

Index

- Λ , 120
- ϵ -fixed arc, 151
- ϵ -optimal, 147, 197
- active node, 59, 67, 162
- admissible arc, 58, 162, 182, 221
- Ahlsweide-Winter theorem, 274
- amortized running time, 21
- ancestor tree, 115
- arc
 - ϵ -fixed, 151
 - admissible, 58, 162, 182, 221
 - fixed, 151
 - holdover, 171
 - nontree, 168
 - saturated, 29
 - special, 123
 - tree, 168
- assignment problem, 180
- augmenting path, 30, 197
 - generalized, 191
 - most improving, 50, 71
- augmenting path algorithm, 32
 - most improving, 47–50
 - shortest, 53–55
- Awerbuch-Leighton algorithm, 240–249
- B (matrix), 258
- B (scalar), 188
- baseball elimination problem, 36–41, 67, 70
- basic cycle, 169
- Bellman-Ford algorithm, 6–10, 205
- bicycle, 199
- binary search, 45
- bisection search, 45
- blocking flow, 116–117
 - algorithm, 119–120, 128–130
 - and dynamic trees, 128–130
 - definition, 116–117
 - in series-parallel graphs, 128
 - in unit capacity graph, 128
- bucket data structure, 64
- C (matrix), 258
- C (scalar), 139
- cactus tree, 115
- cancel-and-tighten, 182
- canceling
 - cycle, 135–136, 221
 - GAP, 191–192
- canonical labeling, 193–194
- capacity constraint, 24, 133, 190
- capacity scaling algorithm
 - for maximum flow problem, 51–53
 - for minimum-cost circulation problem, 154–160
- Cauchy-Schwarz inequality, 244, 268
- characteristic flow, 191
- Chernoff bounds, 289
- circulation, 134–135
 - definition, 132–133
- concentration of measure, 271, 289
- concurrent flow problem, *see* maximum concurrent flow problem
- condition number
 - tree, 278
- conductance, 254
- congestion, 237, 251
- contraction, 90–91
- cost scaling, 185

- cut
 - global minimum, 80
 - s - t , capacity, 26
 - s - t , definition, 26–27
 - sparsifier, 271–273
 - X - t , 82
- cut condition, 225–226
 - two commodities, 227–230
- cut level, 84
- cut-equivalent tree, 100
- cycle, 1
 - basic, 169
 - canceling, 135–136, 221
 - correction, 277
 - flow-absorbing, 191
 - flow-generating, 191
 - minimum-mean, 146
 - negative-cost, 6, 135
 - simple, 1
 - unit-gain, 191
- DAG, 19
- data structure
 - bucket, 64
 - dynamic trees, 67, 128–130, 285
 - Fibonacci heap, 5, 21
 - heap, 4–6
 - pairing heap, 21
 - queue, 9
- deficit, 155
- demand, 133, 254
- density of a graph, 42
- Dijkstra's algorithm, 2–6, 160
- Dilworth's theorem, 77
- Dinitz's algorithm, 116–120
- directed acyclic graph, 19
- discharge, 64
- distance labels, 2, 53
 - X -valid, 83
 - valid, 57
- distance level, 84, 120
 - empty, 84
- dynamic trees, 67, 128–130, 285
- effective resistance, 259–263
- electrical flow
 - definition, 254–255
 - energy, 255
 - optimality conditions, 254–255
- electrical network, 254–257
- energy, 255, 263–265
- error scaling, 218–219
- excess, 57, 155, 190
- excess scaling push-relabel algorithm, 72–73
- Fibonacci heap, 5, 21, 90, 110
- FIFO push-relabel algorithm, 72
- fixed arc, 151
- Fleischer's algorithm, 249–250
- flow
 - characteristic, 191
 - generalized, 190
 - interpretation, 215
 - over time, 170
 - proper, 190
 - s - t , definition, 23–25
 - temporally repeated, 171
 - value, 24, 190
- flow conservation constraint, 24, 133, 190
- flow decomposition, 47, 142, 199
- flow-absorbing cycle, 191
- flow-equivalent tree, 101, 112
 - algorithm, 112–113
- flow-generating cycle, 191
- Foster's theorem, 262
- Fujishige's algorithm, 110
- gain, 188
- gain scaling, 214–215
- GAP, *see* generalized augmenting path
- GAP canceling, 191–192
- gap relabeling, 68–69, 76
- Garg–Könemann algorithm, 236–240, 249–250
- generalized
 - circulation, 222, 292
 - flow, 190
 - proper flow, 190
 - pseudoflow, 189
- generalized augmenting path, 191
- generalized maximum flow problem
 - definition, 188–190
 - GAP canceling algorithm, 196–197
 - optimality condition, 190–198
- generalized minimum-cost circulation
 - problem, 222, 292
- global minimum cut, 80
 - α -approximate, 112
 - in directed graphs, 80
 - in undirected graphs, 80
 - number, 112
- global relabeling, 69

Goldberg-Rao algorithm, 122–128
 Golden Snitch, 70
 Gomory-Hu tree, 292
 algorithm, 104–108
 definition, 100–101
 for symmetric submodular functions, 108–110
 Gusfield's algorithm, 107

 Hall's theorem, 77
 Hao-Orlin algorithm, 82–89
 heap, 4–6
 Fibonacci, 5, 21
 pairing, 21
 highest label push-relabel algorithm, 65
 Hoffman circulation theorem, 71, 133, 139
 holdover arc, 171

 image segmentation problem, 70
 integrality property
 maximum flow problem, 32
 minimum-cost circulation problem, 139
 interpreting flow, 215

