

References

- Aggarwal, J. K. and Y. F. Wang (1991). Sensor data fusion in robotic systems, *Control and Dynamic Systems*, **39**: 435–461.
- Aggerwal, A., B. Chazelle, L. Guibas, C. O'Dúnlaing, and C. Yap (1988). Parallel Computational Geometry, *Algorithmica*, **3**: 293–327.
- Ahmadabi, M. N. and E. Nakano (1996). A cooperation strategy for a group robots, Proc. of IROS'96, Osaka, pp. 125–132.
- Alander, J. T. (1993). On Robot Navigation Using a Genetic Algorithm, *Artificial Neural Networks and Genetic Algorithms Proc.* (ed.) Albrecht, Reeves, and Steele, Springer Verlag.
- Baldwin, J. and N. Guild (1980). Axiomatic approach to implication for approximate reasoning with fuzzy logic, *Fuzzy Sets System*, **3**.
- Banaszak, Z. and E. Roszkowska (1988). Deadlock Avoidance in Concurrent Processes, *Foundations of Control*, **18**(1/2): 3–17.
- Barraquand, J., B. Langlois, and J. C. Latombe (1992). Numerical potential field techniques for robot path planning, *IEEE Trans. System, Man, Cybernetics*, SMC-22(2), 224–240.
- Bedworth, D. D., M. R. Henderson, and P. Wolfe (1991). *Computer Integrated Design and Manufacturing*. McGraw Hill Int. Edit.
- Black, J. T. (1988). The Design of Manufacturing Cells (Integrated Manufacturing Systems), *Proc. of Manufacturing International Conference*, Vol. III, pp. 143–157.
- Born, J. H. and C. H. Menq (1991). Determination of Optimal Measurement Configurations for Robot Calibration Based on Observability Measure, *International Journal of Robotics Research*, **10**(1): 51–63.
- Brady, M., ed. (1986). *Robot Motion: Planning and Control*, MIT Press.
- Brooks, R. (1983). Planning Collision-Free Motions for Pick-and-Place Operations, *International Journal of Robotics Research* **2**(4): 19–44.
- Buzacott, J. A. (1985). Modeling manufacturing systems, *Robotics and Computer Integrated Manufacturing*, **2**: 1.
- Carpenter, G. A. and A. H. Tan (1993). Fuzzy ARTMAP, rule extraction and medical databases, *Proc. of the World Congress on Neural Networks*, Portland, Oregon, Vol. 1, pp. 501–506.
- Carpenter, G. A., S. Grossberg, and D. B. Rosen (1991). Fuzzy ART: Fast stable learning and categorization of analog patterns in an adaptive resonance system, *Neural Network*, **4**: 759–71.
- Chang, P. H. (1987). A closed-form solution for inverse kinematics of robot manipulators with redundancy, *IEEE Journal on Robotics and Automation*, RA-3(5), 393–403.
- Chester, D. L. (1991). Why two hidden layers are better than one, *IJCNN'90-1*, pp. 256–68.
- Chirikjian, G. S. and J. W. Budrick (1990). An obstacle avoidance algorithm for hyper redundant manipulators, *Proc. of IEEE International Conference on Robotics and Automation*, pp. 625–631.
- Chroust, G. (1992). Software-Development Environment, *Informatic Spectrum*, **15**: 165–74.

- Chroust, G. and W. Jacak (1996). Software Process, Work Flow and Work Cell Design Separated by a Common Paradigm? *Lecture Notes in Computer Science*, Springer Verlag, Berlin, Vol. 1030, pp. 403–17.
- Chung, W. J., W. K. Chun, and Y. Youm (1992). Kinematic control of planar redundant manipulators by extended motion distribution scheme, *Robotica*, **10(3)**: 255–262.
- Coffman, E. G., M. Elphick, and A. Shoshani (1971). System Deadlock, *Computing Surveys*, **3(2)**: 67–78.
- Cohen-Tannoudji, B., C. Diu, and F. Laloe (1977). *Quantum Mechanics*, John Wiley and Sons.
- Craig, J. J. (1981). *Introduction to Robotics*. Cambridge, MIT Press.
- Cutkosky, M. R. (1985). *Robotic Grasping and Fine Manipulation*, Kluwer Academic Publishers, Boston.
- Deitel, H. M. (1983). *An Introduction to Operating Systems*. Wesley, Addison.
- Denavit, J. and R. S. Hartenberg (1955). A kinematic notation for lower-pair mechanisms based on matrices, *ASME Tran. Journal of Applied Mechanics*, **77**: 215–221.
- Dubey, R. and J. Y. S. Luh (1987). Redundant robot control for higher flexibility, *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1066–1072.
- Dubois, D. and H. Prade (1985). The generalized modus ponens under sup-min composition, in *Approximate Reasoning in Expert System*, North Holland.
- Duelen, G. and K. Schröer (1991). Robot Calibration-Method and Results, *Robotics and Computer-Integrated Manufacturing*, **8**: 223–231.
- Edelsbrunner, H. (1987). *Algorithms in Combinatorial Geometry*, Springer Verlag Berlin.
- Efstathiou, J. (1987). Rule based process control using fuzzy logic, in: *Approximate Reasoning in Intelligent Systems. Decision and Control*, L. Zadeh (ed.), Pergamon.
- Elsley, R. K. (1988). A learning architecture for control based on back-propagation neural networks *International Conference on Neural Networks ICNN'88-2*.
- Espiau, B. and R. Bouluc (1985). Collision avoidance for redundant robots with proximity sensors, *III Symp. Robotic Research*, pp. 243–251.
- Faverjon, B. (1986). Object Level Programming of Industrial Robots, *IEEE International Conference on Robotics and Automation*, Vol. 2, pp. 1406–1411.
- Feddema, J. T. and Shaheen Ahamed (1985). Determining a Static Robot Grasp for Automated Assembly, *IEEE SMC Conference*.
- Gallant, A. R. and H. White (1988). There exists a neural network that does not make avoidable mistakes, pp. 657–664, *International Conference on Neural Networks ICNN'88-1*.
- Gatrell, L. B. (1989). CAD-Based Grasp Synthesis Utilizing Polygons, Edges Vertexes, pp. 184–89, *IEEE SMC Conference*.
- Geneserth, M. R. and N. J. Nilsson (1987). *Logical Foundations of Artificial Intelligence*, Morgan Kaufmann Publishers.
- Gilbert, E. G., D. W. Johnson, and S. Keerth (1988). A fast procedure for computing the distance between complex objects in three-dimensional space, *IEEE Trans. on Robotics and Automation*, RA-4(2), 193–203.
- Goldenberg, A. A., B. Benhabib, and R. G. Fenton (1985). A complete generalized solution to the inverse kinematics of robots, *IEEE Journal on Robotics and Automation*, RA-1 (1), 14–20.
- Guez, A. and Z. Ahmad (1998a). Accelerated convergence in the inverse kinematics via multilayer feedforward networks, pp. 341–344, *International Conference on Neural Networks ICNN'89-2*.
- (1998b). Solution to the inverse kinematics problem in robotics by neural networks, pp. 617–624, *International Conference on Neural Networks ICNN'88-2*.
- Guo, J. H. and V. Cherkassky (1989). A solution to the inverse kinematic problem in robotics using neural network processing, *IEEE Joint International Conference on Neural Network*, San Diego CA, Vol. 2, pp. 299–304.

