

# Index

## A

- A\* algorithm, 146–148, 164–166
- Absolute utility, 24
- Abstractions, perception and, 472–474
- Adaptation, 481–482
  - emotion-based, 490–492
  - skill-based, 488–490
  - style-based, 492–493
- Agent-driven parameter, 392
  - prone navigation, 401
  - sample structure, 393
  - sidestep navigation, 402–403
  - standing and crouching, 398
  - swim navigation, 401–402
- Agent forces, prototype implementation, 199–200
- Aggressive search, NPC, 306, 308
- AI agent, 389
  - movement path, 400
  - voxel-based navigation mesh/navmesh
    - (see Voxel-based navigation mesh/navmesh)
- AI chromosomes, 252
- AI-driven experience management, 523
  - choosing between stories, 525–526
  - high-level view, 527
  - implementations
    - PaSSAGE, 530–531
    - Player Appraisal Controlling Emotions, 531
    - player in game environment, 524–525
    - Player-Specific Automated Storytelling, 531
    - Sports Commentary Recommendation System, 531
  - modifying narrative, 525
  - technique
    - emotional modeling, 528–529
    - goal inference, 528
    - machine-learned narrative selection, 530
    - narrative generation, 526–527
    - objective function maximization, 529–530
    - play style modeling, 527
- AI Game Programming Wisdom 3*, 366, 381
- Alien Swarm, 491
- Ambience of life
  - buddy AI, *The Last of Us*, 432–433, 441
  - designing, 89–91
- Ambush skill, Infected characters, 413, 415–416
- An Experimental Study of Apparent Behavior*, 473
- Animation, 443
  - additive, 456–457
  - and AI, 7
  - behaviors, 109–112, 458–460
  - blackboard, 36–37
  - blending movement, 452–454
  - complexity explosion, 104–105
  - controller, 112–113
  - delay times, 37–38
  - full-body state machine, 109
  - functional testing, 114

- gameplay and, 107–108
- graphs, 102–104
- interrupting movement, 452–454
- layering, 455–456
- looping (*see* Looping animations)
- masking, 454–455
- mirroring, 455
- NPC realization behaviors (*see* Nonplayer characters (NPCs), behavior realization)
  - one-off (*see* One-off animations)
  - problem of scalability, 104–105
  - SoC, 106–107, 114–115
  - state machine, 104
  - system refactoring, 114–115
  - transition (*see* Transition animations)
  - twinning, 35–36
  - walk, 451–452
- Animation-driven movement, 444
- ANN, *see* Artificial neural network (ANN)
- Anticipation, 195–196
- Anticipatory collision
  - agent forces, 199–200
  - agent state, 196–197
  - avoidance force, 200–201
  - guaranteed collision avoidance, 204–207
  - human motion simulation, 203–204
  - parameter tuning, 202–203
  - predicting, 197–199
  - runtime performance, 202
  - time horizon, 198–199
- Any-angle pathfinding
  - any-angle paths, 162
  - discretize and search, 161
  - grid paths, 162
- A\* paths, 161, 163
- Archetypal analysis (AA), 487
- Archetypes, 487, 492
- Armed combat, buddy AI
  - balancing, 439–440
  - cheating, 440
  - shooting, 439
- Artificial evolution techniques, 489
- Artificial neural network (ANN), 484
- Assassin's Creed, 492
- Attribution theory, 475–476
- Auditory perception, *Tom Clancy's Splinter Cell Blacklist*
  - calculating audio distance, 320–321
  - fairness, 321–322
- Aurora Neverwinter Toolset*, 486, 493
- Autonomous formation, 220
- Avoidance force
  - code, 201
  - computing, 201
  - corner cases, 201
  - direction, 200
  - magnitude, 200–201
- B
- Backtracking, 295–296
- Backward checking, 286–287
- Balanced *k*-d tree, 332
- Balancing, buddy AI, 439–440
- Baldur's Gate II*, 145–146
- Bandit algorithms, 266
- Base maps
  - structure, 349–350
  - type, 350
- Battlefield: Bad Company 2*, 487
- Battle of the Bulge* (iOS), 243–244, 253
- Bayesian networks, 485
- Behavioral Mathematics for Game AI*, 378
- Behavior realization, nonplayer characters, 443
  - action tokens, 458–460
  - blackboards, 460
  - combining actions, 454–456
  - contextual one-off animations, 458
  - decoupling extracted motion, 446
  - displacement direction and orientation, 446–449
  - micro behaviors, 458–459
  - motion correction, 446
  - move and shoot tokens, 459
  - movement models, 444–446
  - on-screen realization, 460
  - per-bone blending, 453–454
  - pose matching, 453
  - pose-only blending, 453–454
  - rank categories, 460
  - speed correction, 450–452
  - tracking, 456–457
  - using additives with idles, 459
- Behavior trees (BTs), 23
  - complexity, 73
  - simple, 71–72
- Bézier spline, 230
- Bit-wise operations, 299
- Blackwell Handbook of Childhood Cognitive Development*, 474

