Algorithms in the Real World: Lecture Notes (Fall 1997)

Guy Blelloch

April 23, 1998

Abstract

This document contains the lecture notes taken by the students in the course "Algorithms in the Real World" taught at UC Berkeley during the Fall semester, 1997. The class covered the following set of ten topics: Compression, Cryptography, Linear Programming, Integer Programming, Triangulation, N-body simulation, VLSI Physical Design, Pattern Matching in Biology, Indexing, and Clustering. Between 2 and 3 lectures were dedicated to each topic. For all topics we looked both at algorithms and at case studies in which the problems are used in real-world applications. The class was a graduate class and assumed that the students came in with a working knowledge of the material covered in a standard algorithms textbook, such as Cormen, Leiserson and Rivest's "Introduction to Algorithms". A goal of the class was to have the students gain an appreciation of how interesting algorithms and data-structures are used in many real-world applications.

The notes contained in this document are based on what was covered in the lectures and are not meant to be complete, and although the scribe takers corrected many of the mistakes in my lectures, I expect many others got through. I must also apologize that the notes contain very few references to the original work on which they rely. Instead I have included at the end of the document a list of primary texts on the topics from which one can find further information. Also, there is a web page associated with the class that has many more online pointers

`http://www.cs.berkeley.edu/~guyb/algs.html`.

I would like to thank all the students who took the notes that appear in this document. They did an excellent job and in many cases went far beyond the call of duty in polishing them and generating figures.

Contents

List of Algorithms **12**

Compression 1 (Ben Zhao) **14**

1 Introduction to Compression **14**
1.1 Information Theory . . . 14
1.2 The English Language . . . 15

2 Basic Compression Algorithms **17**
2.1 Prefix Codes . . . 17
2.2 Huffman Codes . . . 17
2.3 Shannon-Fano Coding . . . 18
2.4 Arithmetic Coding . . . 18
2.5 Deriving Probabilities . . . 19

3 Lempel-Ziv Overview **19**
3.1 LZ78 and LZW . . . 20
3.2 LZ77 (Sliding Windows) . . . 22

Compression 2 (Gabriel Moy) **23**

1 Lossless Compression: review and expansion **23**
1.1 Can we compress all strings? . . . 23
1.2 Huffman Codes Revisited . . . 23
1.3 Arithmetic Coding Revisited . . . 25
1.4 Conditional Entropy and Markov Chains . . . 27
1.5 The JBIG Standard . . . 28
1.6 Tries and use in LZW . . . 28
1.7 Other Lossless Compression Techniques . . . 28

2 Lossy Compression **29**
2.1 Scalar Quantization . . . 29
2.2 Vector Quantization . . . 30
2.3 Transform Coding . . . 30
2.4 Choosing a Transform . . . 31

Compression 3 (Ben Liblit) **33**

1 Case Studies **33**
1.1 JPEG 33
1.2 MPEG 37

2 Other Lossy Transform Codes **44**
2.1 Wavelet Compression 44
2.2 Fractal Compression 47
2.3 Model-Based Compression 50

Cryptography 1 (Tzu-Yi Chen) **51**

1 Definitions and Primitives **51**
1.1 Definitions 51
1.2 Primitives 53

2 Protocols **53**
2.1 Digital Signatures 54
2.2 Key Exchange 55
2.3 Authentication 55
2.4 Some Other Protocols 56

3 Symmetric (private-key) Algorithms **56**
3.1 Symmetric Ciphers (Block Ciphers) 56
3.2 DES (Data Encryption Standard) 57

Cryptography 2 (David Oppenheimer) **59**

1 Block ciphers **59**
1.1 Feistel networks 59
1.2 Security and Block Cipher Design Goals 61
1.3 DES Security 61
1.4 Differential Cryptanalysis 64
1.5 Linear Cryptanalysis 64
1.6 IDEA 65
1.7 Block Cipher Encryption Speeds 67

2 Stream ciphers **67**
2.1 Stream ciphers vs. Block cipher 67
2.2 RC4 68

3 Public-Key Algorithms **69**
3.1 Merkle-Hellman Knapsack Algorithm 69
3.2 RSA 70

Cryptography 3 (Marat Boshernitsan) 72

1 Some group theory 72
1.1 Groups: Basics . . . 72
1.2 Integers modulo n . . . 72
1.3 Generators (primitive elements) . . . 73
1.4 Discrete logarithms . . . 74

