Chapter 13

For Further Study

To deepen your understanding of the methods of this part, first you can study optimization theory, as this forms the basis of all the methods. Two good books on optimization that, among others, have influenced the development here are [68], and the earlier book [337]. To learn more about neural networks and their training, see [130, 238]. The reader wishing to strengthen her or his background in conventional adaptive control should consult [254] (or the earlier books [448, 376, 219, 30]). For an in-depth treatment of stable adaptive estimation and control using fuzzy and neural systems, see [484].

Cognitive Neuroscience of Learning: The descriptions of classical and operant conditioning were based on [152, 223, 268, 269]. The description of the conditioned learning mechanisms in the *Aplysia* was taken from [267, 223, 269]; for more relevant literature in this area, see [130]. It is interesting to note that habituation can occur in microorganisms (e.g., *Vorticella* and nematodes), and it seems that learning of simple behavioral rules can occur in flatworms [161]. The discussion here on Hebbian learning is based on [206, 130]. For more details on modeling and analysis of learning processes from the field of theoretical neuroscience, see [130] and the references therein. Of particular relevance to this book is their coverage of the modified Rescorla-Wagner model used at the neural level for representing classical conditioning, the discussions on modeling of Hebbian learning and its connections with both deterministic and stochastic gradient methods, and the "tuning curves" (e.g., see pp. 14–17) and their connection to function approximation (see pp. 316–321) by viewing them as basis functions. The Rescorla-Wagner model studied in psychology and related mathematical and computer representations of the learning process are studied in [152] (pp. 109–119) and the references therein.

An early study growing from the field of cybernetics is given in [26] where the author seeks to explain the origins of adaptive behavior (learning). While research has often focused on organisms with a neural network when studying learning, there have been studies of microorganisms that can demonstrate behavioral plasticity via training [208].

Function Approximation and System Identification: It is helpful if you study the theory of system identification and for this, it is recommended that you see [331]. It could be helpful to study approximation and regularization theory, and one window into the mathematical literature you may want to consider is [422]. The section on approximation theory, and in particular, the section on whether to use linear or nonlinear in the parameter approximators, used the ideas in [45]. An introduction to the topical area of this part is given in [469, 265], where the authors also cover wavelets, and other approximators and properties in some detail. Wavelet methods for nonlinear identification are studied in [175].

Neural Control: A method that has been popular in the control of robots is the cerebellar model articulation controller (CMAC), which was first introduced in [9], and later applied in a different form in [362, 286]. An early paper on neural networks for control is given in [378]. A very nice introduction to learning control is given in [176]. Although developed independently, the FMRLC approach discussed in Section 9.4, is related to the neural control method in [286]. A nice overview of neural control methods is given in [171] and in [252, 361]. For a method that also adapts the structure of the neural network, see [322]. A related topic is that of "neural dynamic optimization for control," where optimal control laws are approximated [460].

Adaptive Fuzzy Control: The FMRLC was introduced in [300, 301] and uses ideas from the linguistic self-organizing controller (SOC) presented in [429] (with applications in [451, 507, 255, 121, 120, 119, 547]) and ideas in conventional model reference adaptive control. The ship steering application was developed from the work in [30, 376, 301, 412] and other applications of the FMRLC are studied in [555, 297, 370, 302, 300, 560, 303]. Other methods and relevant work are contained in [155, 266, 80, 530, 46, 505, 466, 435, 222, 221, 117, 118, 49, 80, 528, 80, 320], but note that there are many other methods that have been developed and reported in the literature.

Expert, Planning, and Attentive Systems in Adaptive Control: In addition to [193], the authors in [521, 395, 132, 516, 373] study fuzzy supervisory controllers that tune conventional controllers, especially ones that tune PID controllers (there are many conventional PID auto-tuning methods [28, 311]). Conventional gain scheduling has been studied extensively in the literature, especially for a wide range of practical applications. See [461, 443, 462] for some theoretical studies of gain scheduling. The connections between fuzzy supervision and gain scheduling have been highlighted by several researchers. A more detailed mathematical study of the connections is provided in [399]. The idea of using a supervisor for conventional adaptive controllers was studied earlier in [27, 21]. A case study for supervisory control of a two-link flexible robot was presented in [371]. The approach to supervision there bears some similarity to the one in [319]. A case study for a fault-tolerant aircraft control problem, where a rule based system supervises an adaptation mechanism to achieve performance adaptive control, is given in [297]. General issues in hierarchical fuzzy control are discussed in [133].