- Haddadi, A. (1995). *Communication and cooperation in agent system*, Lecture Notes in Computer Science 1056, Springer Verlag, Berlin, New York.
- Han, J. Y. and M. R. Sayeh (1989). Convergence and limit points of neural network and its application to pattern recognition, *IEEE Trans. System, Man, Cybernetics*, SMC-19(5), 1217–1222.
- Han, M. and B. Zhang (1994). Control of robotics manipulators using a CMAC-based reinforcement learning system, *Proc. of IEEE Conference on Intelligent Robots and Systems IROS'94*, Munich, Germany.
- Hecht-Nielsen, R. (1987). Counterpropagation networks, *IEEE First International Conference on Neural Network*, Vol. 2., pp. 19–32.
- Hirsch, M. W. and S. Smale (1974). *Differential Systems and Linear Algebra*, Academic Press, New York.
- Homem De Mello, L. S. and A. C. Sanderson (1990). AND/OR Graph Representation of Assembly Plans, *IEEE Trans. on Robotics and Automation*, 6(2): 188–199.
- Homem De Mello, L. S. and S. Lee (1991). *Computer Aided Mechanical Assembly Planning*, Kluwer Academic Publisher.
- Hornik, K. (1991). Approximation Capabilities of Multilayer Feedforward Networks, *Neural Networks*, 4: 251–257.
- Hornik, K., M. Stinchcombe, and H. White (1990). Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks, *Neural Networks*, 3: 551–60.
- Jacak, W. (1985). Time variant goal space: A fuzzy set approach, *Systems Science*, 11(3).
- (1989a). A Discrete Kinematic Model of Robot in the Cartesian Space, *IEEE Trans. on Robotics and Automation*, 5(4): 435–446.
- (1989b). Strategies for Searching Collision-Free Manipulator Motions: Automata Theory Approach, *Robotica*, 7: 129–138.
- (1990). Robot Task and Motion Planning, *AI, Simulation and Planning in High Autonomy Systems*, IEEE Computer Society Press, 168–176.
- (1991). Discrete Kinematic Modelling Techniques in Cartesian Space for Robotic System, in: *Advances in Control and Dynamics Systems*, (ed.) C. T. Leondes, Academic Press.
- (1994a). Fuzzy Rules Based Control of Intelligent Robotic Workcell, in *Cybernetics and Systems'94*, (ed.) R. Trappl, World Scientific, Vienna, Singapore.
- (1994b). Graph Searching and Neural Processing Approach to Robot Motion Planning, *Technical Report of RISC*, Austria.
- (1995). Hybrid Evolutionary approach to control of intelligent arm, *Proc. of II International Symposium on Methods and Models in Automation and Robotics MMAR'95*, August, Szczecin, Poland.
- (in press). ICARS Simulation based CAPP/CAM System for Intelligent Control Design in Manufacturing Automation, in *The International Handbook on Robot Simulation System*, (ed.) D. W. Wloka, John Wiley and Sons, (in preparation).
- Jacak, W. and A. Senanayake (1995). Neural Network Approach for Optimal Grasping, *Proc. of International Conference on Recent Advances in Mechatronics ICRAM'95*, Istanbul, Turkey, August, pp. 917–22.
- Jacak, W. and B. Buchberger (1996a). Intelligent Robotic Arm based on Reactive Control, *Robotics in Alpe-Adria-Danube Region RAAD'96*, Budapest, Hungary, pp 297–303.
- (1996b). Intelligent Systems Combining Reactive and Learning Capabilities, *Cybernetics and Systems '96* R. Trappl (ed.) Vienna, Austria.
- Jacak, W., B. Buchberger, and R. Muszynski (1995a). Symbolic Computation based Synthesis of Neural Network Dynamics and Controller for Robots, *Proc. of IEEE International Conference on Neural Networks ICNN'95*, November IEEE Press, Perth, Australia pp. 2720–2725.

- Jacak, W. and B. Kapsammer (1993). Automatic Programming in CIM based on Intelligent Tools in: *Information Management in Computer Integrated Manufacturing*, (ed.) H. Adelsberger, V. Marik, Lecture Notes in Computer Science, Vol. 973, Springer Verlag, Berlin, New York, pp. 392–418.
- Jacak, W. and G. Jahn (1995). Distributed Intelligent Control of an Autonomous Robotic Cell, in *Intelligent Autonomous Systems IAS'95*, (ed.) R. Dillman, IOS Press, Amsterdam, Oxford, Karlsruhe, pp. 441–49.
- Jacak, W., H. Praehofer, G. Jahn, and G. Haider (1994a). Supervising Manufacturing System Operation by DEVS-based Intelligent Control, *AI, Simulation and Planning in High Autonomy System*, (ed.) Fishwick B., Gainesville, Florida, USA, IEEE Computer Society Press, Los Alamitos, California.
- Jacak, W., I. Duleba, and P. Rogaliński (1992). A Graph-Searching Approach to Trajectory Planning of Robot's Manipulator, *Robotica*.
- Jacak, W., I. Duleba, and R. Muszynski (1995b). Hybrid Computing Approach to Robot Control in the Presence of Uncertainties, *Proc. of IFAC International Conference on Motion Control*, September, Munich, Germany, pp. 664–671.
- Jacak, W. and J. W. Rozenblit (1992a). Automatic Robot Programming, *Lecture Notes in Computer Science* Vol. 585, Springer Verlag, pp. 168–187.
- (1992b). Automatic Simulation of a Robot Program for a Sequential Manufacturing Process, *Robotica*.
- (1993). Virtual Process Design Techniques for Intelligent Manufacturing, *AIS'93 AI, Simulation and Planning in High Autonomy Systems*, IEEE Computer Soc. Press, Tucson.
- (1994). CAST Tools for Intelligent Control in Manufacturing Automation, *Lecture Notes in Computer Science*, Vol. 763, Springer Verlag, pp. 203–220.
- (in press). Model-based Workcell Task Planning and Control, *IEEE Trans. Robotic and Automation* (submitted).
- Jacak, W. and S. Dreiseitl (1995). Genetic Algorithm based Neural Networks for Dynamical Systems Modeling, *Proc. of IEEE International Conference on Evolutionary Computing ICEC'95*, November IEEE Press, Perth, Australia, pp. 602–607.
- (1996). Hybrid Evolutionary Programming — the Tools for CAST, *Lecture Notes in Computer Science*, Springer Verlag, Berlin, No 1030, pp. 289–305.
- (1996). Robotic Agent Control Combining Reactive and Learning Capabilities, *IEEE International Conference on Neural Networks ICNN'96*, June Washington, USA.
- Jacak, W., S. Dreiseitl, T. Kubik, and D. Schlosser (1995c). Distributed Planning and Control of Intelligent Robot's Arm Motion based on Symbolic and Neural Processing, *IEEE International Conference on Systems, Man and Cybernetics SMC'95*, October, Vancouver, Canada, pp. 2898–2903.
- Jacak, W., T. Kubik, and R. Muszynski (1995d). Neural Processing based Robot Kinematics Modelling and Calibration for Pose Control *XII Conference on Systems Science*, Wroclaw, Poland, September pp. 288–295.
- Jacak, W., and et al. (1994b). The Hybrid Evolutionary Method and System for Intelligent Robot Control, Technical Report RWCP, Linz, Tskuba, Japan.
- (1995e). HYROB System: The module for automatic synthesis of the neural representation of a geometric knowledge and fast distance calculation, Technical Report RWCP, Linz, Tskuba, Japan.
- (1995f). HYROB System: The module for the automatic synthesis of neural models for robot direct and inverse kinematics, Technical Report RWCP, Linz, Tskuba, Japan.
- (1995g). Hybrid Programming Approach to Robot Dynamics, Technical Report RWCP, Linz, Tskuba, Japan.
- (1995h). Neural Network based gain scheduling control system for robots in the presence of uncertainties with one-step trajectory planning, Technical Report RWCP, Linz, Tskuba, Japan.

