

# Index

- ⊕ symbol, controlled qubit in quantum circuit notation, 53
- ⊗ Kronecker product operator, 16
- ★ operator for Kronecker product, 11, 16
- @ operator for matrix multiplication, 11
- 5-qubit error correction code, 291
- 7-qubit Steane code, 291
- 9-qubit Shor error correction code, 285, 289
- Addition
  - Constants, 179
  - Increment operator, 236
  - Quantum arithmetic, 172–177
  - Quantum gates, 172–174
    - qc data structure, 134
  - Testing quantum arithmetic, 178
- Adjoint gates
  - adjoint() function for gates, 26
  - qc data structure, 130
  - Conjugate synonym, 6
- Adjoint operators
  - adjoint() function for operators, 85
- ALAP Scheduling, 307
- Algorithm references, 277
- Amdahl's law, 286
- Amplitude amplification, 227–230
- Amplitude damping, 283
- Amplitude estimation, 230–234
  - Quantum counting, 230–234
- Ancilla qubits (ancillae)
  - About, 66
  - Ancilla registers, 124
    - Code to create and initialize, 125
  - Compiler optimization and, 304
  - Entanglement, 66, 86
  - Error correction trick, 286
  - Multi-controlled gates, 66
  - Quantum computation, 66
  - Quipper programming language, 300
  - Silq programming language, 302
- AND logic gates, 92
- Ansatz, 245
- Arithmetic via quantum gates
  - Decrement circuit, 236
  - Full adder, 89–91
    - Code, 90
    - Constants, 179
    - qc data structure, 134
    - Quantum arithmetic, 172–177
  - Increment circuit, 236
  - Multiplication, 177
  - Powers, 208
  - Subtraction, 177
  - Testing quantum arithmetic, 178
- Arute, Frank, 150
- at (@) operator for matrix multiplication, 11
- Basis states of qubits, 13
  - Constructing a qubit, 15
  - Density matrices
    - Diagonal elements, 24
  - Density matrix diagonal elements, 23
  - Measurement, 73
  - Orthonormal set of basis vectors, 13, 15
    - Superposition as orthonormal basis, 44
  - Projection operators extracting amplitude, 42
  - State as superposition, 17
  - Superposition via Hadamard gates, 44
    - Hadamard basis, 44
- Bell measurement, 98, 99
- Bell states, 63
  - Code, 63
  - Measurement example, 77
  - Quipper programming language, 301
  - Tracing out qubits, 71
- Bell, John S., 60, 63
- Benchmarking
  - Benchmark gaming, 150
  - Cross entropy benchmarking, 150
  - Gate fast application, 139
  - Gate faster application in C++, 143
  - Quantum random circuits, 150
  - Quantum versus classical computers, 150–152
  - Sparse representation, 147

- Bernstein–Vazirani algorithm
  - Oracle form
    - Compiler optimization, 308
- Bernstein–Vazirani algorithm, 105–108
  - Oracle form, 117
  - Phase kick rotation gates, 162
- Beyond Classical, 149
  - About algorithm types, 160
  - Computational complexity theory, 122, 149, 159
  - Google Sycamore processor, 150
    - Benchmarking, 150
    - Benchmarking quantum versus classical computers, 150–152
  - Quantum random circuits, 150
    - Simulation design, 152
    - Simulation evaluation, 158
    - Simulation implementation, 154
    - Simulation metric, 157
    - Simulation run estimation, 155–158
- Binary fractions, 163
- Binary interpretation, 18
- Birthday paradox, 191
- Bit conversion, 24
- Bit index notation for states, 136
- Bit iteration, 25
- Bit order
  - Binary interpretation, 18
  - Qubit order, 18
  - States, 17
  - Two tensored states, 18
- Bit-flip errors
  - Bit-flip channel, 281
  - Bit-flip phase-flip channel, 281
  - Combined phase/bit-flip error, 279
  - Dissipation-induced error, 279
  - Error correction, 286–288
    - Shor’s 9-qubit code, 289
- Bits to binary fractions, 163
- Bits-decimal conversion functions, 24
- bits2frac() for binary fractions, 163
- bits2val() for binary to decimal, 24
- bitstring() function, 22
- Bloch sphere, 55–60
  - About, 35, 55, 242
  - Expectation values, 59, 74, 244
  - Minus sign as global phase, 59
  - Qubit described by, 55–56
    - Computing coordinates for given state, 58
  - Quirk online simulator, 171
  - Relative phase, 35
  - Rotation operators, 35
  - Single-qubit states only, 59
  - Solovay–Kitaev algorithm, 267
  - Two degrees of freedom for superdense coding, 102
  - Universal gates, 266
- Bloch, Felix, 55
- Boolean formulas with quantum gates, 92
- Born rule, 72
  - About projective measurement, 72
- BPP (complexity), 149, 159
- BQP (complexity), 149, 159
- Bra(c)kets, 3
- Brackets, 3
- Branching, *see* Controlled gates
- Bras
  - Dirac notation, 3
  - Inner products, 3
    - Bra-Ket notation, 3
  - Tensor products, 5
- C++
  - Accelerated gate application, 139–145
    - Execution speed, 139, 145
  - Extending Python with, 140
  - Q language C++ class library, 300
    - Transpiler code generation flag, 315
  - Sparse representation, 145
    - Benchmarking, 147
- “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?” (Einstein, Podolsky, and Rosen), 60
- Cartesian coordinates for Bloch sphere, 58
- Channels in information theory, 281
  - Bit-flip channel, 281
  - Bit-flip phase-flip channel, 281
  - Depolarization channel, 282
  - Phase-flip channel, 281
- Circuits
  - About function calls and returns, 298
    - Scaffold programming language, 298
    - Silq programming language, 302
  - Compiler optimization and, 304
  - Decrement circuit, 236
  - Entangler circuits, 61
  - Grover’s algorithm implemented, 224–227
  - Increment circuit, 236
  - Intermediate representation capabilities, 314
  - Libraries of compiler optimization patterns, 309
  - Logic circuits, 91
    - Fan-out in QCL, 296
    - Gan-out circuits, 92
  - Phase kick circuit, 161
  - Quantum circuit data structure, 126
    - Constructor, 127
    - Gates, 129–133
    - Gates applied, 128
    - Measurements, 131
    - Multi-Controlled gates, 132
    - Quantum registers, 127
    - Qubits added, 128
    - Swap and controlled swap gates, 131