- 
- Blending movement, animations,
    - 452–454
  - Blend tree, 102–105, 107–108
  - Blind formation, 219
  - Boolean vision model, 28
  - Borderlands*, 511
  - Bottom-up approach, 512
  - Breadcrumbs, 328
  - Brute force, 285–286
  - Buddy AI, *The Last of Us*, 431
    - armed combat
      - balancing, 439–440
      - cheating, 440
      - shooting, 439
    - combat utility
      - gifting, 438
      - grapples, 438
      - throwing, 437–438
    - cover
      - runtime cover generation, 435–437
      - share, 437
    - finishing touches
      - ambience, 441
      - callouts, 441
      - vocalizations, 440
    - start from scratch
      - ambient following, 432–433
      - approach, 432
      - dodging, 434–435
      - follow positions, 433–434
      - moving, 434
      - plan, 432
      - teleportation, 435
  - C
  - Caesar*, 50
  - Callouts, buddy AI, 441
  - Canabalt*, 503–504, 510
  - Canonical Fortune’s algorithm, 331
  - Cardinality constraint, 296
  - Cascading elitism, 489–490
  - Cautious search, NPC, 305–308
  - Cellular automata, 335–336
  - Character behavior design
    - abstractions, 472–474
    - attribution theory, 475–476
    - perception of motion, 472–474
    - point light animations, 472–473
    - problem of characters as tokens,
      - 476–477
      - psychology
        - development, 474–475
        - practical application, 477
      - shape fitting, 472–473
      - stop-motion animation, 473
      - teleological stance, 474–475
  - Chase skill, Infected characters, 412, 415
  - Cheating, buddy AI, 440
  - Choco, procedural content generation, 515
  - Chokepoints, 373–375
  - Choreographed formations
    - autonomous formation following, 220
    - blind formation following, 219
    - formation design, 218–219
    - slots assignment, 219
  - Civilization V*, 509–511
  - Climb mesh generation, 227–230
  - Climb parser, 230
  - Climb path generation, 229–230
  - Clustering algorithms, 486–488
  - C-means, 486–487
  - Cognitive task reaction time, 33
  - Collaborative filtering (CF) algorithms,
    - 493–494
  - Collision avoidance algorithms, 174, 178
  - Combat Coordinator, 428–429
  - Combat utility, buddy AI
    - gifting, 438
    - grapples, 438
    - throwing, 437–438
  - Combat vector, 428–429
  - Communal analysis
    - clustering algorithms, 486–488
    - goal, 486
    - K-means clustering, 486–488
    - simplex volume maximization, 487–488
  - Company of Heroes*, 369
  - Compositional control, PCG, 510
  - Computer Shogi Association (CSA), 256
  - Congestion maps, 173–174
    - alternatives, 178
    - benefits, 179
    - drawbacks, 179
    - flow fields with, 177–178
    - hysteresis, 180
    - path planning, 175–177
    - performance considerations, 179–180
    - traversal cost computation, 176
  - Consistency, *Tom Clancy’s Splinter Cell*
    - Blacklist*, 314
-

- 
- Constrained optimization, 337
  - Constraint, 285
    - arcs, 301
    - class declaration, 297
    - equality, 289, 291
    - graphical representation, 288
    - inequality, 289, 292
    - at most, 298
    - propagated, 294
  - Constraint-driven methods, 506–507
  - Constraint programming, 283
    - ad hoc algorithm, 284
    - backtracking, 295
    - backward checking, 286–287
    - brute force, 285–286
    - constraint arcs, 301
    - example, 284–285
    - finite-domain representation, 299–301
    - forward checking, 287–290, 295
    - gory implementation, 295–299
    - inconsistencies, detecting, 290–294
    - randomized solutions, 301
    - undo, 295
    - variable ordering, 301
    - work queue, 301
  - Constraint satisfaction problems (CSPs), 285
  - Constructionist approach, 503–504
  - Content explosion, 14–15
  - Context, 32
  - Context Free Art, 515
  - Context steering behaviors, 183
    - avoid behavior, 187–188, 190
    - chase behavior, 187
    - combining and parsing, 188
    - coordinate system, 189
    - danger and interest map, 187
    - drafting behavior, 191
    - final map processing, 188
    - optimizations, 192–193
    - post-processing, 192
    - processing, 191–192
    - racing line behavior, 189–190
    - scalar values, 186
    - subslot calculations, 189
  - Contextual awareness, *Tom Clancy's Splinter Cell Blacklist*, 322, 324–325
  - Contextual one-off animations, 458
  - Counterattacks, 386–387
  - Cover, *The Last of Us*
    - buddy AI, 435–437
    - edge features, 435–437
    - human enemy AI, 424–426
    - post, 424
    - share, 437
  - CryENGINE, 308
  - Cul-de-sacs, 373–375
  - D
  - Data-driven design, *Infected characters*, 411
  - Debugging
    - Infected characters, The Last of Us*, 412
    - procedural content generation, 514
  - Delayed authoring, 534
  - Desert Fox*, 253
  - Diffusion process, 328
  - Direction maps (DMs), 175, 178
  - Direct manipulation, PCG, 510
  - Discretize pathfinding, 161
  - Distractions, *Infected characters*, 410
  - Distraction stimuli, 306
  - DMs, *see* Direction maps (DMs)
  - Dodging, buddy AI, 434–435
  - Domain, 285
  - Dragon Age: Origins*, 145
  - Drive on Moscow*, 253
  - Driver: San Francisco*, 85–86
  - Dual-utility reasoning
    - rank, 24–25
    - weight, 24–25
    - Zoo Tycoon 2*, 25–26
  - Dwarf Fortress*, 503, 511
    - break down and understand system, 520
    - don't overcomplicate, 520
    - real-world analog, 520
    - simulation principles, 519
  - E
  - 1849
    - benefits, 55
    - buildings, 50
    - condition checking frequency, 53–54
    - condition definition language, 54
    - conditions and actions, 51–52
    - data model, 54–55
    - game mechanics, 50
    - game simulation, 50
    - lessons, 55–56
    - related work, 56–57
-