- (1996). HYROB System: Application to the design of intelligent robotic agents acting in a real world environment, Technical Report RWCP, Linz, Tskuba, Japan.
- Jahn, G. (1996). *Event Based Approach to Simulation Based Control of Flexible Manufacturing Systems*, Ph.D. Thesis of Johannes Kepler University Linz.
- Jahn, G. and P. Rogalinski (1993). Simulator of Robotic Workcell: Implementation, Technical Report University Linz, Austria.
- Jones, A. T. and C. R. McLean (1986). A Proposed Hierarchical Control Model for Automated Manufacturing Systems, *Journal of Manufacturing Systems*, **5**(1): 15–25.
- Josin, G., D. Charney, and D. White (1988). Robot control using neural networks, pp. 625–631, *International Conference on Neural Networks ICNN'88-2*.
- Kawato, M., Y. Uno, M. Isobe, and R. Suzuki (1987). A hierarchical model for voluntary movement and its application to robotic, *IEEE First International Conference on Neural Network*, Vol. 4., pp. 573–82.
- King, J. R. (1980). Machine-Component Grouping in Production Flow Analysis: An Approach Using a Rank Order Clustering Algorithm, *International Journal of Production Research*, **20**(2): 213–32.
- Kircanski, M. and O. Timcenko (1992). A geometric approach to manipulator path planning in 3D space in the presence of obstacles, *Robotica*, **10**(4): 321–328.
- Klein, C. A. and C. H. Huang (1983). Review of pseudoinverse control for use with kinematically redundant manipulators, *IEEE Trans. System, Man, Cybernetics*, SMC-13(3), 245–250.
- Kohonen, T. (1988). *Self-Organization and Associative Memory*, Springer Verlag, Berlin.
- Kosuge, K. and T. Oosumi (1996). Decentralized control of multiple robots handling an object, Proc. of IROS'96, Osaka, pp. 318–332.
- Koza, J. R. (1992). *Genetic Programming*, MIT Press.
- Kreinovich, V. Y. (1991). Arbitrary Nonlinearity is Sufficient to Represent All Functions by Neural Networks: A Theorem, *Neural Networks*, **4**: 381–383.
- Krogh, B. H. and Z. Banaszak (1989). Deadlock Avoidance in Pipeline Concurrent Processes, *Proc. of Workshop on Real-Time Programming IFAC/IFIP*.
- Kuc, R. and V. Viard (1991). A physically based navigation strategy for sonar guided vehicles, *International Journal Robotics Research*, **10**: 1–67.
- Kumar, V. R. and K. J. Waldron (1989). Actively Coordinated Vehicle Systems, *ASME Journal of Mech. Trans. and Aut. Design*, **III**(2): June.
- (1991). Force Distribution Algorithms for Multifingered Grippers, *Control and Dynamic Systems*, **39**: Academic Press Inc.
- Kung, S. Y. and J. N. Hwang (1989). Neural network architectures for robotics applications, *IEEE Trans. Robotics Automation*, RA-5(5), 641–657.
- Kusiak, A. (1990). *Intelligent Manufacturing Systems*. Prentice Hall.
- Lambrinos, D. and C. Scheier (1996). Building complete autonomous agent. A case study on categorization, Proc. of IROS'96, Osaka, pp. 170–179.
- Latombe, J. C. (1991). *Robot motion planning*, Kluwer Ac. Pub., Boston, Dordrecht.
- Lee, C. C. (1990). Fuzzy Logic in Control Systems: Fuzzy Logic Controller, *IEEE Trans. SMC*, **20**(2).
- Lee, S. and G. A. Bekey (1991). Applications of neural network to robotics, in *Control and Dynamic System: Advances in Robotic Systems*, (ed.) C. T. Leondes, Academic Press Inc., San Diego, New York, pp. 1–70.
- Lee, S. and J. M. Lee (1990). Multiple task point control of a redundant manipulator, *Proc. of IEEE International Conference on Robotics and Automation*, pp. 988–993.
- Lee, S. and R. M. Kil (1988). Multilayer feedforward potential function network, *IEEE International Conference on Neural Network*, Vol. 1., pp. 641–657.

- (1989). Bidirectional continuous associator based on gaussian potential function network, *IEEE International Joint Conference on Neural Network*, Vol. 1, pp. 45–53.
- (1990). Robot kinematic control based on bidirectional mapping neural network, *IEEE International Joint Conference on Neural Network*, Vol. 3, pp. 327–335.
- Lenz, J. E. (1989). *Flexible Manufacturing*. Marcel Dekker, Inc.
- Liegeois, A. (1977). Automatic supervisory control of the configuration and behavior of multibody mechanisms, *IEEE Trans. System, Man, Cybernetics*, SMC-7(12), 868–871.
- Lifschitz, V. (1987). On the semantics of STRIPS in *Reasoning about Actions and Plans*, M. Georgeff (ed.), San Francisco: Morgan Kaufmann.
- Lin, C. S., P. R. Chang, and J. Y. S. Luh (1983). Formulation and Optimization of Cubic Polynomial Joint Trajectories for Industrial Robots, *IEEE Trans. on Automatic Control*, AC-28(12): 1066–073.
- Lippman, R. P. (1987). An introduction to computing with neural nets, *IEEE ASSP Magazine*, 4: 4–22.
- Lozano-Perez, T. (1981). Automatic Planning of Manipulator Transfer Movements, *IEEE Trans. System, Man, Cybernetics*, SMC-11(10).
- (1986). Motion Planning for Simple Robot Manipulators, *Proc. of the III International Symposium on Robotics Research*, Faugeras O. D. and Giralt G.,
- (1989). Task-Level Planning of Pick-and-Place Robot Motions, *IEEE Trans. on Computers* 38(3): 21–29.
- Luca, A. De, and G. Oriolo (1991). Issues in acceleration resolution of robot redundancy, *Proc. of Symp. on Robot Control, Syroco'91*, Vol. 2, pp. 665–670.
- Lumelsky, V. J. (1985). On fast computation of distance between line segments, *Inform. Process. Lett.*, 21: 55–61.
- Lumelsky, V. J. and K. Sun (1989). A unified methodology for motion planning with uncertainty for 2D and 3D two link robot arm manipulator *International Journal Robotics Research*, 9: pp 89–104.
- Maciejewski, A. A. and C. A. Klein (1985). Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments, *International Journal Robotics Res.* 4(3): 109–17.
- Maimon, O. (1987). Real-time Operational Control of Flexible Manufacturing System, *Journal of Manufacturing Systems*, 6(2): 125–136.
- Martinez, T. M. and K. J. Schulten (1990). Hierarchical neural net for learning control of a robot's arm and gripper, *IEEE Joint International Conference on Neural Network*, San Diego CA, Vol. 2, pp. 747–52.
- Mayer, W. (1986). Distances between boxes: Applications to collision detection and clipping, *Proc. of IEEE International Conference on Robotics and Automation*, pp. 597–602.
- Mayorga, R. V., A. K. C. Wong, and N. Milano (1992). A Fast Procedure for Manipulator Inverse Kinematics Evaluation and Pseudo-Inverse Robustness, *IEEE Tran. on Systems, Man, and Cybernetics*.
- McDermott, D. (1982). A temporal logic for reasoning about processes and plans, *Cognitive Science*, 6.
- McKerrow, P. J. (1991). *Introduction to Robotics*, Addison-Wesley Publ.
- Meystel, A. (1988). Intelligent control in robotics, *Journal Robotic Systems*, 5: 5.
- Michalewicz, Z. (1992). *Genetic Algorithms + Data Structures = Evolution Programs*, Springer Verlag.
- Miller, and W. T. et al. (1990). Real-Time Dynamic Control of an Industrial Manipulator Using a Neural-Network-Based Learning controller, *IEEE Trans. on RA*, 6(1): 1–9.
- Miyamoto, H., M. Kawato, and T. Setoyama (1988). Feedback error learning neural networks for trajectory control of a robotic manipulator, *Neural Networks*, 1: 251–265.