 König-Egerváry theorem, 77
 Kirchoff Current Law, 254
 Kirchoff Potential Law, 255

 labeling, 192–193
 canonical, 193–194
 Laplacian, 257–259
 solver, 277–285
 lossy graph, 210
 low-stretch spanning tree, 289

 m , 2
 MA ordering, 89–90, 110
 for global minimum cut, 90–93
 for maximum flow problem, 110
 for symmetric submodular functions, 111–112
 maximum adjacency ordering, *see* MA ordering
 maximum concurrent flow problem, 224–225
 maximum density subgraph problem, 41–46
 weighted, 71
 maximum flow problem
 augmenting path algorithm, 32
 definition, 23–24
 integrality property, 32
 optimality conditions, 31

 over time, 170–176
 parametric, 76
 maximum flow-minimum cut theorem, 29
 maximum multicommodity flow problem, 224–225
 minimum s - t cut problem, 29
 minimum s -cut problem, 80–81
 minimum X - t cut problem, 82
 minimum cost-to-time ratio cycle problem, 20
 minimum-cost circulation problem, 292–293
 cycle canceling algorithm, 139
 definition, 132–133
 generalized, 222, 292
 integrality property, 139
 optimality conditions, 137
 minimum-cost flow problem, 133–134, 176
 quickest, 176
 minimum-cost perfect matching problem, 177–180
 minimum-mean cycle, 146
 algorithm, 20
 minimum-mean cycle canceling algorithm, 146–154
 most improving
 augmenting path, 47, 50, 71
 negative-cost cycle, 140, 176
 multicommodity flow problem, 292
 cut condition, 225–226
 definition, 224–225
 optimality condition, 226–227
 two commodities, 227–230
 multiplicative weights algorithm, 230–236
 for maximum flow, 266–270
 for multicommodity flow, 236–240
 for packing problems, 233–236
 with costs, 249

 n , 2
 negative-cost cycle, 6, 135
 detection, 11–19
 most improving, 176
 negative-cost GAP, 202
 detection, 205–209
 network reliability problem, 81–82
 network simplex algorithm, 167–170
 node
 active, 59, 67, 162
 contraction, 90–91
 demand, 133
 labeling, 192–193
 potential, 136, 254

- price, 136
- relabeling, 192–193
- sink, 23, 188
- source, 23
- supply, 133
- nonsaturating push, 166
- nontree arc, 168
- $\tilde{O}$, 113, 259
- Ocean's Eleven, 266
- Ohm's Law, 254
- oracle for convex set, 233
 - width, 233
- packing problem, 233
- pairing heap, 21
- parallel composition, 128
- parametric maximum flow, 76
- parent
 - graph, 11
 - pointer, 3
- path, 1
 - augmenting, 30
 - simple, 1
- perfect matching, 177
- polynomial time, 32
 - pseudo-, 32
 - strongly, 33
- potential, 136, 254
 - tree-defined, 256, 277
- potential function, 62–63
- preflow, 56–57
 - X-preflow, 83
 - convert to flow, 67, 76
- price, 136
- price refinement, 183
- proper flow, 190
- pseudoflow, 78, 154
 - definition, 155
 - generalized, 189
- pseudopolynomial time, 32
- push, 58
 - nonsaturating, 62, 166
 - saturating, 62, 166
 - stack, 73
- push-relabel algorithm, 56–69, 291
 - excess scaling, 72–73
 - FIFO, 72
 - highest label, 65
 - wave scaling, 73–76
- queue, 9
- quickest minimum-cost flow problem, 176
- quickest transshipment problem, 176
- quidditch elimination problem, 70
- random contraction algorithm, 93–99
- Rayleigh monotonicity principle, 285
- recursive random contraction algorithm, 97–99
- reduced cost, 136
- relabel, 58
 - set, 182
- relabeling, 192–193
- RELAX, 186
- residual graph, 29–30, 135, 190
- resistance, 254
 - effective, 259–263
- s - t cut
 - capacity, 26
 - definition, 26–27
- s - t flow
 - definition, 23–25
- saturated arc, 29
- saturating push, 166
- scaling
 - capacity, 51–53, 154–160
 - cost, 185
 - error, 218–219
 - gain, 214–215
- scaling parameter, 51
- scan, 8–9
- series composition, 128
- series-parallel graphs, 128
- set relabeling, 182
- shortest augmenting path algorithm, 53–55
- simple
 - cycle, 1
 - path, 1
- sink, 23, 188
- skew symmetry, 24, 134, 189
- source, 1, 23
- sparsifier
 - cut, 271–273
 - spectral, 272–276
- special arc, 123
- spectral sparsifier, 272–276
- stack push, 73
- strongly polynomial time, 33, 292
- submodular function, 102, 110–112
 - symmetric, 102, 110–111
 - Gomory-Hu tree, 108–110
 - minimization algorithm, 111–112

314

successive approximation, 160–167
supply, 133, 254
symmetric submodular function, *see*
 submodular function, symmetric

temporally repeated flow, 171
time-expanded network, 170
transit time, 170
transportation problem, 184
tree
 ancestor, 115
 arc, 168
 cactus, 115
 condition number, 278
 cut-equivalent, 100

Index

 flow-equivalent, 101, 112
 low-stretch, 289
 tree-defined potential, 256, 277
 Truemper's algorithm, 209–214

 U (scalar), 32, 49, 139
 unit capacity graph, 113, 120
 blocking flow, 128
 unit-gain cycle, 191

 value, 24, 190

 Wallacher's algorithm, 140–146
 wave scaling push-relabel algorithm, 73–76
 width, 233