

Index

- ? (question mark)
 - Boolean variable, 123
 - variable, 216–217, 256
 - ;(semicolon), comment delimiter, 172
 - 3D, agent movement, 248–249
 - 3D perspective, 263
 - 3D worlds, 248–252

 - ABM (agent-based modeling). *See also* Hybrid
 - computational models
 - advantages of, 17
 - aggregate patterns, determining from known individual behavior (*see* Integrative understanding)
 - calculus requirements, 17
 - characteristic features of, 101–102
 - complex systems, 23
 - complex systems and emergence, 5–13
 - complex systems theory, 6
 - computational intensity, 37
 - core idea, 32
 - definition, 1
 - differential understanding, 7, 8–9
 - disadvantages of, 17–18
 - vs. EBM (equation-based modeling), 32–34, 162
 - emergence, 6, 29 (*see also* Emergent phenomena)
 - explanatory cognitive pattern (*see* DC [deterministic-centralized] mindset)
 - free parameters, 37
 - future of, 424–425
 - history of (*see* Computational roots of ABM)
 - individual element behavior, determining from known aggregate patterns (*see* Differential understanding)
 - integrative understanding, 7–8
 - introduction, 1–3
 - prerequisites for, 38–39
 - programming language for (*see* NetLogo)
 - restructuration, 4
 - self-organization, 6–7
 - structuration, 4
 - structurational inertia, 4
 - tradeoffs, 36–38, 423–424
 - ABM cycle, hybrid computational models, 388–389
 - ABM design principle, 160–161
 - ABMs (agent-based models)
 - analyzing (*see* Analyzing models)
 - computational, 22
 - conceptual
 - correspondence to implemented model (*see* Verification)
 - definition, 22
 - connecting to the physical world (*see* NetLogoLab environment)
 - controlling (*see* File-based controls methods; Observer/User Interface)
 - correctness (*see* Replication; Validation; Verification)
 - creating (*see* Creating, ABMs)
 - duplication (*see* Replication)
 - glass-box approach, 33
 - with heterogeneous agents, 35–36
 - identifying the need for, 35–36
 - introduction, 14
 - model implementation
 - correspondence to conceptual model (*see* Verification)
 - definition, 22
 - validity for real world phenomena (*see* Validation)
 - vs. other models, 32–34, 35–36
 - path dependency, 30
 - performance, improving, 418–419
 - randomness vs. determinism, 34–35
 - usefulness vs. completeness, 161
 - uses for (*see* Applications for ABMs)
 - verification, 33
- ABMs (agent-based models), components of. *See also* Agents (ABM); Environments; Interactions; Observer/User Interface
 - graphical components, 14
 - Schedule, 204
 - state variables, 14
 - values, 14

- Actions
 - defining new, 209
 - overview, 209–211
 - predefined, 209
- Actor model, computational roots of ABM, 440–442
- Adaptation in Natural and Artificial Systems*, 436
- Adaptive agents
 - agent cognition, 227, 230
 - computational roots of ABM, 436–437
- Agent-agent interactions, 259–260
- Agent-based modeling (ABM). *See* ABM (agent-based modeling)
- Agent-based models (ABMs). *See* ABMs (agent-based models); Agents (ABM)
- Agent buttons, 263
- Agent-centric thinking, 54
- Agent cognition
 - adaptive agents, 227, 230
 - definition, 224–225
 - goal-based, 227–231
 - ML (Machine Learning), 232
 - reflexive agents, 225
 - utility-based, 225–227
- Agent collections
 - agentsets
 - and computation, 217–222
 - creating, 213–214
 - definition, 212
 - empty, 215
 - vs. lists, 214–217
 - no-turtles, 215
 - breeds of agents, 212
 - lists, 214–217
 - types of agents, 211–212
- Agent-environment interactions, 261–262
- Agent variables, 58
- Agents (ABM)
 - actions, 209–211
 - ant foraging model, 22
 - assigning animal behaviors, 167
 - behaviors (*see* Actions)
 - birth events, 257
 - birth processes, 166
 - centering on, 263
 - color (*see* COLOR property)
 - communicating with other agents, 260
 - consuming other agents, 259
 - copying, 210
 - creating (*see* Heroes and Cowards model; Life model; Simple Economy model)
 - death procedures, 51, 166, 167, 258
 - defining new actions, 209
 - definition, 203
 - description, 14
 - direction facing (*see* HEADING property)
 - eating, 167
 - energy levels, 165–166
 - environmental characteristics, 166
 - environmental interactions, 36
 - granularity, 164, 222–224
 - heading, 165–166 (*see also* HEADING property)
 - heterogeneous, 35–36
 - HEV-M (Human, Embedded and Virtual agents in Mediation), 372–375, 376–377
 - history of, 204
 - human, 372–375
 - initial population, specifying, 168
 - internal states (*see* Properties)
 - level of complexity, 222–224
 - location, 165–166
 - location coordinates (*see* XCOR property; YCOR property)
 - meta-agents, 232–233
 - methods (*see* Actions)
 - mixing with human agents (*see* Participatory Simulations)
 - movement (*see* Movement of agents)
 - observing the environment, 261
 - overview, 205
 - pitch property (3D), 248, 250
 - predefined actions, 209
 - properties, 165–166, 205–209
 - proto-agents, 165, 234
 - reproduction, 167, 185–186
 - robotic, 372–375
 - roll property (3D), 248, 251
 - scale of the model, 224
 - self-destruction, 210
 - sensing other agents, 259–260
 - setting up time steps, 167–168
 - stationary agents, 166
 - subagents, 233
 - tracking over time (*see* WHO property)
 - unique IDs (*see* WHO property)
 - virtual, 372–375
 - yaw, 250
- Agents (NetLogo). *See also* Turtles
 - collections of (*see* Agentsets)
 - definition, 15
- Agent-self interactions, 257–258
- Agentsets
 - and computation, 217–222
 - creating, 213–214
 - definition, 92, 212
 - empty of turtles, 215
 - links, 92
 - vs. lists, 214–217
 - no-turtles, 215
 - patches, 92
 - turtles, 92
- Aggregate (macro) behavior, Ant model, 27–28
- Aggregate patterns, determining from known individual behavior. *See* Integrative understanding

- Aggressors and Defenders. *See* Heroes and Cowards model
- Algorithms
 - efficiency of, 220
 - replication, 339
- Alignment rule, 327, 444
- American Express credit card applications, 232
- Analogies from agent-based models, 30
- Analyzing models
 - Spread of Disease model
 - androids, 360
 - batch experiment tool, 289
 - BehaviorSpace tool, 289–291, 292, 293
 - compression of data, 288
 - connections-per-node, effects of, 297–299
 - creating agents, 283–284
 - data inconsistencies, 286–287
 - Disease HubNet model, 360–362
 - edge density of disease trails, 303–304
 - effects of population density, 289, 291
 - environmental data, 301–304
 - environmental decay, effects of, 301–302
 - environmental patterns, 304
 - GIS (Geographical Information Systems), 304
 - graphs, 291–296
 - landscape metrics, 303–304
 - modeling other spreading effects, 285
 - naming experiments, 289–290, 292
 - necessity of multiple runs, 288–291
 - overview, 283–284
 - patterns of infection, 302–304
 - population density, effects of, 285
 - SNA (social network analysis), 296–300
 - statistical analysis, 287–288
 - summary, 305–307
 - time series analysis, 295–296, 297
 - tools for, 288
 - Wolf-Sheep Simple model
 - anomalous/aberrant behavior, 191
 - averaging across runs, 192
 - BehaviorSpace tool, 191
 - collecting results, 191–192
 - data analysis, 192
 - initial number of wolves, 192
 - multiple runs, 191–192
 - oscillatory patterns, 164, 192
 - overview, 189–191
 - parameter sweeping, 191–192
 - plotting data, 192
 - stochastic components, 191–192
 - summarizing data, 192
- Androids, Spread of Disease model, 360
- Anomalous/aberrant behavior, 191
- Ant model
 - aggregate (macro) behavior, 27–28
 - ant agent, 22
 - ant trails, 24, 28
 - complex systems, 23
 - foraging model, 22–27
 - optimal behavior, 28
 - pheromone hypothesis, 24
 - randomness vs. determinism, 34
 - results and observations, 27–28
 - value of, 28–32
- Ant trails, 24, 28
- Anthropology
 - Artificial Anasazi model, 333
 - Kayenta Ancestral Pueblo, 333
 - modeling, 333, 422
- Applications for ABMs
 - anthropology, 422 (*see also* Artificial Anasazi model)
 - biology, 222–223, 421
 - chemistry, 127–128, 421
 - communication, 30, 369–370
 - computer science, 423
 - description, 28–29
 - ecology (*see* Ant model; Fire model; Wolf-Sheep Simple model)
 - economics, 421–422 (*see also* Simple Economy model)
 - education, 30, 371–372 (*see also* Chicago Public Schools model)
 - engineering, 422–423
 - evolution, 196
 - experimentation, 30
 - explanation, 29–30
 - focal objects for scientific dialog, 31
 - math, 423
 - medicine
 - epidemiology, 117–118 (*see also* Spread of Disease model)
 - overview, 421
 - natural sciences, 421
 - organizations, 422
 - persuasion, 370–371
 - physics, 421
 - policy flight simulators, 371
 - politics, 422
 - predictions, 31
 - ray traced images, 420
 - sources of analogy, 30
 - thought experiments, 31
 - traffic jams (*see* Traffic Basic Model)
- Arrow, Kenneth, 437
- Arthur, W. Brian, 141, 437
- Artificial Anasazi model, 333
- Artificial Life models, 445
- Artificial Stock Market, 150
- ASK PATCHES command, 53
- Asynchronous vs. synchronous updates, 270
- Authors, replication factor, 339
- Average rewards, El Farol model, 145–146
- Averaging across runs, 192