2 Public Key Algorithms (continued) 74
2.1 RSA algorithm (continued) . . . 74
2.2 Factoring . . . 75
2.3 ElGamal Algorithm . . . 75

3 Probabilistic Encryption 75
3.1 Generation Random Bits . . . 76
3.2 Blum-Goldwasser . . . 76

4 Quantum Cryptography 77

5 Kerberos (A case study) 77
5.1 How Kerberos Works . . . 78
5.2 Tickets and Authenticators . . . 79
5.3 Kerberos messages . . . 79

Linear Programming 1 (Richard C. Davis) 80

1 Introduction 80
1.1 Some Optimization Problems . . . 80
1.2 Applications of Linear Programming . . . 81
1.3 Overview of Algorithms . . . 82
1.4 Linear Programming Formulation . . . 83
1.5 Using Linear Programming for Maximum-Flow . . . 84

2 Geometric Interpretation 85

3 Simplex Method 86
3.1 Tableau Method . . . 88
3.2 Finding an Initial Corner Point . . . 92
3.3 Another View of the Tableau . . . 93
3.4 Improvements to Simplex . . . 93

Linear Programming 2 (Steven Czerwinski) 95

1 Duality **95**
1.1 Duality: General Case 95
1.2 Example 96
1.3 Duality Theorem 97

2 Ellipsoid Algorithm **97**
2.1 Description of algorithm 98
2.2 Reduction from general case 99

3 Interior Point Methods **99**
3.1 Centering 100
3.2 Optimizing techniques 102
3.3 Affine scaling method 104
3.4 Potential reduction 106
3.5 Central trajectory 107

4 Algorithm summary **107**

5 General Observation **108**

Integer Programming 1 (Andrew Begel) 109

1 Introduction and History **109**

2 Applications **110**
2.1 Knapsack Problem 110
2.2 Traveling Salesman Problem 110
2.3 Other Constraints Expressible with Integer Programming 111
2.4 Piecewise Linear Functions 111

3 Algorithms **112**
3.1 Linear Programming Approximations 112
3.2 Search Techniques 112
3.3 Implicit Enumeration (0-1 Programs) 114

Integer Programming 2 (Stephen Chenney) 116

1 Algorithms (continued) **116**
1.1 Problem Specific Aspects 116
1.2 Cutting Plane Algorithm 117
1.3 Recent Developments 118

2 **Airline Crew Scheduling (A Case Study)** **118**
2.1 The Problem 118
2.2 Set Covering and Partitioning 119
2.3 Generating Crew Pairings 120
2.4 TRIP 122
2.5 New System 123

Triangulation 1 (Aaron Brown) 128

1 **Introduction to Triangulation** **129**

2 **Basic Geometrical Operations** **129**
2.1 Degeneracies and Numerical Precision 129
2.2 The Line-side Test 130
2.3 The In-circle Test 131

3 **Convex Hulls** **132**
3.1 Dual: Convex Polytopes 132
3.2 Algorithms for Finding Convex Hulls 133

4 **Delaunay Triangulations and Voronoi Diagrams** **139**
4.1 Voronoi Diagrams 139
4.2 Delaunay Triangulations 139
4.3 Properties of Delaunay Triangulations 141
4.4 Algorithms for Delaunay Triangulation 141

Triangulation 2 (Franklin Cho) 144

1 **Divide-and-Conquer Delaunay Triangulation** **144**

2 **Incremental Delaunay Triangulation Algorithm** **148**

3 **Mesh Generation** **149**
3.1 Ruppert's Algorithm 150

Triangulation 3 (Michael Downes) 154

1 **Triangulation in Finite Element Methods (Mark Adams)** **154**
1.1 FEM Intro 154
1.2 Approximating the Weak Form with Subspaces 155
1.3 Picking Good Subspaces 156
1.4 Linear System Solvers 156
1.5 Multigrid 159