- 
- resource bins, 51
  - resources, 50–51
  - rule execution, 52–53
  - workers, 50
  - Elite*, 501–502
  - Ellie's Buddy AI, *The Last of Us*, see Buddy AI, *The Last of Us*
  - Emotional modeling, 528–529
  - Emotion-based adaptation, 482
    - Alien Swarm, 491
    - finite-state machine, 491
    - The Journey to Wild Divine, 491
    - Left 4 Dead 2's, 491
    - The Open Racing Car Simulator, 492
    - skin conductance level, 491
    - survivor intensity, 490–491
  - EmotionFX*, 106
  - Ending search, NPC, 312
  - Endless Web*, 511
  - Engineering constraints, 513
  - Environmental awareness, *Tom Clancy's Splinter Cell Blacklist*
    - changes in objects, 319–320
    - connectivity, 317–319
  - Equality constraint, 289, 291, 297
  - Equilibrium distribution, 466
  - Euclidean distance, 146
  - Euro-style board war games, 244–245
  - Evolutionary algorithm, 505
  - Experiential control, PCG, 510
  - Exposure map, 420–421, 427–428
  - Extracting motion from animation, 444
- F**
- Falloff function model
    - partial derivatives, 340
    - point-based influence, 330–331
  - FEAR*
    - bark, 19–20
    - dialogue, 20
    - production budget, 20
  - Finite-domain CSPs, 285
  - Finite-domain representation, 299–301
  - Finite-state machine (FSM), 23, 491
  - First person shooter (FPS), 19, 483
  - Flanking
    - attacks, 385
    - human enemy AI, 428–429
  - Flocks, 210
  - Follow positions, buddy AI, 433–434
  - Follow skill, Infected characters, 415
  - Forced neighbors, 133–134
  - Formation adaptation, 221
  - Forward checking, 287–290, 295–296
  - Functional testing, 114
  - Fuse*
    - climb meshes, 228–229
    - climb mesh generation, 227–228
    - climb path generation, 229–230
    - parsing climb paths, 230
    - traversal setups, 226–227
    - virtual controller input, 230–232
- G**
- Galactic Arms Race*, 506, 508–509
  - Game AI
    - animation, 7
    - autonomy, 5–6
    - content explosion, 14–15
    - conversations, 9–10
    - definition, 4–5
    - design, 4
    - development, 6–7
    - dynamic storylines, 10–11
    - feedback, 4
    - frontiers of field, 8
    - fuzzy border, 5–6
    - interaction, 4
    - modeling and displaying emotion, 12–13
    - online games, 7
    - pathfinding, 8–9
    - player modeling, 11–12
    - replayability, 10
    - requirement, 4
    - scale, 13–14
    - social relationships, 13
    - unexplored, 15
    - usability testing, 11
    - vocabulary, 4
  - Game analytics, 482
    - adaptation, 482
      - emotion-based, 490–492
      - skill-based difficulty, 488–490
      - style-based, 492–493
    - communal analysis, 486–488
    - individual analysis, 483–486
    - recommendation systems, 482, 493–494
    - and team matching techniques, 494–495
  - Game design constraints, 512–513
  - Game-driven movement, 444
-

- 
- Game mechanics, 50
  - Game metrics, 483
  - Game simulation, 50
  - Game state, factorizing, 47–48
  - Game telemetry, 483
  - Gap detection, NPC searching, 311–312
  - Genetic algorithm, evolution and, 251–252
  - Gifting, buddy AI, 438
  - Goal inference, AI-driven experience management, 528
  - Goal-Oriented Action Planning (GOAP), 117
    - action predicates, 122, 125–126
    - actions, 121–122
    - competition tests, 119
    - data structures, 120–121
    - optimization, 118–120
    - plans, 122
    - runtime measurement, 118–119
    - runtime profiling, 123
    - scaling tests, 119
    - state predicates subsets, 123–125
    - states, 123
    - use of profile, 119–120
    - valuable tests, 119
  - Goal velocity, 197
  - GOAP, *see* Goal-Oriented Action Planning (GOAP)
  - Go/no-go time, 32–33
  - Google, procedural content generation, 515
  - Gory implementation, 295–299
  - Gradient descent
    - local minimum, 338
    - optimization method, 337
  - Grammar-based approach, 504–505
  - Graphical user interface (GUI), 240
  - Grapples, combat utility, 438
  - Greedy matching/selection, 63–64
  - Grids, 164
  - Group collision avoidance
    - formation adaptation, 221
    - velocity correction, 221
  - Group navigation
    - behaviors, 212–213
    - choreographed formations
      - autonomous formation following, 220
      - blind formation following, 219
      - formation design, 218–219
      - slots assignment, 219
    - emergent group structure
      - boids and derivatives, 217
      - implementing, 217–218
      - local formations, 217
    - flocks, 210
    - formations, 210–211
    - group collision avoidance
      - formation adaptation, 221
      - velocity correction, 221
    - members relationship model
      - hierarchical entity architecture, 215–216
      - leader, 214
      - virtual group entity, 214–215
    - pathfinding, 216–217
    - pipeline, 213–214
    - social groups, 211–212
  - GUI, *see* Graphical user interface (GUI)
- ## H
- HeBTs, *see* Hinted-execution Behavior Trees (HeBTs)
  - Herd'Em!*, 207–208
  - Hierarchical entity architecture, 215–216
  - Hinted-execution Behavior
    - Trees (HeBTs), 69
  - adaptation, 84–85
  - base behavior, 80–81
  - behavior controllers, 78
  - behavior trees
    - complexity, 73
    - simple, 71–72
  - concept, 73–74
  - conditions, 76–77
  - design/engineering collaboration
    - workflow, 70
  - group behaviors, 85–86
  - higher levels, 78–79
  - high-level tree, 82–83
  - multilevel architecture, 77–79
  - problem, explaining, 70–71
  - prototype idea, 80–82
  - results, analyzing, 83–84
  - selectors, 74–75
- Hostile stimuli, 306
- Human enemy AI, *The Last of Us*, 419
  - behaviors, 426
  - building blocks, 420–421
  - cover, 424–426
  - flanking, 428–429
-