- Quantum random circuits, 150
  - Simulation design, 152
  - Simulation evaluation, 158
  - Simulation implementation, 154
  - Simulation metric, 157
  - Simulation run estimation, 155–158
- Qubits
  - Ordering of qubits, 18
  - qc data structure, 128
  - Quantum circuit notation, 52
  - Shor's 9-qubit error correction, 289
- Cirq commercial system (Google), 303
- Simulators, 321
- Transpilation
  - Code generation flag, 315
  - Dumper function, 319
- Classical arithmetic, *see* Arithmetic via quantum gates
- Classical computers versus quantum, 150–152
  - Summit simulating quantum random circuits, 158
- Clifford gates, 124
- Closed quantum systems, 280
- CNOT, *see* Controlled-Not gate
- CNOT0, *see* Controlled-Not-by-0 gates
- Coin toss operator, 235
- Column vectors
  - Inner products, 3
- Kets
  - Dirac notation, 2
  - Hermitian conjugate of, 2
- Qubits and states as, 2
- Combined phase/bit-flip error, 279
- Commercial systems, 303
- Compiler design challenges, 293, 309
  - About transpilation, 311
  - Intermediate representation, 311
  - Staged compilation, 311
- Compiler optimization, 303
  - About, 303
  - Classical for classical constructs, 304
  - Constant propagation, 304
  - Gate approximation, 310
  - Gate elimination, 305
  - Gate fusion, 306
  - Gate parallelization, 307
  - Gate scheduling, 306
  - High-performance pattern libraries, 309
  - Inlining, 304
  - Logical to physical mapping, 304, 309
    - Resource for information, 310
  - Loop unrolling, 304
  - Peephole optimization, 307
    - Libraries of compiler optimization patterns, 309
    - Relaxed peephole optimization, 308
  - Physical gate decomposition, 310
  - Unentangled qubits, 306
- Complex numbers
  - 2D plane, 2
    - Polar coordinates, 55–56
  - About, 1
  - Conjugates, 1
  - Exponentiation, 2
  - Modulus, 2
  - Norm, 1
  - Python, 2
  - Qubits as column vectors of, 2
  - States as column vectors of, 2
  - Tensor comparisons to values, 12
  - tensor\_type() abstraction, 10
- Complexity classes
  - BPP, 149, 159
  - BQP, 149, 159
  - NP, 149
  - NP-complete, 149
  - NP-hard, 149
  - P, 149
- Complexity of simulation, 151, 155
- Composite kets inner products, 5
- Computation reversed, 66
- Computational complexity theory, 122, 149, 159
- Conditional execution, *see* Controlled gates
- Congruency between numbers, 189
- Conjugate Rotations, 274
- Conjugates
  - Adjoint synonymous with conjugate, 6
  - Complex numbers, 1
  - Denotation not explicit, 3
  - Hermitian conjugate matrix, 6
  - Operator adjoint() function, 26
- Conjugation
  - conjugate() with complex numbers, 2
  - Involutivity, 3
- Constants in quantum addition, 179
- Control register, 124
- Controlled gates
  - About, 46
  - About QCL programming language, 297
  - Controlled rotation gates additive, 161
  - Controlled-Controlled gates, 48, 85
    - Toffoli gates, 49, 85
  - Controlled-Controlled-Not gates, 49
    - Sets of universal gates, 49
  - Controlled-Not gates, 46–49
  - Controlled-Not-by-0 gates, 49
  - Controlled-U1 gate for quantum arithmetic, 172
  - Controlled-Z gates, 81
    - Multi-Controlled-Z gates, 82
  - Function of, 46–49

- Controlled gates (Cont.)
  - Multi-Controlled gates, 86
    - Ancilla qubits, 86
    - Multi-Controlled-Z gates, 82
    - qc data structure, 132
  - Multi-controlled gates
    - Ancilla qubits, 66
  - Nonadjacent controller and controlled qubits, 47
  - Notation for gates involved, 46
  - qc data structure
    - Fast application of gates, 137
    - Faster application with C++, 141
    - Multi-Controlled gates, 132
  - Quantum circuit notation
    - Controlled-X gates, 53
    - Controlled-Not-by-0 gates built, 53
  - Scalability, 48
  - Swap gates, 50, 88
    - Compiler optimization, 308
    - Controlled-Controlled Swap gates, 51
    - qc data structure, 131
    - Quantum circuit notation, 53
- Controlled modular multiplication, 207
- Controlled-Controlled gates, 85
  - Toffoli gates, 49, 85
    - Logic circuits from, 92
- Controlled-Controlled-Not gates (CCX-gate), 49
- Controlled-Not gates (CNOT; CX)
  - Compiler optimization, 308
  - Constructor function, 49
  - Controlled-Z with Hadamard gates, 82
  - Entangler circuits, 62
  - Function, 46–49
  - GHZ states, 64
  - Inverted Controlled-Not gate, 80
  - Logic circuits from, 92
  - Quantum registers for result storage, 67
  - Swap gate action, 51
- Controlled-Not-by-0 gates (CNOT0), 49
  - Quantum circuit notation, 53
- Controlled-U gates under compiler optimization, 308
- Controlled-Z gates, 81
  - Controlled-Not gates via, 82
  - Multi-Controlled-Z gates, 82
- Copenhagen interpretation of quantum mechanics, 61
- Counters
  - Decrement operator, 236
  - Increment operator, 236
- cQASM, 295
- Cross entropy benchmarking (XEB), 150
- Cut on graph, 255
- CX, *see* Controlled-Not gate
- Data registers, 124
  - Code to create and initialize, 125
- Data structure, *see* Quantum circuit (qc) data structure
- Debugging
  - Operator matrices for, 81
  - qc data structure abstraction, 126
  - Reduced density operator, 68
  - Tensors compared to values, 12
- Decoherence times of technologies, 278
- Decoherence-induced phase shift error, 279
- Decrement circuit, 236
- Density matrices
  - About, 9, 68
  - Bloch sphere coordinate computation, 58
  - Cartesian coordinates, 58
  - Outer product of state with itself, 23, 68
  - Partial trace derivation, 68
    - Code, 69
    - Tracing out other qubits, 69
  - Probabilities of measuring a basis state, 24
  - Quantum computing theory as, 24, 68
  - Trace of, 24, 70
- Depolarization, 282
- Depolarization channel, 282
  - Depolarization definition, 282
- Deutsch algorithm, 108–117
- Deutsch–Jozsa algorithm, 118–121
- Diagonal matrices
  - Tensor products, 5
- Diffusion Operator, 220
- Dirac notation
  - Bras, 3
  - Kets, 2
  - Qubits
    - 0-state and 1-state, 13
    - Two tensored states, 18
- Discrete phase gates, 39, 164
- Dissipation-induced error, 279
- Dot products, *see* Inner products
- Dual vectors for a ket, 3
- Dumper function for all relevant info, 21
  - Transpilation, 315
- EGCD, *see* Extended Euclidean algorithm
- Eigenstates
  - About, 7
  - Compiler optimization of gates, 306
- Eigenvalues, 7
  - About, 7
  - Compiler optimization of gates, 306
  - Hamiltonians, 240, 248
  - Hermitian and Pauli matrices, 35
  - Phase estimation, 180
  - Trace of a matrix, 8
  - variational quantum eigensolver, 248

- Eigenvectors
  - About, 7
  - Hamiltonians in Schrödinger equation, 240
  - Phase estimation, 180
- Einstein, Albert
  - Hidden state, 60
  - Spooky action at a distance, 60
- Electron decoherence time
  - Electron spin, 278
  - Gallium arsenide, 278
  - Gold, 278
- Entanglement, 60–65
  - About, 60
  - Analysis by Scaffold, 299
  - Ancilla qubits, 66, 86
  - Bell states, 63
    - Code, 63
    - Tracing out qubits, 71
  - Code
    - Bell states, 63
    - Entangler circuit, 62
    - GHZ states, 64
  - Compiler optimization and, 306
  - Copenhagen interpretation, 61
  - Entangler circuits, 61
    - Code, 62
  - GHZ states, 64
    - Code, 64
    - Error correction trick, 286
  - Maximal entanglement, 64
  - Mixed states, 68
    - Depolarization, 282
  - Product states, 61
  - Quantum teleportation, 97–102
  - Superdense coding, 102–105
  - Swapping, 102
  - Teleportation, 97–102
  - Tracing out qubits, 71
- Entangler circuits, 61
- Environmental challenges of quantum computing, 278–284
  - Closed versus open quantum systems, 280
- Equal superposition of adjacent qubits, 45
- Erasure of information resulting in heat dissipation, 66
- Error correction
  - About, 278, 284
  - Bit-flip errors, 286–288
    - Shor's 9-qubit code, 289
  - Channels, 281
  - Compiler optimization and, 304
  - Error correction code memory, 284
  - Error syndrome, 286
  - Phase-flip errors, 288
    - Shor's 9-qubit code, 289
  - Quantum computing challenges, 285
  - Quantum noise, 278–284
  - Repetition code, 284
    - Majority voting, 285
    - No-cloning theorem, 285
    - Quantum repetition code, 285
  - Resources for information, 291
  - Shor's 9-qubit error correction code, 285, 289
- Error correction code memory (ECC), 284
- Error injection to model quantum noise, 282
  - Checking bit-flip error correction, 287
  - Gates as quantum noise source, 283
- Error syndrome, 286
- Euler's formula
  - Complex exponentiation, 2
  - Phase gate derivation, 38, 165
- Euler's identity, 37
- Expectation values, 74
  - Bloch sphere, 59
  - Variational quantum eigensolver, 244
- Exponentiation
  - Complex numbers, 2
  - Operators, 36
- Extended Euclidean algorithm, 202
- Factorization, 190
- Fan-out circuits, 92
  - QCL programming language, 296
- Fast Gate Application, 134
- Feynman, R., xi
- Fidelity, 269
- Flexible phase gates
  - Discrete phase gates, 39
  - U1( $\lambda$ ) gates, 40
- Fourier transform, *see* Quantum Fourier transform (QFT)
- Fractions
  - Binary fractions, 163
  - $\pi$  fractions transpilation output, 316
- Fredkin gates, 51
- Full adder, 89–91
  - Code, 90
    - qc data structure, 134
  - Quantum arithmetic, 172–177
- Fused gates, 306
- Gallium arsenide (GaAs) electron decoherence time, 278
- Gate equivalences
  - About, 79
  - Compiler optimization, 306, 309
  - Controlled gate equivalencies listed, 87
    - Code to validate, 88
  - Controlled-phase symmetry, 82
  - Controlled-Z gates, 81
    - Controlled-Not gates via, 82
  - Inverted Controlled-Not, 80
  - Multi-Controlled gates, 86