- Axelrod, R., 437
 Axelrod-Hammond Ethnocentrism model, 336–337
 Axtell, Robert, 87–88
- BACH group, 437
 BACKWARD command, 210
 Backward movement of agents, 262
 Bak, Per, 104, 431
 Barabasi-Albert model, 433
 Bass model, 414
 Batch experiment tool, 289
 Batch running, 268
 Batch vs. interactive, 267–268
 Behaviors of agents. *See* Actions
 BehaviorSpace tool, 191, 289–291, 292, 293
 Big-O notation, 220
 Binomial distributions, 173–174
 Biology
 computational roots of ABM, 431
 modeling (*see also* Ant model; Flocking model)
 AIDS model, 223–224
 overview, 421
 Tumor model, 222–223
 Birds in flight, rules of, 327, 444. *See also* Flocking model
 Birth events, 257
The Blind Watchmaker, 85
 Boat traffic, modeling, 407
 Boids model, 443–445
 Boltzmann, Ludwig, 95
 Boltzmann equation, 95
 Boltzmann-Gibbs law, 93, 95
 Bonabeau, Eric, 68–69
 Books and publications. *See also* Documentation
 Adaptation in Natural and Artificial Systems, 436
 The Blind Watchmaker, 85
 Calculating Space, 65
 Changing Minds, 4
 “Computer Recreations,” 68
 Dialogues Concerning Two New Sciences, 5
 Emergence, 438
 Growing Artificial Societies, 88
 Hidden Order, 438
 The Magic Machine, 68
 Micromotives and Macrobehavior, 432
 A New Kind of Science, 68
 Organization of Behavior, 436
 Six Degrees, 300
 Theory of Self Reproducing Automata, 434
 Bottom-up modeling, 159–160
 Boundary conditions, 240–241
 Bounded rationality, 437
 Bounded topology, 240–241
 Box, George, 161
 BREED property, 206
 Breeds of agents, 212
 Bugs vs. emergent behavior, 317–321, 325
- Burks, Arthur W., 434, 437
 Buttons, 264
- CA (cellular automata), 65–68, 433–435
Calculating Space, 65
 Calculus requirements for ABM, 17
 Calibration
 definition, 332
 extracting rules
 by data mining, 358
 from empirical data, 357–360
 from participatory simulations, 360–363
 through machine learning, 357
 Cars, modeling traffic flow. *See also* Traffic Basic model
 adjusting speed, 210–211
 Cruising model, 407
 free-flow state, 231
 initializing speed, 208–209
 interactions between, 210–211
 interactions with the environment, 224–230
 modifying actions and strategies, 230
 speed regulation, 258
 Cellular automata (CA), 65–68, 433–435
 Centering on an agent, 263
Changing Minds, 4
 Chemistry
 ABM applications for, 127–128
 modeling, 421
 Chicago Public Schools model, 407
 Chinese streamers, 85–86
 Choosers, 265
 City growth model, 334–335
 CLEAR-ALL command, 51
 CLEAR-TURTLES command, 258
 Climate Change Game model, 407
 Clock
 advancing, 55–60
 resetting, 54
 Closures. *See* TASKS
 Club of Rome model, 432
 CM (connection machines), 442
 Code, definition, 271
 Code tab, 48, 272
 Coding guidelines, clarity over efficiency, 220
 Cognitive pattern underlying ABM. *See* DC
 (deterministic-centralized) mindset
 Cognitive psychology, participatory simulations, 360
 Cohen, M. D., 437
 Cohesion rule, 327–328, 444
 Collecting results, 191–192
 Collections of data. *See* Agentsets; Lists; TASKS
 Collective behavior, modeling. *See* Heroes and Cowards model
 Color coding
 agents, 143–145
 grass, 182–183

- NetLogo command blocks, 109
- NetLogo commands, 109
- COLOR property, 206, 208
- Command blocks, color coding, 109
- Command center, 264
- Commands. *See also specific commands*
 - re-executing, 264
 - running lists of, 393–395
- Commands, NetLogo
 - color coding, 109
 - creating your own, 109
 - definition, 109
 - naming conventions, 109
- Commenting your code, 172
- Communication
 - agents with agents, 260
 - model implementers and model authors, 313
 - modeling, 369–370
 - replicators and model author(s), 342
 - through agent-based models, 30
- Complex adaptive systems, 435–438
- Complex systems
 - Ant model, 23
 - definition, 23
 - and emergence, 5–13
- Complex systems theory, 6
- Compositional understanding. *See* Differential understanding
- Compression of data, 288
- Computational complexity, 220
- Computational intensity of ABM, 37
- Computational models
 - definition, 22
 - hybrid
 - ABM cycle, 388–389
 - EC (evolutionary computation), 389–390
 - integrated cycle, 388–389
 - Journey to El Yunque project, 385
 - ML (Machine Learning) and ABM, 386, 388–389
 - ML cycle, 388–390
 - SDM (System Dynamics Modeling) and ABM, 383–386
 - Tabonuco-Yagrumo Hybrid model, 385, 387
 - Wolf-Sheep Predation Docked model, 385
- Computational roots of ABM
 - Actor model, 440–442
 - adaptive agents, 436–437
 - alignment rule, 444
 - Arrow, Kenneth, 437
 - Arthur, W. Brian, 437
 - Artificial Life models, 445
 - Axelrod, R., 437
 - BACH group, 437
 - Bak, Per, 431
 - Barabasi-Albert model, 433
 - biology, 431
 - birds, behavioral rules, 444
 - Boids model, 443–445
 - bounded rationality, 437
 - Burks, Arthur W., 434, 437
 - cellular automata, 433–435
 - Club of Rome model, 432
 - CM (connection machines), 442
 - Cohen, M. D., 437
 - cohesion rule, 444
 - complex adaptive systems, 435–438
 - computer graphics, 443–445
 - Conway, John Horton, 434
 - C-Star language, 442
 - cybernetics, 432
 - Dahl, Ole-Johan, 440
 - data parallelism, 442–443
 - document retrieval, 443
 - ECHO models, 438
 - El Farol Bar problem, 437
 - engineering, 431–432
 - Feynman, Richard, 443
 - first computational agent, 439
 - fluid flow application, 443
 - Forrest, Stephanie, 438
 - GA (genetic algorithms), 435–438
 - Game of Life, 434
 - game theory models, 432
 - Garbage Can Model of Organizations, 437
 - Gardner, Martin, 434
 - Goucher, Adam, 434
 - Hebb, Donald, 435–436
 - Hewitt, Carl, 441
 - Hillis, Danny, 442
 - Holland, John H., 434, 435–438
 - human networks, 433
 - IBM methodology, 431, 441
 - implicit fitness function, 438
 - Individual-Based Modeling, 431
 - Ising models, 431
 - Kay, Alan, 440
 - Langton, Chris, 445
 - Lisp language, 440
 - Logo, 439–440
 - McCarthy, John, 440
 - Milgram, Stanley, 433
 - Minsky, Marvin, 439
 - Mitchell, Melanie, 438
 - Monte Carlo simulation, 432
 - network theory, 432–433
 - Nygaard, Kristen, 440
 - object-oriented programming, 440–442
 - object recognition, 443
 - Papert, Seymour, 439–440
 - particle systems, 443–445
 - perceptrons, 436
 - physics, 431
 - point representations, 444
 - process engineering, 431–432