- 
- lethality, 427–428
  - perception, 421–423
  - polish, 429
  - posts, 424–426
  - skills, 426
  - stealth, 426–427
  - Human in the loop approach, 505–506
  - Hunters, 407–410, 413–414
  - I
  - Idle animations, 459
  - The Illusion of Life*, 473
  - Inconsistencies, detecting, 290–294
  - Individual analysis
    - goals, 483–484
    - K-means clustering, 485
    - player modeling
      - artificial neural network, 484
      - Bayesian networks, 485
      - numeric weight vectors, 486
      - support vector machines, 484–485
  - Inequality constraint, 289, 292, 298
  - Infected-canvass behavior, 413
    - on-fire skill, 416
    - requirements, 413
    - selecting animation, 414
    - sleep skill, 417
  - Infected characters, *The Last of Us*, 407
    - ambush skill, 415–416
    - characteristics, 408
    - chase skill, 415
    - data-driven design, 411
    - debugging, 412
    - distractions, 410
    - follow skill, 415
    - implementation, 412
    - logical sounds, 418–419
    - move-to behavior, 410, 415
    - on-fire skill, 416
    - philosophy, 410–411
    - search skill, 413–415
    - senses, 408–409
    - skills, 412
    - sleep skill, 417
    - throw skill, 416
    - types, 415–416
    - vision type, 411
    - wander skill, 416–417
  - Infinite-resolution influence mapping,  
  *see* Influence map
  - Influence map, 327
    - breadcrumbing, 328
    - construction, 328
    - example implementation, 340
    - handling obstacles, 335–336
    - influence source density, 340
    - k-d tree
      - application, 333
      - construction, 333
      - left-hand branch, 332
      - range query, 333–334
      - right-hand branch, 332
      - splitting axis, 332–333
      - temporal propagation, 335
    - limitations, 329
    - modular tactical (*see* Modular tactical  
  influence maps)
    - nontrivial topologies, 335–336
    - optimization queries, 336–339
    - point-based influence, 330–331
    - propagation methods, 328
    - query point, 330, 341
    - representing AI guess, 328–329
    - spatial reasoning, 375–377
      - border calculations, 381–382
      - force estimates over regions, 378–380
      - intelligence illusion, 379–381
      - propagation calculations, 377–378
    - statistical approximations, 341
    - suitability considerations, 340
    - temporal influence propagation, 333–335
    - 2D to 3D geometry, 339–340
    - update frequency, 341
  - Influence propagation algorithm, 380
  - Influence source density, 340
  - Information retrieval, 354–355
  - Initial stimulus-based trigger, NPC
    - searching, 306
  - Interest search, 256
    - analysis, 262–263
    - classifying moves, 258–259
    - dynamic calculation, 261–262
    - Japanese chess (Shogi), 260–262
    - quantifying, 257–258
    - rethinking selective search, 257
    - selection by variation, 257
    - tree, 258
    - Treebeard chess, 260
  - Interrupting movement, 452–454
  - Investigate skill, human enemy AI, 426
-

---

## J

- The Journey to Wild Divine, 491
- JPS, *see* Jump point search (JPS)
- Jump points, 132
  - diagonal, 136
  - primary, 134–135
  - southwest diagonal, 139
  - straight, 134–136
  - westward straight, 138
- Jump point search (JPS), 131–132
  - forced neighbors, 133–134
  - jump points, 134–136
  - map preprocess implementation, 137–139
  - pruning strategy, 132–133
  - runtime implementation, 139–142
  - wall distances, 136–137

## K

- k*-d tree
  - application, 333
  - construction, 333
  - influence source density, 340
  - left-hand branch, 332
  - range query, 333–334
  - right-hand branch, 332
  - splitting axis, 332–333
  - temporal propagation, 335
- Kernel density estimation, 336
- K-means clustering
  - communal analysis, 486–488
  - individual analysis, 485
- Knowledge representation
  - influence mapping, 327
  - procedural content generation
    - components, 508
    - experiential chunks, 507
    - mixing-and-matching paradigms, 508–509
    - subcomponents, 508
    - templates, 507–508
- Kohan II: Kings of War*, 366
  - attackable obstacles, 385
  - greedy flood-fill-based approach, 368
  - illusion of intelligence, 381
  - tile based, 368