- Mooring, B. W., Z. S. Roth, and M. R. Driels (1991). *Fundamentals of Manipulator Calibration*, John Wiley and Sons.
- Nakamura, Y. and H. Hanafusa (1987). Optimal redundancy control of robot manipulators, *International Journal Robotics Res.* **6**(1): 32–42.
- Nilsson, N. J. (1980). *Principles of Artificial Intelligence*, Tioga, Palo Alto, CA.
- Noble, B. (1976). *Methods for Computing the Moore–Penrose Generalized Inverse and Related matters*, (ed.) M. Z. Nashed, Academic Press.
- Park, J. and S. Lee (1990). Neural computation for collision-free path planning, *IEEE International Joint Conference on Neural Network*, Vol. 2, pp. 229–232.
- Parker, J. K. and D. E. Goldberg (1989). Inverse Kinematics of Redundant Robots using Genetic Algorithms, *Proc. of the IEEE International Conference on Robotics and Automation*.
- Pau, L. F. (1988). Sensor data fusion, *Journal Intelligent Robotic Systems*, **1**: 103–16.
- Paul, R. P. (1981). *Robot Manipulators: Mathematics, Programming and Control*. Cambridge, MIT Press.
- Pearl, J. (1984). *Heuristics*, Reading, MA: Addison-Wesley.
- (1988). *Probabilistic Reasoning in Intelligent Systems*, San Francisco, Morgan Kaufmann.
- Pichler, F. (1989). CAST Modelling Approaches in Engineering Design, *Lecture Notes in Computer Science* Vol. 410, pp. 52–68.
- Prasad, B., (ed.) (1989). *CAD/CAM Robotics and Factories of the Future*. Springer Verlag.
- Preparata, F. P. and M. I. Shamos (1985). *Computational Geometry: An Introduction*, Springer Verlag New York.
- Ranky, P. G. and C. Y. Ho (1985). *Robot Modeling. Control and Applications with Software*, Springer Verlag.
- Red, W. E. (1983). Minimum distances for robot task simulation, *Robotica*, **1**: 231–38.
- Rogalinski, P. (1994). An approach to automatic robots programming in the flexible manufacturing cell, *Robotica*, **11**.
- Rozenblit, J. W. and B. P. Zeigler (1988). Design and Modeling Concepts, in: *International Encyclopedia of Robotics, Applications and Automation*, (ed.) Dorf R., John Wiley and Sons, New York, pp. 308–22.
- Rumelhart, D. E., G. E. Hinton, and J. L. McClelland (1986a). A General Framework for Parallel Distributed Processing, *Parallel Distributed Processing*, MIT Press, Cambridge.
- Rumelhart, D. E., G. E. Hinton, and R. J. Williams (1986b). Learning Internal Representations by Error Propagation, *Parallel Distributed Processing*, MIT Press, Cambridge.
- Sacerdot, E. D. (1981). Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, **15**(2).
- Sanderson, A. C., L. S. Homem De Mello, and H. Zhang (1990). Assembly Sequence Planning, *AI Magazine*, **11**(1): Spring.
- Saridis, G. N. (1983). Intelligent robotic control, *IEEE Trans. Automatic Control* **AC-28**: 5.
- (1989). Analytical formulation of the principle of increasing precision with decreasing intelligence for intelligent machines, *Automatica*.
- Schwartz, J. T. (1981). Finding the minimum distance between two convex polygon, *Inform. Process. Lett.*, **13**: 168–170.
- Sciavicco, L. and B. Siciliano (1988). A solution algorithm to the inverse kinematic problem for redundant manipulators, *IEEE Trans. on Robotics and Automation*, RA-4(4), 403–410.
- Shin, K. and N. McKay (1985). Minimum Time Control of Robotic Manipulator with Geometric Path Constrains, *IEEE Trans. on Automatic Control*, **30**(6): 531–541.
- (1986). A Dynamic Programming Approach to Trajectory Planning of Robotic Manipulators, *IEEE Trans. on Automatic Control*, **31**(6): 491–500.
- Sierocki, I. (1996). An Automata Approach to Design of Feedback Control for an autonomous Agent, *Proc. of Intern. Conference EMCSR'96*, Vienna.