- Gate equivalences (Cont.)
  - Negate Y-gate, 83
  - Pauli matrices, 83
  - Rotation axis changed, 84
  - Squaring root of gate equals gate, 80
- Gates
  - About operators as gates, 25
  - About ordering of gate applications, 54
  - About Scaffold built-ins, 298
    - Classical-To-Quantum-Circuit tool, 298
  - Adjoint gates, 130
  - Application, 27–29
    - apply() function, 32, 128, 142
    - Density matrices, 68
    - Fast application, 134–139
    - Fast application generalized, 137–139
    - Faster application with C++, 139–145
    - Fastest Benchmarked, 147
    - Fastest with sparse representation, 145–147
    - Multiple operators in sequence, 31
    - Multiple qubits, 29–31
    - Noise reduction via compiler optimization, 304
    - Norm preserving, 6, 25
    - Notation for qubit index applied to, 31
    - Padding operators, 31, 48
    - Projection operators extracting subspace, 73
    - Quantum computation, 66
    - to state  $\psi$  at qubit index, 32
  - Compiler optimization
    - Gate approximation, 310
    - Gate elimination, 305
    - Gate fusion, 306
    - Gate parallelization, 307
    - Gate scheduling, 306
    - Logical to physical mapping, 304, 309
    - Logical to physical mapping resource for information, 310
    - Noise reduction, 304
    - Physical gate decomposition, 310
    - Unentangled qubits, 306
  - Controlled gates, *see* Controlled gates
  - Equivalences, *see* Gate equivalences
  - Flexible phase gates
    - Discrete phase gates, 39
    - Phase shift or kick gate, 40
  - Hadamard gates, 44–45
  - Identity gates, 26, 33
    - Applied to multiple qubits, 30
  - Multi-qubit gates
    - Controlled gates, 46–51
    - Hadamard gates, 44–45
    - Single-qubit constructors for, 33
  - Outer product representation of operator, 43
  - Phase gates, 38
    - Discrete phase gates, 39
    - Phase inversion operator, 219
    - Phase shift or kick gates, 40
    - Phases via various gates, 40
    - Square root of as T-gate, 41
  - Projection operators, 42
  - qc data structure, 129
    - Gates applied, 128
    - libq implementation, 324–326
  - Quantum circuit notation, 53
    - About ordering of gate applications, 54
    - Applying operator, 52
  - Quantum noise source, 283
  - Qubit on Bloch sphere, 56–58
  - Rk gates, 39
  - RotationZ-gates versus phase gates, 38
  - Sets of universal gates, 49
  - Single-qubit gates, 33–45
  - Solovay–Kitaev theorem, 266
  - T-gates
    - Phases via, 40, 165
    - Rotation axis changed, 84
    - Square root of phase gates, 41
    - Universal gates, 266
  - U1(lambda) gates, 40
  - V-gates
    - Square root of X-gates, 85
  - V-gates as square roots of X-gates, 41
  - X-gates, 26, 34
  - Y-gates, 34
    - Negate Y-gate, 83
    - Square root of, 41
  - Yroot gates, 41
- GCD, *see* Greatest common divisor
- GHZ states, 64
  - Error correction trick, 286
- Global phase, 59
  - Bloch sphere, 59
  - Phase invariance, 59
  - Relative phase versus, 60
  - Rotation axis changed, 84
- “Going Beyond Bell’s Theorem” (Greenberger, Horne, and Zeilinger), 64
- Gold (Au) electron decoherence time, 278
- Google
  - Cirq commercial system, 303
  - Simulators, 321
  - Transpilation, 315, 319
- Coding style, 12
  - Underscore in function names, 12
- Quantum random circuits, 150, 158
  - Simulation design, 152
  - Simulation evaluation, 158
  - Simulation implementation, 154
  - Simulation metric, 157
  - Simulation run estimation, 155–158
- Sycamore processor supremacy, 150–152

- Gradient descent, 247
- Greatest common divisor (GCD), 190
- Greenberger, Daniel M., 64
- Ground state energy
  - About variational quantum eigensolver, 240
  - Variational principle, 242–245
- Grover's algorithm
  - About, 210
  - Examples
    - Simple numerical, 214
    - Two-qubit, 214
  - Grover operator, 211, 217
    - Constructing, 233
    - Implementing, 226
    - Quantum counting, 231
  - High-level overview, 211
  - Implementation, 224–227
  - Inversion about the mean, 213
    - Implementation, 219–223
    - Operator, 223
  - Iteration count, 216–218
    - Multiple solutions, 228–230
  - Multiple solutions, 227–230
  - Phase inversion, 212
    - Implementation, 218
    - Multiple solutions, 227–230
    - Operator, 219, 232
  - Quantum amplitude amplification, 227–230
  - Quantum counting, 230–234
- Hadamard basis, 44
  - Measuring in, 104
- Hadamard gates, 44–45
  - Entangler circuits, 61
  - Function call syntax via Pauli matrices, 83
  - Hadamard basis, 44
    - Measuring in, 104
  - Hadamard coin, 235
  - Its own inverse, 45
  - Quantum circuit notation
    - Gate applied, 52
  - Qubit on Bloch sphere, 56–58
  - Random number generator, 78
  - Rotation axis changed, 84
  - Universal gates, 266
- Hamiltonian
  - Definition, 241
  - Eigenvalues
    - About VQE algorithm, 160, 240, 242
    - Schrödinger equation derivation, 240
    - Variational principle, 242–245
    - Variational principle measurements, 248
  - Ising spin glass model, 254
    - Hamiltonian constructed, 258–260
  - Operator, 241
    - Hermitian, 241
- Hash table in libq, 327, 332
- Haskell programming language, 300
  - Quipper as embedded DSL, 300
    - Oracle automatic construction, 301
  - Silq as embedded DSL, 302
- Heisenberg uncertainty principle, 241
- Helper functions
  - Bit conversion, 24
  - Bit iteration, 25
- Hermitian conjugate vector, 2
- Hermitian matrices
  - About, 6
  - Checking if tensor is Hermitian, 12
  - Eigenvalues as real, 7
  - Hermitian adjoint matrices, 6
    - Expressions, 6
  - Hermitian conjugate matrices
    - Adjoint synonymous with conjugate, 6
  - Projection operators as, 43
  - Real vector space basis, 35
  - Trace of, 8
- Hermitian projector, 43
- Hidden state, 60, 63
- Hierarchical QASM, 299
- High-Performance Computing (HPC) techniques, 124
- Horne, Michael A., 64
- IBM
  - Qiskit commercial system, 303
    - ALAP scheduling of gates, 307
    - Algorithm reference, 277
    - QASM support, 311
    - Simulators, 321
  - Sycamore supremacy challenged, 150
    - Summit supercomputer, 158
- Idempotent projection operators, 43
- Identity gates, 26, 33
  - Applied to multiple qubits, 30
  - Controller and controlled qubits not adjacent, 47
  - Hermitian matrix real vector space, 35
  - Phases via, 40
- Increment circuit, 236
- Increment modulo 9 circuit, 237
- Indirect measures of similarity between states, 93–97
  - Swap test code, 96
- Information
  - Erasure resulting in heat dissipation, 66
  - Quantum circuit double lines, 54
  - Quantum teleportation, 97–102
  - Superdense coding, 102–105
  - Teleportation, 97–102
- Inner products
  - About, 3
  - Tensors, 5