## L

- The Last of Us*, 21, 28–29
  - buddy AI, 431
    - armed combat, 439–440
    - combat utility, 437–438
    - cover, 435–437
    - finishing touches, 440–441
    - start from scratch, 432–435
  - human enemy AI, 419
    - behaviors, 426
    - building blocks, 420–421
    - cover, 424–426
    - flanking, 428–429
    - lethality, 427–428
    - perception, 421–423
    - polish, 429
    - posts, 424–426
    - skills, 426
    - stealth, 426–427
- Infected characters, 407
  - ambush skill, 415–416
  - characteristics, 408
  - chase skill, 415
  - data-driven design, 411
  - debugging, 412
  - distractions, 410
  - follow skill, 415
  - implementation, 412
  - logical sounds, 418–419
  - move-to behavior, 410, 415
  - on-fire skill, 416
  - philosophy, 410–411
  - search skill, 413–415
  - senses, 408–409
  - skills, 412
  - sleep skill, 417
  - throw skill, 416
  - types, 415–416
  - vision type, 411
  - wander skill, 416–417
- Launchpad level generator, 505, 512
- Layers of animation, 455–456
- League of Legends*, 12
- Left 4 Dead, 10, 490, 528–529
- Left 4 Dead 2's (L4D2), 491
- Lethality, human enemy AI, 427–428
- Limbic brain, 245, 251



- 
- Linear programming (LP), 207, 241
  - Little Red Riding Hood*, 523–525, 530–531
  - Living Scenes, 89–90
    - execution, 91, 94
    - NPCs, 95
    - performing through smart zones, 90–91
    - realization, 94
    - trigger management, 93
  - Locomotion, 109
    - animations, 36, 403
    - context, 110
    - forward, 103
    - NPC, 115
  - Looping animations
    - correcting speed, 451
    - displacement direction and orientation, 446–448
    - pose matching, 453
  - Losing target triggering method, NPC
    - searching, 306–307
  - LP, *see* Linear programming (LP)
  - M
  - Machine-learned narrative selection
    - technique, 530
  - Machine learning algorithms, 484
  - Marine Mania*, 25
  - Mario-like game, 513
  - Masking animations, 454–455
  - MCTS algorithms, *see* Monte Carlo tree search (MCTS) algorithms
  - Mecanim*, 106
  - Merchant’s inventory modeling, M/M/1
    - queues, 467
    - add\_rate parameters, 468–469
    - computational considerations, 469–470
    - remove\_rate parameters, 468–469
    - three-item inventory changes with time, 467–468
    - virtual observation, 468
  - Micro behaviors, NPC, 458–459
  - Minecraft*, 511
  - Minimax, 255–256
  - Mirrored animations, 455
  - Mirror’s Edge*, 225
  - M/M/1 queues, 463
    - equilibrium distribution, 466
    - modeling merchant’s inventory, 467
      - add\_rate parameters, 468–469
      - computational considerations, 469–470
      - remove\_rate parameters, 468–469
      - three-item inventory changes with time, 467–468
      - virtual observation, 468
    - stationary distribution, 464–466
    - transient distribution, 464–466
  - Modular tactical influence maps
    - advantage, 344–345
    - allied agent, 347
    - architecture, 343–344
      - base map structure, 349–350
      - templates, 350–352
      - types of base maps, 350
      - working maps, 352–353
    - implementations, 346–347
    - information retrieval, 354–355
    - location for area of effect attack, 362
    - movement to safer spot, 362
    - nearest battlefield location, 362–364
    - population of map data, 353–354
    - propagation, 348–349
    - querying information, 356–357
    - usage
      - gathering information, 356–357
      - positioning, 360–361
      - targeting, 357–360
  - Molotov cocktail, 410
  - Monte Carlo tree search (MCTS) algorithms, 255, 276
    - applying to games, 280
    - enhancements and variations, 279
    - implementation, 279
  - Morpheme*, 106
  - Move-to behavior, Infected characters, 410, 415
  - Moving
    - buddy AI, *The Last of Us*, 434
    - translation logic to animgraph, 107–108
  - Multiobjective evolutionary
    - algorithm, 484
  - Multiplayer online battle arena, 494
-

---

## N

Narrative generation technique, 526–527  
Navigation behaviors, 212–213  
Navigation pipeline, 213–214  
Neocortex, 245  
    state evaluation, 249–251  
    tree-search algorithm, 248–249  
*Neverwinter Nights*, 489  
Nonnegative matrix factorization (NMF), 486–487  
Nonplayer characters (NPCs)  
    behavior realization, 443  
        action tokens, 458–460  
        blackboards, 460  
        combining actions, 454–456  
        contextual one-off animations, 458  
        decoupling extracted motion, 446  
        displacement direction and orientation, 446–449  
        micro behaviors, 458–459  
        motion correction, 446  
        move and shoot tokens, 459  
        movement models, 444–446  
        on-screen realization, 460  
        per-bone blending, 453–454  
        pose matching, 453  
        pose-only blending, 453–454  
        rank categories, 460  
        speed correction, 450–452  
        tracking, 456–457  
        using additives with idles, 459  
    factorizing game state, 47–48  
    human enemy AI, *The Last of Us* (see Human enemy AI, *The Last of Us*)  
    locomotion, 115  
    perception testing, 421–423  
    possibility maps, 42–45  
    searching  
        aggressive search, 306, 308  
        cautious search, 305–308  
        ending search, 312  
        gap detection, 311–312  
        initial stimulus-based trigger, 306  
        losing target triggering method, 306–307  
        performing, 309–310  
        search spot, 309–311  
        techniques, 237–238  
        triggering, 306–307

    search map locations, 427

*Tom Clancy's Splinter Cell Blacklist* (see *Tom Clancy's Splinter Cell Blacklist*)

