-16 . . . 77
4.3.6 UTF-32 . . . 78
4.3.7 Text in Java . . . 78
4.3.8 Text in Python . . . 79
4.4 Representation of general information . . . 81
4.4.1 Text . . . 81
4.4.2 Numbers . . . 82
4.4.3 General information . . . 83
4.4.4 Computer programs . . . 83
4.5 A fundamental principle of computing . . . 83

5 Computer arithmetic and round-off errors 87
5.1 Integer arithmetic and errors . . . 88
5.2 Floating-point arithmetic and round-off error . . . 88
5.2.1 Truncation and rounding . . . 89
5.2.2 A simplified model for computer arithmetic . . . 90
5.2.3 An algorithm for floating-point addition . . . 91
5.2.4 Observations on round-off errors in addition/subtraction . . . 93
5.2.5 Multiplication and division of floating-point numbers . . . 95
5.2.6 The IEEE standard for floating-point arithmetic . . . 97
5.3 Measuring the error . . . 99
5.3.1 Absolute error . . . 99
5.3.2 Relative error . . . 99
5.3.3 Properties of the relative error . . . 100
5.3.4 Errors in floating-point representation . . . 102
5.4 Rewriting formulas to avoid rounding errors . . . 104

II Sequences of Numbers 109

6 Numerical Simulation of Difference Equations 111
6.1 Why equations? . . . 111
6.2 Difference equations defined . . . 112

6.2.1 Initial conditions . . . 115
6.2.2 Linear difference equations . . . 116
6.2.3 Solving difference equations . . . 116
6.3 Simulating difference equations . . . 118
6.4 Review of the theory for linear equations . . . 121
6.4.1 First-order homogenous equations . . . 121
6.4.2 Second-order homogenous equations . . . 122
6.4.3 Linear homogenous equations of general order . . . 126
6.4.4 Inhomogenous equations . . . 127
6.5 Simulation of difference equations and round-off errors . . . 130
6.5.1 Explanation of example 6.27 . . . 133
6.5.2 Round-off errors for linear equations of general order . . . 136
6.6 Summary . . . 139

7 Lossless Compression **141**
7.1 Introduction . . . 142
7.1.1 Run-length coding . . . 143
7.2 Huffman coding . . . 144
7.2.1 Binary trees . . . 146
7.2.2 Huffman trees . . . 146
7.2.3 The Huffman algorithm . . . 148
7.2.4 Properties of Huffman trees . . . 152
7.3 Probabilities and information entropy . . . 154
7.3.1 Probabilities rather than frequencies . . . 154
7.3.2 Information entropy . . . 155
7.4 Arithmetic coding . . . 157
7.4.1 Arithmetic coding basics . . . 157
7.4.2 An algorithm for arithmetic coding . . . 159
7.4.3 Properties of arithmetic coding . . . 163
7.4.4 A decoding algorithm . . . 165
7.4.5 Arithmetic coding in practice . . . 167
7.5 Lempel-Ziv-Welch algorithm . . . 169
7.6 Lossless compression programs . . . 169
7.6.1 Compress . . . 170
7.6.2 gzip . . . 170

8 Digital Sound **171**
8.1 Sound . . . 171
8.1.1 Loudness: Sound pressure and decibels . . . 172
8.1.2 The pitch of a sound . . . 175

8.1.3 Any function is a sum of sin and cos . . . 177
8.2 Digital sound . . . 180
8.2.1 Sampling . . . 180
8.2.2 Limitations of digital audio: The sampling theorem . . . 181
8.2.3 Reconstructing the original signal . . . 182
8.3 Simple operations on digital sound . . . 184
8.4 More advanced sound processing . . . 189
8.4.1 The Discrete Cosine Transform . . . 189
8.5 Lossy compression of digital sound . . . 190
8.6 Psycho-acoustic models . . . 193
8.7 Digital audio formats . . . 194
8.7.1 Audio sampling — PCM . . . 194
8.7.2 Lossless formats . . . 195
8.7.3 Lossy formats . . . 196

9 Polynomial Interpolation 199
9.1 The Taylor polynomial with remainder . . . 200
9.1.1 The Taylor polynomial . . . 200
9.1.2 The remainder . . . 202
9.2 Interpolation and the Newton form . . . 207
9.2.1 The interpolation problem . . . 208
9.2.2 The Newton form of the interpolating polynomial . . . 210
9.3 Divided differences . . . 214
9.4 Computing with the Newton form . . . 219
9.5 Interpolation error . . . 221
9.6 Summary . . . 223