- Computational roots of ABM (cont.)
 - Reed, John, 437
 - Repast platform, 440
 - Reynolds, Craig, 444
 - Rosenblatt, Frank, 436
 - Sandpile model, 431
 - Schelling, Thomas, 432
 - SDM (System Dynamics Modeling), 432
 - Segregation model, 432
 - separation rule, 444
 - seven bridges of Königsberg problem, 432–433
 - SIMD (Single Instruction, Multiple Data), 442
 - Simon, Herbert, 437
 - Simula language, 440
 - Smalltalk language, 440
 - SOC (self-organizing criticality), 431
 - social sciences, 432
 - StarLisp language, 442
 - Swarm toolkit, 437, 440
 - Thinking Machines company, 442–443
 - Tucker, Lew, 443
 - turtle objects, 439
 - Ulam, Stanislaw, 433
 - von Neumann, John, 433–435
 - Watts and Strogatz small-worlds network models, 433
 - Wolfram, Stephen, 434, 435, 443
- Computer graphics, computational roots of ABM, 443–445
- Computer languages, replication factor, 338
- “Computer Recreations,” 68
- Computer science, modeling, 423
- Conceptual ABMs
 - correspondence to implemented model (*see* Verification)
 - definition, 22
 - describing, 314–315
- Connected Chemistry application, 127–128
- Connection machines (CM), 442
- Connections-per-node, effects of, 297–299
- Conservation law of money, 93
- Continuous spaces, 238–239
- Controller. *See* Observer
- Conway, John Horton, 45, 434
- Cook, Matthew, 68
- Copper aggregates, modeling, 119–121
- Copying
 - agents, 210
 - models (*see* Replication)
- Correctness of a model. *See* Replication; Validation; Verification
- CREATE-NETWORK command, 410
- CREATE-PREFERENTIAL-ATTACHMENT command, 411
- CREATE-RANDOM-NETWORK command, 410
- CREATE-TURTLES command, 71
- Creating
 - ABMs (*see also specific models*)
 - communication between model implementers and model authors, 313
 - describing conceptual models, 314–315
 - flowcharts, 314
 - pseudo-code, 314–315
 - UML (Unified Modeling Language), 315
 - Agents (*see* Heroes and Cowards model; Life model; Simple Economy model; Wolf-Sheep Simple model)
 - agentsets, 213–214
 - GO buttons (*see* GO buttons, examples)
 - GO procedures (*see* GO procedures, examples)
 - grass, 180–181
 - HubNet models, 363–369
 - NetLogo commands, 109
 - NetLogo reporters, 109
 - SETUP buttons, 54
 - SETUP procedures (*see* SETUP procedure, examples)
- Critical threshold, 103. *See also* Fire model
- Cruising model, 407
- C-Star language, computational roots of ABM, 442
- Cybernetics, computational roots of ABM, 432
- Dahl, Ole-Johan, 440
- Data analysis, 192
- Data inconsistencies, analyzing, 286–287
- Data measures, selecting, 169
- Data mining, extracting rules, 358
- Data parallelism, computational roots of ABM, 442–443
- Data sources
 - advanced mathematical analysis
 - Mathematica™, 415–417
 - Mathematica Link, 416–417
 - MatLab™, 415–416
 - GIS (Geographical Information Systems), 402–407
 - NetLogo Networks extension
 - CREATE-NETWORK command, 410
 - CREATE-PREFERENTIAL-ATTACHMENT command, 411
 - CREATE-RANDOM-NETWORK command, 410
 - importing data, 406, 408–409
 - Network extension toolkit, 409
 - Simple Viral Marketing example, 409–411
 - specifying data to import, 412–413
 - physical sensor data, 414–415
 - SNA (social network analysis), 406–415
- Data structures. *See* Agentsets; Lists; TASKS
- Dawkins, Richard, 85
- DC (deterministic-centralized) mindset, 10, 12
- Death procedures, 51, 166, 167, 258
- Debugging, 312
- Decision trees, 358
- Determinism vs. randomness, 34–35

- Dewdney, A. K., 68
Dialogues Concerning Two New Sciences, 5
 DIE command, 210
 Differential understanding, 7, 8–9
 Diffusion of innovation model, 300. *See also* Spread of Disease model
 Direction of agents. *See* HEADING property
 Discrete space
 description, 235–236
 mapping to continuous, 239
 Disease model, 360–362
 Distributional equivalence, 339
 Diversity thresholds, Segregation model, 136–137
 Diversity-seeking individuals, Segregation model, 137–140
 DLA (diffusion-limited aggregation) model
 advanced modeling applications, 127–128
 copper aggregates, 119–121
 description, 119–121
 extension 1: probabilistic sticking, 121–122
 extension 2: neighbor influence, 121–125
 GO procedure, 120–121, 123–124
 overview, 118–119
 SETUP procedure, 125–127
 source for, 120
 summary of, 127
 Document retrieval, computational roots of ABM, 443
 Documentation. *See also* Books and publications
 annotating models, 265
 commenting your code, 172
 definition, 271
 Info tab, 273–274
 NetLogo, 15
 organization, 274
 Dot, final behavior, 78, 83
 Dragulescu, A., 96
 Duplicating ABMs. *See* Replication
- EAT procedure, 181
 EBM (equation-based modeling, equational modeling)
 vs. ABM (agent-based modeling), 2, 17–18, 32–34, 36–37, 162
 free parameters, 37
 Lotka-Volterra differential equations, 15
 in phenomena-based modeling, 159
 and validation, 331
 EC (evolutionary computation), 389–390
 ECHO models, computational roots of ABM, 438
 Ecology, modeling. *See* Ant model; Fire model; Wolf-Sheep Simple model
 Economics, modeling. *See also* El Farol model;
 Simple Economy model
 Artificial Stock Market, 150
 Minority Game, 150
 Oil Cartel model, 150
 overview, 421–422
 participatory simulations, 360
 Root Beer Supply Chain, 150
 stock market simulation, 150
 Econophysics, 96
 Edge density of disease trails, 303–304
 Education
 Chicago Public Schools model, 407
 modeling, 371–372
 participatory simulations, 360
 through agent-based models, 30
 Einstein, Albert, 160
 El Farol Bar problem, computational roots of ABM, 437
 El Farol model. *See also* Minority Game
 advanced modeling applications, 150–151
 average min and max rewards, 145–146
 color-coding agents, 143–145
 description, 141–143
 histograms for reward values, 146–149
 overview, 141
 source for, 142
 summary of, 149–150
 Elaborated Realistic Models (ER), 353–355
 Electromagnetism, modeling, 128
 Element behavior, determining from known aggregate patterns. *See* Differential understanding
 Embracing Complexity conference, 68
 Emergence
 and complex systems, 5–13
 definition, 6
 emergent phenomena, 29
Emergence, 438
 Emergent behavior vs. bugs, 317–321, 325
 Emergent phenomena
 definition, 6
 identifying rules for, 29
 levels of, 12–13
 obstacle to thinking about, 12
 self-organizing patterns, 102
 thinking in levels, 12–13
 Empirical data, extracting rules, 357–360
 Empirical validation, 325, 331–334
 Empty agentsets, 215
 END1 property, 206
 END2 property, 206
 Energy variable, 174–177
 ENERGY-GAIN-FROM-GRASS parameter, 183–184
 ENERGY-GAIN-FROM-SHEEP parameter, 188–189
 Engineering
 computational roots of ABM, 431–432
 modeling, 422–423
 Environmental data, analyzing, 301–304
 Environmental decay, effects on spread of disease, 301–302
 Environmental patterns in spread of disease, 304
 Environment-environment interactions, 261

