

Chapter 2

Theory and Background

In this chapter we present some basic concepts about the work to understand better the idea and the context of this book.

2.1 Fuzzy Logic

Fuzzy logic can be conceptualized as a generalization of classical logic. Modern fuzzy logic was developed by Lotfi Zadeh in the mid-1960s to model those problems in which imprecise data must be used or in which the rules of inference are formulated in a very general way making use of fuzzy categories. In fuzzy logic, which is also sometimes called multiple-valued logic, there are not just two alternatives but a whole continuum of truth values for logical propositions. A proposition A can have the truth value of 0.4 and its complement A^c the truth value of 0.5. According to the type of negation operator that is used, the two truth values must not be necessarily add up to 1. Fuzzy logic has a weak connection to probability theory. Probabilistic methods that deal with imprecise knowledge are formulated in the Bayesian framework, but fuzzy logic does not need to be justified using a probabilistic approach. The common route is to generalize the findings of multi-valued logic in such a way as to preserve part of the algebraic structure [1–7].

Fuzzy set theory corresponds to fuzzy logic and the semantic of fuzzy operators can be understood using a geometric model. The geometric visualization of fuzzy logic will give us a hint as to the possible connection with neural networks. Fuzzy logic can be used as an interpretation model for the properties of neural networks, as well as for giving a more precise description of their performance. Fuzzy logic can also be used to specify networks directly without having to apply a learning algorithm. An expert in a certain field can sometimes produce a simple set of control rules for a dynamical system with less effort than the work involved in training a neural network. A classical example proposed by Zadeh to the neural network community is developing a system to park a car. It is straightforward to formulate a set of fuzzy rules for this task, but it is not immediately obvious how to build a network to do the same or how to train it. Fuzzy logic is now being used in

many products of industrial and consumer electronics for which a good control system is sufficient and where the question of optimal control does not necessarily arise [8–13].

2.1.1 Fuzzy Set

The difference between crisp (i.e., classical) and fuzzy sets is established by introducing a membership function. Consider a finite set $X = \{x_1, x_2 \dots x_n\}$ which will be considered the universal set in what follows. The subset A of X consisting of the single element x_i can be described by the n-dimensional membership vector $Z(A) = (1, 0, 0 \dots 0)$, where the convention has been adopted that a 1 at the i-th position indicates that x_i belongs to A. The set B composed of the elements x_i and x_n is described by the vector $Z(B) = (1, 0, 0 \dots 1)$. Any other crisp subset of X can be represented in the same way by an n-dimensional binary vector. But what happens if we lift the restriction to binary vectors? In that case we can define the fuzzy set C with the following vector description: $Z(C) = (0.5, 0, 0 \dots 0)$ [14–18].

In classical set theory such a set cannot be defined. An element belongs to a subset or it does not. In the theory of fuzzy sets we make a generalization and allow descriptions of this type. In our example the element x_1 belongs to the set C only to some extent. The degree of membership is expressed by a real number in the interval $[0, 1]$, in this case 0.5. This interpretation of the degree of membership is similar to the meaning we assign to statements such as “person x_1 is an adult”. Obviously, it is not possible to define a definite age which represents the absolute threshold to enter into adulthood. The act of becoming mature can be interpreted as a continuous process in which the membership of a person to the set of adults goes slowly from 0 to 1. There is many other examples of such diffuse statements. The concepts “old” and “young” or the adjectives “fast” and “slow” are imprecise but easy to interpret in a given context. In some applications, such as expert systems, for example, it is necessary to introduce formal methods capable of dealing with such expressions so that a computer using rigid Boolean logic can still process them. This is what the theory of fuzzy sets and fuzzy logic tries to accomplish [19–23].

2.1.2 Fuzzy Logic and Control

A fuzzy controller is a regulating system whose modus operandi is specified with fuzzy rules. In general it uses a small set of rules. The measurements are processed in their fuzzified form, fuzzy inferences are computed, and the result is defuzzified, that is, it is transformed back into a specific real number. The example of the electrical heater is explained to understand the idea in control and fuzzy logic: Assume that the room temperature is a number between 0 and 40 °C.