10 Zeros of Functions 225
10.1 The need for numerical root finding . . . 226
10.1.1 Analysing difference equations . . . 226
10.1.2 Labelling plots . . . 227
10.2 The Bisection method . . . 228
10.2.1 The intermediate value theorem . . . 228
10.2.2 Derivation of the Bisection method . . . 229
10.2.3 Error analysis . . . 232
10.2.4 Revised algorithm . . . 234
10.3 The Secant method . . . 236
10.3.1 Basic idea . . . 237
10.3.2 Testing for convergence . . . 239
10.3.3 Revised algorithm . . . 240

10.3.4 Convergence and convergence order of the Secant method . 241
10.4 Newton's method 243
10.4.1 Basic idea 243
10.4.2 Algorithm 244
10.4.3 Convergence and convergence order 245
10.5 Summary 249

11 Numerical Differentiation **251**
11.1 Newton's difference quotient 252
11.1.1 The basic idea 253
11.1.2 The truncation error 254
11.1.3 The round-off error 256
11.1.4 Optimal choice of h 262
11.2 Summary of the general strategy 264
11.3 A symmetric version of Newton's quotient 265
11.3.1 Derivation of the method 265
11.3.2 The error 267
11.3.3 Optimal choice of h 269
11.4 A four-point differentiation method 271
11.4.1 Derivation of the method 271
11.4.2 The error 272
11.5 Numerical approximation of the second derivative 274
11.5.1 Derivation of the method 274
11.5.2 The error 275
11.5.3 Optimal choice of h 277

12 Numerical Integration **279**
12.1 General background on integration 280
12.2 The midpoint rule for numerical integration 283
12.2.1 A detailed algorithm 284
12.2.2 The error 286
12.2.3 Estimating the step length 290
12.3 The trapezoidal rule 291
12.3.1 The error 293
12.4 Simpson's rule 296
12.4.1 Derivation of Simpson's rule 296
12.4.2 Composite Simpson's rule 298
12.4.3 The error 300
12.5 Summary 302

13 Estimating Errors **303**
13.1 Different types of errors . . . 303
13.2 Some useful mathematical tools . . . 303
13.3 Truncation errors . . . 303
13.4 Round-off errors . . . 303

14 Numerical Solution of Differential Equations **305**
14.1 What are differential equations? . . . 305
14.1.1 An example from physics . . . 305
14.1.2 General use of differential equations . . . 307
14.1.3 Different types of differential equations . . . 308
14.2 First order differential equations . . . 308
14.2.1 Initial conditions . . . 309
14.2.2 A geometric interpretation of first order differential equations 310
14.2.3 Conditions that guarantee existence of one solution . . . 312
14.2.4 What is a numerical solution of a differential equation? . . . 313
14.3 Euler's method . . . 315
14.3.1 Basic idea and algorithm . . . 315
14.3.2 Geometric interpretation . . . 317
14.4 Error analysis for Euler's method . . . 319
14.4.1 Round-off error . . . 320
14.4.2 Local and global error . . . 320
14.4.3 Untangling the local errors . . . 322
14.5 Differentiating the differential equation . . . 325
14.6 Taylor methods . . . 328
14.6.1 The quadratic Taylor method . . . 328
14.6.2 Taylor methods of higher degree . . . 331
14.6.3 Error in Taylor methods . . . 332
14.7 Midpoint Euler and other Runge-Kutta methods . . . 334
14.7.1 Euler's midpoint method . . . 334
14.7.2 The error . . . 336
14.7.3 Runge-Kutta methods . . . 337
14.8 Systems of differential equations . . . 340
14.8.1 Vector notation and existence of solution . . . 340
14.8.2 Numerical methods for systems of first order equations . . . 343
14.8.3 Higher order equations as systems of first order equations . 344
14.9 Final comments . . . 347

III Functions of two variables **351**

15 Functions of two variables **353**
15.1 Basics . . . 353
15.1.1 Basic definitions . . . 353
15.1.2 Differentiation . . . 355
15.1.3 Vector functions of several variables . . . 357
15.2 Numerical differentiation . . . 358

16 Digital images and image formats **363**
16.1 What is an image? . . . 363
16.1.1 Light . . . 363
16.1.2 Digital output media . . . 364
16.1.3 Digital input media . . . 365
16.1.4 Definition of digital image . . . 365
16.1.5 Images as surfaces . . . 367
16.2 Operations on images . . . 369
16.2.1 Normalising the intensities . . . 369
16.2.2 Extracting the different colours . . . 370
16.2.3 Converting from colour to grey-level . . . 371
16.2.4 Computing the negative image . . . 372
16.2.5 Increasing the contrast . . . 372
16.2.6 Smoothing an image . . . 374
16.2.7 Detecting edges . . . 376
16.2.8 Comparing the first derivatives . . . 379
16.2.9 Second-order derivatives . . . 380
16.3 Image formats . . . 381
16.3.1 Raster graphics and vector graphics . . . 381
16.3.2 Vector graphics formats . . . 383
16.3.3 Raster graphics formats . . . 384

Answers to some exercises **387**

Part I

Numbers