- Environments
 - 3D worlds, 248–252
 - boundary conditions, 240–241
 - bounded topology, 240–241
 - continuous spaces, 238–239
 - discrete spaces, 235–236
 - GIS (Geographical Information Systems), 252–257
 - grid graphs, 235–236
 - hex lattices, 237–238
 - implementing, 234–235
 - infinite-plane topology, 241
 - interaction topologies, 247–248, 252–253
 - lattice graphs, 235–236
 - mesh graphs, 235–236
 - Moore neighborhoods, 236–237
 - network-based
 - Erdős-Rényi network, 245
 - examples, 242–243
 - links, 241
 - nodes, 241
 - preferential attachment, 245–246
 - random networks, 244–245
 - scale-free networks, 245–246
 - small-world networks, 246–247
 - patches, 235–236
 - spatial, 235
 - square lattices, 236–237
 - topologies, 240
 - toroidal topology, 240
 - triangle lattices, 235
 - von Neumann neighborhoods, 236–237
 - wrapping, 240–241
- Environment-self interactions, 258–259
- Epidemiology, modeling, 117–118. *See also* Spread of Disease model
- Epstein, Josh, 87–88
- Equation-based modeling (EBM). *See* EBM (equational modeling)
- ER (Elaborated Realistic Models), 353–355
- Erdős-Rényi network, 245
- Erosion, modeling, 335
- Ethnicities, adding to Segregation model, 134–135
- Ethnocentrism model, 266–267, 336–337
- Evolution, modeling, 195
- Evolutionary algorithms, 395–398
- Evolutionary computation (EC), 389–390
- Examining models. *See* Analyzing models
- Examples. *See also specific models*
 - American Express credit card applications, 232
 - determining aggregate patterns from known individual behavior (*see* Integrative understanding)
 - determining individual element behavior from known aggregate patterns (*see* Differential understanding)
 - differential understanding, 7, 8–9
 - flocking of birds, 8–9, 10, 11–12 (*see also* V formation of goose flocks)
 - forest fires, 18–20
 - integrative understanding, 7–8
 - Italian shopkeepers, 4
 - ML (Machine Learning), 395–398
 - predator-prey interactions, 15–18
 - pulsating circle, 8, 9
 - Roman to Hindu-Arabic numeracy transparency, 3–5
 - slime molds, 13
 - time series, 295–296
 - traffic jams, 9, 12 (*see also* Traffic Basic model)
 - V formation of goose flocks, 8–9, 10, 13
 - voting behavior, 312–317
 - wolf and moose populations, 163–164
- Excusing people from prejudice, 129
- Experimentation through agent-based models, 30
- Experiments, naming, 289–290, 292
- Explanation through agent-based models, 29–30
- Exploratory modeling, 129, 159. *See also* Segregation model
- Extending ABMs. *See* DLA (diffusion-limited aggregation) model; El Farol model; Fire model; Segregation model
- EXTENSIONS command, 401–402
- Extracting rules
 - data mining, 358
 - decision trees, 358
 - from empirical data, 357–360
 - ML (Machine Learning), 357
 - overview, 356
 - from participatory simulations, 360–363
- FACE command, 263
- Face validation, 325, 331–334
- FACEXYZ command, 263
- Feynman, Richard, 443
- Fibonacci, 4
- File-based controls methods, 265
- FILTER primitive, 391–393
- Fire model
 - advanced modeling applications, 117–118
 - control parameters, 104–105
 - description, 104–110
 - epidemiology, modeling, 117–118
 - extension 1: probabilistic transitions, 110–112
 - extension 2: adding wind, 112–115
 - extension 3: long-distance transmission, 115–116
 - forest density, 104–105
 - GO procedure, 107–108
 - initializing, 105–106
 - percolation, modeling, 117–118
 - REPORT primitive, 108
 - report procedures, 108–109
 - self-organizing criticality, 104
 - SETUP procedure, 105–107
 - source for, 104
 - summary of, 116–117

- tipping point, 103–104, 114–115
- TO-REPORT primitive, 108
- First computational agent, 439
- First-class functions, 400. *See also* TASKS
- Fish ladders, effects on salmon populations, 195
- Fish populations, modeling, 195
- Floating-point numbers, 180
- Flocking model
 - alignment rule, 327
 - cohesion rule, 327–328
 - rules of bird movement, 327
 - separation rule, 327–328
 - V formation of goose flocks, 8–9, 10, 13
- Flocking model, examples of
 - ABM, 8–9, 10, 11–12
 - validation, 326–328
- Flowcharts, 314
- Fluid flow application, 443
- Focal objects for scientific dialog, 31
- FOLLOW command, 263
- Food webs, modeling, 195–196
- Foraging model, 22–27
- FOREACH command, 216
- Forest density, Fire model, 104–105
- Forest fire model. *See* Fire model
- Forever button, 264
- Forrest, Stephanie, 438
- FORWARD command, 210
- Forward movement of agents, 262
- Fractals, modeling. *See* DLA (diffusion-limited aggregation) model
- Fratelli theater group, 68
- Fredkin, Ed, 67
- Free parameters, 37
- Frequency of sampling values during replication, 340
- Friends and Enemies. *See* Heroes and Cowards model
- Frozen, final behavior, 78, 82
- Full Spectrum Modeling, 353–355

- GA (genetic algorithms), computational roots of
 - ABM, 435–438
- Galileo, 4–5
- Game models. *See* Heroes and Cowards model; Life model
- Game of Life. *See also* Life model
 - computational roots of ABM, 434
 - history of, 45
 - simulating itself, 65
 - as Turing machine, 65
- Game theory models, computational roots of ABM, 432
- Garbage Can Model of Organizations, computational roots of ABM, 437
- Gardner, Martin, 45, 434
- GIS (Geographical Information Systems), 252–257, 304, 402–407
- Glass-box approach, 33

- Glider guns, 65
- Global variables, 58
- GO buttons, examples
 - Heroes and Cowards model, 74
 - Life model, 57
- GO procedures, examples
 - DLA (diffusion-limited aggregation) model, 120–121, 123–124
 - Fire model, 107–108
 - Heroes and Cowards model, 74
 - Life model, 55–60
 - Schedule, 269–270
 - Segregation model, 133
 - Simple Economy model, 91–93
 - Wolf-Sheep Simple model, 172–173
- Goal-based agent cognition, 227–231
- Gosper, Bill, 65
- Goucher, Adam, 434
- Grand Canyon model, 253–257, 406
- Granularity of agents, 222–224
- Graphical components, agent-based models, 14
- Graphs. *See* Plotting
- Grass
 - choosing agents for, 166
 - color coding, 182–183
 - creating, 180–181
 - growth/regrowth, 167–168
 - regrowing, 181–183
 - sheep eating, 181
- GRASS-REGROWTH-RATE parameter, 183
- Grid graphs, 235–236
- Grid size, changing, 49
- Groupthink, 343
- Growing Artificial Societies: Social Science from the Bottom Up*, 88
- Guerin, Stephen, 69
- Guns, 65

- Hardware, replication factor, 338
- HATCH command, 210
- Heading, of agents, 165–166
- HEADING property, 206, 208
- Hebb, Donald, 435–436
- Heroes and Cowards model
 - brave behavior, 78–80, 84
 - Chinese streamers, 85–86
 - cowardly behavior, 78–80, 84
 - dot, final behavior, 78, 83
 - final behaviors, 76, 78, 85–86
 - frozen, final behavior, 78, 82
 - game rules, 68–69
 - GO button, 74
 - GO procedure, 74
 - inspectors, 74–75, 77
 - links, 74
 - personalities, setting, 74–80
 - pseudo-code, 79

- Heroes and Cowards model (cont.)
 - random number generation, 76, 81
 - SETUP procedure, 69, 71–73
 - slinky, final behavior, 78, 82
 - spiral, final behavior, 78, 83
 - strategies, 70
 - turtle monitors, 74–75, 77
 - user interface, 73–78
- Heterogeneous agents, 35–36
- HEV-M (Human, Embedded and Virtual agents in Mediation), 372–375, 376–377
- Hewitt, Carl, 441
- Hex lattices, 237–238
- Hidden Order*, 438
- HIDDEN? property, 206
- Higher-order operators, 391–401
- Hillis, Danny, 442
- HISTOGRAM command, 88, 147
- Histograms, 88, 146–149. *See also* Plotting
- History of ABM. *See* Computational roots of ABM
- Holland, John H., 434, 435–438
- Housing choices, modeling. *See* Segregation model
- HubNet models
 - creating, 363–369
 - description, 150
 - Disease model, 360–362
- Human, Embedded and Virtual agents in Mediation (HEV-M), 372–375, 376–377
- Human agents, 372–375
- Human networks, computational roots of ABM, 433
- Hybrid computational models
 - ABM cycle, 388–389
 - EC (evolutionary computation), 389–390
 - integrated cycle, 388–389
 - Journey to El Yunque project, 385
 - ML (Machine Learning) and ABM, 386, 388–389
 - ML cycle, 388–390
 - SDM (System Dynamics Modeling) and ABM, 383–386
 - Tabonuco-Yagrumo Hybrid model, 385, 387
 - Wolf-Sheep Predation Docked model, 385
- IBM methodology, computational roots of ABM, 431, 441
- Ideas, modeling the spreading of, 285. *See also* Spread of Disease model
- Implicit fitness function, computational roots of ABM, 438
- Importing data from social networks, 406, 408–409
- Individual-based modeling, 193–195, 431
- Infinite-plane topology, 241
- Info tab, 48, 273–274
- Initializing
 - Fire model, 105–106
 - properties, 209
 - sheep energy, 174–177
 - traffic speed, 208–209
 - wolf energy, 186–189
- Input boxes, 265
- Input controls, 265
- Inspecting properties, 206–207
- Inspectors, turtle monitors, 74–75, 77
- Integrated cycle, hybrid computational models, 388–389
- Integrative understanding, example, 7–8
- Interaction topologies, 247–248, 252–253
- Interactions. *See also* Observer/User Interface
 - agent-agent, 259–260
 - agent-environment, 261–262
 - agent-self, 257–258
 - birth events, 257
 - death procedures, 166, 167, 258
 - definition, 203–204
 - environment-environment, 261
 - environment-self, 258–259
 - speed regulation, 258
 - between the user and the model, 204
- Interactive running, 267–268
- Interface, definition, 271
- Interface controls, 265. *See also* Observer/User Interface
- Interface tab, 48, 274–275
- Internal states of agents. *See* Properties
- Invariant results, 334–335
- Ising models, computational roots of ABM, 431
- Italian shopkeepers, example, 4
- Iterating through a list, 216
- Iterative Modeling, 355–356
- Journey to El Yunque project, 385
- Kay, Alan, 440
- Kayenta Ancestral Pueblo, 333
- Kill command. *See* CLEAR-TURTLES command
- Kolmogorov-Smirnov test, 324
- LABEL property, 206
- LABEL-COLOR property, 206
- Lambdas. *See* TASKS
- Landscape metrics, 303–304
- Langton, Chris, 445
- Language change, modeling, 285. *See also* Spread of Disease model
- Languages, replication factor, 338
- Lattice graphs, 235–236
- LEFT command, 210
- Level of complexity of agents, 222–224
- Lewes, G. H., 6
- Life model
 - agent-centric thinking, 54
 - cells
 - changing color, 60
 - live, monitoring, 60

