Chapter 2
MISO Takagi-Sugeno Fuzzy System with Linear Membership Functions

Although we will be especially interested in Takagi-Sugeno models [180] called TS models for short which use linear or polynomial membership functions, we begin our considerations with the single-input and single-output system (SISO TS) which uses nonlinear membership functions. The problem involves determining the fuzzy rules which exactly model a nonlinear function belonging to some class of functions.

2.1 Perfect Approximation of Nonlinear Functions Using the Simplest Takagi-Sugeno Model

Below we will consider the problem of perfect approximation of nonlinear functions using the simplest Takagi-Sugeno model in the context of interpretability of fuzzy sets.

Suppose the input variable of a TS system is $z \in [-\alpha, \beta]$ and its output is S as shown in Fig. 2.1.

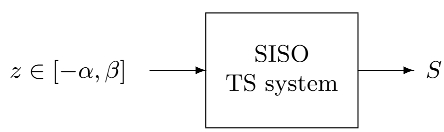

Fig. 2.1 Single-input-single-output TS system defined by the rules (2.1)

We assume that $\alpha + \beta \neq 0$. By N and P we denote two fuzzy sets which will be identified both with their linguistic labels and membership functions: $N(z)$ and $P(z)$, respectively. Thus, $N, P : [-\alpha, \beta] \rightarrow [0, 1]$. The TS system is defined by two fuzzy rules

$$\left.\begin{aligned} R_1 &: \text{If } z \text{ is } N, \text{ then } S = q_1, \\ R_2 &: \text{If } z \text{ is } P, \text{ then } S = q_2. \end{aligned}\right\} \tag{2.1}$$

J. Kluska: Analytical Methods in Fuzzy Modeling and Control, STUDFUZZ 241, pp. 3–24.
springerlink.com

The natural requirements concerning the fuzzy sets are as follows

1. $N(z)$ is a continuous, nonincreasing function of z,
2. $N(-\alpha) = 1$ and $N(\beta) = 0$,
3. $P(z) = 1 - N(z)$.

Observe that P is a continuous, increasing function of z which satisfies boundary conditions: $P(\beta) = 1$ and $P(-\alpha) = 0$. Continuity, monotonicity and preservation of boundary conditions ensure a clear linguistic interpretation of both membership functions.

Suppose some continuous and monotonic function $f(z) : [-\alpha, \beta] \to \mathbb{R}$ is given. The problem is "How to obtain membership functions for the fuzzy rule-based TS system, such that its output is exactly the same, i.e. $S(z) = f(z)$ for any $z \in [-\alpha, \beta]$?" First of all the following conditions

$$q_1 = f(-\alpha), \qquad q_2 = f(\beta), \tag{2.2}$$

$$N(z) = \frac{f(z) - f(\beta)}{f(-\alpha) - f(\beta)}, \tag{2.3}$$

must be satisfied, since the output of the TS system is computed as follows [180]

$$S(z) = \frac{q_1 N(z) + q_2 P(z)}{N(z) + P(z)} = f(z), \qquad \text{for} \quad z \in [-\alpha, \beta]. \tag{2.4}$$

Example 2.1. The model (2.1) exactly approximates the following monotonic and continuous function (see Fig. 2.2)

$$f(z) = -\frac{\cos z}{z + \pi/4}, \qquad \text{for} \quad z \in \left[-\frac{\pi}{6}, \frac{\pi}{2}\right]. \tag{2.5}$$

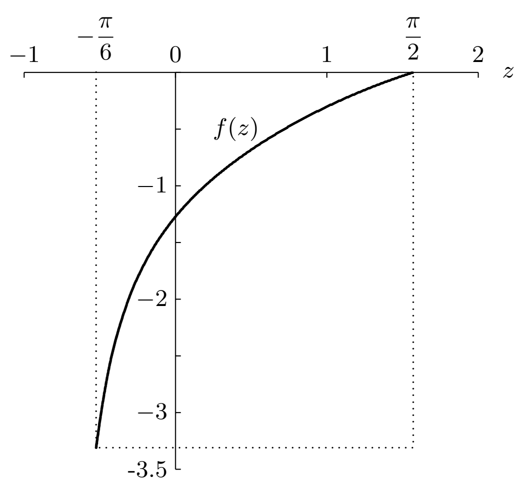

Fig. 2.2 Plot of the monotonic function (2.5) which can be exactly expressed by a TS system

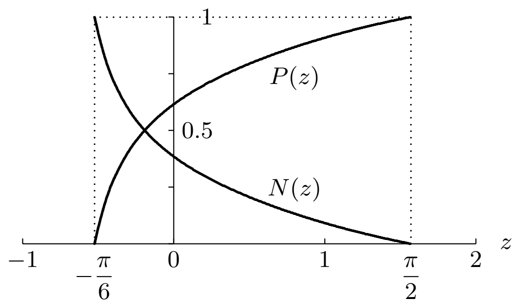

Fig. 2.3 Plot of the membership function $N(z)$ defined by (2.6) and its complement $P(z) = 1 - N(z)$

This is true if, and only if the membership function $N(z)$ from the class of functions defined above is given by the function shown in Fig. 2.3

$$N(z) = -\frac{\pi\sqrt{3}}{18} f(z), \qquad \text{for} \quad z \in \left[-\frac{\pi}{6}, \frac{\pi}{2}\right]. \tag{2.6}$$

Monotonicity of the membership functions of fuzzy sets is an important requirement. The question arises whether this requirement can be substituted by a local or global sector nonlinearity condition as suggested in [184] (p. 10)?

Example 2.2. Let us consider the function depicted in Fig. 2.4

$$f(z) = z(\sin z + 2), \qquad \text{for} \quad z \in [-1, 5]. \tag{2.7}$$

This function is a sector bounded nonlinearity. It is clear that for $z \in [-1, 5]$ the equation

$$N(z) = \frac{z(\sin z + 2) - 5\sin 5 - 10}{\sin 1 - 5\sin 5 - 12} \tag{2.8}$$

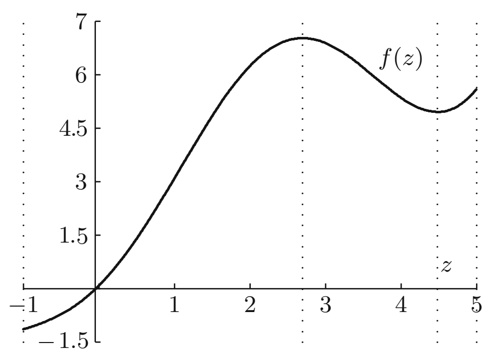

Fig. 2.4 Plot of a sector bounded function (2.7) which cannot be exactly expressed by a single TS system

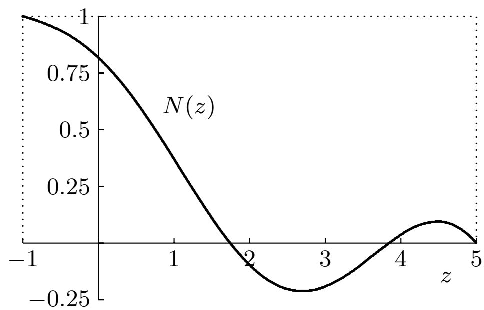

Fig. 2.5 Plot of the function (2.8)

must be satisfied, but the condition $N(z) \in [0,1]$ is not true for all $z \in [-1,5]$. Therefore $N(z)$ cannot be viewed as a membership function of some fuzzy set defined on the universe of discourse $[-1,5]$, (see Fig. 2.5). Of course, the function (2.7) can be exactly expressed in the form of three rule-based TS systems, where every system is designed in the monotonicity region of the original function $f(z)$ (see Fig.2.4).

Monotonicity of the membership functions of fuzzy sets is very important requirement from the interpretability point of view.