A survey of model predictive control is given in [192]. Adaptive fuzzy model predictive control is studied for a process control problem in [389] and there have been other similar studies for fuzzy model predictive control on which some of this work was based [237, 236]. The section on dynamically focused learning, an attentive mechanism for adaptive fuzzy systems, is based on [296].

Next, note that multisensor integration [339] and a variety of applications [11] utilize a concept called "world modeling" where a model of the environment is built while the system operates and information from the model is used in decision-making. While there are some relationships between systems that exploit a world model, and those in adaptive model predictive control, general world modeling is an important topic in its own right as it represents a very general philosophy on model building.

Finally, note that the area of learning automata is relevant to the topics studied in this part (e.g., in modeling learning systems and analysis of stochastic learning systems). For an introduction to that area, see [380]. The area of learning Bayesian networks from data is covered in [383].

Linear Least Squares: There are many methods to train neural networks and fuzzy systems. There are many books on neural networks (see, e.g., [238, 262]). For other methods to train fuzzy systems, consider [412], [262], [530, 531], or [242, 37].

The idea of using least squares to train fuzzy systems was first introduced in [504] and later studied in [532] and other places. Numerical issues for least squares methods are discussed in [331, 103, 332] and model validity is studied in [70]. The controller construction problem where process operator data is used was taken from [498], as was the CO_2 estimation problem for the gas furnace studied in an exercise at the end of Chapter 10. Issues in how to determine which inputs to use for an estimator are discussed in [331, 104, 260, 498, 262].

Gradient Methods: If you are interested in connections between gradient methods and learning in neuroscience, the first area to study is modeling Hebbian learning [241], specifically when it is modeled as a gradient method. For this, you can study [130, 238] and the references therein.

For more details on gradient methods, see [337, 68]. The brief discussion on the stochastic gradient method is based on [69]. For more background on stochastic optimization, see [439], where the "stochastic approximation" method was introduced, and [293] (the classical backpropagation method is a stochastic gradient approach, since it uses a steepest descent gradient approach and random presentation of data from the training data set).

The hybrid methods (e.g., methods that may use one optimization method for the nonlinear part of the approximator structure and another for the linear part) have been used by a variety of researchers; a particularly nice set of applications were studied in [262, 259, 261]. Some clustering methods are overviewed in [43]. A method that combines an online clustering and least squares method is given in [102]. A variety of clustering methods are discussed in [147], where the authors also focus on construction of local models that are useful for the development of control systems.

Some other methods related to the topics in this part are given in [251, 48, 545, 306, 1, 253, 378, 80].

Stable Adaptive Fuzzy/Neural Control: Here, the treatment was only meant to introduce the topic of stable adaptive fuzzy/neural control. A recent text that covers the full details of many stable adaptive neural/fuzzy methods is [484]. There, more general direct and indirect adaptive control methods are introduced, the output feedback and multivariable cases are discussed in detail, many examples and applications/implementations are provided, and discrete-time and decentralized adaptive control are covered. It seems that the field of stable neural control started with [424] and has been significantly affected by the work in [423, 424, 441, 172, 174, 167, 321, 552, 316, 379, 447, 101, 546, 445, 425, 105], where the authors make use of neural networks as approximators of nonlinear functions. In [497, 248, 529, 304, 99, 530, 486], the authors use fuzzy systems for the same purpose and [441, 379] use dynamical neural networks. The neural and fuzzy approaches are most of the time equivalent, differing between each other mainly in the structure of the approximator chosen. Indeed, to try to bridge the gap between the neural and fuzzy approaches, several researchers (e.g., in [486]) introduce adaptive schemes using a class of parameterized functions that include both neural networks and fuzzy systems. Linear in the parameter approximators are used in much of the above-referenced work, and for example, [497, 248, 423, 424, 92, 172, 167, 99, 447, 445, 486] and nonlinear in, for example, [321, 552, 316, 379, 101, 546, 425]. Note that most of the papers deal with indirect adaptive control, whereas very few authors use the direct adaptive control approach (see, however, [486, 442]). Research on decentralized adaptive neural/fuzzy control is given in [487] and for the MIMO case (both direct and indirect) in [396]. Persistency of excitation issues are studied in [173, 172]. An interesting study on issues related to the use of local (finite support) approximators in adaptive control can be found in [170]. Implementation studies for adaptive neural fuzzy control are given in [397]. For more information on multiple model adaptive control, see [31, 347, 333, 377, 375] and the references therein.