Nontrivial topologies, influence map, 335–336

NPCs, see Nonplayer characters (NPCs)

Numeric weight vectors, 486

## O

Object identification certainty, 27–28

    camouflage, 29

    identification certainty, 29–30

    incorporating movement, 29

Objective function maximization technique, 529–530

Obstacles

    attackable, 385–386

    handling, influence map, 335–336

    reciprocal velocity obstacle, 206, 220–221

Offline AI, 266

Offline UCB1, 273–275

One-off animations

    contextual, 458

    displacement direction and orientation, 446, 448–449

    micro behaviors, 458–459

On-fire skill, Infected characters, 416

Online AI, 265–266

Online UCB1, 266–268

    applying to games, 268–270

    javascript, 269

On-screen realization, nonplayer characters, 460

Open post, 424

The Open Racing Car Simulator, 492

OpportunisticShooter, 428

Optimal reciprocal collision

    avoidance (ORCA), 204–208

Optimization-based generator, PCG, 505–506

Optimization query, influence map, 336–339

## P

PAM, see Predictive avoidance method (PAM)

Parameterized control, PCG, 509

Path caching, 372–373

Pathfinding

    group navigation, 216–217

    idealized environment, 174–175

Path planning, 371

    congestion maps, 175–177

    efficient state representation, 239–240

- 
- Path smoothing, 177–179
  - PCA, *see* Principle component analysis (PCA)
  - PCG, *see* Procedural content generation (PCG)
  - Per-bone blending, 453–454
  - Perception
    - auditory
      - calculating audio distance, 320–321
      - fairness, 321–322
    - human enemy AI, *The Last of Us*, 421–423
    - motion, 472–474
    - visual, 314–317
  - Personality in triune architecture, 246
  - Photon mapping, 336
  - Placement mechanism, 328
  - Planning Domain Definition
    - Language (PDDL), 524, 527–528
  - Platformers, 483
  - Platoon leaders, 365
  - Player Appraisal Controlling
    - Emotions (PACE), 531
  - Player experience modeling, 506
  - Player interaction, with PCG, 510–511
  - Player modeling, 11–12
    - artificial neural network, 484
    - Bayesian networks, 485
    - definition, 484
    - numeric weight vectors, 486
    - support vector machines, 484–485
  - Player-Specific Automated
    - Storytelling (PAST), 531
  - Player-Specific Stories via Automatically
    - Generated Events (PaSSAGE)
      - AI-driven experience management, 530–531
      - style-based adaptation, 492–493
  - Play style modeling, 527
  - Point-based influence, 330–331
  - Point light animations, 472–473
  - Polish, *The Last of Us*, 429
  - Polygons, voxel-based navmesh
    - generation, 392
    - sloping side after simplification, 395
    - staircase side before simplification, 395
    - triangulating, 394–396
  - Portal*, 15
  - Pose matching, 451, 453
  - Pose-only blending, 453–454
  - Position, agent state, 197
  - Possibility maps, 42
    - NPC's location, 43
    - probability maps, combining, 46–47
    - using, 42–45
  - Posts
    - definition, 424
    - selector, 424–426
    - types, 424
  - Potassco, 515
  - Predictive avoidance method (PAM), 203–204
  - Preference control, PCG, 509
  - Prince of Persia*, 268, 273
  - Principle component analysis (PCA), 486–487
  - Probability map, 42
    - guard, 46
    - possibility and, combining, 46–47
    - static, 47
    - updating, 45–46
  - Procedural content generation (PCG), 15, 501
    - academic research, 515
    - bottom-up approach, 512
    - Choco, 515
    - constraint-driven methods, 506–507
    - constructionist approach, 503–504
    - Context Free Art, 515
    - debugging, 514
    - Elite*, 501–502
    - engineering constraints, 513
    - evolutionary algorithm, 505
    - game design constraints, 512–513
    - Google, 515
    - grammar-based approach, 504–505
    - knowledge representation
      - components, 508
      - experiential chunks, 507
      - mixing-and-matching paradigms, 508–509
      - subcomponents, 508
      - templates, 507–508
    - mechanical role, 509–510
    - optimization-based generator, 505–506
    - player interaction with, 510–511
    - Potassco, 515
    - relationship with art, 513
    - simulation-based approaches, 503
    - StructureSynth, 515
    - top-down approach, 512
    - tuning, 514
    - wiki, 515
-