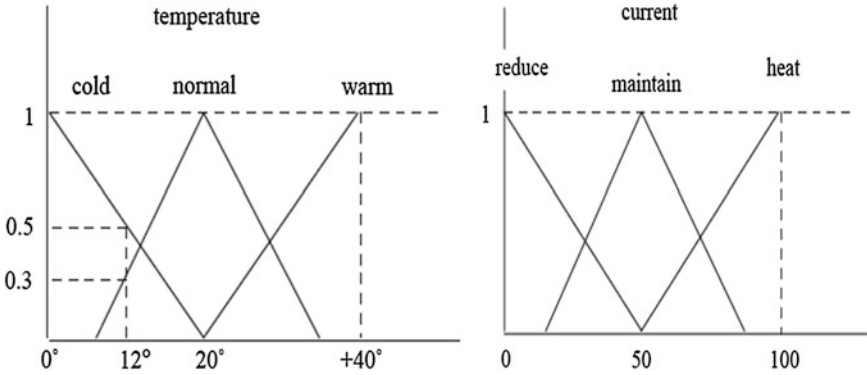


Fig. 2.1 Membership functions for temperature and electric current categories

The controller can vary the electrical power consumed between 0 and 100 (in some suitable units), whereby 50 is the normal stand-by value. The temperature of 12° corresponds to the fuzzy number $T = \text{cold}/0.5 + \text{normal}/0.3 + \text{warm}/0.0$. These values lead to the previously computed inference action = $\text{heat}/0.5 + \text{maintain}/0.3 + \text{reduce}/0.0$. Membership degree of 0.5 represent a set of cold temperature, membership degree of 0.3 represent a set of normal temperature and membership degree of 0.0 represent a set of warm temperature. The controller must transform the result of this fuzzy inference into a definite value. The surfaces of the membership triangles below and the inferred degree of membership are calculated (see Fig. 2.1).

A fuzzy controller is just a system for the rapid computation of an approximation of a coarsely defined control surface. The fuzzy controller computes a control variable according to the values of the variables x and y . Both variables are transformed into fuzzy categories. Assume that each variable is transformed into a combination of three categories. There are nine different combinations of the categories for x and y . For each of these nine combinations the value of the control variable is defined using a fuzzy rule [24, 25].

2.2 Granularity

Information granulation has emerged as one of the fundamental concepts of information processing giving rise to the discipline of Granular Computing. The concept itself permeates through a large variety of information systems the underlying idea is intuitive and appeals to our commonsense reasoning. We perceive the world by structuring our knowledge, perceptions, and acquired evidence in terms of information granules-entities, which are abstractions of the complex word and phenomena. By being abstract constructs, information granules and their ensuing processing done under the umbrella of Granular Computing, provides a

conceptual and algorithmic framework to deal with an array of decision-making, control, and prediction problems. Granular Computing supports human-centric processing, which becomes an inherent feature of intelligent systems. The required level of detail becomes conveniently controlled by making suitable adjustments to the size of information granules and their distribution in the space in which the problem at hand is being positioned and handled. In this sense, an inherent flexibility, which comes hand in hand with Granular Computing, can be effectively exploited in various ways. Three general tendencies encountered in Granular Computing can be identified: (a) a design of information granules of higher order, (b) a development of information granules of higher type, and (c) a formation of hybrid information granules.

The basic concepts of graduation and granulation form the core of fuzzy logic, and are the main distinguishing features of fuzzy logic. More specifically, in fuzzy logic everything is or is allowed to be graduated, i.e., be a matter of degree or, equivalently, fuzzy. Furthermore, in fuzzy logic everything is or is allowed to be granulated, with a granule being a clump of attribute values drawn together by in distinguishability, similarity, proximity, or functionality. The concept of a generalized constraint serves to treat a granule as an object of computation. Graduated granulation, or equivalently fuzzy granulation, is a unique feature of fuzzy logic. Graduated granulation is inspired by the way in which humans deal with complexity and imprecision. The concepts of graduation, granulation, and graduated granulation play key roles in granular computing. Graduated granulation underlies the concept of a linguistic variable, i.e., a variable whose values are words rather than numbers. In retrospect, this concept, in combination with the associated concept of a fuzzy if–then rule, may be viewed as a first step toward granular computing.