- Instruction Set Architecture (ISA) of quantum computers, 25
- Intel Quantum Simulator, 320
- Intermediate representation (IR)
  - About circuit capabilities, 314
  - Scaffold, 298
    - Classic and quantum mix, 305
  - Scalability, 293
  - Transpilation, 311
    - About transpilation, 311
    - IR base class, 312
    - IR nodes, 311
    - Quantum circuit extensions, 313
    - Uncomputation, 315
- Inversion about the mean, 213
  - Implementation, 219–223
  - Operator, 223
- Involutivity, 3
  - Hadamard gates, 45
  - Pauli matrices, 35
  - Rotations, 36
- Ion trap decoherence time, 278
- IR, *see* Intermediate representation (IR)
- ISA (Instruction Set Architecture) of quantum computers, 25
- Ising
  - Hamiltonian, 254
  - Spin Glass, 254
- Junk qubits
  - Quantum computation, 66
- KD-Tree, 270
- Kets
  - About, 72
  - Composite kets inner products, 5
  - Dirac notation, 2
  - Dual vectors for, 3
  - Hermitian conjugate of, 2
  - Inner products, 3
    - Bra-Ket notation, 3
    - Composite kets, 5
  - Outer products, 4
    - Trace of, 8
  - Tensor products, 5
- Knuth, D. E., xi
- Krauss operators, 281
- Kronecker power function (kpow), 11
- Kronecker product, 5, 10
  - $\otimes$  operator symbol, 16
  - $\star$  operator for, 11, 16
- Landauer's principle, 66
- Landauer, D., 66
- Least significant bit, *see* Bit order
- Libq, 145
- Implementation
  - About, 322
  - Controlled gates, 325
  - Gate application, 328
  - Hash table, 327, 332
  - Register file, 322
  - Superposition-preserving gates, 324
  - Superpositioning gates, 326
- libquantum basis, 145
- Transpilation
  - Code generation flag, 315
  - Dumper function, 317
- libquantum, 145
  - Library for sparse representation, 145
- Libquantum library for sparse representation
  - Simulation, 320
- Libraries of compiler optimization patterns, 309
- Logic circuits, 91
  - Fan-out circuits, 92
    - QCL programming language, 296
- “Logical Reversibility of Computation” (Bennett), 66
- Majority voting for repetition code, 285
- Mathematical notation of gate application, 55
- Matrices
  - $\star$  operator for Kronecker product, 11
  - 2-dimensional index via projection operators, 43
  - Density matrices, 23
  - Diagonalization function, 274
  - Eigenvalues, 7, 180
  - Hermitian, *see* Hermitian matrices
  - Matrix multiplication ordering of gate application, 55
  - Pauli matrices, 34
    - Hermitian matrix real vector space, 35
    - Involutivity, 35
  - Permutation matrices, 12
  - Scalability, 122
  - Tensoring together, 11
  - Trace of matrix, 8
  - Transposition, 2
  - Unitary, 6
- Maximal entanglement, 64
- Maximally mixed state, 64
- Maximum cut algorithm
  - About, 254
  - Cut definition, 255
  - Ising formulations of NP algorithms, 254
  - Maximum cut definition, 255
  - Quantum approximate optimization algorithm, 253
  - Variational quantum eigensolver, 260
  - Weighted maximum cut, 255
    - Computing maximum cut, 257

- Graphs constructed, 255
  - Hamiltonian constructed, 258–260
- Measurement gates, 54
- Measurements
  - By peek-a-boo, 131
  - By peek-a-boo, Grover’s algorithm, 226
  - Entanglement, 60
  - Error detection challenges, 285
  - Expectation values, 59, 74, 244
  - Implementation, 75
  - Indirect measures of similarity between states, 93–97
  - Measuring in Hadamard basis, 104
  - Pauli bases, 242
  - Projective, 72
    - Examples, 76
    - Implementation, 75
  - qc data structure, 131
    - Statistical sampling function, 131
  - Quantum circuit notation, 54
  - Quantum Fourier transform and, 169
  - Quantum mechanics postulates, 72
  - States collapsing on measurement, 13, 60
    - Born rule, 72
    - Measurement definition, 72
    - Renormalization, 74
- Mermin, David, 61
- Microsoft Q# commercial system, 303
  - Quantum Developer Kit, 303
  - Simulators, 321
- Microwave cavity decoherence time, 278
- Minimum cut problems, 255
- Mixed states, 68
  - Depolarization, 282
  - Tracing out qubits, 71
- MLPerf benchmarks, 150
- Modular arithmetic, 189
  - Controlled modular multiplication, 207
  - Modular addition, 205–207
- Modular inverse, 202
- Modulus of complex numbers, 2
- Most significant bit, *see* Bit order
- Multi-Controlled gates, 86
  - Ancilla qubits, 86
  - qc data structure, 132
- Multi-qubit gates
  - About controlled gates, 46
  - About single-qubit constructors, 33
  - Hadamard gates, 44–45
- Multiplication, 177
  - Testing quantum arithmetic, 178
- NAND logic gates, 92
- nbits property of Tensor class, 16
  - qc data structure, 128
- No-cloning theorem, 65
  - Fan-out circuits and, 92
  - Repetition code for error control, 285
  - Uncomputation not violating, 67
- Node class for transpilation, 311
- Noise, *see* Quantum noise
- Noisy Intermediate Scale Quantum Computers (NISQ), 240, 293
- Norm
  - Complex numbers, 1
  - Product of two states, 16
  - Unitary matrices as norm preserving, 6, 25
- Vectors
  - Normalization, 20
- Not-gates, *see also* X-gates
  - Logic circuits from, 92
- Nuclear spin decoherence time, 278
- numpy
  - ★ operator for Kronecker product, 11
  - About, 9
  - adjoint() function for operators, 26
  - allclose() for Tensor comparisons, 12
  - Complex number support, 10
  - Eigenvalues of matrices, 180
  - ndarray, 10
    - Base for Tensor, 9
  - ndarray base for Tensor, 9
    - Instantiating, 10
  - Path to, 140
  - print configuration, 27
- “On the Einstein Podolsky Rosen paradox” (Bell), 60, 63
- Open quantum systems, 280
- Open-source simulators, 320
- OpenPulse, 295
- OpenQASM, 295
  - Transpilation
    - About QASM, 311
    - Code generation flag, 315
    - Dumper function, 316
- Operator class
  - adjoint() function, 26
  - Dumper function for matrix structure, 26
  - Gate applied to state  $\psi$  at qubit index, 32
  - Gate function returning Operator object, 32
  - Tensor class parent, 26
- Operator-sum representation, 281
- Operators
  - About, 9, 25
  - Application, 27–29
    - apply() function, 32, 128, 142
  - Density matrices, 68
  - Fast application, 134–139
  - Fast application generalized, 137–139
  - Faster application with C++, 139–145

