CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

Index

abstract data types, 19, 20, 147, 149
abstract data types and algorithm design, 11
academic vs. industrial research, 323
action systems, 234
adaptability, 150
AI, 33, 89, 116, 125, 144, 155, 317
AI and experiment, 44
AI and mathematics, 37, 41, 43, 86
AI and statistics, 42
AI and vision, 41
AI as source of ideas, 38, 46
AI, fragmentation of, 38
AI, general purpose view, 39
AI, invisibility of, 40
AI, nature of, 36
AI, quality of, 36
AI, the future, 45
algebraic specification, 19
algorithm, 1, 2
algorithm development, 77
algorithm efficiency, 7
algorithms engineering, 4
algorithms engineering and algorithms science, 14
algorithms in NLP, 89
Alvey Programme, 34
analytical models, 301
architecture, limits of, 145
archiving, 31, 121, 138
asynchronous system, 192
ATM networks, 289

Bachmann diagram, 113
benchmarking, 328
Birch–Swinnerton-Dyer conjectures, 67
Boolean circuit, 18
bottom-up process, 6
bounds on algorithms, 3, 79
Buchberger's algorithm, 75, 76

calculus of design, 176
calculus of processes, 276
causal independence, 276
CD-ROM, 25
chess, 39
Chomskyan transformational rules, 96
classification, 42
client–server model, 115
cognitive science, 37
combinatorial algorithms, 12, 13
combinatorial explosion, 13
communicating system, 269
communication in parallel programming, 60
complete induction, 7
complexity, 79, 80, 199, 201, 249
complexity, in safety-critical systems, 199
complexity, measures of, 199
complexity classes, 80
complexity of algorithms, 2, 18
complexity of knowledge, 112
compliance, 226, 227
computability, 141, 249
computational geometry, 13
computational linguistics, 44, 88
computational morphology, 95
computational number theory, 15, 16, 68, 79
computational psycholinguistics, 89
computational skeletons, 58
computer algebra and calculus, 73
computer architecture, 221
computer networks, 269
computer science and algebraic geometry, 75
computer science and geometry, 84
computer science and mathematics, 158, 159, 165
computer science and observation, 163

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

computer science as engineering, 249, 302, 307
computer science as science, 160, 249
computer science, influence on mathematics, 74
computer science, nature of, 159
computer science, theoretical, 161
computer science; engineering or science, 255
computer vision, 47
computers in algebra, 67
computers in combinatorics, 68
computers, effect on pure mathematics, 67
computing system architecture, 230
concrete domains, 274
concurrency, 276
concurrency and time, 190
concurrent computing, 282
concurrent object-oriented programming, 282
concurrent process, 256
concurrent programming languages, 279
configuration skeletons, 56
constraint, integrity, 118
constraints, 118, 120
coordination languages, 53
copyright and electronic publishing, 31
CSCW, 305
customizing, 29

data longevity, 121, 138
data maintenance, 138, 150
data transmission, 284
data, accessibility, 133
data, compatibility of, 137
database, 127, 129, 132, 135, 325
database system architecture, 110
databases and knowledge bases, 111
databases and networks, 115, 121
deadlock, 163, 166, 170, 174, 185
dense domain, 192
dependability, 200
dependability evaluation, 201
dependability and human error, 208
dependability and probability, 201, 203, 207, 208, 210
dependability evaluation, 199, 203, 206, 211
dependability modelling, 206, 211
dependability, ultra-high, 206
design by objectives, 228
determinism, 192, 195
dialogues, 137
digitizable specification, 194
discrete processes , 277
discrete time, 192
distributed computing, 145, 293
distributed database, 123
distributed systems, 279
Divide and Conquer, 6, 7, 15, 17
document maintenance, 31
domain, 262, 266
domain, algebraic, 266
domains as information systems, 266
duration calculus, 193
dynamic priority allocation, 194
dynamic programming, 5–7, 42

effective methods, 70, 71, 74
effectivity and computer science, 71
effectivity in calculus, 72
efficiency of an algorithm, 3
efficient algorithms, 2, 73
efficient methods, 74
electronic publishing, 25, 28, 30, 66
Elliptic Curve Factoring Algorithm, 15
entity–relationship diagram, 113
entity-relation view, 142
error propagation, 11
ethernet, 286
Euclid's algorithm, 2, 70
event structure, 276
execution time efficiency, 17
execution time function, 3
execution time of an algorithm, 2, 3, 16
experiment in computer science, 211
expert systems, 34
exploratory programming, 37
external choice, 174

factorization, 79
failure, 166, 168
fast Fourier transform, 14, 15, 80
fault-tolerance, 196, 204–206
fault-transformation, 196
feature structures, 100
Fibonacci numbers, 4
Fifth Generation Project, 34
fixed point, 175
formal methods, 50, 147, 155, 200, 209, 210, 212, 254, 255
formal methods in practice, 51
Fortran 90, 59
Four-Colour Theorem, 67
fractals, 41
frames , 117
functional languages, 65
functional programming, 49–51, 270, 324
functional programming, advantages, 52, 325
functional programming, disadvantages, 330