- Speed, R. (1987). Off-line Programming of Industrial Robots, *Proc. of ISIR 87*, 2110–123.
- Spong, M. and M. Vidyasagar (1989). *Robot Dynamics and Control*. Cambridge, MIT Press, Cambridge.
- Sugeno, M. (1985). *Industrial application of fuzzy logic*, North-Holland, Amsterdam.
- Thrun, S. (1994). A lifelong learning perspective for mobile robot control, *Proc. of IEEE Conference on Intelligent Robots and System IROS'94*, Munich, Germany.
- Thrun, S. and T. Mitchell (1993). Integrating inductive neural network learning and explanation-based learning, *Proc. of IJCAI'93*, Chamberry, France.
- Tilli, T. (1992). *Fuzzy Logic*, Franzis Press, Munich.
- Valavanis, K. P. and H. M. Stellakis (1991). A general organizer model for robotic assemblies, *IEEE Trans. SMC* **21**(2).
- Vukobratovic, M. and M. Kircanski (1984). A Dynamic Approach to Nominal Trajectory Synthesis for Redundant Manipulator, *IEEE Trans. on System, Man, Cybernetics*, **SMC-14**(4).
- Vukobratovic, M. and N. Kircansky (1985). *Scientific Fundamentals of Robotics, Real-Time Dynamics of Manipulation Robots*. Springer Verlag.
- Wang, H. P. and J. Li (1991). *Computer Aided Process Planning*. Elsevier.
- Wang, L. and J. Mendel (1992). Generating Fuzzy Rules by Learning from Examples, *IEEE Trans. SMC*, **22**(6).
- Wang, Z., E. Nakano, and T. Matsukawa (1996). Realizing cooperative object manipulation using multiple behaviour based robots, *Proc. of IROS'96*, Osaka, pp. 310–317.
- White, H. (1990). Connectionist Nonparametric Regression: Multilayer Feedforward Networks Can Learn Arbitrary Mappings, *Neural Networks*, 535–549.
- Whitney, D. E. (1972). The Mathematics of Coordinated Control of Prosthetic Arms and Manipulators, *ASME International Journal of Dynamic Systems, Measurement, and Control*, **94**(12).
- William, H. (1990). *Numerical recipes in C*, Cambridge, Univ. Press.
- Wloka, D. W., ed. (1991). *Robot Simulation*, Springer Verlag, (in german).
- Wolter, J. D., R. A. Volz, and A. C. Woo (1985). Automatic Generation of Gripping Positions, *IEEE Trans. System, Man Cybernetics*, **SMC-15**(2): Mar.
- Wooldridge, M., J. P. Mueller, and M. Tambe (ed) (1996). *Intelligent Agents*, Lecture Notes in Computer Science 1037, Springer Verlag, Berlin, New York.
- Wu, C. H., J. Ho, and K. Y. Young (1988). Design of Robot Accuracy Compensator After Calibration, *Proc. of the IEEE International Conference on Robotics and Automation, Volume 1* Computer Society Press.
- Wu, C. M., B. C. Jiang, and C. H. Wu (1993). Using neural networks for robot positioning control, *Robotics and Computer Integrated Manufacturing*, **10**(3): 153–168.
- Yeung, D. T. and G. A. Bekey (1989). Using a context-sensitive learning network for robot arm control, *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1441–1447.
- Yoshikawa, H. and T. Holden (ed.) (1990). *Intelligent CAD*. North Holland.
- Yoshikawa, T. (1985). Manipulability of robotics mechanisms, *International Journal Robotics Res.* **4**(2): 3–9.
- Zadeh, L. A. (1973). Outline of a new approach to the analysis complex system and decision processes, *IEEE Trans. SMC*, **3**.
- Zeigler, B. P. (1984). *Multifaceted Modeling and Discrete Event Simulation*, Academic Press.
- Ziegert, J. and P. Datsaris (1988). Basic Considerations for Robot Calibration, *Proc. of the IEEE International Conference on Robotics and Automation*, IEEE, New York.

Index

- Acceleration, trajectory planner, 127
- Acceptor, workcell state recognizer, 211–213
- Action interpreter, 225, 237–240
- Actions, *see also* Execution level
 - learning for prediction and coordination, 230–233
 - planning, 6, 14, 225; *see also* Planning cell actions
- Active learning, 203, 228
- Active mode
 - monitoring, 255–256
 - neural model of robot dynamics, 109–112
- Active model, workstation, 158–162
- Advance, sensor model, 181, 182
- Agent, event-based modeling and control of, 165–169
- Allocation relation, 35, 36
- Assembly tasks, 34, 35–40
- Attractive potential, fine motion planning, 129
- Automatic planning
 - action, 14
 - collision-free path planning, 147
 - task-level, 54
- Autonomous agents, 10, 15–17
 - classification of, 2
 - distributed control, 219–221
- Backpropagation network, 60, 115–117, 205
- Behavior of workcell, 17–19
- Bidirectional search, trajectory planning, 122–123, 127
- BRAEN, 40
- Calibrated kinematic model, 72–74
- Calibrated neural model, computed torque controller, 207
- Calibration
 - kinematics, measurement data interface, 150
 - neural model of robot dynamics, 109–118
 - neural network kinematic model, 67–72
- CAP/CAM systems, 102, 141–151
 - HyRob system structure and design process 148–151
 - intelligent design with ICARS, 143–148
 - structure of ICARS, 141–143
- CARC (computer-assisted robotic cell) control 11–12
- Cartesian space
 - collision-free robot search process, 97
 - discretization of, 56
 - kinematic matrix expressions, 66–67
 - path planning, 82–83
- Category neurons, 229–230
- Cell controller, 4, 12; *see also* Control system
- Cell design: *see* CAP/CAM systems: Workcells
- Cell level of control, 3, 12, 13
- Centralized agent, 15–17
- Centralized robotic system coordinator, 213–219
- Circular-wait-deadlock, 43, 44
- Classification of workcells, 1–2
- Collision-free condition, 215
- Collision-free path planning, 55–99, *see also* Path planner
 - Groplan, 147
 - optimal trajectory of motion, 119–123
- Collision-free path search, 77–78, 91–97
- Collision-free robot configurations, 93–97
- Collision relation, 214–215
- Common-sense rule, 115–116
- Communications networks, 3
- Computational geometry modeling methods, 26–32
- Computed torque controller, 205, 206–209
- Computer-assisted robotic cell (CARC) control task, 11–12
- Conceptual state space, 225, 227–230
- Concurrent process planning, 41
- Conditional path planning, 128–129
- Configuration feasibility testing, 78
- Constant-velocity submodes, 110

- Context-sensitive networks, 60
- Continuous track calculation, 98–99, 100
- Control files, object-oriented implementation of fuzzy organizer, 286–287
- Control system, 3, 4, 10, 12–15, 23; *see also*
 - Execution level; Organization level of system
 - ICARS, 141–148, 149, 151
 - machining task planning, 40
 - multiagent system: *see* Coordination of multiagent system
 - robotic workcell components, 9
 - unit module, path planner in presence of unknown objects in environment, 188
- Control task, computer assisted robotic cell (CARC), 11–12
- Convergence, path planning, 85–86
- Coordination level of control, CARC, 7, 12, 13
- Coordination level of knowledge, 14
- Coordination of multiagent system, 211–240
 - acceptor, workcell state recognizer, 211–213
 - centralized, 213–219
 - distributed, 219–221
 - lifelong-learning-based, real-world systems, 221–240
 - action interpreter, 237–240
 - conceptual state space, 227–230
 - learning of agent actions for prediction and coordination, 230–233
 - Q-learning-based action planner, 233–237
 - structure, 223–227
- Coordinator, centralized, 15, 213–219
- COPLANNER, 40
- Cost
 - organization inputs, 242
 - robot motion and, 120
 - time–trajectory planning, 122
- Cubic spline interpolation, 127
- Deadlock, 43–46
 - path planner in presence of unknown objects in environment, 188
 - sensor model, 181, 182
- Decision making, organization level, 246–253
- Decision system, 171, 188
 - path planner in presence of unknown objects in environment, 188
 - sensor model, 181–182
- Decomposition approach, assembly–disassembly sequences, 39–40
- Decoupled network topology network, 62
- Deforming condition, grasp, 138
- Denavit–Hartenberg model, 72, 104, 149
- Design systems: *see* CAP/CAM systems
- Detour, sensor model, 181
- DFA, 40
- Diagnostic system, 18
- Direct kinematics
 - sinusoidal neural network, 64–67
 - training module, 150–151
- Disassembly sequences, 39
- Discrete event system (DEVS), 18–19; *see also*
 - Event-based modeling; Object-oriented discrete-event simulator
 - control system creation, 155–157
 - distributed control, 219–221
 - event-based modeling and control, 157–164
 - monitoring and updating, 256–259
- Discrete model of robot kinematics, 57–58, 127
- Discretization of space, 56–57
- Distance computing problem, 27–31
- Distributed controls, 16–17
- Distributed coordination of multiagent system, 219–221
- Duration of movement, 175, 192
- Dynamic control, 15, 17
 - machining task planning, 40
 - optimal trajectory of motion, 119, 120, 121
 - time–trajectory planning, 121–122, 126; *see also* Time–trajectory planner
- Dynamic linearization module, neural network-based executor, 203–205
- Encoder resolution, neural network kinematic model calibration, 69
- Equilibrating forces, object-gripper system, 135–136
- Euler–Lagrange equation, 101, 103–107
- Evaluation function, time–trajectory planning, 123
- Event-based modeling
 - agent, 165–169
 - motion planning, 176–182
 - production store, 164–165
 - workstation, 157–164
- Event coordination, 14
- Excluded surfaces, 134
- Execution level, 7
 - agent action in presence of uncertainty, 203–209
 - CARC, 12, 13
 - event-based modeling and control of agent, 165–169
 - event-based modeling and control of workstation, 157–164
 - event-based modeling of production store, 164–165