Example 2.3. For the continuous, smooth and highly nonlinear function

$$f(z) = e^{\pi} - (e^{\pi} - \pi^{e}) \sin^2(5\pi z/2) \exp\left(-\sin^2(9\pi z)\right), \quad z \in [0,1], \quad (2.9)$$

one can find the fuzzy rules in the form of (2.1) and the fuzzy sets, such that $S(z) = f(z)$ for $z \in [0,1]$. The consequents of the fuzzy rules are constants $q_1 = e^{\pi}$, $q_2 = \pi^{e}$ and the membership functions of fuzzy sets N and P satisfy the boundary conditions ($P(0) = 0$, $P(1) = 1$, $N(0) = 1$ and $N(1) = 0$). The membership functions are as follows (see Fig. 2.6)

$$N(z) = 1 - P(z), \quad P(z) = \sin^2(5\pi z/2) \exp\left(-\sin^2(9\pi z)\right), \quad z \in [0,1]. \quad (2.10)$$

Even though the output S of the TS system is exactly the same as the function (2.9) for all points from the universe of discourse and the membership functions satisfy the boundary conditions, the fuzzy sets are not easy for interpretation.

In the fuzzy modeling we should rather avoid nonmonotonic membership functions. Similar investigation to the one in the above section can be made for exact modeling of nonlinear systems with many input variables. Some ideas on this subject are included in [184], where however, there is no systematic procedure for converting a general nonlinear system to the TS form, even for nonlinear systems with nonlinearities that are polynomials of input variables [31].

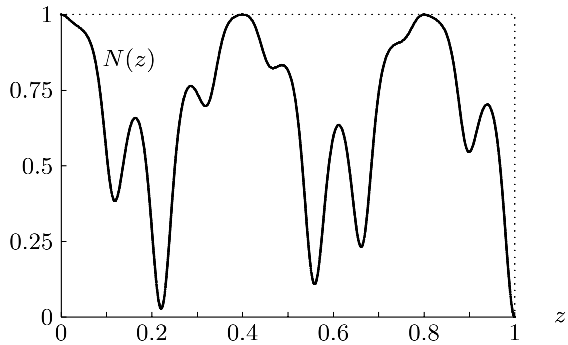

Fig. 2.6 Plot of the membership function $N(z)$ defined by (2.10)

2.2 Assumptions and Linguistic Interpretation of Linear Membership Functions

We will mainly use linear membership functions for input variables. They are conceptually the simplest, have a clear interpretation and play a crucial role in many applications in the fuzzy modeling and control. We will show further on mathematically and by examples that they are sufficient for modeling complex highly nonlinear static or dynamic, continuous or discrete-time systems.

Let us consider a multiple-input and single-output rule-based system (MISO system for short) with *input variables* $z_1, z_2, \ldots, z_n$. For every input $z_k \in [-\alpha_k, \beta_k]$ we require that there is no interval degenerated to a single point, i.e. we assume $\alpha_k + \beta_k \neq 0$ for $k = 1, 2, \ldots, n$, throughout the book. For any z_k, we define two fuzzy sets with *linear membership functions* $N_k(z_k)$, and $P_k(z_k)$, where P_k is an *algebraic complement* to N_k (see Fig. 2.7)

$$N_k(z_k) = \frac{\beta_k - z_k}{\alpha_k + \beta_k}, \tag{2.11}$$

$$P_k(z_k) = 1 - N_k(z_k), \qquad k = 1, 2, \ldots, n. \tag{2.12}$$

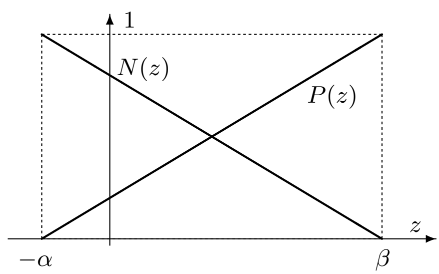

Fig. 2.7 Linear membership functions of two fuzzy sets

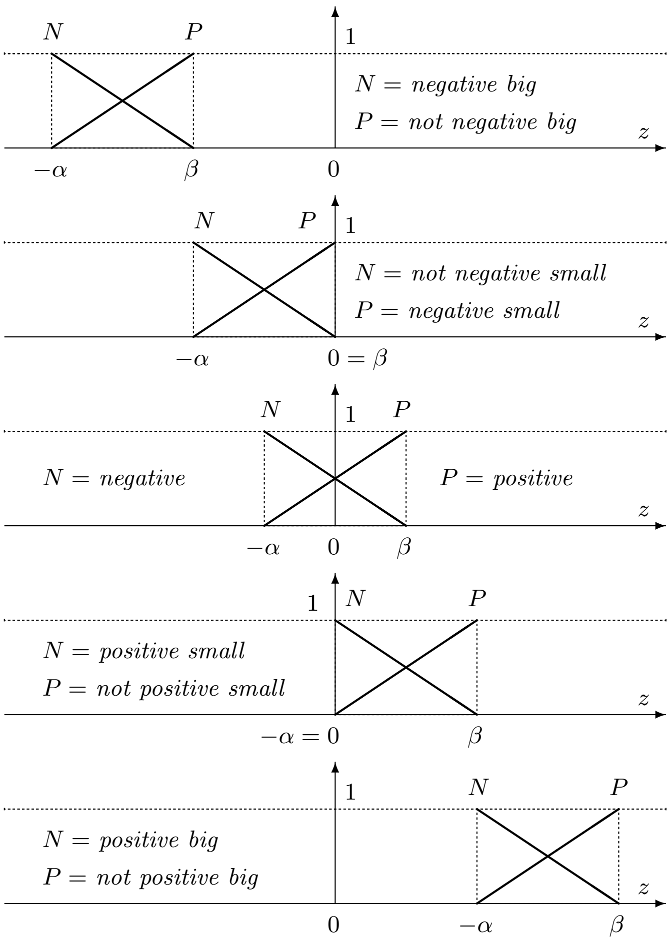

Fig. 2.8 Examples of linguistic interpretation of the fuzzy sets $N = N(z)$ and $P = P(z)$ for $z \in [-\alpha, \beta]$

It should be noted that using some linear transformation, the intervals $[-\alpha_k, \beta_k]$ could be replaced by different "standardized intervals". The unity interval $[0, 1]$ or symmetric around zero interval $[-1, 1]$ and many others belong to them. Such substitution would greatly simplify all mathematical descriptions and proofs. However, we will mainly use intervals $[-\alpha_k, \beta_k]$ further on, because for them it is possible to distinguish five cases, in which the terms N_k and P_k have different linguistic interpretations (see Fig. 2.8):

1. If $-\alpha_k < \beta_k < 0$, then N_k can be interpreted as *negative big*, and P_k - *not negative big*,

2. If $-\alpha_k < \beta_k = 0$, then N_k can be interpreted as *not negative small*, and P_k - *negative small*,
3. If $\alpha_k \approx \beta_k > 0$, then N_k can be interpreted as *negative*, and P_k - *positive*,
4. If $0 = -\alpha_k < \beta_k$, then N_k can be interpreted as *positive small*, and P_k - as *not positive small*,
5. If $0 < -\alpha_k < \beta_k$, then N_k can be interpreted as *positive big*, and P_k - as *not positive big*.

Obviously, depending on the context or specific application, the linguistic terms can be substituted by more suitable, adequate for the considered problem. For example the term *positive* can be replaced by *positive small* or *positive big*. We will use symbolic intervals $[-\alpha_k, \beta_k]$, where $-\alpha_k < \beta_k$. Thanks to this our analytical results will be more general than those obtained in other works, e.g. [168], [207].

Observe that for the functions (2.11)-(2.12) the inequalities

$$\frac{dN_k}{dz_k} < 0 \qquad \text{and} \qquad \frac{dP_k}{dz_k} > 0,$$

are satisfied, since $\alpha_k + \beta_k > 0$ for $k = 1, \ldots, n$. Therefore the symbol N_k refers to the membership function with *negative slope* and analogously P_k refers to the function with *positive slope*.