Granular Computing (GrC) is a general computation theory for effectively using granules such as subsets, neighborhoods, ordered subsets, relations (subsets of products), fuzzy sets (membership functions), variables (measurable functions), Turing machines (algorithms), and intervals to build an efficient computational model for complex with huge amounts of data, information and knowledge [26, 27].

2.3 Genetic Algorithms

Genetic algorithms (GAs) were proposed by John Holland in the 1960s and were developed by Holland and his students and colleagues at the University of Michigan in the 1960s and the 1970s [8]. In contrast with evolution strategies and evolutionary programming, Holland’s original goal was not to design algorithms to solve specific problems, but rather to formally study the phenomenon of adaptation as it occurs in nature and to develop ways in which the mechanisms of natural adaptation might be imported into computer systems. Holland’s 1975 book *Adaptation in Natural and Artificial Systems* presented the genetic algorithm as an

abstraction of biological evolution and gave a theoretical framework for adaptation under the GA. Holland's GA is a method for moving from one population of "chromosomes" (e.g., strings of ones and zeros, or "bits") to a new population by using a kind of "natural selection" together with the genetics—inspired operators of crossover, mutation, and inversion. Each chromosome consists of "genes" (e.g., bits), each gene being an instance of a particular "allele" (e.g., 0 or 1). The selection operator chooses those chromosomes in the population that will be allowed to reproduce, and on average the fitter chromosomes produce more offspring than the less fit ones. Crossover exchanges subparts of two chromosomes, roughly mimicking biological recombination between two single—chromosome ("haploid") organisms; mutation randomly changes the allele values of some locations in the chromosome; and inversion reverses the order of a contiguous section of the chromosome, thus rearranging the order in which genes are arrayed. (Here, as in most of the GA literature, "crossover" and "recombination" will mean the same thing.)

Holland's introduction of a population—based algorithm with crossover, inversion, and mutation was a major innovation. (Rechenberg's evolution strategies started with a "population" of two individuals, one parent and one offspring, the offspring being a mutated version of the parent; many—individual populations and crossover were not incorporated until later. Fogel, Owens, and Walsh's evolutionary programming likewise used only mutation to provide variation.) Moreover, Holland was the first to attempt to put computational evolution on a firm theoretical footing [28]. Until recently this theoretical foundation, based on the notion of "schemas," was the basis of almost all subsequent theoretical work on genetic algorithms [29–32]. Genetic algorithms (GAs) for rule discovery can be divided into two broad approaches, based on how rules are encoded in the population of individuals ("chromosomes"). In the Michigan approach each individual encodes a single prediction rule, whereas in the Pittsburgh approach each individual encodes a set of prediction rules [8].

2.4 Aviation

There are four fundamental basic flight maneuvers upon which all flying tasks are based: straight and level flight, turns, climbs, and descents. All controlled flight consists of either one, or a combination or more than one, of these basic maneuvers. If a student pilot is able to perform these maneuvers well, and the student's proficiency is based on accurate "feel" and control analysis rather than mechanical movements, the ability to perform any assigned maneuver will only be a matter of obtaining a clear visual and mental conception of it. The pilot should always be considered the center of movement of the airplane, or the reference point from

which the movements of the airplane are judged and described [33–37]. The following will always be true, regardless of the airplane's attitude in relation to the Earth:

- When back pressure is applied to the elevator control, the airplane's nose rises in relation to the pilot.
- When forward pressure is applied to the elevator control, the airplane's nose lowers in relation to the pilot.
- When right pressure is applied to the aileron control, the airplane's right wing lowers in relation to the pilot.
- When left pressure is applied to the aileron control, the airplane's left wing lowers in relation to the pilot.
- When pressure is applied to the right rudder pedal, the airplane's nose moves (yaws) to the right in relation to the pilot.
- When pressure is applied to the left rudder pedal, the airplane's nose moves (yaws) to the left in relation to the pilot.