- cellular automata, 65–68
- clock
 - advancing, 55–60
 - resetting, 54
- configuring settings, 48, 50, 51
- documentation, 48
- game rules, 45–48
- glider guns, 65
- GO button, creating, 57
- GO procedure
 - creating, 55–60
 - ending, 56
- grid size, changing, 49
- guns, 65
- Model Settings dialog, 48, 50, 51
- NetLogo elements, 48–49
- Oscillators, 63
- patches
 - agent monitors, 60–61
 - color, setting, 51, 52
 - as game cells, 50
 - inspectors, 60–61
 - issuing commands, 52
 - making requests, 52
 - neighbors, 56–57
- sample trajectories, 62
- SETUP button, creating, 54
- SETUP procedure, creating, 50–55
- Spaceships, 63–65
- stable patterns, 60, 63
- Still Lives, 60, 63
- user interface, 48
- the view
 - clearing, 51
 - definition, 48
 - updating on each tick, 57
- wrapping, 49, 52
- Links
 - agentsets, 92
 - definition, 241
 - network-based environments, 241
 - properties, 206
 - between turtles, 74
- Lisp language, computational roots of ABM, 440
- Lists
 - vs. agentsets, 214–217
 - definition, 92
 - working with, 391–393
- Logo, computational roots of ABM, 439–440
- Long-distance fire transmission, Fire model, 115–116
- Looping. *See* Iterating
- Lotka-Volterra equations, 15, 193–195
- Machine Learning (ML). *See* ML (Machine Learning)
- Macro (aggregate) behavior, Ant model, 27–28
- Macrobehavior derived from micromotives, 129. *See also* Segregation model
- Macro-empirical validation, 332, 334
- Macro-face validation, 332, 334
- Macrovalidation, 325, 329–331
- The Magic Machine*, 68
- MAP primitive (reporter), 391–393
- Math, modeling, 423
- Mathematica™, 415–417
- Mathematica Link, 416–417
- Mathematical analysis, advanced. *See also* Statistical analysis
 - Mathematica™, 415–417
 - Mathematica Link, 416–417
 - MatLab™, 415–416
 - MatLab™, 415–416
- max/min rewards, El Farol model, 145–146
- Maxwell-Boltzmann distribution, 95
- McCarthy, John, 440
- Medicine, modeling
 - epidemiology, 117–118 (*see also* Spread of Disease model)
 - overview, 421
- Mesh graphs, 235–236
- Meta-agents, 232–233
- Methods of agents. *See* Actions
- Micro-empirical validation, 332, 334
- Micro-face validation, 332, 334
- Micromotives and Macrobehavior*, 432
- Microvalidation, 325, 329–331
- Milgram, Stanley, 300, 433
- min/max rewards, El Farol model, 145–146
- Minority Game, 150. *See also* El Farol model
- Minsky, Marvin, 439
- Mitchell, Melanie, 438
- ML (Machine Learning)
 - and ABM, 386, 388–389
 - agent cognition, 232
 - example, 395–398
 - extracting rules, 357
- ML cycle, hybrid computational models, 388–390
- Model clock, incrementing, 173
- Model design guidelines
 - ER (Elaborated Realistic Models), 353–355
 - Full Spectrum Modeling, 353–355
 - Iterative Modeling, 355–356
 - POM (pattern-oriented modeling), 353–355
- Model implementation, 22
 - correspondence to conceptual model (*see* Verification)
 - validity for real world phenomena (*see* Validation)
- Model output, 264–265
- Model Settings dialog, 48, 50, 51
- Model visualization, 265–267
- Modeler. *See* Observer

- Models. *See also* ABMs (agent-based models); *specific models*
 computational models, definition, 22
 conceptual models, 22
 equational models (EBM), 2, 17–18, 32, 37, 159, 162, 331
 hybrid models, 388–389
 systems dynamics models, 383–386
 textual models, 21–22
- Monetary wealth, modeling. *See* Simple Economy model
- Monitors, 265
- Monte Carlo simulation, computational roots of
 ABM, 432
- Moore, Edward F., 236
- Moore neighborhoods, 56, 236–237
- MOVE procedure, 173, 176–177, 179
- Movement of agents
 3-D, 248–249
 agent-environment interaction, 262
 backward/forward, 262
 movement cost, 174–179
 randomized, 173–174
 rules of bird movement, 327
 turtles, 210
 Wolf Sheep Simple model, 167
- Multiple runs, 191–192, 288–291
- Naming conventions, NetLogo commands, 109
- Naming experiments, 289–290, 292
- Nano-sheep problem, 194–195
- Nano-wolf problem, 32
- Natural sciences, modeling, 421
- Neighbor influence, in DLA model, 121–125
- Neighborhood Tipping model, 129
- NEIGHBORS primitive, 56
- NEIGHBORS4 primitive, 56
- NetLogo
 advanced concepts
 data structures, 395–398
 evolutionary algorithms, 395–398
 FILTER primitive, 391–393
 higher-order operators, 391–401
 MAP primitive, 391–393
 ML (Machine Learning) example, 395–398
 REDUCE primitive, 391–393
 RUN primitive, 393–395
 running lists of commands, 393–395
 RUNRESULT primitive, 393–395
 storing code for later use, 398, 400–401
 TASKS, 400–401
 working with lists, 391–393
 command blocks, color coding, 109
 commands
 color coding, 109
 creating your own, 109
 definition, 109
 naming conventions, 109
 continuous space vs. discrete, specifying, 239
 developer of, 14
 documentation, 15
 elements of
 actions, 209–210
 agents, 15 (*see also* Turtles)
 Code tab, 48
 Info tab, 48
 Interface tab, 48
 predefined actions, 209
 turtles, 15
 primitives, definition, 109
 reporter blocks, color coding, 109
 reporters
 color coding, 109
 creating your own, 109
 definition, 109
 source for, 14
 NetLogo 2D, 3-D Perspective, 263
 NetLogo 3D, 248
 NetLogo Controlling API, 401–402
 NetLogo Extension API, 401–402
 NetLogo GIS Extension, 401–402
 NetLogo Networks extension
 CREATE-NETWORK command, 410
 CREATE-PREFERENTIAL-ATTACHMENT command, 411
 CREATE-RANDOM-NETWORK command, 410
 importing data, 406, 408–409
 Network extension toolkit, 409
 Simple Viral Marketing example, 409–411
 specifying data to import, 412–413
 NetLogo turtles. *See* Turtles
 NetLogoLab environment
 Arduino extension, 379–383
 bifocal modeling, 383–384
 description, 375
 GoGo Board, 375, 378
 overview, 377–383
 Network extension toolkit, 409
 Network science, 241
 Network theory, computational roots of ABM, 432–433
 Network-based environments
 Erdős-Rényi network, 245
 examples, 242–243
 links, 241
 nodes, 241
 preferential attachment, 245–246
 random networks, 244–245
 scale-free networks, 245–246
 small-world networks, 246–247
A New Kind of Science, 68
 NEW-SEED command, 81
 NIELS application, 128

- Nodes, 241
- Notes, 265
- Notes, in models, 265
- No-turtles agentsets, 215
- Numerical identity, 339
- Nygaard, Kristen, 440