- 
- Production systems, 59
    - debugging and tuning, 66–67
    - decision making, 60–61
    - design considerations, 60
    - greedy matching/selection, 63–64
    - left-hand side, 62
    - matching system, 63–64
    - right-hand side
      - complex, 66
      - execution and design, 65–66
      - recording, 66
    - rules
      - authoring, 62
      - multiple, 64–65
      - recorded system, 62–63
      - representation, 61–62
    - terminology, 60
  - Prone navigation, 401
  - Propagation methods, 328
  - Proximity map, 350, 353
    - high point of enemy, 356–357, 361
    - templates, 351
  - Psychology, character behavior design
    - development, 474–475
    - practical application, 477
- Q**
- Quadtree approach, 329, 332
  - Queued narrowing operations, 300–301
- R**
- Radius, 197, 345
  - Range query, *k*-d tree, 333–334
  - Readable character, 471
  - Real-time strategy (RTS) game, 240, 366, 494
  - Recast, 390
  - Reciprocal velocity obstacle (RVO), 206, 220–221
  - Recognition, 32–33
  - Reference node, 444
  - Reference speeds, 451
  - Regret matching, 270–271
    - applying to games, 271–273
    - javascript, 272–273
  - Regret-minimizing algorithms, 266
  - Reinforcement learning (RL) algorithms, 489
  - Relative utility, 24
  - Reptilian brain, 245
  - Robot Unicorn Attack*, 503, 507, 511
  - Rock–paper–scissors (RPS), 266
  - Role-playing games (RPGs), 237, 270
    - game metrics, 483
    - modular tactical influence map, 343–344, 352, 354
  - Root node, 444
    - cyclical movement animations, 448
    - motion extraction, 445
  - RTS, *see* Real-time strategy game (RTS)
- S**
- Searching, NPC
    - aggressive search, 306, 308
    - cautious search, 305–308
    - ending search, 312
    - gap detection, 311–312
    - initial stimulus-based trigger, 306
    - losing target triggering method, 306–307
    - performing search, 309–310
    - search spot, 309–311
    - techniques
      - alternate uses, 240
      - bottlenecks, 240–241
      - efficient state representation, 239–240
      - illustrative example, 237–238
      - operators, 239
      - requirements, 239
    - triggering, 306–307
  - Search map, 427
  - Search pathfinding, 161
  - Search skill, Infected characters, 412–415
  - Search spot, NPC
    - generation, 309
    - selection, 310–311
  - Senses, Infected characters, 408–409
  - Separation of concerns (SoC), 106–107, 114–115
  - Set operations implementation, 299
  - Shape fitting, 472–473
  - Shogi, 260–262
  - Shooting, armed combat, 439
  - SI, *see* Survivor intensity (SI)
  - Sidestep navigation, 402–403
  - The Sim*, 6, 12
  - Simple reaction time, 32–33
  - Simple subgoal graphs (SSGs), 147–148
    - constructing, 148–150
    - direct-h-reachable, 149–153
    - h-reachability, 148–149
    - searching using, 150–151
  - Simplex volume maximization (SIVM), 487–488
-

- 
- Simulation-based approaches, 503
  - Simulation principles, *Dwarf Fortress*, 519–521
  - Skill-based adaptation, 482, 488
    - artificial evolution techniques, 489
    - cascading elitism, 489–490
    - reinforcement learning algorithms, 489
    - weighted behavior selection techniques, 489
  - Skin conductance level (SCL), 491
  - Sleep skill, Infected characters, 417
  - Smart Objects, 90
  - Smart Zones
    - behavior orchestration, 94
    - definition by game designers, 91–93
    - juggler spectacle, 98
    - NPC behaviors, 95
    - overlapping zones management, 95
    - performing Living Scenes, 90–91
    - practice, 91–95
    - role assignment, 93–94
    - role interruption, 95
    - runtime architecture, 93
    - scenario, 96–97
    - Smart Zone 1, 96
    - Smart Zone 2, 96
    - Smart Zone 3, 96
    - Smart Zone 4, 96
    - Smart Zone 5, 96
    - trigger management, 93
    - Unity3D, 97–99
  - SoC, *see* Separation of concerns (SoC)
  - Social awareness, 322–324
  - Social groups, 211–212
  - Spatial analysis, 327
  - Spatial coverage, extending voxel-based navmesh
    - metadata use, 399–401
    - reintroducing discarded space, 397–399
  - Spatial partitioning, 366–367
    - homogeneity, 366–367
    - region generation
      - designer defined, 367
      - navmesh based, 368
      - tile based, 368–369
    - static *vs.* dynamic regions, 369
  - Spatial reasoning, 327, 366
    - characteristics
      - attackable obstacles, 385–386
      - avenues of approach, 383–385
      - consolidation, 386–387
      - counterattacks, 386–387
      - flanking attacks, 385
      - high-traffic areas, 383
      - scent of death, 383
    - influence maps, 375–377
      - border calculations, 381–382
      - force estimates over regions, 378–380
      - intelligence illusion, 379–381
      - propagation calculations, 377–378
    - working with regions, 369
      - chokepoints, 373–375
      - cul-de-sacs, 373–375
      - distance estimation, 371–372
      - path caching, 372–373
      - path planning, 371
      - picking places to attack, 370
      - picking places to scout/explore, 370
      - picking unit positions, 370–371
  - Splitting axis, *k-d* tree, 332–333
  - Sports Commentary Recommendation System (SCoReS), 531
  - Squad leaders, 365
  - SSGs, *see* Simple subgoal graphs (SSGs)
  - Starcraft*, 145, 273
  - Start from scratch, buddy AI
    - ambient following, 432–433
    - approach, 432
    - dodging, 434–435
    - follow positions, 433–434
    - moving, 434
    - plan, 432
    - teleportation, 435
  - State machines, 103
  - Stationary distribution, M/M/1 queues, 464–466
  - Stealth, human enemy AI, 426–427
  - Steering behaviors, 183
    - Band-Aids, 184
    - child behaviors, 184
    - flocks *vs.* groups, 185
    - lack of context, 185–186
    - prioritization, 185
    - weighting, 184
  - Stop-motion animation, 473
  - Strategy games, 483
    - attackable obstacles, 385
    - RTS game, 240, 366, 494
    - tile-based, 366
  - StructureSynth, procedural content generation, 515
  - Style-based adaptation, 482, 492–493
-