The aircraft wing rock model used in the design problem in Section 12.7 was taken from [382, 165], which is based on wind tunnel data as studied in [309]. The aircraft wing rock design problem at the end of Section 12.7 was taken from [289] and is based on [247].

Approximator Structure Learning: For an overview of methods for automatically constructing or pruning neural networks, see [295, 431] and for discussion on some methods to adjust structure of fuzzy systems, see [412]. The issue of structure choice is treated in [103] for linear in the parameter approximators where you want to choose the number of basis functions. There, least squares methods are used to eliminate basis functions that do not significantly contribute to making the approximation accurate. Such methods could be employed in another method to initialize the nonlinear portion of the approximator [331] where you simply pick a very large approximator structure, then eliminate the parts that do not contribute in a significant way. Some work in the direction of trying to tune structure of an approximator in an adaptive control system is contained in [198] (but also see the references there).

Immune Networks: While not discussed in this chapter, there has been recent interest in biomimicry of immune systems and subsequent engineering applications (e.g., in pattern recognition and control) [124]. For more recent work, see [125]. Some in fact think of the immune system as a second type of "brain" in the human body, with the ability to learn (e.g., it encounters a type of pathogen, then "learns" how to more easily recognize it the next time) and make decisions (e.g., how and when to attack foreign invaders). Immune networks are models of immune systems and some such networks are "connectionist" in similar ways to how neural networks are [168], and some types of immune networks have underlying mechanisms that are sometimes thought of as being similar to evolutionary optimization (e.g., the genetic algorithm) [182], since their learning strategy can be viewed as a type of nongradient stochastic optimization process. There have been several studies of underlying mechanisms for learning in immune networks [169, 417, 66, 63, 67, 249, 250, 65, 471] and the application of these ideas to control [63, 64] has been considered.

A recent study [146] focuses on tuning approximator structure and parameters and these ideas are more firmly connected with the ideas and methods of this part.

Temporal Difference Learning and Neuro-Dynamic Programming: An introduction to the area of reinforcement learning, and in particular "temporal difference learning" is given in [499], and connections to neuroscience are discussed. Related methods and analysis of temporal difference learning are studied in the area of "neuro-dynamic programming" [69], where the authors also study the application to a number of multi-stage decision-making system problems. In neuro-dynamic programming, an approximator, such as a neural network, is used to approximate the "optimal cost to go" in the dynamic programming methodology, and then it is used to make choices of decision variables. Many other references are available in this general area, so you should search the current literature.

Part IV

EVOLUTION

Part Contents

Sequence of Essential Concepts

- Evolution is a type of search that continually and incrementally redesigns the structure and parameters of organisms to maximize organism fitness for survival in an uncertain environment. To do this, it tends to optimize the design of the organism for typical characteristics of its environment (i.e., some type of average environment it encounters) and thereby produce an organism that is "robust" for survival in its habitat. (Extinction processes are due to insufficient adaptation rate or "traps" in the search space due to coupled constraints like physiology and environment.) Engineering design is analogous to evolution, but for technological products and systems.

- Genetic algorithms simulate evolution and hence, can serve as a general tool for parallel stochastic nongradient based optimization. There are, however, many closely related deterministic and stochastic conventional approaches (i.e., not biologically motivated), including response surface, pattern search, simplex, and stochastic optimization methods. These provide insights into the operation of biologically motivated optimization methods, such as the genetic algorithm (or the foraging algorithms of Part V). Moreover, they provide practical approaches for solving engineering problems that involve robust optimal design.

- Evolution is best viewed as a type of global optimization process ("global" in time and population space) that can act on all aspects of the organism, including its ability to perform learning, which can be viewed as a "local" adaptive search. The environment is the fundamental driver of this optimization process. Learning is "local" in time, since it applies to a single generation and local in space, since it occurs in a single individual (but of course, culture has more global influences on learning in groups of organisms). Learning can accelerate evolution (the "Baldwin effect") and evolution can shape learning (it can design every aspect of the learning system). Evolution may produce an optimal balance between instincts and learning capabilities that is dependent on characteristics of the environment (e.g., the stochastic nature of the environment) and organism. These ideas provide some principles in the design of robust optimal complex decision-making systems.

- Genetic algorithms are optimization processes that can be employed to tune approximators in closed-loop systems and hence, can achieve real-time adaptive control. Direct and indirect adaptive controllers and adaptive model predictive controllers can be designed using genetic algorithms.