- Object recognition, computational roots of ABM, 443
- Object-oriented programming, computational roots of
 - ABM, 440–442
- Observer, 262
- Observer buttons, 263
- Observer/User Interface. *See also* Schedule
 - 3D perspective, 263
 - agent buttons, 263
 - batch running, 268
 - batch *vs.* interactive, 267–268
 - buttons, 264
 - centering on an agent, 263
 - choosers, 265
 - command center, 264
 - forever button, 264
 - input boxes, 265
 - input controls, 265
 - interactive running, 267–268
 - interface, definition, 271 (*see also* Interface tab)
 - interface controls, 265
 - model output, 264–265
 - model visualization, 265–267
 - monitors, 265
 - notes, 265
 - observer, 262
 - observer buttons, 263
 - output boxes, 265
 - output controls, 265
 - plots, 265
 - re-executing commands, 264
 - sliders, 265
 - switches, 265
 - user input, 264–265
- Oil Cartel model, 150
- ONE-OF primitive, 72
- Optimal behavior, Ant model, 28
- Organization of Behavior*, 436
- Organizations, modeling, 422. *See also* El Farol
 - model; Heroes and Cowards model
- Original models, 337–338
- Oscillators, 63
- Oscillatory patterns, 164, 192
- Output boxes, 265
- Output controls, 265

- Papert, Seymour, 370, 439–440
- Parallel actions *vs.* sequential, 270–271
- Parameter sweeping, 191–192
- Pareto distribution, 95
- Participatory Simulations, 150
- Participatory simulations, extracting rules,
 - 360–363
- Particle systems, computational roots of ABM,
 - 443–445
- Patches
 - agent monitors, 60–61
 - agentsets, 92
 - color (*see* PCOLOR property)
 - color, setting, 51, 52
 - as game cells, 50
 - inspectors, 60–61
 - issuing commands, 53
- Life model
 - agent monitors, 60–61
 - color, setting, 51, 52
 - as game cells, 50
 - inspectors, 60–61
 - issuing commands, 52
 - making requests, 52
 - neighbors, 56–57
 - making requests, 53
 - moving, 211
 - neighbors, 56–57
 - properties, 206
 - representing discrete spaces, 235–236
 - setting colors, 172
 - vs.* turtles, 211
- Path dependency, 30
- Pattern-oriented modeling (POM), 332, 334,
 - 353–355
- Patterns in nature, modeling. *See* Cellular automata;
 - DLA (diffusion-limited aggregation) model
- Patterns of infection, 302–304
- PCOLOR property, 206
- PEN-MODE property, 206
- PEN-SIZE property, 206
- Perceptrons, computational roots of ABM, 436
- Percolation, modeling, 117–118
- Percolation model, 2D. *See also* Fire model
 - vs.* 3D version, 252
 - code sample, 250
- Percolation model, 3D
 - vs.* 2D version, 252
 - code sample, 251
 - example, 249
- Percolation phenomena, 104. *See also* Fire model
- Performance, improving, 418–419
- Persuasion, modeling, 370–371
- Phase transition. *See* Tipping point
- Phenomena-based modeling. *See also* Segregation
 - model
 - definition, 128
 - in equation-based modeling, 159
 - goal of, 159
 - reference patterns, 128–129
- Pheromone hypothesis, 24
- Physical sensor data, 414–415

- Physics
 - computational roots of ABM, 431
 - modeling, 421
- Pictures generated from models, 420
- Pitch property (3D), 248, 250
- PLABEL property, 206
- PLABEL-COLOR property, 206
- PLOT widget, 88–89
- Plotting
 - code-based method, 177–178
 - for data analysis, 192
 - histograms, 88, 146–149
 - in model analysis, 291–296
 - output controls, 264
 - programmatic method, 177–179
 - sheep population, 177–179
 - widget-based method, 177–178
- Wolf-Sheep Simple data
 - analyzing results, 192
 - sheep population, 177–179
 - wolf population, 189
- Point representations, computational roots of ABM, 444
- Policy flight simulators, modeling, 371
- Political science, participatory simulations, 360
- Politics, modeling, 422
- POM (pattern-oriented modeling), 332, 334, 353–355
- Population density, effects on spread of disease, 285, 289, 291
- Population fluctuations, modeling, 194–195
- Predator-prey interactions
 - advanced modeling applications, 195–196
 - example, 15–18
 - fish ladders, effects on salmon populations, 195
 - fish populations, modeling, 195
 - food webs, modeling, 195–196
 - individual-based modeling, 193–195
 - Lotka-Volterra equations, 193–195
 - modeling as a continuous distribution, 194–195
 - nano-sheep problem, 194–195
 - population fluctuations, modeling, 194–195
 - wolf and moose populations, 163–164
 - wolf and sheep populations (*see* Wolf-Sheep Simple model)
- Predefined actions, 209
- Predictions through agent-based models, 31
- Preferential attachment, 245–246
- Prejudice, excusing people from, 129
- Primitives, NetLogo, 109
- Probabilistic sticking, in DLA model, 121–122
- Probabilistic transitions, Fire model, 110–112
- Process engineering, computational roots of ABM, 431–432
- Properties. *See also specific properties*
 - agents, 205–209
 - changing during runtime, 209
 - initializing, 209
 - inspecting, 206–207 (*see also* Inspectors; Monitors)
 - links, 206
 - NetLogo turtles, 206
 - patches, 206
- Proto-agents, 234
- Pseudo-code, 79, 314–315
- Pseudo-random numbers, 81
- Pulsating circle, example, 8, 9
- PXCOR property, 206
- PYCOR property, 206
- Question mark (?)
 - Boolean variable, 123
 - variable, 216–217, 256
- Questions. *See* Research questions
- RANDOM command, 54
- Random networks, 244–245
- Random number generators (RNGs), 76, 81
- Random numbers
 - NEW-SEED command, 81
 - ONE-OF command, 72
 - pseudo-random numbers, 81
 - RANDOM command, 54
 - RANDOM-FLOAT primitive, 122
 - RNGs (random number generators), 76, 81
 - seeds, 81
- RANDOM-FLOAT primitive, 122
- Randomized agent movement, 173–174
- Randomness vs. determinism, 34–35
- Ray traced images, 420
- RECOLOR-GRASS procedure, 182–183
- REDUCE primitive, 391–393
- Reed, John, 437
- Reference patterns, 128–129, 159
- Reflexive agents, agent cognition, 225
- REGROW-GRASS procedure, 182
- Regrowing grass, 181–183
- Relational alignment, 340
- Repast platform, computational roots of ABM, 440
- Replicated models, 337–338
- Replication. *See also* Validation; Verification
 - Axelrod-Hammond Ethnocentrism model, 336–337
 - benefits of, 340–341
 - communication with the model author(s), 342
 - definition, 337
 - dimensions
 - algorithms, 339
 - authors, 339
 - computer languages, 338
 - hardware, 338
 - languages, 338
 - time, 338
 - toolkits, 338–339
 - frequency of sampling values, 340
 - groupthink, 343
 - original models, 337–338