- 
- Subgoal graphs, 145
    - core graph, 157
    - preliminaries, 146–147
    - simple (*see* Simple subgoal graphs (SSGs))
    - three-level graph, 156
    - two-level (*see* Two-level subgoal graphs (TSGs))
    - variants on game maps, 146
    - visibility graphs, 148
  - Support vector machines (SVMs), 484–485
  - Survivor intensity (SI), 490–491
  - Swim navigation, 401–402
  - System refactoring, 114–115
  - T
  - Tactical Environment Awareness
    - System (TEAS), 318, 324
  - Tactical point system (TPS), 308–309
  - Team balancing, 482
  - Team matching techniques, 494–495
  - Telemetry, 483, 487
  - Teleological stance, 474–475
  - Teleportation, buddy AI, 435
  - Templates
    - modular tactical influence maps, 350–352
    - procedural content generation, 507–508
    - proximity map, 351
    - threat map, 352
  - Temporal influence propagation, 333–335
  - A Theory of Fun for Game Design*, 365
  - Theta\*, 163–164
    - algorithm, 166–167
      - example trace, 167
      - pseudocode, 166
    - Dijkstra’s algorithm, 178
    - paths, 168–170
    - problem formalization, 164
  - Threat map, 350, 352
  - Throwing, combat utility, 437–438
  - Throw skill, Infected characters, 413, 416
  - Tile-based map, 368–369
  - Tile-based strategy games, 284, 366
  - Time horizon, 198–199
  - Time-to-collision, 197–199
  - Tokens, characters problem, 476–477
  - Tomb Raider*, 225
  - Tom Clancy’s Splinter Cell Blacklist*, 28
    - auditory perception
      - calculating audio distance, 320–321
      - fairness, 321–322
    - consistency, 314
    - contextual awareness, 322, 324–325
    - disappearing NPC problem, 325–326
    - environmental awareness
      - changes in objects, 319–320
      - connectivity, 317–319
    - fairness, 313
    - good feedback, 314
    - intelligence, 314
    - social awareness, 322–324
    - visual perception, 314–317
  - Top-down approach, procedural content
    - generation, 512
  - Tracking
    - additive aiming poses, 456–457
    - inverse kinematics, 457
  - Transient distribution, M/M/1 queues, 464–466
  - Transition animations, 444, 451
    - correcting speed, 451
    - displacement direction and orientation, 446, 448–449
    - pose matching, 453
  - Traversal setups, 225–227
  - Treebeard chess, 260
  - Tree-search algorithm, 248
  - Triune brain model, 245–246
  - TSGs, *see* Two-level subgoal graphs (TSGs)
  - Tuning, procedural content generation, 514
  - Two-level grid graphs, 157
  - Two-level property (TLP), 153
  - Two-level subgoal graphs (TSGs)
    - constructing, 153–154
    - n-level graphs, 156–158
    - searching using, 154–155
  - U
  - UCB1
    - offline, 273–275
    - online (*see* Online UCB1)
  - UCT, 276–278
    - applying to games, 280
    - enhancements and variations, 279
    - implementation, 279
    - pseudocode, 278
    - selection and expansion phases, 276–277
    - simulation and propagation phases, 277–278
  - Unity3D, 97–99

- 
- Usability testing, 11
  - User-based CF approach, 494
  - Utility-based AI, 23–26
  - V
  - Value semantics, 299
  - Variables, 285
    - class declaration, 297
    - graphical representation, 288
    - ordering, 301
    - undo, 295
  - VCI, *see* Virtual controller input (VCI)
  - Velocity, 197
  - Velocity correction, 221
  - Vertex contraction, 158
  - Virtual controller input (VCI), 230–232
  - Virtual group entity, 214–215
  - Visibility graphs, 163
  - Vision sweet spot, 28
  - Vision zones, 27–28
    - camouflage, 29
    - identification certainty, 29–30
    - incorporating movement, 29
  - Visual perception, *Tom Clancy's Splinter Cell Blacklist*, 314–317
  - Vocalizations, buddy AI, 440
  - Voronoi diagrams, 331
  - Voxel-based navigation mesh/navmesh, 389
    - agent-driven parameter (*see* Agent-driven parameter)
    - border voxels, 393–394
    - crouch-only triangles, 397–400
    - enter voxelization, 391–394
    - extending spatial coverage
      - metadata use, 399–401
      - reintroducing discarded space, 397–399
    - goals, 390–391
    - multiple generation parameter passes, 403
    - nonwalkable voxels, 392
    - playing with heuristic, 403–404
    - prone navigation, 401
    - recap, 396–397
    - sidestep navigation, 402–403
    - stand-only triangles, 397–399
    - swim navigation, 401–402
    - triangulating polygons, 394–396
    - walkable voxels, 392
  - W
  - Walk animation, 451–452
  - Wander skill, Infected characters, 412–413, 416–417
  - Warcraft II*, 145
  - Warzone 2100*, 506
  - Weighted behavior selection techniques, 489
  - Wiki, PCG, 515
  - Working maps, 349, 352–353
  - Work queue, 301
  - World of Warcraft*, 6, 487
  - Z
  - Zeus*, 50
  - Zoo Tycoon 2*, 25–26