1.6 Unstructured Multigrid Algorithms 159
1.7 Parallel Multigrid Solver for FE 160

2 Distance Metrics for Voronoi Diagrams and Delaunay Triangulations 161

3 Surface Modeling Using Triangulation 162
3.1 Terrain Modeling 162

N-Body Simulations 1 (Tajh Taylor) 165

1 Delaunay Triangulation - more applications 165

2 N-Body Introduction 166
2.1 Basic Assumptions 166
2.2 Applications 167
2.3 History of Algorithms 168

3 Mesh-based Algorithms 170

4 Tree-based Algorithms 172
4.1 Barnes-Hut Algorithm 172
4.2 Fast Multipole Method 176

N-body Simulations 2 (Steve Gribble) 183

1 Fast Multipole Method (Review) 183
1.1 The Four Ideas 183
1.2 Interaction Lists 184

2 Callahan and Kosaraju's Algorithm 185
2.1 Well Separated Pair Decompositions 185
2.2 Definitions 186
2.3 Algorithm: Tree Decomposition 187
2.4 Algorithm: Well-Separated Realization 189
2.5 Bounding the size of the realization 190

3 N-body Summary 194

VLSI Physical Design 1 (Rich Vuduc) 195

1 Some notes regarding the homework assignment 195
1.1 Representing a Delaunay triangulation 195
1.2 Incremental Delaunay 198

1.3 Ruppert Meshing 199
1.4 Surface Approximation 199

2 Intro to VLSI Physical Design Automation **201**
2.1 Overview 202
2.2 The role of algorithms 207
2.3 Evolution 208

3 Summary **208**

VLSI Physical Design 2 (Mehul Shah) 210

1 Introduction to VLSI Routing **210**

2 Global Routing Algorithms **211**

3 Two-Terminal Nets: Shortest Paths **213**
3.1 Dijkstra's Algorithm 213
3.2 Maze Routing 215

4 Multi-Terminal Nets: Steiner Trees **216**
4.1 Steiner Trees 216
4.2 Optimal S-RST for separable MSTs 217

VLSI 3 & Pattern Matching 1 (Noah Treuhaft) 220

1 Integer Programming For Global Routing **220**
1.1 Concurrent Layout 220
1.2 Hierarchical Layout 221

2 Detailed Routing (Channel Routing) **221**
2.1 Introduction 221
2.2 Left Edge Algorithm (LEA) 222
2.3 Greedy 223

3 Sequence Matching And Its Use In Computational Biology **224**
3.1 DNA 224
3.2 Proteins 224
3.3 Human Genome Project 224
3.4 Sequence Alignment 225

Pattern Matching 2 (Felix Wu) 226

1 Longest Common Subsequence **226**

2 Global Sequence Alignment **227**
2.1 Cost Functions for Protein Matching 228

3 Sequence Alignment Algorithms **228**
3.1 Recursive Algorithm 228
3.2 Memoizing 229
3.3 Dynamic Programming 229
3.4 Space Efficiency 230

4 Gap Models **231**

5 Local Alignment **232**

6 Multiple Alignment **233**

7 Biological Applications **233**

Indexing and Searching 1 (Helen J. Wang) 234

1 Pattern Matching (continued) **234**
1.1 Ukkonen's Algorithm 234

2 Introduction to Indexing and Searching **236**

3 Inverted File Indexing **239**
3.1 Inverted File Compression 239
3.2 Representing and Accessing Lexicons 242

Indexing and Searching 2 (Ben Horowitz) 246

1 Evaluating information retrieval effectiveness **246**

2 Signature files **247**
2.1 Generating signatures 247
2.2 Searching on Signatures 248
2.3 Query logic on signature files 249
2.4 Choosing signature width 249
2.5 An example: TREC 250

3 Vector space models **250**
3.1 Selecting weights 251
3.2 Similarity measures 251
3.3 A simple example 252

3.4 Implementation of cosine measures using inverted lists 253
3.5 Relevance feedback 254
3.6 Clustering 254

4 Latent semantic indexing (LSI) 254
4.1 Singular value decomposition (SVD) 255
4.2 Using SVD for LSI 256
4.3 An example of LSI 257
4.4 Applications of LSI 257
4.5 Performance of LSI on TREC 260

Evolutionary Trees and Web Indexing (Amar Chaudhary) 262

1 Evolutionary Trees (Andris Ambainis) 262
1.1 Parsimony 263
1.2 Maximum-Likelihood 264
1.3 Distance methods 266

2 Finding authoritative web pages using link structure (David Wagner) 267
2.1 A naive approach using adjacency 268
2.2 An algorithm for finding communities within topics 268