- Execution level (*cont.*)
 - lifelong learning-based coordinator, 227
 - neural and fuzzy computation-based agents, 169–209
 - intelligent and reactive behavior in presence of uncertainty, 172–176
 - multisensor image processing-based world modeling and decision-making systems, 176–182
 - on-line geometric path planner in presence of unknown object, 182–189
 - on-line time trajectory planner, 189–203
 - reactive executor of agent action in presence of uncertainty, 203–209
 - structure of workcell, 170
- Execution level of knowledge, 14
- Extended model, neural kinematic, 70
- External selection rule
 - coordination of nonautonomous actions, 217–218
 - distributed control, 220–221
- Facility level of flexible manufacturing systems, 3
- Feedback network, update rule, 89, 90
- Feedforward network topology, 60
- Final state machine model of robot kinematics, 74–77
- Fine motion planner, 128–131
- Finger sensors, 140
- Finite state machine model of robot kinematics, 74–77
- First free buffer strategy, 242
- Flexible manufacturing systems (FMS), 2–3, 9–12
- Flow time, 41
- Force, grasp, 137–138
- Force factor, gripper, 138–139
- Forward kinematics, 60–63
- Frames Table, 55, 157
- Friction condition, grasp, 137
- Fundamental plan
 - lifelong learning-based coordinator, 224
 - workcell action, 41
- fuzzyART and fuzzyARTMAP algorithm, 228
- Fuzzy computation-based agents, 169–209
 - intelligent and reactive behavior in presence of uncertainty, 172–176
 - multisensor image processing-based world modeling and decision-making systems, 176–182
 - on-line geometric path planner in presence of unknown object, 182–189
- Fuzzy computation-based agents (*cont.*)
 - on-line time trajectory planner, 189–203
 - reactive executor of agent action in presence of uncertainty, 203–209
- Fuzzy organization level, 7
- Fuzzy organizer, object-oriented simulation, 285–293
- Fuzzy reasoning, organization level, 242–246
- Fuzzy tuner, time–trajectory planner, 195–203
- Gain, fuzzy decision making, 251, 252, 253
- Gear behavior, 69
- Generalization algorithm, neural implementation of, 228–229
- Geometrical model, 24–26, 27, 32
- Geometric control, 15
 - machining task planning, 40
 - time–trajectory planning, 126
- Geometric path planning
 - conditional, 128–129
 - in presence of unknown object, 182–189
- Global methodology of obstacle avoidance, 56
- Global state of workcell statistics, 245
- Global updating, 257–258
- Gradient algorithm, 85–86, 94, 95
- Graphics, GRIM module, 141, 142; *see also* CAP/CAM systems
- Graphics modeling, 26
- GRASP, 40
- Grasping/gripper, 131–140
 - grasp learning, 132–133
 - optimal grasping forces, 133–140
 - selection of mating surfaces, 133–135
 - stability of object in, 135–136
 - symbolic computation-based, 132–133
- Grim module, 141, 142, 143, 144
- GROPLAN, 142, 143, 145–148
- Heuristic function
 - monotone restriction and, 80–82
 - time–trajectory planning, 123–126
- Hierarchical architecture of flexible manufacturing systems, 3, 4
- Hierarchical control, 9, 10, 12–15
- Hopfield-type networks, 60
- HyRob system structure and design process, 148–151
- ICARS
 - HyRob system, 149, 151
 - intelligent design with, 143–148
 - structure of, 141–143
- If–then rules, 195–203, 246

- Inertia submode, 110
- Intelligent controller, 171
- Intelligent control of autonomous robotic system (ICARS): *see* ICARS
- Intelligent and reactive behavior in presence of uncertainty, 172–176
- Intelligent robotic workcell, 2–7, 9–12
 - centralization versus autonomy, 15–17
 - components and definitions, 9–12
 - distributed control of, 16–17
 - hierarchical control, 12–15
 - structure and behavior of, 17–19
 - workcell, 9–12
- Interaction forces, object-gripper system, 135, 136–137
- Internal selection rule, 214–216
- Intersection detection problem, 31–32
- Inverse dynamics, neural network-based executor, 206–209
- Inverse kinematics, 60–63, 82–90; *see also* Neural networks
 - HyRob system, 151
 - with on-line obstacle avoidance, 185–186
 - path planner, 127
 - in Cartesian space, redundant manipulators, 86–92
 - in presence of unknown objects in environment, 188–189
 - sensor data combination algorithm, 178–179
- Jacobian
 - calibrated kinematic model application to, 72–74, 75
 - inverse kinematics builder HyRob system, 151
 - path planning in Cartesian space, 83
 - sigmoid neural network-based training, 63, 64–67
- Jerk minimization, 193, 199, 200, 201, 202, 203
- Job, defined, 42
- Joint motion
 - execution level of system, 175
 - limits, computation of, 191–192
- Joint space discretization, 56
 - collision-free robot search process, 96
 - path planner structure, 80–81
- Joint space position calculation, 174–175
- Joint velocity, 82–83
- Just-in-time strategy, 242, 244
 - fuzzy organizer, 288, 290
 - rule base, 247
- Kinematic models, 57–58
 - neural network, *see also* Neural networks
 - calibration of, 67–72
 - symbolic computation-based neural model of, 67–69, 149
 - successor set generation, 92
 - synthesizer, 150
 - training module, 150–151
- Knowledge base, 10, 14
- Knowledge bottlenecks, 223
- Layout modeling, 26, 32, 144
- Learning, *see also* Neural networks
 - fuzzy decision making, 249
 - neural model of robot dynamics, 109–118
- LEGA, 40
- Lifelong-learning-based systems, real-world, 221–240
 - action interpreter, 237–240
 - conceptual state space, 227–230
 - learning of agent actions for prediction and coordination, 230–233
 - Q-learning-based systems, action planner, 233–237
 - structure, 223–227
- Linear approximation method, sensor model, 178, 179, 180
- Link parameter errors, 69
- Local area networks, 3
- Local methodology of obstacle avoidance, 56
- Local updating, 258
- Logical control, 14, 40
- Logical model, 24, 32
- Lyapunov function, 88, 186–187
- Machine model, final state, 74–77
- Machine selection, 4, 11
- Machining
 - assembly task specification, 37
 - control systems, 4
 - defined, 33
 - flexible manufacturing system architecture, 3
 - operations design, 40
 - planning cell actions, 34–35, 40–43
- Manipulator models
 - position error reduction, 72
 - sensor data combination, 178
- Material handling system, 9
- Material selection, 4
- Materials flow, 5
- Mathematica, 149
- Maximum force condition, grasp, 137–138