2.3 Compact Description of the MISO TS System

In order to allow the numbering of fuzzy rules by natural numbers, and to give more compact descriptions, we introduce a convenient indexing. Let us consider a MISO TS system with the inputs $z_1, \ldots, z_n$ and the output S (see Fig. 2.9). This system is defined by 2^n rules in the form of implications

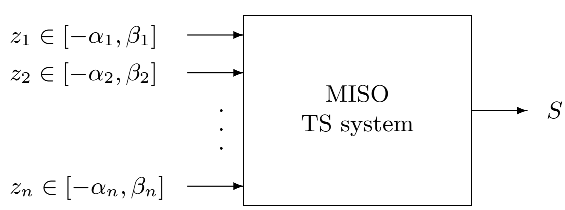

Fig. 2.9 The inputs and the output of MISO TS system

$$\text{If } P_{(i_1,\ldots,i_n)}, \text{ then } S = q_{(i_1,\ldots,i_n)}, \tag{2.13}$$

where $(i_1, \ldots, i_n) \in \{0,1\}^n$ and each *antecedent* $P_{(i_1,\ldots,i_n)}$ of an implication is the statement of the form

$$P_{(i_1,\ldots,i_n)} = \text{“}z_1 \text{ is } A_{i_1} \text{ and ... and } z_n \text{ is } A_{i_n}\text{”}, \tag{2.14}$$

and

$$A_{i_k} = \begin{cases} N_k, & \text{for} \quad i_k = 0 \\ P_k, & \text{for} \quad i_k = 1 \end{cases}, \qquad k = 1, \ldots, n. \tag{2.15}$$

If it is not stated differently, we assume that the *consequents* $q_{(i_1,\ldots,i_n)}$ of the rules in (2.13) do not depend on the input variables, i.e. we will consider a *zero-order Takagi-Sugeno model* [180]. In more general TS systems, the consequents are polynomials of the first or higher order or more complicated functions of input variables.

The rule-based system (2.13)-(2.15) we will call *P1-TS system* to emphasize that membership functions of fuzzy sets for input variables are polynomials of the first order.

Now we introduce indexing which allows the ordering of the fuzzy rules. For any n-tuple of indices $(i_1, \ldots, i_n) \in \{0,1\}^n$ we define the corresponding index v, which is formally a function of the sequence of indices $(i_1, \ldots, i_n)$:

$$v = 1 + \sum_{k=1}^{n} i_k 2^{n-k}, \qquad i_k \in \{0,1\}, \qquad k = 1, \ldots, n. \tag{2.16}$$

Any v from the set $\{1, 2, \ldots, 2^n\}$ corresponds to only one antecedent of the fuzzy "If–then" rule. When the bijection (2.16) holds we will simply write $v \leftrightarrow (i_1, \ldots, i_n)$, e.g. $182 \leftrightarrow (1,0,1,1,0,1,0,1)$.

The rules (2.13) can be rewritten as

$$\text{If } P_v, \text{ then } S = q_v, \tag{2.17}$$

where $v \leftrightarrow (i_1, \ldots, i_n)$. For the inputs $z_1, \ldots, z_n$, the output is S and it is defined by the formula [180]

$$S(z_1, \ldots, z_n) = \frac{\sum_{v=1}^{2^n} q_v h_v(z_1, \ldots, z_n)}{\sum_{v=1}^{2^n} h_v(z_1, \ldots, z_n)}, \tag{2.18}$$

where

$$h_v(z_1, \ldots, z_n) = \top \left(A_{i_1}(z_1), \ldots, A_{i_n}(z_n)\right)_v, \tag{2.19}$$

the operator $\top$ denotes an algebraic t-norm: $\top(x, y) = xy$ [202], the indices v and $(i_1, \ldots, i_n)$ are in the one-to-one correspondence (2.16), and $A_{i_k}(z)$ are membership functions of the fuzzy sets, i.e. $A_{i_k} \in \{N_k, P_k\}$ for $i_k \in \{0,1\}$ and $k = 1, \ldots, n$. The value h_v can be interpreted as a *degree of fulfilment* (or *degree of firing level*) of the vth rule by the given inputs $z_1, \ldots, z_n$. One can check that

$$\sum_{v=1}^{2^n} h_v(z_1, \ldots, z_n) = \prod_{i=1}^{n} \left(N_i(z_i) + P_i(z_i)\right), \tag{2.20}$$

and therefore, if the complementary property (2.12) is satisfied, then (2.18) reduces to

$$S = \sum_{v=1}^{2^n} q_v h_v(z_1, \ldots, z_n). \tag{2.21}$$

The function $h_v(z_1, \ldots, z_n)$ can be viewed as a *normalized* membership function of many variables or as a fuzzy relation.

The set

$$D^n = [-\alpha_1, \beta_1] \times \ldots \times [-\alpha_n, \beta_n], \tag{2.22}$$

where $\times$ denotes the Cartesian product, we will call a *hypercuboid*. Its vertices are the vectors

$$\boldsymbol{\gamma}_v = [\gamma_1, \ldots, \gamma_n]^T \in \{-\alpha_1, \beta_1\} \times \ldots \times \{-\alpha_n, \beta_n\}, \tag{2.23}$$

where $v \leftrightarrow (i_1, \ldots, i_n) \in \{0, 1\}^n$, and they can be ordered according to (2.16) as shown in Fig. 2.10. The length L_k of the interval $[-\alpha_k, \beta_k]$ and the volume V_k of the hypercuboid D^k are defined by

$$L_k = \alpha_k + \beta_k \ , \qquad k = 1, 2, ..., n, \tag{2.24}$$

$$V_k = \prod_{i=1}^{k} L_i \ , \qquad k = 1, 2, ..., n. \tag{2.25}$$

They will be helpful in the future for the interpretation of some results.

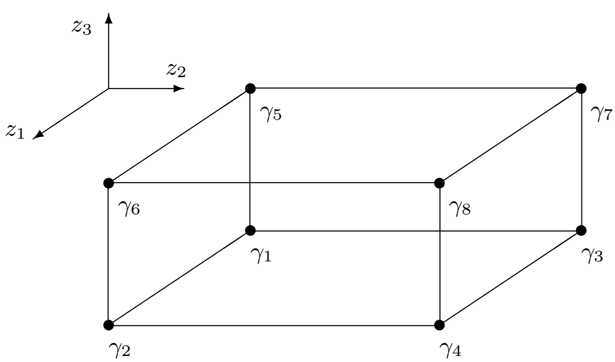

Fig. 2.10 Vertices of the hypercuboid D^n for $n = 3$

2.4 Crisp Output of the Zero-Order MISO P1-TS System

In this section we prove the main theorem concerning modeling of systems using the Takagi-Sugeno rule scheme, which uses two complementary linear membership functions for each input variable.

Theorem 2.4. *Define for the vector variable* $\mathbf{z} = [z_1, \ldots, z_n]^T$, *the following multilinear function* $f_0 : D^n \to \mathbb{R}$,

$$f_0(\mathbf{z}) = \sum_{(p_1, p_2, \ldots, p_n) \in \{0,1\}^n} \theta_{p_1, p_2, \ldots, p_n} z_1^{p_1} z_2^{p_2} \cdots z_n^{p_n} \ , \tag{2.26}$$

where 2^n coefficients $\theta_{00...0}$, $\theta_{10...0}$, $\theta_{01...0}$, ..., $\theta_{11...1}$, are real numbers. For every function of the type (2.26) there exists a zero-order MISO P1-TS system such that $S(\mathbf{z}) = f_0(\mathbf{z})$ for all $\mathbf{z} \in D^n$ and

(i) the inputs of the system are components of $\mathbf{z} \in D^n$ and the output is S (see Fig. 2.9),
(ii) two linear membership functions defined by (2.11)-(2.12) are assigned to each component of the vector $\mathbf{z}$,
(iii) the system is defined by 2^n fuzzy rules in the form of (2.13)-(2.15).