- overview, 336–337
- recommendations for
 - model authors, 344–346
 - replicators, 341–344
- replicated models, 337–338
- RS (replication standards)
 - distributional equivalence, 339
 - numerical identity, 339
 - relational alignment, 340
- shared understanding, 341
- source code, 343
- success criteria, 339
- Replication standards (RS). *See* RS (replication standards)
- REPORT primitive, 108
- Report procedures, 108–109
- Reporter blocks, color coding, 109
- Reporters
 - color coding, 109
 - creating your own, 109
 - definition, 109
- REPRODUCE procedure, 186
- Reproduction among agents, 167, 185–186
- Research questions
 - choosing, 161–163
 - example, wolf and moose populations, 163–164
 - scope of, 159
 - stating formally, 162–163
 - suitability for our model, 162
- RESET-TICKS command, 54, 71, 72
- Restructuration, 4
- Results and observations, Ant model, 27–28
- Reynolds, Craig, 444
- RIDE command, 263
- RIGHT command, 210
- RNGs (random number generators), 76, 81
- Robotic agents, 372–375
- Robustness, 321–324
- Roll property (3D), 248, 251
- Roman to Hindu-Arabic numeracy transparency,
 - example, 3–5
- Root Beer Supply Chain, 150
- Roots of ABM. *See* Computational roots of ABM
- Rosenblatt, Frank, 436
- RS (replication standards)
 - distributional equivalence, 339
 - numerical identity, 339
 - relational alignment, 340
- Rules
 - of birds in flight, 327, 444
 - extracting
 - data mining, 358
 - decision trees, 358
 - from empirical data, 357–360
 - ML (Machine Learning), 357
 - overview, 356
 - from participatory simulations, 360–363
- RUN primitive, 393–395, 419
- RUNRESULT primitive, 393–395
- Sandpile model, computational roots of ABM, 431
- Scale of the model, 224
- Scale-free networks, 245–246
- Schedule. *See also* Observer/User Interface
 - asynchronous vs. synchronous updates, 270
 - description, 268
 - GO procedure, 269–270
 - sequential vs. parallel actions, 270–271
 - SETUP procedure, 268–269
 - simulated concurrency, 270–271
- Schelling, Thomas, 129, 432
- SDM (System Dynamics Modeling), 383–386, 432
- Seeds for random numbers, 81
- Segregation model
 - advanced urban modeling, 140
 - computational roots of ABM, 432
 - description, 131–133
 - extension 1: adding ethnicities, 134–135
 - extension 2: diverse thresholds, 136–137
 - extension 3: diversity-seeking individuals, 137–140
 - failure to reach equilibrium, 139
 - GO procedure, 133
 - history of, 128–131
 - SETUP procedure, 132
 - source for, 131
 - summary of, 140
- Self-destruction of agents, 210
- Self-organizing criticality, 104
- Semicolon (;), comment delimiter, 172
- Sensitivity analysis, 321–324
- Separation rule, 327–328, 444
- Sequential vs. parallel actions, 270–271
- SETUP button, creating, 54
- SETUP procedure, examples
 - DLA (diffusion-limited aggregation) model, 125–127
- Fire model, 105–107
- Heroes and Cowards model, 69, 71–73
- Life model, 50–55
- Schedule, 268–269
- Segregation model, 132
- Simple Economy model, 88–91
- Wolf-Sheep Simple model, 172
- SETXYZ command, 263
- Seven bridges of Königsberg problem, 432–433
- SHAPE property, 206
- Shared understanding, 341
- SIMD (Single Instruction, Multiple Data),
 - computational roots of ABM, 442
- Simon, Herbert, 437

- Simple Economy model
 - analyzing results, 93–96
 - Boltzmann-Gibbs law, 93, 95
 - configuring the view, 89–91
 - conservation law of money, 93
 - description, 87–88
 - GO procedure, 91–93
 - HISTOGRAM command, 88
 - Pareto distribution, 95
 - PLOT widget, 88–89
 - running the model, 93
 - SETUP procedure, 88–91
 - visualizing wealth, 90
- Simple Viral Marketing model, 406–415
- Simula language, computational roots of ABM, 440
- Simulated concurrency, 270–271
- Six Degrees*, 300
- SIZE property, 206
- Sliders, 265
- Slime molds, example, 13
- Slinky, final behavior, 78, 82
- Small World networks, 300
- Smalltalk language, computational roots of ABM, 440
- Small-worlds network models, 246–247, 433
- SNA (social network analysis), 296–300, 406–415
- SOC (self-organizing criticality), computational roots of ABM, 431
- Social psychology, participatory simulations, 360
- Social sciences, computational roots of ABM, 432
- Sociology, participatory simulations, 360
- Source code, for replication, 343
- Spaceships, 63–65
- Spatial environments, 235
- Speed, increasing, 418–419
- Spiral, final behavior, 78, 83
- Spread of Disease model
 - androids, 360
 - Disease HubNet model, 360–362
- Spread of Disease model, analyzing
 - batch experiment tool, 289
 - BehaviorSpace tool, 289–291, 292, 293
 - compression of data, 288
 - connections-per-node, effects of, 297–299
 - creating agents, 283–284
 - data inconsistencies, 286–287
 - edge density of disease trails, 303–304
 - effects of population density, 289, 291
 - environmental data, 301–304
 - environmental decay, effects of, 301–302
 - environmental patterns, 304
 - GIS (Geographical Information Systems), 304
 - graphs, 291–296
 - landscape metrics, 303–304
 - modeling other spreading effects, 285
 - naming experiments, 289–290, 292
 - necessity of multiple runs, 288–291
 - overview, 283–284
 - patterns of infection, 302–304
 - population density, effects of, 285
 - SNA (social network analysis), 296–300
 - statistical analysis, 287–288
 - summary, 305–307
 - time series analysis, 295–296, 297
 - tools for, 288
- Spread runs, 324
- Square lattices, 236–237
- StarLisp language, computational roots of ABM, 442
- State variables, agent-based models, 14
- Statistical analysis. *See also* Mathematical analysis,
 - advanced
 - invariant vs. variant results, 334–335
 - Kolmogorov-Smirnov test, 324
 - of models, 287–288 (*see also* Analyzing models)
 - Student's *t*-test, 324
 - verification, 324
- Still Lives, 60, 63
- Stochastic components, analyzing, 191–192
- Stock market simulation, 150
- Storing code for later use, 398, 400–401
- Structuration, 4
- Structurational inertia, 4
- Student's *t*-test, 324
- Subagents, 233
- SugarScape, 87–88. *See also* Simple Economy model
- Summarizing results data, 192
- Swarm toolkit, computational roots of ABM, 437, 440
- Swarms, modeling. *See* Heroes and Cowards model
- Switches, 265
- Swords and Shields. *See* Heroes and Cowards model
- Synchronous vs. asynchronous updates, 270
- System Dynamics Modeling (SDM), 383–386
- Tabonuco-Yagrumo Hybrid model, 385, 387
- TASKS, 400–401, 419
- Termites model, 271
- Terrain, specifying, 168
- Textual models, 22
- Theory of Self Reproducing Automata*, 434
- THICKNESS property, 206
- Thinking in levels, 12–13
- Thinking Machines company, computational roots of ABM, 442–443
- Thought experiments
 - from agent-based models, 31
 - Roman to Hindu-Arabic numeracy transparency, 3–5
- 3D, agent movement, 248–249
- 3D perspective, 263
- 3D worlds, 248–252
- TICK command, 173
- TIE-MODE property, 206
- Time, replication factor, 338

- Time series analysis, 295–296, 297
- Tipping point, 103–104, 114–115. *See also* Fire model
- Top-down modeling, 159–160
- Topologies
 - bounded, 240–241
 - infinite-plane, 241
 - interaction, 247–248, 252–253
 - toroidal, 240
 - wrapping, 166, 240–241
- TO-REPORT primitive, 108
- Toroidal topology, 240
- Tracking agents over time. *See* WHO property
- Tradeoffs of ABM, 36–38, 423–424
- Traffic Basic model
 - causes of traffic jams, 9, 12
 - controlling. *See* File-based controls methods; Observer/User Interface
 - history of, 204
 - model visualization, 266
- Traffic Basic model, agent cognition
 - adaptive agents, 227, 230
 - definition, 224–225
 - goal-based, 227–231
 - ML (Machine Learning), 232
 - reflexive agents, 225
 - utility-based, 225–227
- Traffic Basic model, agent collections
 - agentsets
 - and computation, 217–222
 - creating, 213–214
 - definition, 212
 - empty, 215
 - vs. lists, 214–217
 - no-turtles, 215
 - breeds of agents, 212
 - lists, 214–217
 - types of agents, 211–212
- Traffic Basic model, agents. *See also* Traffic Basic model, interactions
 - actions, 209–211
 - behaviors (*see* Actions)
 - birth events, 257
 - communicating with other agents, 260
 - composed of other agents (*see* Meta-agents)
 - composing other agents (*see* Subagents)
 - consuming other agents, 259
 - copying, 210
 - death procedures, 258
 - defining new actions, 209
 - definition, 203
 - granularity, 222–224
 - history of, 204
 - level of complexity, 222–224
 - meta-agents, 232–233
 - movement, 262
 - moving forward and backward, 210
 - moving left and right, 210
 - observing the environment, 261
 - overview, 205
 - as placeholders (*see* Proto-agents)
 - predefined actions, 209
 - properties, 205–209
 - property awareness (*see* Agent cognition)
 - proto-agents, 234
 - scale of the model, 224
 - self-destruction, 210
 - sensing other agents, 259–260
 - subagents, 233
- Traffic Basic model, cars
 - adjusting speed, 210–211
 - free-flow state, 231
 - initializing speed, 208–209
 - interactions between, 210–211
 - interactions with the environment, 224–230
 - modifying actions and strategies, 230
 - speed regulation, 258
- Traffic Basic model, environments. *See also* Traffic Basic model, interactions
 - 3D worlds, 248–252
 - boundary conditions, 240–241
 - bounded topology, 240–241
 - continuous spaces, 238–239
 - discrete spaces, 235–236
 - GIS (Geographical Information Systems), 252–257
 - grid graphs, 235–236
 - hex lattices, 237–238
 - implementing, 234–235
 - infinite-plane topologies, 241
 - interaction topologies, 247–248, 252–253
 - lattice graphs, 235–236
 - mesh graphs, 235–236
 - Moore neighborhoods, 236–237
 - network-based
 - Erdős-Rényi network, 245
 - examples, 242–243
 - links, 241
 - nodes, 241
 - preferential attachment, 245–246
 - random networks, 244–245
 - scale-free networks, 245–246
 - small-world networks, 246–247
 - patches, 235–236
 - spatial, 235
 - square lattices, 236–237
 - topologies, 240
 - toroidal topologies, 240
 - triangle lattices, 235
 - von Neumann neighborhoods, 236–237
 - wrapping, 240–241