- Maximum processing time, 289
- Maximum wait time strategy, 226, 247, 289
- Measurement data interface, robot kinematics trainer, 150
- Minimal energy method, 130–131
- Minimum processing time strategy, 289
- Minimum setup time strategy, 242, 244, 247, 288
- Minimum transfer cost strategy, 242
- Model-based approach, robot group control, 17
- Modeler, lifelong learning-based coordinator, 225
- Modeling of workcell, 4; *see also* Virtual robotic cells
- Monitoring, 4, 7, 255–261
 - object-oriented simulator, 268
 - on-line, 18
 - prediagnosis, 256–261
 - tracing active state of system, 255–256
- Monotone restriction, 80–82
- Motion controller, 4, 192
 - execution level, 172–175
 - neural network dynamics, passively acquired, 203–205
- Motion planning, 5, 6, 7, 19
 - control system problems, 11
 - GROPLAN, 142, 143
 - object-oriented simulator, 264
 - off-line: *see* Robot motion, off-line planning
- Motion track, search technique for, 78–80
- Multiagent system coordination, 16; *see also* Coordination of multiagent system
- Multilayer network topology, 60
- Multisensor systems
 - data combination, path planning, 188–189
 - neural and fuzzy computation-based, 176–182
- Network, 10, 23
- Neural gradient algorithm, 89, 90
- Neural models of robot kinematics, 57–58
- Neural networks
 - execution, 169–209
 - intelligent and reactive behavior in presence of uncertainty, 172–176
 - multisensor image processing-based world modeling and decision-making systems, 176–182
 - on-line geometric path planner in presence of unknown object, 182–189
 - on-line time trajectory planner, 189–203
 - reactive executor of agent action in presence of uncertainty, 203–209
 - HyRob system, 151
 - kinematic model calibration, 67–72
 - Neural networks (*cont.*)
 - kinematics output, 150
 - path planning, conditional, 128–129
 - path planning in Cartesian space, 82–99
 - continuous track calculation, 98–99
 - inverse model of robot kinematics, 82–90
 - search for collision free path, 91–97
 - state transition of system, 91
 - path planning in joint space, 58–82
 - discrete model of robot kinematics, 58–74
 - finite state machine model of robot kinematics, 74–77
 - search strategies for collision-free robot movements, 77–82
 - path planning in presence of unknown objects in environment, 182–189
 - inverse kinematics with on-line obstacle avoidance, 185–186
 - Lyapunov function, method based on, 186–187
 - one-step path planning based on multisensor data combination, 188–189
 - steepest descent method, 187
 - virtual points, 184
 - tuning, backpropagation algorithm, 115–117
- Newton equations, 101
- Nominal kinematics, 69
- Non-autonomous agents, 1, 10
- Null rule, 116–117
- Object modeling, 25
 - Grim module, 144
 - virtual cell modeling, 25
- Object-oriented discrete-event simulator, 261–293
 - fuzzy organizer implementation, 285–293
 - object classes, 289–290
 - optimal values, 290–293
 - organizer, 285–289
 - object classes, 269–285, 289–290
 - buffer, 270, 271
 - device, 273
 - equipment, 270, 271, 272–273
 - event, 283–284
 - InputConveyer, 276–277
 - operation, 280–281
 - OutputConveyer, 277–278
 - part, 282–283
 - Robot, 278–279, 280
 - store, 275–276
 - task, 281–282
 - workstation, 274, 275
 - specification, 262–269

- Objects/obstacles, environmental
 - neural network-based path planner, 182–189
 - sensor model, 181–182
- Obstacle avoidance, 56
- Off-line planning, 171; *see also* Robot motion, off-line planning
- Off-line simulation, 17
- One-step path planning based on multisensor data combination, 188–189
- One-step trajectory planner, 190–195
- On-line obstacle avoidance, 185–186
- On-line planning
 - geometric path planner in presence of unknown object, 182–189
 - time trajectory planner, 189–203
- Operating plan, 4, 5
- Operational control, 14
- Optimal grasping forces, 133–140
- Optimization, time–trajectory planning, 122
- Organization level of knowledge, 14
- Organization level of system, 7, 241–253
 - control, 12, 13
 - fuzzy reasoning, 242–246
 - rule base and decision making, 246–253
 - task of organizer, 241–242
- Organizer input fuzzification, 245
- Output function g modeling, 59–60
- Overshoot minimization, 194, 199, 203

- Parameter tuning, 113–114
- Partial derivative computation, 102
- Partially autonomous agents. 1, 10
- Passive learning
 - computed torque controller, 207, 208, 209
 - generalization algorithm implementation, 228
 - neural network-based executor, 203–205
- Passive mode, neural model of robot dynamics, 109, 112–113
- Passive model, workstation, 163–164
- Passive state register of robot model, 168–169
- Path planner, 127, 147
 - execution level of system, 171
 - motion controller, 172–173
 - segment planning step, 173–175
 - of manipulator, 55–99
 - neural and discrete models of robot kinematics, 57–58
 - neural network-based planning in Cartesian space, 82–99; *see also* Neural networks
 - neural network-based planning in joint space, 58–82; *see also* Neural networks
 - optimal trajectory of motion, 119–123
- Path planner (*cont.*)
 - neural network and fuzzy computation-based in presence of unknown object, 182–189
 - inverse kinematics with on-line obstacle avoidance, 185–186
 - Lyapunov function, method based on, 186–187
 - one-step path planning based on multisensor data combination, 188–189
 - steepest descent method, 187
 - virtual points, 184
- Pause, sensor model. 181, 182
- Penalty function, 80, 93
- Performance function, 94
- Planning, 4, 17, 18; *see also* Execution level; Path planners; Robot motion, off-line planning
- Planning cell actions, 33–54
 - methods for, 38–43
 - assembly task, 38–40
 - machining, 40–43
 - on-line: *see* On-line planning
 - production routes, fundamental plans of action, 43–54
 - process route, algorithm for, 48–50
 - process route interpreter, 50–54
 - route planning, quality criterion, 43–48
 - task specification, 33–38
 - assembly, 35–37
 - machining, 34–35
- Polyoptimization problem, 194
- Potential field method, 129–130
- Precedence relation, 34, 35, 36
- Prediagnostics, 256–261
- Priority-based start-stop synchronization, 215, 216
- Process definition, 3
- Processing steps, 4
- Processing time, 41, 289
- Process planner, 19, 141, 142, 143, 144
- Process route, 40
 - algorithm for, 48–50
 - interpreter, 50–54
- Process sequencing, 4
- Product flow simulation, 5
- Production route planning, 43–54
 - machining task, 42
 - process route, algorithm for, 48–50
 - process route interpreter, 50–54
 - quality criterion, 43–48
- Production store, event-based modeling of, 164–165