## Operators (Cont.)

- Fastest benchmarked, 147
- Fastest with sparse representation, 145–147
- Multiple operators in sequence, 31
- Multiple qubits, 29–31
- Noise reduction via compiler optimization, 304
- Norm preserving, 6, 25
- Notation for qubit index applied to, 31
- Padding operators, 31, 48
- Projection operators extracting subspace, 73
- Quantum computation, 66
- To state  $\psi$  at qubit index, 32
- Cloning qubits impossible, 65
- Outer product representation, 43
- qc data structure, 129
  - Gates applied, 128
- Quantum Fourier transform operator, 169
  - Inverse, 170
- Tensor class parent, 26
- Unitary, 25
  - Invertable, 26

## Optical cavity decoherence time, 278

## Optimization

- Gate application iteration lesson, 331
- Gate application special cases, 143–145
- Hamiltonians constructed for, 254
- Hash table reconstruction, 332

## OR logic gates, 92

## Oracles

- Bernstein–Vazirani algorithm, 106, 117
- Deutsch algorithm, 108–117
- Deutsch–Jozsa algorithm, 118–121
- General oracle operator, 117
- Phase inversion implementation, 218
- Quipper automatic construction of, 301
- RevKit for constructing reversible, 321
- Silq construction of, 302

## Oracles–Bernstein–Vazirani algorithm

- Compiler optimization, 308

## Order finding, 192

- About, 192
- Order finding quantum algorithm, 196–200
  - Continued fractions, 208
  - Experimentation, 209
  - Main program, 200
  - Modular addition, 205–207
  - Support routines, 202–205

## Orthogonal vectors, 4

## Outer product representation, 43

## Outer products

- About, 4
- Density matrices as, 23, 68
- Outer product representation of operator, 43
- Projection operators, 42
- Trace of
  - State vectors, 70

## Two kets, 8

Overloading  $\star$  operator, 11Parallelism, *see* Quantum parallelism

## Partial-trace procedure

- Bloch spheres in many-qubit system, 59
- Code, 69
  - Experimenting with, 70
- Derivation for reduced density operator, 68
- Maximal entanglement, 64
- Tracing out other qubits, 69
  - Entangled states, 71
  - Environment traced out, 281
  - Experimenting with, 70
  - Mixed states, 71
  - Pure states, 71
- Quirk qubits on Bloch sphere, 171

## Path to numpy, 140

## Pauli commutators, 83

## Pauli matrices

- About, 34
- Eigenvalues, 180
- Function call syntax via Hadamard gates, 83
  - Pauli commutators, 83
- Hermitian matrix real vector space, 35
- Involutivity, 35
- Measurement in Pauli bases, 242
- Pauli X-gates, *see also* X-gates
- Pauli Y-gates, 34
- Pauli Z-gates
  - Phase-flip gates, 34, 37
- Quantum noise modeling, 282
- Rotation operators via exponentiation, 35

## Peephole optimization, 307

- Libraries of compiler optimization patterns, 309
- Relaxed peephole optimization, 308

## Performance

- Compiler optimization and, 304
- Quantum versus classical computers, 150–152

## Period of function, 192

- Order finding quantum algorithm, 208
- Period finding quantum algorithm, 196–200
  - Experimentation, 209
  - Main program, 200
  - Modular addition, 205–207
  - Support routines, 202–205

## Permutation matrices

- About, 12
- Checking if tensor is permutation, 12
- Controlled-Not gate, 46, 47

## Phase damping, 283

Phase estimation, *see* Quantum phase estimation (QPE)

## Phase gates, 38, 164

- Discrete phase gates, 39, 164
- Phase inversion operator, 219
- Phase shift or kick gates, 40, 164
- Phases via various gates, 40
- RotationZ-gates versus, 38
- Square root of as T-gate, 41
- U1( $\lambda$ ) gates, 40, 164
  - Controlled-U1 gate for quantum arithmetic, 172
- Phase invariance, 59
- Phase inversion, 212
  - Implementation, 218
  - Multiple solutions, 227–230
  - Operator, 219
    - Quantum counting, 232
- Phase kick circuit, 161
- Phase of qubits, 21
- Phase shift error, decoherence-induced, 279
- Phase-flip errors, 279
  - Bit-flip phase-flip channel, 281
  - Error correction, 288
    - Shor's 9-qubit code, 289
  - Phase-flip channel, 281
- Phase-flip gates, 37
- Phase/bit-flip errors, 279
- $\pi$  (pi) fractions transpilation output, 316
- Planck constant, 241
- Podolsky, B., 60
- Polar coordinates for qubit, 55–56
  - Moving about sphere, 56
- Postulates of quantum mechanics, 72
- Power arithmetic, 208
- Power function via Kronecker products, 11
- Preskill, John, 149
- Probabilistic Turing machines, 149
- Probability amplitudes, 13
  - Binary addressing, 19
  - Ket definition, 72
  - Maximally mixed state, 64
  - Measurement, 72, 73
  - Qubits as states, 15, 16
    - Equal superposition with same amplitude, 45
  - Projection operators extracting amplitude, 42
- State class code, 19
- State collapsing on measurement, 13, 72
- State vectors and unitary operators, 25
- Swap gates, 50
- Product states, 61
- Programming languages
  - About hierarchy of abstractions, 295
  - FORTTRAN, 292
  - Haskell, 300
    - Quipper as embedded DSL, 300
    - Quipper oracle automatic construction, 301
    - Silq as embedded DSL, 302
  - PSI probabilistic, 302
  - Q language C++ class library, 300
  - Q#, 303
    - Silq comparison, 302
  - QASM tool, 295
    - Addition via QFT circuit, 173
  - QCL, 296–298
    - Quipper comparison, 301
  - Quipper, 300
    - Oracle automatic construction, 301
    - Proto-Quipper follow-ups, 301
    - QCL comparison, 301
    - Silq comparison, 302
  - Resources for information, 303
  - Scaffold, 298
    - Classical and quantum constructs, 305
    - Entanglement analysis, 299
    - Hierarchical QASM, 299
    - Transpiler, 298
  - Silq, 302
    - Code snippet showcasing, 303
    - Oracle construction, 302
- Projection operators, 42
  - 2-dimensional index into matrix, 43
- Controller and controlled qubits not adjacent, 47
- Hermitian, 43
- Not unitary or reversible, 43
- Outer product representation, 43
- Projective measurements and, 73
- Projective measurements, 72
  - About projective, 73
- ProjectQ commercial system, 303
  - Simulator, 321
- $\psi$  as qubit state space, 13
  - operator applied to  $\psi$  at qubit index, 32
- PSI probabilistic programming language, 302
- Pure states, 68
  - Compiler optimization, 306
  - Trace of density matrix, 24
  - Tracing out qubits, 71
- Python
  - About numpy, 9
  - About ordering of gate applications, 54
- C++
  - Accelerated gate application, 139–145
  - Execution speed, 139, 145
  - Extending Python with, 140
  - Sparse representation, 145
  - Sparse representation benchmarked, 147
- Complex numbers, 2
- conjugate() function, 2
- Operator application, 27–29
- Tensor class
  - About, 9