One can find all consequents q_1, q_2, ..., q_{2^n} of the fuzzy rules by solving 2^n linear equations. For a nonzero volume of the hypercuboid D^n, the unique solution always exists.

Proof. First we identify the class of functions performed by the TS system. The t-norm in (2.18) is an algebraic product and any function h_v in (2.19) is a product of the first order polynomials. Thus,

$$S = \sum_{i_n=1}^{2} \cdots \sum_{i_1=1}^{2} \prod_{k=1}^{n} (a_{i_k} z_k + b_{i_k})\, q_{(i_1,\ldots,i_n)} \ ,$$

where a_{i_k}, b_{i_k}, and $q_{(i_1,\ldots,i_n)}$ are real numbers. This means that $S(\mathbf{z})$ is a multilinear function which can be written in the form of (2.26).

Now assume that some function f_0 in the form of (2.26) is given. Our goal is to express all consequents of the rules for the fixed in advance collection of coefficients $\theta_{00...0}$, $\theta_{10...0}$, $\theta_{01...0}$, ..., $\theta_{11...1}$. The function f_0 is the scalar product $f_0(\mathbf{z}) = \boldsymbol{\theta}^T \mathbf{g}(\mathbf{z})$, where

$$\boldsymbol{\theta} = [\theta_{00...0}, \theta_{10...0}, \theta_{01...0}, ..., \theta_{p_1...p_n}, ..., \theta_{11...1}]^T \in \mathbb{R}^{2^n}, \tag{2.27}$$

$$\mathbf{g}(\mathbf{z}) = [1, \ldots, (z_1^{p_1} \cdots z_n^{p_n}), \ldots, (z_1 \cdots z_n)]^T \ , \tag{2.28}$$

with $p_k \in \{0, 1\}$ for $k = 1, \ldots, n$. For a given $\mathbf{z}$, the vector $\mathbf{g}(\mathbf{z})$ we will call a *generator*. It is continuous nonlinear mapping, which transforms the points $\mathbf{z} \in D^n$ into 2^n-dimensional space, whereas the function f_0 is a *linear function with respect to parameters* $\theta_{00...0}$, $\theta_{10...0}$, $\theta_{01...0}$, ..., $\theta_{p_1,p_2,...,p_n}$, ..., $\theta_{11...1}$. The generator $\mathbf{g} = \mathbf{g}(z_1, z_2, \ldots, z_n)$ contains 2^n components of the form "$z_{i_1} z_{i_2} \ldots z_{i_k}$" being elements of the polynomial "$(1 + z_1) \times \cdots \times (1 + z_n)$" written in the expanded additive form, when substituting in the *monomials* of this polynomial all coefficients by "1".

The equation $f_0(z_1, \ldots, z_n) = \boldsymbol{\theta}^T \mathbf{g}(z_1, \ldots, z_n)$ must be satisfied for all points in the hypercuboid D^n, especially in its vertices. Thus, the following 2^n linear equations

$$\boldsymbol{\theta}^T \mathbf{g}(\boldsymbol{\gamma}_v) = q_v, \qquad v = 1, 2, \ldots, 2^n, \tag{2.29}$$

must be satisfied, or equivalently

$$\mathbf{q} = \boldsymbol{\Omega}^T \boldsymbol{\theta}, \quad \boldsymbol{\Omega} = \left[\mathbf{g}\left(\boldsymbol{\gamma}_1\right), \ldots, \mathbf{g}\left(\boldsymbol{\gamma}_{2^n}\right) \right]_{2^n \times 2^n}, \tag{2.30}$$

where the consequents of the rules constitute the vector $\mathbf{q} = [q_1, \ldots, q_{2^n}]^T$. Thus,

$$\mathbf{q} = \begin{bmatrix} \boldsymbol{\theta}^T \mathbf{g}\left(\boldsymbol{\gamma}_1\right) \\ \vdots \\ \boldsymbol{\theta}^T \mathbf{g}\left(\boldsymbol{\gamma}_{2^n}\right) \end{bmatrix} = \begin{bmatrix} f_0\left(\boldsymbol{\gamma}_1\right) \\ \vdots \\ f_0\left(\boldsymbol{\gamma}_{2^n}\right) \end{bmatrix}. \tag{2.31}$$

The equations (2.30)-(2.31) formulate necessary conditions, under which the system of fuzzy rules is equivalent to (2.26). Now we prove that they are sufficient as well. Sufficiency requires that the $2^n \times 2^n$ matrix $\boldsymbol{\Omega}$ containing the columns $\mathbf{g}\left(-\alpha_1, -\alpha_2, \ldots, -\alpha_n\right)$, ..., $\mathbf{g}\left(\beta_1, \beta_2, \ldots, \beta_n\right)$ is a nonsingular one. Observe that the output S of the TS system with n inputs z_1, z_2, ..., z_n can be defined by the following rule base, which is equivalent to (2.13)-(2.15):

$$\left.\begin{array}{l} R_1 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2 \text{ and } \ldots \text{ and } z_n \text{ is } N_n, \text{ then } S = q_1, \\ R_2 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2 \text{ and } \ldots \text{ and } z_n \text{ is } N_n, \text{ then } S = q_2, \\ R_3 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2 \text{ and } \ldots \text{ and } z_n \text{ is } N_n, \text{ then } S = q_3, \\ R_4 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2 \text{ and } \ldots \text{ and } z_n \text{ is } N_n, \text{ then } S = q_4, \\ \qquad\qquad\qquad\qquad \vdots \\ R_{2^n} : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2 \text{ and } \ldots \text{ and } z_n \text{ is } P_n, \text{ then } S = q_{2^n}. \end{array}\right\} \tag{2.32}$$

The matrix $\boldsymbol{\Omega}$ we will call the *fundamental matrix* throughout the book. It contains as elements the values of the generator $\mathbf{g}$ in such vertices of the hypercuboid D^n that exactly correspond to the labels used in the antecedents of the rules. This means that "$-\alpha_k$" in $\mathbf{g}$ corresponds to N_k, and "β_k" in $\mathbf{g}$ corresponds to P_k, where N_k and/or P_k are used in the antecedent of the rule. The vector $\mathbf{q} = [q_1, q_2, \ldots, q_{2^n}]^T$ contains successive consequents of the rules. Thus, both the order of vertices $\boldsymbol{\gamma}_v$ used for computing $\boldsymbol{\Omega}$, and the order of elements of $\mathbf{q}$ are strictly defined. Now we prove inductively that $\boldsymbol{\Omega}$ is nonsingular if, and only if $\alpha_k + \beta_k \neq 0$ for $k = 1, 2, \ldots, n$. In the case of n input variables, the generator $\mathbf{g}$ and the matrix $\boldsymbol{\Omega}$ will have a subscript, i.e. $\mathbf{g} = \mathbf{g}_n$ and $\boldsymbol{\Omega} = \boldsymbol{\Omega}_n$. Formally, we define an artificial generator $\mathbf{g}_0 = 1$ and the corresponding artificial matrix $\boldsymbol{\Omega}_0 = 1$. First, consider the case with $n = 1$. The rule base structure is as follows

$$\left.\begin{array}{l} R_1 : \text{If } z_1 \text{ is } N_1, \text{ then } S = q_1, \\ R_2 : \text{If } z_1 \text{ is } P_1, \text{ then } S = q_2, \end{array}\right\} \tag{2.33}$$

and the corresponding generator $\mathbf{g}_1 = \mathbf{g}_1\left(z_1\right)$ is given by

$$\mathbf{g}_1 = \begin{bmatrix} \mathbf{g}_0 \\ z_1 \mathbf{g}_0 \end{bmatrix} = \begin{bmatrix} 1 \\ z_1 \end{bmatrix}. \tag{2.34}$$