The preceding explanations should prevent the beginning pilot from thinking in terms of “up” or “down” in respect to the Earth, which is only a relative state to the pilot. It will also make understanding of the functions of the controls much easier, particularly when performing steep banked turns and the more advanced maneuvers. Consequently, the pilot must be able to properly determine the control application required to place the airplane in any attitude or flight condition that is desired. The flight instructor should explain that the controls will have a natural “live pressure” while in flight and that they will remain in neutral position of their own accord, if the airplane is trimmed properly. With this in mind, the pilot should be cautioned never to think of movement of the controls, but of exerting a force on them against this live pressure or resistance. Movement of the controls should not be emphasized; it is the duration and amount of the force exerted on them that effects the displacement of the control surfaces and maneuvers the airplane. The amount of force the airflow exerts on a control surface is governed by the airspeed and the degree that the surface is moved out of its neutral or streamlined position.

Since the airspeed will not be the same in all maneuvers, the actual amount the control surfaces are moved is of little importance; but it is important that the pilot maneuver the airplane by applying sufficient control pressure to obtain a desired result, regardless of how far the control surfaces are actually moved. The controls should be held lightly, with the fingers, not grabbed and squeezed. Pressure should be exerted on the control yoke with the fingers. A common error in beginning pilots is a tendency to “choke the stick.” This tendency should be avoided as it prevents the development of “feel,” which is an important part of aircraft control. The pilot's feet should rest comfortably against the rudder pedals. Both heels should support the weight of the feet on the cockpit floor with the ball of each foot touching the individual rudder pedals. The legs and feet should not be tense; they must be relaxed just as when driving an automobile [38–40].

2.4.1 Attitude Flying

Attitude is the angular difference measured between an airplane's axis and the line of the Earth's horizon. Pitch attitude is the angle formed by the longitudinal axis, and bank attitude is the angle formed by the lateral axis. Rotation about the airplane's vertical axis (yaw) is termed an attitude relative to the airplane's flight path, but not relative to the natural horizon. In attitude flying, airplane control is composed of four components: pitch control, bank control, power control, and trim [41].

- Pitch control is the control of the airplane about the lateral axis by using the elevator to raise and lower the nose in relation to the natural horizon.
- Bank control is control of the airplane about the longitudinal axis by use of the ailerons to attain a desired bank angle in relation to the natural horizon.
- Power control is used when the flight situation indicates a need for a change in thrust.
- Trim is used to relieve all possible control pressures held after a desired attitude has been attained.

2.4.2 Level Turns

A turn is made by banking the wings in the direction of the desired turn. A specific angle of bank is selected by the pilot, control pressures applied to achieve the desired bank angle, and appropriate control pressures exerted to maintain the desired bank angle once it is established. All four primary controls are used in close coordination when making turns. Their functions are as follows:

- The ailerons bank the wings and so determine the rate of turn at any given airspeed.
- The elevator moves the nose of the airplane up or down in relation to the pilot, and perpendicular to the wings. Doing that, it both sets the pitch attitude in the turn and "pulls" the nose of the airplane around the turn.
- The throttle provides thrust, which may be used for airspeed to tighten the turn.
- The rudder offsets any yaw effects developed by the other controls. The rudder does not turn the airplane.

For purposes of this discussion, turns are divided into three classes: shallow turns, medium turns, and steep turns:

- Shallow turns are those in which the bank (less than approximately 20°) is so shallow that the inherent lateral stability of the airplane is acting to level the wings unless some aileron is applied to maintain the bank.
- Medium turns are those resulting from a degree of bank (approximately 20° – 45°) at which the airplane remains at a constant bank.

- Steep turns are those resulting from a degree of bank (45° or more) at which the “overbanking tendency” of an airplane overcomes stability, and the bank increases unless aileron is applied to prevent it [42, 43].