- Q language C++ class library, 300
  - Transpiler code generation flag, 315
- Q# commercial system (Microsoft), 303
  - Q# programming language, 303
    - Silq comparison, 302
  - Quantum Developer Kit, 303
- QASM tool, 295
  - Addition via QFT circuit, 173
  - cQASM, 295
  - Hierarchical QASM, 299
  - OpenQASM, 295
  - Transpilation
    - About QASM, 311
    - Code generation flag, 315
    - Dumper function, 316
- qc (quantum circuit) data structure
  - About abstraction, 126, 134
  - Constructor, 127
  - Full adder example, 134
  - Gates, 129
    - Adjoint, 130
    - Applying, 128, 142
    - Fast application, 134–139
    - Fast application generalized, 137–139
    - Faster application with C++, 139–145
    - Multi-Controlled gates, 132
    - Swap and controlled swap gates, 131
  - Measurements, 131
    - Statistical sampling function, 131
  - nbits property, 128
  - Quantum registers, 127
  - Qubits added, 128
  - Sparse representation, 145–147
    - Benchmarking, 147
  - Transpilation extension of, 313
    - Code generation flags, 315
    - Eager mode, 313, 314
- QCL programming language, 296–298
  - Quipper comparison, 301
- QFT, *see* Quantum Fourier transform
- qHipster simulator, 320
- Qiskit commercial system (IBM), 303
  - ALAP scheduling of gates, 307
  - Algorithm reference, 277
  - QASM support, 311
  - Simulators, 321
- QPE, *see* Quantum phase estimation
- QRAM model of quantum computing, 293, 294
  - Gate approximation, 310
- qsim simulator (Google), 321
- qsimh simulator (Google), 321
- Quadratic programming problem, 254
- Quantum advantage, 149
- Quantum algorithm zoo, 277
- Quantum amplitude amplification (QAA), 227–230
- Quantum approximate optimization algorithm (QAOA), 253
- Quantum circuit (qc) data structure
  - About abstraction, 126, 134
  - Constructor, 127
  - Full adder example, 134
  - Gates, 129
    - Adjoint, 130
    - Applying, 128, 142
    - Fast application, 134–139
    - Fast application generalized, 137–139
    - Faster application with C++, 139–145
    - Multi-Controlled gates, 132
    - Swap and controlled swap gates, 131
  - Measurements, 131
    - Statistical sampling function, 131
  - nbits property, 128
  - Quantum registers, 127
  - Qubits added, 128
  - Sparse representation, 145–147
    - Benchmarking, 147
  - Transpilation extension of, 313
    - Code generation flags, 315
    - Eager mode, 313, 314
- Quantum circuit notation
  - About ordering of gate applications, 54
  - Controlled gates
    - Controlled-X gates, 53
    - Controlled-Z gates, 53
    - Controlled-Not-by-0 gates, 53
    - More than one qubit controlling, 54
  - Entangler circuits, 61
  - Fan-out circuits, 92
  - Full adder, 89
  - Information flow double lines, 54
  - Logic circuits, 92
  - Measurement, 54
  - Oracle for Bernstein-Vazirani algorithm, 106
  - Qubit order, 51
  - Single-qubit operator applied, 52, 53
  - State as tensor product combined state, 52
  - State change depiction, 52
  - State initialization, 52
  - Swap test, 93
  - X-gates, 53
- Quantum computers
  - Arithmetic via full adder, 89–91
    - Quantum arithmetic, 172–177
  - Classical computers controlling, 293, 294
  - Classical computers simulated by, 149
  - Commercial systems, 303
  - Compiler design challenges, 293
  - Density matrices for theory of, 24, 68
  - Environmental challenges, 278–284
  - Error correction challenges, 285
  - Flow control via controlled gates, 46
    - QCL programming language, 297
    - Silq programming language, 302

- Logic circuits, 91
- Noisy Intermediate Scale Quantum Computers
  - era, 240, 293
- Operators as ISA of, 25
- QRAM model, 293, 294
  - Gate approximation, 310
- Quantum computation, 66
  - $\lambda$ -calculus to express, 300
- Quantum registers, 67, 124–126
- Simulation, 124
- Uncomputation, 66
  - QCL programming language, 296
  - Silq programming language, 302
  - Transpilation intermediate representation, 315
  - Trick for saving result, 67
- Quantum counting, 230–234
- Quantum Developer Kit (QDK), 303
  - Simulators, 321
- Quantum dot decoherence time, 278
- Quantum error conditions, 279
- Quantum error correction
  - About, 278, 284
  - Bit-flip errors, 286–288
    - Shor's 9-qubit code, 289
  - Channels, 281
  - Compiler optimization and, 304
  - Error correction code memory, 284
  - Error syndrome, 286
  - Phase-flip errors, 288
    - Shor's 9-qubit code, 289
  - Quantum computing challenges, 285
  - Quantum noise, 278–284
  - Repetition code, 284
    - Majority voting, 285
    - No-cloning theorem, 285
    - Quantum repetition code, 285
  - Resources for information, 291
  - Shor's 9-qubit error correction code, 285, 289
- Quantum fidelity, 269
- Quantum Fourier transform (QFT)
  - About, 169
  - Algorithm detail
    - About, 163
    - Binary fractions, 163
    - Quantum Fourier transform, 165–167
    - Two-qubit QFT, 167–169
    - Two-qubit QFT online simulation, 170
  - Measurement, 169
  - Online simulation, 170
  - Phase kick circuits, 161
  - QCL programming language, 298
  - QFT operator, 169
    - Inverse, 170
  - Quantum arithmetic
    - Addition, 172–177
    - Multiplication, 177
    - Subtraction, 177
    - Testing, 178
- Quantum information, *see* Information
- Quantum IO Monad, 300
- Quantum mechanics
  - Copenhagen interpretation, 61
  - Hidden state, 60, 63
  - Postulates, 72
- Quantum noise, 278–284
  - Amplitude damping, 283
  - Channels, 281
    - Bit flip and phase flip, 281
    - Depolarization, 282
  - Compiler optimization and noise reduction, 304
  - Error correction, 284–291
  - Error injection to model, 282
    - Checking bit-flip error correction, 287
    - Gates as quantum noise source, 283
  - Gates imprecise, 283
  - Phase damping, 283
  - Quantum operations, 280
    - Operation element, 281
    - Operator-sum representation, 281
  - Simulation, 320, 321
- Quantum operations
  - Operation element, 281
  - Operator-sum representation, 281
- Quantum parallelism, 108, 112
- Quantum phase estimation (QPE)
  - Detailed derivation, 182–186
  - Eigenvalues and eigenvectors, 180
  - Hamiltonian eigenvalues, 240
  - Implementation, 186–189
  - Phase estimation, 181
    - Definition, 181
  - Quantum counting, 231
- Quantum programming languages
  - About hierarchy of abstractions, 295
  - Haskell, 300
    - Quipper as embedded DSL, 300
    - Quipper oracle automatic construction, 301
    - Silq as embedded DSL, 302
  - PSI probabilistic, 302
  - Q language C++ class library, 300
  - Q#, 303
    - Silq comparison, 302
  - QASM tool, 295
    - Addition via QFT circuit, 173
  - QCL, 296–298
    - Quipper comparison, 301
  - Quipper, 300
    - Oracle automatic construction, 301
    - Proto-Quipper follow-ups, 301
    - QCL comparison, 301
    - Silq comparison, 302
  - Resources for information, 303