The fundamental matrix $\boldsymbol{\Omega}_1$ is a concatenation of 2 columns $\mathbf{g}_1$. It is generated as follows

$$\boldsymbol{\Omega}_1 = \left[\mathbf{g}_1\left(-\alpha_1\right), \mathbf{g}_1\left(\beta_1\right)\right] = \begin{bmatrix} 1 & 1 \\ -\alpha_1 & \beta_1 \end{bmatrix}. \tag{2.35}$$

For $n = 2$ the rule-base structure is the following

$$\left.\begin{array}{r} R_1 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_1, \\ R_2 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_2, \\ R_3 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_3, \\ R_4 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_4, \end{array}\right\} \tag{2.36}$$

and the corresponding generator $\mathbf{g}_2 = \mathbf{g}_2\left(z_1, z_2\right)$ is given by

$$\mathbf{g}_2 = \begin{bmatrix} \mathbf{g}_1 \\ z_2\mathbf{g}_1 \end{bmatrix} = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ z_1 z_2 \end{bmatrix}. \tag{2.37}$$

The fundamental matrix $\boldsymbol{\Omega}_2$ is a concatenation of 4 columns

$$\begin{aligned} \boldsymbol{\Omega}_2 &= \left[\mathbf{g}_2\left(-\alpha_1, -\alpha_2\right), \mathbf{g}_2\left(\beta_1, -\alpha_2\right), \mathbf{g}_2\left(-\alpha_1, \beta_2\right), \mathbf{g}_2\left(\beta_1, \beta_2\right)\right] \\ &= \begin{bmatrix} 1 & 1 & 1 & 1 \\ -\alpha_1 & \beta_1 & -\alpha_1 & \beta_1 \\ -\alpha_2 & -\alpha_2 & \beta_2 & \beta_2 \\ \alpha_1\alpha_2 & -\alpha_2\beta_1 & -\alpha_1\beta_2 & \beta_1\beta_2 \end{bmatrix}. \end{aligned} \tag{2.38}$$

For $n = 3$ the rule base structure is

$$\left.\begin{array}{r} R_1 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2 \text{ and } z_3 \text{ is } N_3, \text{ then } S = q_1, \\ R_2 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2 \text{ and } z_3 \text{ is } N_3, \text{ then } S = q_2, \\ R_3 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2 \text{ and } z_3 \text{ is } N_3, \text{ then } S = q_3, \\ R_4 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2 \text{ and } z_3 \text{ is } N_3, \text{ then } S = q_4, \\ R_5 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2 \text{ and } z_3 \text{ is } P_3, \text{ then } S = q_5, \\ R_6 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2 \text{ and } z_3 \text{ is } P_3, \text{ then } S = q_6, \\ R_7 : \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2 \text{ and } z_3 \text{ is } P_3, \text{ then } S = q_7, \\ R_8 : \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2 \text{ and } z_3 \text{ is } P_3, \text{ then } S = q_8, \end{array}\right\} \tag{2.39}$$

and the corresponding generator $\mathbf{g}_3 = \mathbf{g}_3\left(z_1, z_2, z_3\right)$ is given by

$$\mathbf{g}_3(z_1, z_2, z_3) = \begin{bmatrix} \mathbf{g}_2 \\ z_3\mathbf{g}_2 \end{bmatrix} = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ z_1z_2 \\ z_3 \\ z_1z_3 \\ z_2z_3 \\ z_1z_2z_3 \end{bmatrix}. \tag{2.40}$$

The fundamental matrix $\mathbf{\Omega}_3$ is a concatenation of $2^3 = 8$ columns $\mathbf{g}_3$ and is generated as

$$\begin{aligned}\mathbf{\Omega}_3 = [&\mathbf{g}_3(-\alpha_1, -\alpha_2, -\alpha_3), \ \mathbf{g}_3(\beta_1, -\alpha_2, -\alpha_3), \ \mathbf{g}_3(-\alpha_1, \beta_2, -\alpha_3), \\ &\mathbf{g}_3(\beta_1, \beta_2, -\alpha_3), \ \mathbf{g}_3(-\alpha_1, -\alpha_2, \beta_3), \ \mathbf{g}_3(\beta_1, -\alpha_2, \beta_3), \\ &\mathbf{g}_3(-\alpha_1, \beta_2, \beta_3), \ \mathbf{g}_3(\beta_1, \beta_2, \beta_3)] \\ = &\begin{bmatrix} 1 & -\alpha_1 & -\alpha_2 & \alpha_1\alpha_2 & -\alpha_3 & \alpha_1\alpha_3 & \alpha_2\alpha_3 & -\alpha_1\alpha_2\alpha_3 \\ 1 & \beta_1 & -\alpha_2 & -\alpha_2\beta_1 & -\alpha_3 & -\beta_1\alpha_3 & \alpha_2\alpha_3 & \alpha_2\beta_1\alpha_3 \\ 1 & -\alpha_1 & \beta_2 & -\alpha_1\beta_2 & -\alpha_3 & \alpha_1\alpha_3 & -\alpha_3\beta_2 & \alpha_1\alpha_3\beta_2 \\ 1 & \beta_1 & \beta_2 & \beta_1\beta_2 & -\alpha_3 & -\beta_1\alpha_3 & -\alpha_3\beta_2 & -\beta_1\alpha_3\beta_2 \\ 1 & -\alpha_1 & -\alpha_2 & \alpha_1\alpha_2 & \beta_3 & -\alpha_1\beta_3 & -\alpha_2\beta_3 & \alpha_1\alpha_2\beta_3 \\ 1 & \beta_1 & -\alpha_2 & -\alpha_2\beta_1 & \beta_3 & \beta_1\beta_3 & -\alpha_2\beta_3 & -\alpha_2\beta_1\beta_3 \\ 1 & -\alpha_1 & \beta_2 & -\alpha_1\beta_2 & \beta_3 & -\alpha_1\beta_3 & \beta_2\beta_3 & -\alpha_1\beta_2\beta_3 \\ 1 & \beta_1 & \beta_2 & \beta_1\beta_2 & \beta_3 & \beta_1\beta_3 & \beta_2\beta_3 & \beta_1\beta_2\beta_3 \end{bmatrix}^T,\end{aligned} \tag{2.41}$$

and so forth. In general,

$$\begin{aligned} &\mathbf{g}_0 = 1, \qquad \mathbf{\Omega}_0 = 1, \\ &\mathbf{g}_{k+1} = \begin{bmatrix} \mathbf{g}_k \\ z_{k+1}\mathbf{g}_k \end{bmatrix} = \begin{bmatrix} 1 \\ z_{k+1} \end{bmatrix} \otimes \mathbf{g}_k \in \mathbb{R}^{2^{k+1}}, \qquad k = 0, 1, 2, \ldots, n-1, \end{aligned} \tag{2.42}$$

where the symbol "$\otimes$" denotes the Kronecker product (see Appendix A or [43], [83]). One can easily check that

$$\mathbf{\Omega}_{k+1} = \begin{bmatrix} \mathbf{\Omega}_k & \mathbf{\Omega}_k \\ -\alpha_{k+1}\mathbf{\Omega}_k & \beta_{k+1}\mathbf{\Omega}_k \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ -\alpha_{k+1} & \beta_{k+1} \end{bmatrix} \otimes \mathbf{\Omega}_k \in \mathbb{R}^{2^{k+1} \times 2^{k+1}}, \tag{2.43}$$

for $k = 0, 1, 2, \ldots, n-1$. From (A.6a) given in Appendix A we immediately obtain

$$\det \mathbf{\Omega}_{k+1} = (\beta_{k+1} + \alpha_{k+1})^{2^k} (\det \mathbf{\Omega}_k)^2 .$$