References

1. Cervantes, L., Castillo, O.: Design of a fuzzy system for the longitudinal control of an F-14 airplane. In: *Soft Computing for Intelligent Control and Mobile Robotics*, vol. 318, pp. 213–224. Springer, Berlin (2011)
2. Cervantes, L., Castillo, O.: Intelligent control of nonlinear dynamic plants using a hierarchical modular approach and type-2 fuzzy logic. In: *Lecture Notes in Computer Science*, vol. 7095, pp. 1–12. Springer, Berlin (2011)
3. Cervantes, L., Castillo, O.: Hierarchical genetic algorithms for optimal type-2 fuzzy system design. In: *Annual Meeting of the North American Fuzzy Information Processing Society*, pp. 324–329 (2011)
4. Cervantes, L., Castillo, O.: Automatic design of fuzzy systems for control of aircraft dynamic systems with genetic optimization. In: *World Congress and AFSS International Conference*, pp. OS-413-1–OS-413-7 (2011)
5. Cervantes, L., Castillo, O.: Comparative study of type-1 and type-2 fuzzy systems for the three-tank water control problem. In: *LNAI*, vol. 7630, pp. 362–373. Springer, Berlin (2013)
6. Cervantes, L., Castillo, O.: Genetic design of optimal type-1 and type-2 fuzzy systems for longitudinal control of an airplane. *J. Intell. Autom. Soft Comput.* **20**(2), 213–227 (2014)
7. Chalupa, P., Novák, J., Bobál, V.: Detailed Simulink model of real time three tank system. In: *Proceedings of the 2nd International Conference on Circuits, Systems, Communications and Computers 2011, CSCC'11*, pp. 161–166 (2011)
8. Holland, J.H.: *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor (1975)
9. Lam, H., Leung, F.: Stability analysis of discrete-time fuzzy-model-based control systems with time delay: time delay-independent approach. *Fuzzy Sets Syst.* **159**(8), 990–1000 (2008)
10. Lam, H., Leung, F.: Fuzzy rule-based combination of linear and switching state-feedback controllers. *Fuzzy Sets Syst.* **156**(2), 153–184 (2005)
11. Li, I.-H., Lee, L.-W.: A hierarchical structure of observer-based adaptive fuzzy-neural controller for MIMO systems. *Fuzzy Sets Syst.* **185**(1), 52–82 (2011)
12. Malla, S., Bhende, C.: Voltage control of stand-alone wind and solar energy system. *Int. J. Electr. Power Energy Syst.* **56**, 361–373 (2014)
13. Niemann, H., Stoustrup, J.: Passive fault tolerant control of a double inverted pendulum a case study. *Control Eng. Pract.* **13**(8), 1047–1059 (2005)
14. Oh, S.-K., Jung, S.-H., Pedrycz, W.: Design of optimized fuzzy cascade controllers by means of hierarchical fair competition-based genetic algorithms. *Expert Syst. Appl.* **36**(9), 11641–11651 (2009)
15. Ornelas-Tellez, F., Sanchez, E., Loukianov, A., Rico, J.: Robust inverse optimal control for discrete-time nonlinear system stabilization. *Eur. J. Control* **20**(1), 38–44 (2014)
16. Sung, H., Kim, D., Park, J., Joo, Y.: Robust digital control of fuzzy systems with parametric uncertainties: LMI-based digital redesign approach. *Fuzzy Sets Syst.* **161**(6), 919–933 (2010)
17. Warren, P.: *Mechanics of Flight*, 2nd edn. Wiley, Hoboken, New Jersey (2010)
18. Castillo, O., Melin, P.: A review on the design and optimization of interval type-2 fuzzy controllers. *Appl. Soft Comput.* **12**(4), 1267–1278 (2012)
19. Castillo, O., Melin, P.: New fuzzy-fractal-genetic method for automated mathematical modelling and simulation of robotic dynamic systems. In: *IEEE International Conference on Fuzzy Systems*, vol. 2, pp. 1182–1187 (1998)