- Quantum programming languages (Cont.)
  - Scaffold, 298
    - Classical and quantum constructs, 305
    - Entanglement analysis, 299
    - Hierarchical QASM, 299
    - Transpiler, 298
  - Silq, 302
    - Code snippet showcasing, 303
    - Oracle construction, 302
- Quantum random circuits (QRC), 150
  - Simulation design, 152
  - Simulation evaluation, 158
  - Simulation implementation, 154
  - Simulation metric, 157
  - Simulation run estimation, 155–158
- Quantum random walk, 234–240
  - 1D walk, 235
  - 2D walk, 237
  - About, 234, 238
  - Coin toss, 235
  - Walking the walk, 237
- Quantum registers, 124–126
  - Code to create and initialize, 125
    - qc data structure, 127
  - Compiler optimization and, 304
  - For result storage, 67
  - QCL programming language, 296
  - Reg class, 124
- Quantum supremacy, 149
  - “Quantum supremacy using a programmable superconducting processor” (Arute et al.), 150
- Quantum teleportation, 97–102
  - Error correction trick, 286
- Quantum Turing machines, 149
- Qubits
  - About the state of a qubit, 13
    - Basis states, 13, 15
    - Basis states orthonormal, 15
    - Collapsing on measurement, 13, 60, 72
    - Communicating state of two with one, 102–105
    - Equal superposition of adjacent qubits, 45
    - Measurement, 75
    - Measurement examples, 76
    - Probability amplitudes, 13, 15, 16
    - State class constructors, 22
    - Superposition via Hadamard gates, 44
    - Tensor product combined state, 52
  - Ancilla qubits, 66
  - Binary addressing, 19
  - Bloch sphere describing, 55–56
    - Computing coordinates for given state, 58
  - Cloning or copying impossible, 65
  - Column vectors of complex numbers, 2, 15
  - Compiler optimization via recycling, 309
  - Complexity of scaling up, 122
  - Constructing in code, 14
    - Data structure, 14
  - Entanglement, 60–65
  - Junk qubits, 66
  - Operator application, 27–29
    - Applied at index specified, 31
    - Controller and controlled qubits, 46–51, 53
    - Multiple operators in sequence, 31
    - Multiple qubits, 29–31
    - Nonadjacent controller and controlled qubits, 47
    - Norm preserving, 6, 25
    - Notation for qubit index applied to, 31
    - Projection operators extracting subspace, 73
    - Quantum computation, 66
    - Qubit ordering, 137
  - Order of qubits, 18
    - Operator application, 137
    - Quantum circuit notation, 51
  - Phase, 21
  - Polar coordinates describing
    - Moving about sphere, 56
  - Tensors constructing, 9, 14
    - Code, 15
    - nbits property, 16
    - n* qubits, 16
- QuEST (Quantum Exact Simulation Toolkit), 321
- Quipper programming language, 300
  - Oracle automatic construction, 301
  - Proto-Quipper follow-ups, 301
  - QCL comparison, 301
  - Silq comparison, 302
- Quirk online simulations, 170
- QX Simulator, 320
- Random circuits, *see* Quantum random circuits (QRC)
- Random number generator, 78
  - Coin toss, 235
  - Random combination of 0 or 1 states, 23
- Random walk, 234–240
  - 1D walk, 235
  - 2D walk, 237
  - About, 234, 238
  - Coin toss, 235
  - Walking the walk, 237
- Reduced density operator, 68
  - Debugging tool for inspecting states, 68
  - Partial trace derivation, 68
    - Code, 69
  - Quirk qubits on Bloch sphere, 171
- Reg class, 124
- Registers, 67, 124–126
  - Code to create and initialize, 125

- libq implementation, 322
- qc data structure, 127
- Compiler optimization and, 304
- QCL programming language, 296
- Reg class, 124
- Relative phase, 35
  - Global phase versus, 60
- Relaxed peephole optimization, 308
- Renormalization
  - States collapsing on measurement
  - Renormalization, 74
- Repetition code, 284
  - Majority voting, 285
  - No-cloning theorem, 285
  - Quantum repetition code, 285
- Resources for information
  - Algorithms, 277
  - Computational complexity theory, 149
  - Logical to physical mapping, 310
  - Quantum error correction, 291
  - Quantum programming languages, 303
  - Quirk online simulator, 170
  - Schrödinger equation, 241
  - Simulators available, 320
- RevKit for reversible oracles, 321
- Rk gates, 39
  - Phases via various gates, 40
- Roots of gates, 41
  - Square roots of gates
  - About, 41
- Rosen, N., 60
- Rotation Axis, 37
- Rotation operators, 35
  - Axis of rotation, 37
  - Controlled rotation gates additive, 161
  - Discrete phase gates, 39
  - Quantum counting, 231
  - Qubit on Bloch sphere, 56
  - Rotation axis changed, 84
  - Square roots of, 41
- Row vectors
  - Bras in Dirac notation, 3
  - Inner products, 3
- RSA encryption algorithm, 189
- S-gates, *see also* Phase gates, 165
- Scaffold programming language, 298
  - Classical and quantum constructs, 305
  - Entanglement analysis, 299
  - Hierarchical QASM, 299
  - transpiler, 298
- Scalability
  - About, 122
  - Complexity of scaling up, 122, 293
  - Controlled gates, 48
  - Gate fast application, 134–139
  - Hierarchical QASM, 299
  - Quipper programming language, 300
- Scalar products, *see* Inner products
- Scheduling of gates, 306
- Schrödinger–Feynman Simulation, 124
- Schrödinger equation
  - qsim simulator, 321
  - Resource for more information, 241
  - Time-independent for state evolving, 72
    - Drivation, 240
    - Variational principle, 242–245
- Schrödinger–Feynman path histories, 145, 151, 159
  - qsimh simulator, 321
- scipy
  - Installing, 42
  - sqrtn() function, 42
- Sets of universal gates, 49
- Shor’s 9-qubit error correction code, 285, 289
- Shor’s integer factorization algorithm
  - About, 189
  - About phase estimation, 163
  - Classical
    - Experimentation, 193
    - Factorization, 190
    - Greatest common divisor, 190
    - Modular arithmetic, 189
    - Period finding, 192
  - Period finding quantum algorithm, 196–200
    - Continued fractions, 208
    - Controlled modular multiplication, 207
    - Experimentation, 209
    - Main program, 200
    - Modular addition, 205–207
    - Support routines, 202–205
  - Sparse representation Benchmarked, 147
- Silq programming language, 302
  - Code snippet showcasing, 303
  - Oracle construction, 302
- Simon’s algorithm, 121
- Simon’s generalized algorithm, 121
- Simulation
  - About scalability, 122
  - Available simulators, 320
  - Complexity, 122, 151, 155
  - Online simulators, 170
  - Open-source simulators, 320
  - Parallelization of gates, 307
  - Quantum Fourier transform online simulation, 170
  - Quantum random circuits
    - Google team, 151, 158
    - Metric, 157
    - Simulation design, 152
    - Simulation evaluation, 158
    - Simulation implementation, 154
    - Simulation run estimation, 155–158