- Production system, robotic workcell components, 9
- Production time, organization inputs, 242
- Productivity, organization inputs, 242
- Programming, 5, 14
- Projection method, sensor model, 179, 180
- Proportional excitement rule, 116
- Pseudoinertia matrices, 104, 105, 106
- PUMA-like robot
 - one-step trajectory planning, 197
 - robot dynamics modeling, 102–107
- Push strategy, 242, 244
 - object-oriented implementation of fuzzy organizer, 288
 - rule base, 247
- Q-learning-based systems, action planner, 233–237
- Quasistatic submode, 110
- Ranging, sonar, 177–178
- Reactive behavior in presence of uncertainty, 172–176, 203–209
- Real cell, 23
- Real-time monitoring: *see* Monitoring
- Recall mode, 230
- Recurrent loop, 29
- Redundant manipulators, path planning in Cartesian space, 86–92
- Registration, workstation model, 163–164
- Resource allocation policy, route planning, 43–46
- Resource management, 4
- Resources, object-oriented simulator, 262–264
- Reward function, Q-learning, 234
- Robot group cooperation, 16–17
- Robotic workcells: *see* Workcells
- Robot motion, off-line planning, 55–140
 - collision-free path planning of manipulator, 55–99
 - neural and discrete models of robot kinematics, 57–58
 - neural network-based planning in Cartesian space, 82–99; *see also* Neural networks
 - neural network-based planning in joint space, 58–82; *see also* Neural networks
 - fine motion, 128–131
 - grasping, 131–140
 - time trajectory planner, 99–126, 127
 - modeling of robot dynamics, 99, 101–107
 - neural network-computed dynamics for, 121–126, 127
- Robot motion, off-line planning (*cont.*)
 - time trajectory planner (*cont.*)
 - optimal trajectory planning problem, 118–121
 - symbolic and neural network-computed robot dynamics, 107–117
- Route planning
 - production: *see* Production route planning
 - Taskplan, 145
- RPY angles, 73
- RPY coordinates, 66–67, 74
- Rule base, organization level, 246–253
- Scheduling, 4
- Scheduling fuzzification, 244
- Search strategies, 91–97
 - collision-free robot movements, 77–82
 - for collision-free robot movements, 77–78
 - time–trajectory planning, 122–123
- Search technique for motion track, 78–80
- Selection rules, coordination of
 - nonautonomous actions, 214–218
- Sensor data combination, 178–181
- Sensors
 - conceptual states, 232, 233
 - grasp planning, 140
 - multisensor image processing-based world modeling and decision-making systems, neural and fuzzy computation-based, 176–182
 - one-step path planning based on multisensor data combination, 188–189
- Shortest processing time strategy, 289
- Sigmoidal neural network approach, 60–63
- Similarity radius, 228
- Simple model, neural kinematic, 70
- Simulation, *see also* Virtual robotic cells
 - fuzzy decision making, 249, 250
 - GRIM, 141, 142, 143, 144
 - object-oriented: *see* Object-oriented discrete-event simulator
 - off-line, 17
 - on-line, 18
- Sinusoidal function, 108
- Sinusoidal neural network, direct kinematic modeling, 64–67
- Sonar ranging, 177–178
- Stage function, machining process, 42
- Start–stop synchronization, priority-based, 215, 216
- State transition function, 91
- Statistics, object-oriented simulator, 268–269

- Status functions, machining process, 42
- Steepest descent method, 187
- Successor set generation, 92
- Surfaces, unreachable, 134
- Symbolic calculation-based grasp learning, 132–133
- Symbolic calculation-based grasp planning, 140
- Symbolic calculation-based kinematics, 67–69, 149
- Synchronization, workstation model, 164

- Task fuzzification, 243
- Task management, 3, 4
- Taskplan, 144
 - ICARS, 141, 142
 - route planning, 145
- Task planning, 5
- Tasks
 - assembly and machining, 33–38
 - object-oriented simulator, 265–267
 - time–trajectory planning, 126
- Technological operations, machining task specification, 34
- Technological task, 23
- Technological task fuzzification, 243
- Thermal effects, neural network kinematic model calibration, 69
- Time planning
 - Groplan, 147–148
 - organization inputs, 242
- Time–trajectory planner, 99–126, 127
 - Groplan, 147–148
 - modeling of robot dynamics, 99, 101–107
 - neural network and fuzzy logic-based, 189–203
 - fuzzy tuner, 195–203
 - one-step trajectory planner, 190–195
 - neural network-computed dynamics for, 121–126, 127
 - on-line, 189–203
 - optimal trajectory planning problem, 118–121
 - symbolic and neural network-computed robot dynamics, 107–117
- Tooling, machining task planning, 40
- Tool selection, 4
- Torque controller, 205, 206–209
- Tracking, sensor model, 181
- Tractability bottlenecks, 223
- Training mode, 230
- Training module, direct kinematics, 150–151
- Training pattern preparation, grasping forces, 139–140

- Trajectory planning, 127, 192–193; *see also* Time–trajectory planner
 - execution level of system, 171, 175–176
 - one-step, path planner connection, 189
 - optimal motion, 119–123
- Tuning
 - fuzzy organizer optimization, 290
 - neural network-based executor, 205
- Two-dimensional manipulator, 60–63

- Uncertainty
 - reactive behavior in presence of, 172–176
 - reactive executor of agent action in presence of, 203–209
- Unknown objects, neural network-based path planner, 182–189
- Unreachable surfaces, 134
- Update ratio, 30
- Update rule for feedback network, 89, 90
- Updating, monitoring, 257–258

- Value function, Q-learning, 234
- Velocities, trajectory planner, 127
- Virtual points, path planning, 184
- Virtual robotic cells, 4, 23–32
 - computational geometry methods, 26–32
 - defined, 23
 - distance computing problem, 27–31
 - geometrical model, 24–26, 27
 - intersection detection problem, 31–32
 - layout modeling, 26
 - logical model, 24
 - object modeling, 25
- Virtual workcell: *see* CAP/CAM systems

- Wait function, 42, 43–46
- Waiting time, 41, 242
- Workcells, 1–7, 9–12; *see also* Intelligent robotic workcell
 - object modeling, 25
 - virtual: *see* CAP/CAM systems; Virtual robotic cells
- Work-in-process, 268, 285, 290
- Workspace point selection, 174
- Workstation, 3, 4, 23–24
 - event-based modeling and control of, 157–164
 - interaction with agent, modeling, 160–162

- XAP, 40

- Zero-reference position model, 72
- Zones, route planning, 43