- Traffic Basic model, interactions
 - agent-agent, 259–260
 - agent-environment, 261–262
 - agent-self, 257–258
 - birth events, 257
 - death procedures, 258
 - environment-environment, 261
 - environment-self, 258–259
 - speed regulation, 258
- Traffic Basic model, properties
 - agent awareness of (*see* Agent cognition)
 - agents, 205–209
 - changing during runtime, 209
 - initializing, 209
 - inspecting, 206–207
 - links, 206
 - NetLogo turtles, 206
 - patches, 206
- Traffic jams. *See* Traffic Basic model
- Treatments, 324
- Triangle lattices, 235
- Tucker, Lew, 443
- Turtle monitors, 74–75, 77
- Turtle objects, computational roots of ABM, 439
- Turtles
 - accessing patch properties, 206
 - actions, 209–210
 - agentsets, 92
 - CLEAR-TURTLES command, 258
 - color (*see* COLOR property)
 - color properties, 206
 - CREATE-TURTLES command, 71
 - definition, 15
 - empty agentsets, 215
 - links, 74
 - movement, 210
 - order of operations, 216
 - vs.* patches, 211
 - properties, inspecting, 74
- Turtles, all the way down, 233
- Ulam, Stanislaw, 433
- UML (Unified Modeling Language), 315
- Unique IDs for agents. *See* WHO property
- User input, 264–265
- User interface. *See also* Observer/User Interface
 - Heroes and Cowards model, 73–78
 - Life model, 48
- Utility-based agent cognition, 225–227
- V formation of goose flocks, modeling, 8–9, 10, 13.
 - See also* Flocking model
- Validation. *See also* Replication; Verification
 - benefits of, 335–336
 - definition, 325
 - empirical validation, 325, 331–334
 - face validation, 325, 331–334
 - macro-empirical validation, 332, 334
 - macro-face validation, 332, 334
 - macrovalidation, 325, 329–331
 - micro-empirical validation, 332, 334
 - micro-face validation, 332, 334
 - microvalidation, 325, 329–331
 - with participatory simulation, 363
 - pattern-oriented modeling, 332, 334
 - questions about, 335–336
 - validation spectrum, 331
- Validation spectrum, 331
- Validation testing
 - invariant results, 334–335
 - path dependence, 334–335
 - statistical analysis
 - invariant *vs.* variant results, 334–335
 - Kolmogorov-Smirnov test, 324
 - Student's *t*-test, 324
 - verification, 324
 - stochasticity, 334–335
 - variant results, 334–335
- Values, agent-based models, 14
- Variables
 - agent, 58
 - built into NetLogo, 58
 - defining your own, 58
 - global, 58
 - observing over time, 265
 - overview, 58
- Variant results, 334–335
- Venice model, 407
- Verification. *See also* Replication; Validation
 - benefits, 324–325
 - communication between model implementers and model authors, 313
 - debugging, 312
 - definition, 33, 161
 - describing conceptual models, 314–315
 - emergent behavior *vs.* bugs, 317–321, 325
 - issues, 324–325
 - Kolmogorov-Smirnov test, 324
 - robustness, 321–324
 - sensitivity analysis, 321–324
 - spread runs, 324
 - statistical analysis, 324
 - Student's *t*-test, 324
 - treatments, 324
 - voting behavior, example, 312–317
- Verification testing
 - ANT (Active Nonlinear Testing), 324
 - with BehaviorSpace, 323–324
 - online/offline unit testing, 317
 - overview, 315–317
 - unit testing, 317

- View, Life model
 - clearing, 51
 - definition, 48
 - updating on each tick, 57
- Virtual agents, 372–375
- Visualization, 265–267, 416, 418
- von Neumann, John
 - artificial reproducing machine, 433–435
 - computational roots of ABM, 433–435
 - Universal Constructor, 433–434
- von Neumann neighborhoods, 56, 236–237
- Voting behavior, example, 312–317
- WATCH command, 263
- Watts, Duncan, 300
- Watts and Strogatz small-worlds network models, 433
- Wealth gap, modeling. *See* Simple Economy model
- Weather, modeling. *See* Climate Change Game model
- WHO property, 206, 208
- WIGGLE command, 173
- Wilensky, Uri, 14
- Wind, Fire model, 112–115
- Wolf and moose populations, 163–164
- Wolfram, Stephen, 67–68, 434, 435, 443
- Wolf-Sheep Predation Docked model, 385
- Wolf-Sheep Simple model, analyzing
 - anomalous/aberrant behavior, 191
 - averaging across runs, 192
 - BehaviorSpace tool, 191
 - collecting results, 191–192
 - data analysis, 192
 - initial number of wolves, 192
 - multiple runs, 191–192
 - oscillatory patterns, 164, 192
 - overview, 189–191
 - parameter sweeping, 191–192
 - plotting data, 192
 - stochastic components, 191–192
 - summarizing data, 192
- Wolf-Sheep Simple model, building
 - version 1
 - assigning initial sheep properties, 172
 - binomial distributions, 173–174
 - clearing the world, 172
 - commenting your code, 172
 - controlling sheep and wolves, 171, 173
 - creating a sheep breed, 171
 - creating individual sheep, 172
 - GO procedure, 172–173
 - incrementing the model clock, 173
 - MOVE procedure, 173
 - randomized movement, 173–174
 - setting patch colors, 172
 - SETUP procedure, 172
 - sheep procedures, 173
 - version 2
 - energy variable, 174–177
 - initializing sheep energy, 174–177
 - MOVE procedure, 176–177, 179
 - movement cost, 174–179
 - plotting sheep population, 177–179
 - stop when no more sheep, 177
 - version 3
 - color-coding grass, 182–183
 - creating grass, 180–181
 - EAT procedure, 181
 - ENERGY-GAIN-FROM-GRASS parameter, 183–184
 - gaining energy for sheep, 180–185
 - GRASS-REGROWTH-RATE parameter, 183
 - RECOLOR-GRASS procedure, 182–183
 - REGROW-GRASS procedure, 182
 - regrowing grass, 181–183
 - sheep eating grass, 181
 - version 4
 - REPRODUCE procedure, 186
 - sheep reproduction, 185–186
 - version 5
 - creating wolves, 186–189
 - ENERGY-GAIN-FROM-SHEEP parameter, 188–189
 - gaining energy for wolves, 188–189
 - initializing wolf energy, 186–189
 - plotting wolf population, 189
 - specifying wolf behaviors, 187–189
- Wolf-Sheep Simple model, designing. *See also* Model design guidelines
 - ABM design principle, 160–161
 - from the bottom up, 159–160
 - choosing agents (wolves and sheep)
 - animal behaviors, 167
 - birth processes, 166
 - death, 166, 167
 - eating, 167
 - energy level, 165–166
 - environmental characteristics, 166
 - granularity, 164
 - heading, 165–166
 - initial population, specifying, 168
 - location, 165–166
 - movement, 167
 - properties for, 165–166
 - proto-agents, 165
 - reproduction, 167
 - setting up time steps, 167–168
 - stationary agents, 166
 - data measures, selecting, 169
 - design summary, 169–170
 - exploratory modeling, 159
 - exploring emerging patterns (*see* Exploratory modeling)

- Wolf-Sheep Simple model, designing (cont.)
 - grass
 - choosing agents for, 166
 - growth/regrowth, 167–168
 - model parameters, 168–169
 - phenomena-based modeling, 159
 - reference patterns, 159
 - research questions
 - choosing, 161–163
 - example, wolf and moose populations, 163–164
 - scope of, 159
 - stating formally, 162–163
 - suitability for our model, 162
 - from a target phenomenon to underlying causes
(*see* Phenomena-based modeling)
 - terrain, specifying, 168
 - from the top down, 159–160
 - verification, 161
 - wrapping topology, 166
- Wrapping environments
 - description, 52
 - Game of Life, 49
 - NetLogo default, 166
 - topologies, 240–241
- XCOR property, 206, 208
- Yakovenko, V. M., 96
- Yaw, 250
- YCOR property, 206, 208
- Zuse, Konrad, 65, 67