- Simulation (Cont.)
  - Quantum registers, 124–126
  - Quantum simulating classical computers, 149
- Single-qubit gates, 33–45
  - About constructing multi-qubit operators, 33
  - About operators, 9
  - Applying operator, 52
  - Hadamard gates, 44–45
  - Identity gates, 26, 33
    - Applied to multiple qubits, 30
  - Phase gates, 38
    - Discrete phase gates, 39
    - Phase shift or kick gates, 40
    - Phases via various gates, 40
  - Projection operators, 42
  - Reversed by conjugate transpose, 26
  - Rk gates, 40
  - Rotation operators, 35
  - RotationZ-gates versus phase gates, 38
  - Square roots of gates, 41
  - T-gates, 40
    - Phases via, 40
  - U1( $\lambda$ ) gates, 40
  - X-gates, 26, 34
    - Applied to multiple qubits, 30
    - State initialization to 0- or 1-state, 22
  - Y-gates, 34
  - Z-gates, 34
- Sleator–Weinfurter construction, 85
- Solovay–Kitaev (SK) theorem, 266
- Solovay–Kitaev (SK) algorithm
  - About, 266
  - Algorithm, 270–272
  - Balanced group commutator, 272–274
    - Matrix diagonalization function, 274
  - Bloch sphere angle and axis, 267
  - Evaluation, 274
  - Pre-computing gates, 269
  - Random gate sequences, 276
  - Similarity metrics
    - Quantum fidelity, 269
    - Trace distance, 268
  - Theorem, 266
  - Universal gates, 266
    - SU(2) group, 267
- Sparse representation, 145–147
  - About, 122
  - Benchmarking, 147
  - libquantum library, 145
    - Simulation, 320
- SPEC benchmarks, 150
- Spooky action at a distance, 60, 63
  - Quantum teleportation, 97–102
- sqrtm() function of scipy, 42
- Square roots of gates
  - scipy sqrtm() function, 42
  - Squaring root of gate equals gate, 80
- State class
  - Constructing qubits in code, 14
    - Qubit data structure, 14
  - Constructors, 22
    - All 0-states or 1-states, 22
  - density() function, 23
  - Member functions, 19–21
    - Dumper function for all relevant info, 21
  - Probability and amplitudes, 19
  - Tensor class parent, 15
    - nbits property, 16
- States
  - About, 9
  - About bit order, 17, 18
    - Binary interpretation, 18
    - Bit index notation, 136
  - Basis states of qubits, *see* Basis states
  - Cloning, 65
  - Collapsing on measurement, 13, 60
    - Born rule, 72
    - Measurement definition, 72
    - Renormalization, 74
  - Density matrices, 68
  - Entanglement, 60–65
  - Kets representing state of system, 72
    - State evolving via operators, 72
  - Maximally mixed state, *see also* Probability amplitudes
  - Operator application, 27–29
    - Multiple qubits, 29–31
  - Projection operators extracting amplitude, 42
  - Quantum circuit notation
    - Single-qubit operators applied, 52
    - State change depiction, 52
    - State initialization, 52
  - Quantum operations, 280
  - Similarity via swap test, 93–97
    - Code, 96
  - Single-qubit 0 and 1 state constants, 23
  - Tensors constructing, 9
  - Tensors constructing qubits, 14
    - Code, 15
    - Qubit data structure, 14
    - $n$  qubits, 16
    - Tensor product combined state, 52
- Vectors
  - Binary interpretation, 18
  - Column vectors of complex numbers, 2, 14
  - Complexity of scaling up, 122
  - Kets representing state of system, 72
  - Normalization, 13, 20
  - Normalized vectors and, 4
  - Operator application, 27–29, 72
  - Unitary operators as norm preserving, 25
- Steane code, 291

- Steane, Andrew, 291
- SU(2) group, 267
- Subset-sum algorithm
  - About, 262
  - Experiments, 264
  - Implementation, 263
- Subtraction, 177
  - Decrement operator, 236
  - Testing quantum arithmetic, 178
- Summit supercomputer simulating quantum
  - random circuits, 158
- Superdense coding, 102–105
- Superposition
  - About, 44
  - About measurement, 72
  - Hadamard gates on qubits, 44
    - Equal superposition of adjacent qubits, 45
  - Linear combination of basis states, 13
  - Maximally mixed state, 64
  - State after operator applied, 52
- Swap gates, 50, 88
  - Compiler optimization, 308
  - Controlled-controlled Swap gates, 51
  - qc data structure, 131
- Swap test, 93–97
  - Code, 96
- Sycamore processor, 150
- T-gates
  - Phases via, 40, 165
  - Rotation axis changed, 84
  - Square root of phase gates, 41
  - Universal gates, 266
- Teleportation, 97–102
  - Error correction trick, 286
- Tensor class
  - About array behavior, 9
  - Comparing to values, 12
  - Instantiating, 10
    - tensor\_type() abstraction, 10
  - Kronecker product member function, 11
  - Operators derived from, 26
  - Qubit states
    - Code, 15
  - State class derived from, 15
    - nbits property, 16
- Tensor products, 5
  - Binary interpretation, 19
  - Distributive, 5
  - Kronecker product as, 10
  - Multiplication with scalar, 5
  - Operators applied to multiple qubits, 29–31
    - Multiple operators in sequence, 31
  - Product states, 61
  - State of two or more qubits, 16
  - Trace of matrix, 8
- Tensors, 9
  - About array behavior, 9
  - Checking if Hermitian or unitary, 12
  - Comparing to values, 12
  - Inner products, 5
  - Instantiating, 10
    - tensor\_type() abstraction, 10
  - Kronecker product, 10
- Testing
  - Benchmarking, *see* Benchmarking
  - Gate fast application, 139
  - Controlled gate equivalencies validated, 88
  - Debugging, 12
  - Quantum arithmetic, 178
  - Tracing out state of one qubit, 70
- Toffoli gates, 49, 85
  - Logic circuits from, 92
  - Multi-Controlled X-gates, 86
- Trace distance, 268
- Trace of matrix, 8
  - Hermitian matrices, 8
  - Partial-trace procedure, 59
  - Tensor product, 8
  - Trace of outer product two kets, 8
- Transpilation
  - About, 292, 311
  - Code generation flags, 315
  - Dumper function, 315
    - Cirq, 319
    - libq, 317
    - QASM, 316
  - Intermediate representation, 311
    - Circuit capabilities of, 314
    - IR base class, 312
    - IR nodes, 311
    - Quantum circuit extensions, 313
    - Uncomputation, 315
  - $\pi$  fractions output, 316
  - Scaffold transpiler, 298
- Transposition
  - Involutivity, 3
  - Matrix, 2
- Two-qubit quantum Fourier transform, 167–169
  - Online simulator, 170
- U1(lambda) gates, 40, 164
  - Controlled-U1 gate for quantum arithmetic, 172
- Uncomputation, 66, 315
  - QCL programming language, 296
  - Silq programming language, 302
  - Transpilation intermediate representation, 315
  - Trick for saving result, 67
- Underscore in function names, 12
- Unitary matrices
  - About, 6
  - Checking if tensor is unitary, 12

- Unitary matrices (Cont.)
  - Norm preserving, 6, 25
  - Tensoring together, 11
- Unitary operators, *see also* Gates; Operators
  - Invertable, 26
- Universal gates, 49
  - Definition, 266
  - QRAM model of quantum computing, 294
  - Solovay–Kitaev theorem, 266
  - SU(2) group, 267
- Universal gates in quantum computing, 49
- V-gates
  - Square root of X-gates, 41, 85
- val2bits() for decimal to binary, 24
- Variational quantum eigensolver (VQE)
  - About, 160, 240
  - Algorithm, 245–248
  - Expectation values, 244
  - Hamiltonian type, 242
  - Measurement in Pauli bases, 242
  - Measuring eigenvalues, 248
  - Multiple qubits, 250–252
  - Quantum phase estimation, 240
  - Schrödinger equation, 240
  - Variational principle, 242–245
- Vectors
  - Binary interpretation, 18
  - Complex numbers, 2
  - Norm, 4
  - Orthogonal, 4
  - States
    - Basis states of qubits, 13
    - Complexity of scaling up, 122
    - Initializing with normalized vector, 23
    - Kets representing state of system, 72
    - Operator application, 27–29, 72
    - Unitary operators as norm preserving, 25
  - Tensor products, 5
  - Unitary matrices as norm preserving, 6, 25
- VQE, *see* Variational quantum eigensolver algorithm
- Weighted maximum cut, 255
- Wilczek, F., xi
- Wire optimization, 309
- X-gates, 26, 34
  - Applied to multiple qubits, 30, 135
  - Controlled-Controlled X-gates (CCX-gates), 49, 85
  - Logic circuits from, 92
  - Multi-Controlled X-gates, 86
  - Not-gate, 34
  - Quantum circuit notation
    - Controlled-X gates, 53
    - Controlled-Not-by-0 gates built, 53
  - X-gates, 53
  - Qubit on Bloch sphere, 56
  - Square root of as V-gate, 41, 85
  - State initialization to 0- or 1-state, 22
- Y-gates, 34
  - Negate Y-gate, 83
  - Square root of, 41
- Z-gates, 34
  - Controlled-Z gates, 81
    - Controlled-Not gates via, 82
    - Multi-Controlled-Z gates, 82
  - Phase-flip gates, 34, 37
  - Phases via, 40
  - Quantum circuit notation, 53
    - Controlled Z-gates, 53
  - Qubit on Bloch sphere, 56–58
- Z90-gates, *see also* Phase gates
- Zeilinger, Anton, 64