Taking into account $\mathbf{\Omega}_0 = 1$ already defined, we obtain

$$\det \mathbf{\Omega}_n = \prod_{i=1}^{n} (\beta_i + \alpha_i)^{2^{n-1}} = (V_n)^{2^{n-1}} .$$

Thus, $\det \boldsymbol{\Omega}_n \neq 0$ if, and only if the volume of the heperrectangle D^n in the space $\mathbb{R}^n$ is not zero or, equivalently, for every input variable z_k, $(k = 1, 2, \ldots, n)$, the interval $[-\alpha_k, \beta_k]$ is not degenerated to a single point. Finally, from (2.30) we obtain the vector of coefficients of the function (2.26)

$$\boldsymbol{\theta} = \left(\boldsymbol{\Omega}^T\right)^{-1} \mathbf{q}. \tag{2.44}$$

Thus, the crisp output of the P1-TS system is given by

$$S(\mathbf{z}) = \mathbf{g}^T(\mathbf{z}) \left(\boldsymbol{\Omega}^T\right)^{-1} \mathbf{q} = f_0(\mathbf{z}). \tag{2.45}$$

This ends the proof of Theorem 2.4. □

Remark 2.5. The components of the vector $\boldsymbol{\theta}$ in (2.26) depend on $2^n + 2n$ parameters, i.e. on 2^n coefficients q_v and $2n$ boundaries of intervals $[-\alpha_k, \beta_k]$, $(k = 1, 2, \ldots, n)$.

Remark 2.6. Suppose the hypercuboid $D^n \subset \mathbb{R}^n$ is established. The function

$$f_1(\mathbf{z}) = \prod_{k=1}^{n} (r_k z_k + s_k), \tag{2.46}$$

where r_k, s_k are real numbers, is a special case of the function (2.26) for $n > 2$, since it contains $2n$ parameters r_k and s_k, whereas the function (2.26) contains 2^n coefficients $\theta_{00\ldots0}$, $\theta_{10\ldots0}$, $\theta_{01\ldots0}$, ..., $\theta_{11\ldots1}$.

As a conclusion of Theorem 2.4, whose interpretation is important, we obtain

Corollary 2.7. *Suppose a function $f : D^n \to \mathbb{R}$ is known and it belongs to the class of functions (2.26). In other words $f(\mathbf{z}) = f_0(\mathbf{z})$, where $\mathbf{z} \in D^n$ for some collection of coefficients of the vector $\boldsymbol{\theta}$ as in (2.27). A necessary and sufficient condition under which the considered TS system is equivalent to $f(\mathbf{z})$ for any $\mathbf{z} \in D^n$, is as follows*

$$q_v = f(\boldsymbol{\gamma}_v), \quad \text{for} \quad v = 1, 2, \ldots, 2^n. \tag{2.47}$$

This means that by formulating the consequents of the fuzzy rules, the only information needed by an expert are values of the function f in all vertices of the hypercuboid D^n.

What is more, Theorem 2.4 says that we can always obtain an equivalent TS system to the given function (2.26).

2.5 Completeness and Noncontradiction in Rule-Based Systems Defined by Metarules

The rule-base is usually assumed to have the form of (2.32). Such system contains *complete* and *noncontradictory* rules [92]. The system of “If-then” fuzzy rules will be called *complete* if every rule contains all possible antecedents in its "If" part, which results in 2^n rules as in (2.13)-(2.15). The system of rules is a *contradictory* one if there are at least two rules which have the same antecedent but different consequents. By such definitions, the system (2.32) is both complete and noncontradictory. The same notions can be defined in the fuzzy sense, i.e. the rules can be viewed as complete or noncontradictory to some degree. However, we will consider them as bivalent notions, i.e. the systems of rules will be treated as complete (contradictory) or not, throughout the book.

When the number of inputs is large, we can use the *metarules*, i.e. the rules which are equivalent to some subset of the rules, where each single rule is in the form of (2.13)-(2.15). Most frequently we have to do with the metarule if some "If-then" rule in its "If" part contains the word ANY_k (or ANY without a subscript). By the term ANY_k we mean any label from the bivalent set $\{N_k, P_k\}$, $(k = 1, \ldots, n)$. Sometimes the set of the rules may be generated by a metarule for other reasons.

Remark 2.8. The fragments "z_k is ANY" in the antecedents of the rules will be sometimes omitted. For example, we can simplify the fuzzy rule "If z_1 is N_1 and z_2 is ANY_2, then $S = q_1$" into the shorter one "If z_1 is N_1, then $S = q_1$".

Example 2.9. Let us consider three P1-TS systems with two inputs and one output, which are equivalently presented in Tables 2.1 a) – c).

a) The system of rules is defined by (2.36) and shown in Table 2.1 a). It is complete and noncontradictory. This case is simple and does not need a comment.
b) The system of rules:

$$\left.\begin{array}{l} R_1: \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_1, \\ R_2: \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_2, \\ R_4: \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_4, \end{array}\right\}$$

is shown in Table 2.1 b). Observe that there is no consequent for the antecedent "z_1 is N_1 and z_2 is P_2". This system is noncomplete and noncontradictory.

Table 2.1 Look-up-tables for the P1-TS system from Example 2.9: a) Complete and noncontradictory rules, b) Noncomplete and noncontradictory rules, c) Noncomplete and contradictory rules

a)

z_1, $z_2 \rightarrow$ ↓	N_2	P_2
N_1	q_1	q_3
P_1	q_2	q_4

b)

z_1, $z_2 \rightarrow$ ↓	N_2	P_2
N_1	q_1	–
P_1	q_2	q_4

c)

z_1, $z_2 \rightarrow$ ↓	N_2	P_2
N_1	q_1	q_1, q_3
P_1	q_2	–

c) The system of rules:

$$\left.\begin{aligned} R_1 &: \text{If } z_1 \text{ is } N_1, \text{ then } S = q_1, \\ R_2 &: \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_2, \\ R_3 &: \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_3, \end{aligned}\right\}$$

is equivalent to (see Remark 2.8)

$$\left.\begin{aligned} R_1' &: \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_1, \\ R_1'' &: \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_1, \\ R_2 &: \text{If } z_1 \text{ is } P_1 \text{ and } z_2 \text{ is } N_2, \text{ then } S = q_2, \\ R_3 &: \text{If } z_1 \text{ is } N_1 \text{ and } z_2 \text{ is } P_2, \text{ then } S = q_3. \end{aligned}\right\}$$

Although there are four single rules, there is no consequent for the antecedent "z_1 is P_1 and z_2 is P_2". For $q_1 \neq q_3$, the metarule R_1'' contradicts (more or less) the rule R_3. Thus, this system is both contradictory and noncomplete.

The above example shows that detection of completeness or noncontradiction in the rule-base is a very simple task if we use the look-up tables.

2.6 Matrix Description of the MIMO Fuzzy Rule-Based System

In this section we generalize the concept of MISO fuzzy rule-based systems into the multiple-input and multiple-output (MIMO) systems. In the systems with many outputs there are no cross-feedback loops. Therefore the procedure of computing a single output is the same as for the MISO TS systems.