20. Chen, G., Pham, T.: Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems. CRC Press, Boca Raton (2001)
21. Dadios, E.: Fuzzy Logic-Controls, Concepts, Theories and Applications (2012)
22. Sefer, K., Omer, C., Okyay, K.: Adaptive neuro-fuzzy inference system based autonomous flight control of unmanned air vehicles. *Expert Syst. Appl. J.* **37**(2), 1229–1234 (2010)
23. Sepulveda, R., Castillo, O., Melin, P., Montiel, O.: An efficient computational method to implement type-2 fuzzy logic in control application. In: *Analysis and Design of Intelligent System Using Soft Computing Techniques 2007*, pp. 45–52 (2007)
24. Zadeh, L.: Fuzzy Sets and Fuzzy Information Granulation Theory. Beijing Normal University Press, Beijing (2000)
25. Zadeh, L.A.: Some reflections on soft computing, granular computing and their roles in the conception, design and utilization of information/intelligent systems. *Soft Comput.* **2**, 23–25 (1998)
26. Yu, Y., Chen, L., Sun, F., Wu, C.: Matlab/Simulink-based simulation for digital-control system of marine three-shaft gas-turbine. *Appl. Energy* **80**(1), 1–10 (2005)
27. Ouyang, P.R., Acob, J., Pano, V.: PD with sliding mode control for trajectory tracking of robotic system. *Robot. Comput. Integr. Manuf.* **30**(2), 189–200 (2014)
28. Castillo, O., Martinez-Marroquin, R., Melin, P., Valdez, F., Soria, J.: Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot. *Inf. Sci.* **192**, 19–38 (2012)
29. Cázarez, N., Aguilar, L., Castillo, O.: Fuzzy logic control with genetic parameters optimization for the output regulation of a servomechanism with nonlinear backlash. *Expert Syst. Appl.* **37**(6), 4368–4378 (2010)
30. Haupt, R., Haupt, S.: Practical Genetic Algorithm. Wiley Interscience, Hoboken (2004)
31. Hidalgo, D., Melin, P., Castillo, O.: An optimization method for designing type-2 fuzzy inference systems based on the footprint of uncertainty using genetic algorithms. *Expert Syst. Appl.* **39**(4), 4590–4598 (2012)
32. No, T.S., Mina, B.M., Stone, R.H., Wong, K.C.: Control and Simulation of Arbitrary Flight Trajectory-Tracking, pp. 560–756. Department of Aerospace Engineering, Chonbuk National University, Deokjin Dong, Chonju NSW (2006)
33. Caughey, D.: Introduction of Aircrafts Stability and Control Course Notes for M&AE 5070. Sibley School of Mechanical and Aerospace Engineering, Cornell University, New York, pp. 14853–17501 (2011)
34. No, T.S., Kim, J.-E., Moon, J.H., Kim, S.J.: Modeling, control, and simulation of dual rotor wind turbine generator system. *Renew. Energy* **34**(10), 2124–21322 (2009)
35. Pedrycz, W., Chen, S.: Granular Computing and Intelligent System, Design with Information Granules of Higher Order and Higher Type. *Intelligent Systems Reference Library*, vol. 13 (2011)
36. Rachman, E., Jaam, J., Hasnah, A.: Non-linear simulation of controller for longitudinal control augmentation system of F-16 using numerical approach. *Inf. Sci. J.* **164**(1–4), 47–60 (2004)
37. Sanchez, E., Becerra, H., Velez, C.: Combining fuzzy, PID and regulation control for an autonomous mini-helicopter. *J. Inf. Sci.* **177**(10), 1999–2022 (2007)
38. Schmidt, D.: Modern Flight Dynamics. McGraw Hill, New York (2012)
39. Song, Q., Song, Y.D.: Generalized PI control design for a class of unknown nonaffine systems with sensor and actuator faults. *Syst. Control Lett.* **64**, 86–95 (2014)
40. Song, Y., Wang, H.: Design of flight control system for a small unmanned tilt rotor aircraft. *Chin. J. Aeronaut.* **22**(3), 250–256 (2009)
41. Tao, C., Taur, J., Wang, C., Chen, U.: Fuzzy hierarchical swing-up and sliding position controller for the inverted pendulum–cart system. *Fuzzy Sets Syst.* **159**(20), 2763–2784 (2008)
42. Zadeh, L.A.: Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. Syst. ManCybern. SMC* **3**, 28–44 (1973)
43. Zhang, Y., Wang, Q., Dong, Ch., Jiang, Y.: H_∞ output tracking control for flight control systems with time-varying delay. *Chin. J. Aeronaut.* **26**(5), 1251–1258 (2013)

Hierarchical Type-2 Fuzzy Aggregation of Fuzzy
Controllers

Cervantes, L.; Castillo, O.

2016, VIII, 69 p. 77 illus., 7 illus. in color., Softcover

ISBN: 978-3-319-26670-1