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

functional programming, non-strict, 324, 327, 328
functional programming, strict, 327
fuzzy logic, 40, 117

Galois connection, 177
game theory, 269
garbage collection, 332
general purpose lexicon, 97
general purpose processors, 293
general purpose programming languages, 184
general purpose reasoning, 34
general purpose view, 34
general purpose vs. specific, 30
general-purpose, all embracing vs. user-specific, 28
general-purpose, purely-functional, 325
genetic algorithm, 8, 9, 10
global optimisation, 121
Gröbner base, 75, 80
grammar, 96, 98
grammar, probabilistic, 98, 101
greedy algorithms, 9
greedy heuristic, 8

Hamiltonian circuit, 18
hardware design and mathematics, 181
hardware reliability, 198
Hensel's lemma, 14, 15, 81
High Performance Fortran, 55
homonymy, 104
Horn clause, 44, 116
human genome project, 11, 110, 112, 125, 143
human–computer interaction, 30, 46, 148, 297, 303–305
human–computer interaction in the computing curriculum, 296
human–computer interaction, benefits of, 309
human–computer interface, 132
hybrid systems, 193
hypertext, 24, 27, 29, 122

image processing, 41
incompleteness, 248
inductive design, 7, 8
information, 129
information and computers, 127
information and secrecy, 134, 140
information as an agent of change, 143
information flow, 247, 265, 267, 269, 282
information flow and domains, 266
information repository, 135
information representation, 28
information vs. raw data, 128
information, access to, 130
information, accessing, 135
information, ambiguity of, 140
information, changing view of, 149
information, incompatible, 133
information, incorrect, 132
information, meaning of, 156
information, science of, 153
information, verifying, 133
inheritance, 115
integer overflow, 6
integrity constraint, 118
interaction of commerce and academic research, 26
interactive computing, 282, 297
interactive computing and software engineering, 315
interactive computing, benefits of, 298
interactive systems, 295
Intermedia, 28
Internet, 146
intuitionism, 71
intuitionism vs. non-effectivity, 71
iteration, 5, 6

Karmarkar's algorithm, 15
Khachian's ellipsoid method, 15
knowledge broker, 122
knowledge compilation, 118
knowledge re-use, 111, 113, 115, 116, 120, 121, 125
knowledge re-use and programming languages, 120
nowledge representation, 42, 96
knowledge servers, 116, 123
Knowledge Sharing Effort, 122
knowledge storage, 118
knowledge transformation, 111, 122

λ-calculus, 272, 279, 281
λ-notation, 263, 271
language acquisition, 90
language and mathematics, 165, 257
language for computer science, 165
language generation, 90
language generation, probabilistic work, 108
language understanding, 90
language, formal deifintion of, 256
lattice reduction algorithm, 69, 74
lazy learning, 100
learning, 36
least slack ordering, 194
lexicon, 97
lexicon, general purpose, 97
linguistic computation, 89
Liouville's Principle, 73

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

Local area networks, 286
logic, 248, 250
logic and computer science, 71
logic and programming, 250
logic and programming languages, 266
logic programming, 42, 50
logic and real-time computing, 193

machine learning, 42, 47
machine translation, 89, 96, 104
Markov chain, 9, 12
mathematics and computer science, 277
mathematics and language, 257
mathematics and programming features, 261
mathematics and programming languages, 263
mathematics and software engineering, 200
matroid, 9
Maximum Segment Sum problem, 7
mental vitalism, 36, 40
Mertens' conjecture, 69
metric temporal logic, 193
mobility, 279
model-checking, 194
models of parallel programming, 64
monads, 329
monotonicity, 263, 267
morphology, 105
multi-precise arithmetic, 11
multimedia, 25

natural language processing; see also computational linguistics, 88
NLP and databases, 95
NLP and discourse, 108
NLP and the lexicon, 95
NLP, algorithms, 89
NLP, ambiguity in, 103
NLP, applied, 89
NLP, context problem in, 94
NLP, difficulties in building system, 90
NLP, probabilistic, 98
NLP, probabilistic approach, 105, 106
NLP, probabilistic methods in, 91
NLP, probabilistic work, 108
NLP, probabilistic work in, 97, 105
NLP, simulated annealing, 104
NLP, symbolic, 98
NLP, symbolic and probabilistic approaches, 103
NLP, symbolic approach, 106
NLP, symbolic methods in, 91
NLP, unifying symbolic and probabilistic approaches, 98
negmajority, 174
networks, 131
neural nets, 42, 44
Newton–Raphson iteration, 14
non-determinism, 20, 174, 178, 180, 181, 201, 275
non-determinism and parallelism, 180
non-monotonic logic, 43
non-well-founded set, 43, 276
notation, 165
NP, 13, 17, 18, 80, 195
numerical algorithms, 11

object classes, 113
object-oriented database, 113, 115, 143
observations and environment, 162
on-line documents, 23
open system, 150
operations research, 189
optimisation, 8, 9, 42, 188, 189
ordering of polynomials, 77