Our goal in this section is to develop yet another compact and convenient description of the rule-based system, i.e. the *model in the matrix form.* Let us consider a TS system with the inputs $z_1, \ldots, z_n$ and the outputs $S_1, \ldots, S_m$, as shown in Fig. 2.11. By a MIMO P1-TS system we mean the system with $m \geq 2$ outputs, in which the membership functions of fuzzy sets for all inputs are linear as defined in (2.11)-(2.12). Such a system is described by the following 2^n fuzzy rules:

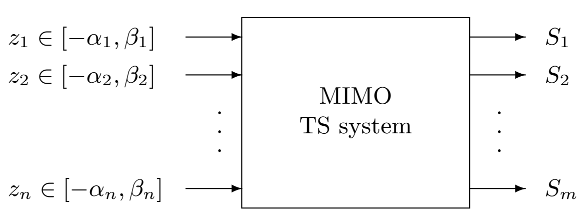

Fig. 2.11 The inputs and the outputs of a MIMO TS system given in the matrix form (2.48)-(2.53)

R_1 : If z_1 is N_1 and z_2 is N_2 and ... and z_n is N_n,
then $S_1 = q_{1,1}$, ..., $S_m = q_{1,m}$,

$\vdots$

R_v : If z_1 is A_{i_1} and z_2 is A_{i_2} and ... and z_n is A_{i_n},
then $S_1 = q_{v,1}$, ..., $S_m = q_{v,m}$,

$\vdots$

R_{2^n} : If z_1 is P_1 and z_2 is P_2 and ... and z_n is P_n,
then $S_1 = q_{2^n,1}$, ..., $S_m = q_{2^n,m}$,

where $A_{i_k} \in \{N_k, P_k\}$, ($k = 1, 2, \ldots, n$ and $i_k \in \{0, 1\}$), as defined in (2.15). Equivalently, this system can be described by the following single *"If-then" rule in the matrix form*

$$\text{If } [z_1, \ldots, z_n] \text{ is } \mathbf{M}, \text{ then } [S_1, \ldots, S_m] \text{ is } \mathbf{Q}, \tag{2.48}$$

where we assume that

- the *antecedents matrix* **M** contains the labels of fuzzy sets and has 2^n rows and n columns
$$\mathbf{M} = \begin{bmatrix} N_1 & \cdots & N_{n-1} & N_n \\ \vdots & \ddots & \vdots & \vdots \\ A_{i_1} & \cdots & A_{i_{n-1}} & A_{i_n} \\ \vdots & \ddots & \vdots & \vdots \\ P_1 & \cdots & P_{n-1} & P_n \end{bmatrix}, \tag{2.49}$$
where $\left(A_{i_1}, \ldots, A_{i_{n-1}}, A_{i_n}\right) \in \{N_1, P_1\} \times \ldots \times \{N_{n-1}, P_{n-1}\} \times \{N_n, P_n\}$,
- the *consequents matrix* **Q** contains m columns, and every column $\mathbf{q}_j$ corresponds to the output S_j of the rule-based system

$$\mathbf{Q} = [\mathbf{q}_1, \cdots, \mathbf{q}_j, \cdots, \mathbf{q}_m] = \begin{bmatrix} q_{1,1} & \cdots & q_{1,j} & \cdots & q_{1,m} \\ q_{2,1} & \cdots & q_{2,j} & \cdots & q_{2,m} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ q_{2^n,1} & \cdots & q_{2^n,j} & \cdots & q_{2^n,m} \end{bmatrix} \in \mathbb{R}^{2^n \times m}. \tag{2.50}$$

For such systems we formulate the following

Theorem 2.10. *Suppose the MIMO P1-TS system with the inputs constituting the vector* $[z_1, \ldots, z_n]^T = \mathbf{z} \in D^n$ *and the outputs* $S_1, \ldots, S_m$, *is defined by* 2^n *fuzzy "If-then" rules or equivalently – by a single rule in the matrix form (2.48)-(2.50). The row vector of crisp outputs* $\mathbf{S}(\mathbf{z}) = [S_1, \ldots, S_m]$ *can be computed by the formula*

$$\mathbf{S}(\mathbf{z}) = \mathbf{g}^T(\mathbf{z}) \left(\boldsymbol{\Omega}^T\right)^{-1} \mathbf{Q}, \tag{2.51}$$

where the consequents matrix

$$\mathbf{Q} = \boldsymbol{\Omega}^T \boldsymbol{\Theta}, \tag{2.52}$$

and

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_m] \in \mathbb{R}^{2^n \times m},$$

$$\boldsymbol{\theta}_j = [\theta_{j,00\ldots0}, \theta_{j,10\ldots0}, \theta_{j,01\ldots0}, ..., \theta_{j,11\ldots1}]^T \in \mathbb{R}^{2^n}, \qquad j = 1, \ldots, m. \tag{2.53}$$

Every column $\boldsymbol{\theta}_j$ is assigned to a single system output S_j, ($j = 1, \ldots, m$). The successive components of the generator $\mathbf{g}(\mathbf{z})$ are in accordance with the rows of the antecedents matrix $\mathbf{M}$ defined by (2.49). The fundamental matrix $\boldsymbol{\Omega}$ of the system is a concatenation of 2^n columns which are values of the generator $\mathbf{g}$ for the vertices of the hypercuboid D^n, where every vertex corresponds to the appropriate antecedent of the rule.

Proof. The proof is straightforward and will be omitted, since the procedure for computing every output S_j applies in the same manner as in the proof of Theorem 2.4. The formal proof of Theorem 2.10 comes down to substituting the vector of consequents of the rules $\mathbf{q}$ by the matrix $\mathbf{Q}$ in the equations (2.45) and (2.30). □

2.7 Equivalence Problem in the Rule-Based Systems

The problem of equivalence between the systems of fuzzy "If-then" rules is important especially when one compares the outcomes obtained by various experts or designers, and the number of inputs is greater than two. In the systems with n inputs, there are 2^n fuzzy rules. Thus, the number of ways of ordering "If" parts is $(2^n)!$ and the number of generators and fundamental matrices is $(2^n)!$ as well. For example, for $n = 2$ we have $\left(2^2\right)! = 24$, but for $n = 3$ there are 40 320 possibilities of writing the rules. The systems of rules can be equivalent or not.

We call two rule-based systems *equivalent* if, and only if, their crisp outputs are the same for the same inputs from the universe of discourse D^n. In order to avoid mistakes in computations, which may occur especially for systems with $n \geq 3$ inputs, the designer must know exactly the relationship between the fuzzy "If-then" rules containing antecedents and consequents, and their algebraic counterparts in the form of generators, fundamental matrices, and consequents of the rules. We will show that the relationship between elements of generators and the particular consequents of the rules plays a key role; they must correspond to each other.

More precisely, our goal is to explain why the results formulated by Theorems 2.4 and 2.10 are valid independently of the order of fuzzy "If-then" rules. Without loss of generality we consider two MISO systems, which are defined by two pairs of matrices describing antecedents and consequents of the rules: $\left(\mathbf{M}_{[1]}, \mathbf{Q}_{[1]}\right)$ and $\left(\mathbf{M}_{[2]}, \mathbf{Q}_{[2]}\right)$, respectively. The systems differ from each other in two rows, namely, the rth row in matrices $\mathbf{M}_{[1]}$ and $\mathbf{Q}_{[1]}$ is the same as the sth row of $\mathbf{M}_{[2]}$ and $\mathbf{Q}_{[2]}$, respectively, and vice-versa:

$$\mathbf{M}_{[1]} = \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_r \\ \vdots \\ \mathbf{m}_s \\ \vdots \\ \mathbf{m}_{2^n} \end{bmatrix}, \quad \mathbf{Q}_{[1]} = \begin{bmatrix} q_1 \\ \vdots \\ q_r \\ \vdots \\ q_s \\ \vdots \\ q_{2^n} \end{bmatrix}, \quad \mathbf{M}_{[2]} = \begin{bmatrix} \mathbf{m}_1 \\ \vdots \\ \mathbf{m}_s \\ \vdots \\ \mathbf{m}_r \\ \vdots \\ \mathbf{m}_{2^n} \end{bmatrix}, \quad \mathbf{Q}_{[2]} = \begin{bmatrix} q_1 \\ \vdots \\ q_s \\ \vdots \\ q_r \\ \vdots \\ q_{2^n} \end{bmatrix},$$

where $\mathbf{m}_1, \ldots, \mathbf{m}_{2^n} \in \{N_1, P_1\} \times \cdots \times \{N_n, P_n\}$, and $q_1, \ldots, q_{2^n} \in \mathbb{R}$.