Clustering (Josh MacDonald) 272

1 Principle Component Analysis 272

2 Introduction to Clustering 274
2.1 Applications 274
2.2 Similarity and Distance Measures 275

3 Clustering Algorithms 275
3.1 Bottom-Up Hierarchical Algorithms 276
3.2 Top-Down Hierarchical Algorithms 277
3.3 Optimization Algorithms 278
3.4 Density Search Algorithms 279
3.5 Summary 279

Guest lecture Eric Brewer on HotBot (Carleton Miyamoto) 280

1 Comment on Assignment (Guy) 280

2 Goals of HotBot 282

3 Internal Architecture **282**
3.1 Semantics 282
3.2 Hardware 282
3.3 Data Base Model 283
3.4 Score 284
3.5 Filters 284

4 Optimizations **285**
4.1 Caching 285
4.2 Compression 285
4.3 Data Base 285
4.4 Parallelism 285
4.5 Misc. 287

5 Crawling **287**

6 Summary **288**

Other Material 289

1 References by Topic **289**
1.1 Compression 289
1.2 Cryptography 289
1.3 Linear and Integer Programming 290
1.4 Triangulation 290
1.5 N-body Simulation 291
1.6 VLSI Physical Design 291
1.7 Pattern Matching in Biology 292
1.8 Indexing and Searching 292
1.9 Clustering 292

2 Assignments **292**
Assignment 1 (Compression) 293
Assignment 2 (Cryptography) 295
Assignment 3 (Integer and Linear Programming) 297
Assignment 4 (Triangulation and N-body) 299
Assignment 5 (VLSI and Pattern Matching) 302

List Of Algorithms

The following algorithms are covered to some extent in these notes, ranging from a short overview to a full lecture.

Compression

Huffman Coding 17
Arithmetic Coding 18
LZ78 and LZW 20
LZ77 (Sliding Windows) 22
JPEG Coding 33
MPEG Coding 37
Wavelet Compression 44

Cryptography

DES 57
RC4 68
Merkle-Hellman Knapsack Algorithm 69
RSA 70
ElGamal 75
Blum-Goldwasser 75

Linear Programming

Simplex 86
Ellipsoid Algorithm: Khachian 97
Interior Point: Affine Scaling 104
Interior Point: Potential Reduction 106
Interior Point: Central Trajectory 107

Integer Programming

Branch and Bound Enumeration 112
Implicit Enumeration 114
Cutting Plane 117
Crew Scheduling: Anbil, Tanga and Johnson 118

Triangulation

Convex Hull: Giftwrapping 134
Convex Hull: Graham Scan 134
Convex Hull: Mergehull 136
Convex Hull: Quickhull 137
Delaunay Triangulation: Blelloch, Miller, and Talmor 144
Delaunay Triangulation: Incremental 148
Meshing: Ruppert's Algorithm 150
Surface Modeling: Garland and Heckbert 163

N-body Simulation

Particle Mesh (using FFT . . . 170
Barnes-Hut . . . 172
Fast Multipole Method: Greengard and Rokhlyn . . . 176
Callahan and Kosaraju . . . 185

VLSI Physical Design

Shortest paths: Dijkstra's Algorithm . . . 213
Shortest paths: Hadlock's Algorithm . . . 215
Optimal Rectilinear Steiner Trees . . . 217
Routing with Integer Programming . . . 220
Detailed Routing: Left Edge Algorithm . . . 222
Detailed Routing: Greedy Algorithm . . . 223

Pattern Matching in Biology

Sequence Alignment: Memoizing . . . 229
Sequence Alignment: Dynamic Programming . . . 229
Sequence Alignment: Hirschberg . . . 230
Alignment with Gaps: Waterman-Smith-Beyer . . . 231
Alignment with Gaps: Gotoh . . . 231
Local Alignment: Smith-Waterman . . . 232
Sequence Alignment: Ukkonen . . . 234

Indexing and Searching

Inverted File Compression . . . 239
Searching Signature Files . . . 248
Vector Space Searching . . . 253
Latent Semantic Indexing (LSI) . . . 254
Authoritative Pages: Kleinberg . . . 268

Clustering

Bottom-up Hierarchical Clustering . . . 276
Splinter Group Clustering . . . 277
Clustering with Linear Programming . . . 278
Density Search Clustering . . . 279