π-calculus, 281
P and NP problems, 17, 18
P, NP, 80
parallel computation, 46, 49, 51–53, 269
parallel Fortran, 60
parallel optimisation, 54
pattern matching, 11
persistence, 113, 141
Petri net, 181, 275, 276
pipelining, 272
polynomial factorization, 72, 80
polysemy , 104
portability, 53
portability of parallel programs, 54
powerdomain, 265
Prüfer encoding, 12
prefixing, 170
principle of data independence, 115, 120
principle of data independence and constraints, 120
Priority Ceiling Protocol, 195
probabilistic grammars, 105, 108
process algebra, 182, 191, 233, 234, 277, 278
process algebra, timed, 193
program evaluation, 272
program logics, 266
programming and time, 190
proof and computer science, 84
proof in mathematics, 70
prototype, 301, 322

query language, 325
query optimisation, 119
query processing, 135

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

randomisation, 9
raw data, 128
re-use, 200, 237
re-usability and parallelism, 54
reactive model, 274
reactive systems, 147
real-time computing, 148, 188, 199, 286
real-time computing and parallel programs, 190
real-time computing and sequential programs, 190
real-time systems, 220, 221, 230
recognition, 41
recovery program, 196
recurrence, 16
recursion, 4, 5, 42
relational database, 135, 142
reliability and environment, 213
reliability and security, 214
reliability growth, 204, 209, 211
reliability measures, 204
reliability modelling., 215
reliability, hardware , 198
reliability, software, 198, 205
representation theory, 19
requirements, 221, 222, 300
requirements and architecture, 231
requirements and design, 222
research and industry, 154
reverse engineering, 251, 252, 254
Riemann hypothesis, 69
risk, 219
robustness, 108
RSA encryption scheme, 15
RSA public-key cryptographic scheme, 68
RSA scheme, 79
rule bases, 116

safety-critical systems, 199, 201, 202, 205, 206, 209, 217, 224, 230, 232, 238
scalability, 111
scheduling, 194, 195
scientific simulation, 135
searching algorithms, 10
security, stochastic models of, 215
sequential algorithm, 268
sequential function, 268
sequential programming language, 172
sequential vs. non-sequential computation, 272
simple induction, 7
simulated annealing, 8, 9, 10
simulated annealing and NLP, 104
software dependability, 299
software design, 27
software design and mathematics, 181, 254
software development, 51, 53, 222, 223
software engineering and mathematics, 200
software engineering, stochastic methods in, 204
software failure, 163
software maintenance, 30, 251, 254
software production, 238
software re-use, 51
software reliability, 198, 213
software testing, 178, 212
software tools, 223, 306
specification, 164, 176
specification of real-time programs, 195
speech processing, 47, 89
stable function, 268
stochastic geometry, 41
stochastic issues, 213
stochastic modelling in evaluation, 214
stream model, 272, 273
strong AI, 36
strong synchrony, 191
structural complexity theory, 17
structured information, 127
structured operational semantics, 181
structured parallel programming, 49, 52, 55
symbolic algebra, 12, 19, 40, 91
synchrony hypothesis, 191
systems engineering, 220

tagging, 102, 108
tagging algorithms, 102
term rewriting, 183
termination, 164
theorem proving, 34, 44, 47, 84, 86, 189
theoretical computer science, 248
timed process algebras, 193
tools for development, 223
top-down process, 6, 225
trace, 163, 170
Travelling Salesman Problem, 17
treebank, 101
Turing machine, 250
types, 264

ultra-high reliability, 215
uncertain knowledge, 117
uncertain reasoning, 43
universal language, impossibility of, 257
universities and industry, 243
user feedback, 30
uzzy matching, 112

validation, 221

© Cambridge University Press www.cambridge.org

CAMBRIDGE

Cambridge University Press
978-0-521-46085-9 - Computing Tomorrow: Future Research Directions in Computer Science
Edited by Ian Wand and Robin Milner
Index
More information

verification, 221, 231
vision, 41
vocabulary, 107
von Neumann model, 53

weak inverse, 177
World Wide Web, 146

Zipf's law, 93, 105, 106

© Cambridge University Press www.cambridge.org