From the proof of Theorem 2.4 (equations (2.29)-(2.31)), we immediately obtain that the outputs of such systems are the same. Of course, the above systems have different generators

$$\mathbf{g}_{[1]}(z_1, \ldots, z_n) = \begin{bmatrix} g_0 \\ \vdots \\ z_1^{r_1} \cdots z_n^{r_n} \\ \vdots \\ z_1^{s_1} \cdots z_n^{s_n} \\ \vdots \\ g_{2^n} \end{bmatrix}, \qquad \mathbf{g}_{[2]}(z_1, \ldots, z_n) = \begin{bmatrix} g_0 \\ \vdots \\ z_1^{s_1} \cdots z_n^{s_n} \\ \vdots \\ z_1^{r_1} \cdots z_n^{r_n} \\ \vdots \\ g_{2^n} \end{bmatrix},$$

where the powers $r_1, \ldots, r_n$ and $s_1, \ldots, s_n$ are from the set $\{0, 1\}$. The power r_k or s_k is 0, if kth element of the row $\mathbf{m}_r$ contains the label N_k, or 1, if kth element of the row $\mathbf{m}_r$ contains the label P_k. Consequently, the fundamental matrices $\mathbf{\Omega}_{[1]}$ and $\mathbf{\Omega}_{[2]}$ are different, according to their own generators. In spite of this, the above systems of rules are equivalent, i.e. they generate the same output, which can be expressed using formerly proved Theorems 2.4 and 2.10. We omit the formal proof, because it is a simple consequence of Theorem 2.4. Instead of this let us consider an example.

Example 2.11. Consider two rule bases for the systems with $n = 3$ inputs and $m = 1$ output, using the matrix description of the rules. Let the generator of the first system $\mathbf{g}_{[1]}(z_1, z_2, z_3)$ be the same as in (2.40). Thus,

$$\mathbf{M}_{[1]} = \begin{bmatrix} N_1 & N_2 & N_3 \\ P_1 & N_2 & N_3 \\ N_1 & P_2 & N_3 \\ P_1 & P_2 & N_3 \\ N_1 & N_2 & P_3 \\ P_1 & N_2 & P_3 \\ N_1 & P_2 & P_3 \\ P_1 & P_2 & P_3 \end{bmatrix}, \quad \mathbf{Q}_{[1]} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \\ q_6 \\ q_7 \\ q_8 \end{bmatrix}, \quad \mathbf{g}_{[1]}(z_1, z_2, z_3) = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ z_1 z_2 \\ z_3 \\ z_1 z_3 \\ z_2 z_3 \\ z_1 z_2 z_3 \end{bmatrix}, \tag{2.54}$$

and the corresponding fundamental matrix is given by (2.41). The second system differs from the first one in the replacement of the 4th row in the matrix $\mathbf{M}_{[1]}$ with the 5th one, i.e. we replace the row (P_1, P_2, N_3) for (N_1, N_2, P_3) and vice-versa. Thus,

$$\mathbf{M}_{[2]} = \begin{bmatrix} N_1 & N_2 & N_3 \\ P_1 & N_2 & N_3 \\ N_1 & P_2 & N_3 \\ N_1 & N_2 & P_3 \\ P_1 & P_2 & N_3 \\ P_1 & N_2 & P_3 \\ N_1 & P_2 & P_3 \\ P_1 & P_2 & P_3 \end{bmatrix}, \quad \mathbf{Q}_{[2]} = \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_5 \\ q_4 \\ q_6 \\ q_7 \\ q_8 \end{bmatrix}, \quad \mathbf{g}_{[2]}(z_1, z_2, z_3) = \begin{bmatrix} 1 \\ z_1 \\ z_2 \\ z_3 \\ z_1 z_2 \\ z_1 z_3 \\ z_2 z_3 \\ z_1 z_2 z_3 \end{bmatrix}. \tag{2.55}$$

By generating the fundamental matrix for the second system we take the following concatenation of the columns

$$\begin{aligned} \mathbf{\Omega}_{[2]} = [&\mathbf{g}_{[2]}(-\alpha_1, -\alpha_2, -\alpha_3), \mathbf{g}_{[2]}(\beta_1, -\alpha_2, -\alpha_3), \mathbf{g}_{[2]}(-\alpha_1, \beta_2, -\alpha_3), \\ &\mathbf{g}_{[2]}(-\alpha_1, -\alpha_2, \beta_3), \mathbf{g}_{[2]}(\beta_1, \beta_2, -\alpha_3), \mathbf{g}_{[2]}(\beta_1, -\alpha_2, \beta_3), \\ &\mathbf{g}_{[2]}(-\alpha_1, \beta_2, \beta_3), \mathbf{g}_{[2]}(\beta_1, \beta_2, \beta_3)] \end{aligned}$$

$$= \begin{bmatrix} 1 & -\alpha_1 & -\alpha_2 & -\alpha_3 & \alpha_1\alpha_2 & \alpha_1\alpha_3 & \alpha_2\alpha_3 & -\alpha_1\alpha_2\alpha_3 \\ 1 & \beta_1 & -\alpha_2 & -\alpha_3 & -\alpha_2\beta_1 & -\beta_1\alpha_3 & \alpha_2\alpha_3 & \alpha_2\beta_1\alpha_3 \\ 1 & -\alpha_1 & \beta_2 & -\alpha_3 & -\alpha_1\beta_2 & \alpha_1\alpha_3 & -\alpha_3\beta_2 & \alpha_1\alpha_3\beta_2 \\ 1 & -\alpha_1 & -\alpha_2 & \beta_3 & \alpha_1\alpha_2 & -\alpha_1\beta_3 & -\alpha_2\beta_3 & \alpha_1\alpha_2\beta_3 \\ 1 & \beta_1 & \beta_2 & -\alpha_3 & \beta_1\beta_2 & -\beta_1\alpha_3 & -\alpha_3\beta_2 & -\beta_1\alpha_3\beta_2 \\ 1 & \beta_1 & -\alpha_2 & \beta_3 & -\alpha_2\beta_1 & \beta_1\beta_3 & -\alpha_2\beta_3 & -\alpha_2\beta_1\beta_3 \\ 1 & -\alpha_1 & \beta_2 & \beta_3 & -\alpha_1\beta_2 & -\alpha_1\beta_3 & \beta_2\beta_3 & -\alpha_1\beta_2\beta_3 \\ 1 & \beta_1 & \beta_2 & \beta_3 & \beta_1\beta_2 & \beta_1\beta_3 & \beta_2\beta_3 & \beta_1\beta_2\beta_3 \end{bmatrix}^T. \tag{2.56}$$

As one can see $\mathbf{\Omega}_{[1]} \neq \mathbf{\Omega}_{[2]}$. From the logical point of view both systems describe the same mapping $D^3 \to \mathbb{R}$, but we want to prove this using formerly formulated theorems. From (2.51) we have $\mathbf{S}(\mathbf{z}) = \mathbf{Q}^T \mathbf{\Omega}^{-1} \mathbf{g}(\mathbf{z})$ and therefore

$$\mathbf{S}_{[1]}(\mathbf{z}) = \mathbf{Q}_{[1]}^T \mathbf{\Omega}_{[1]}^{-1} \mathbf{g}_{[1]}(\mathbf{z}), \qquad \mathbf{S}_{[2]}(\mathbf{z}) = \mathbf{Q}_{[2]}^T \mathbf{\Omega}_{[2]}^{-1} \mathbf{g}_{[2]}(\mathbf{z}). \tag{2.57}$$

After computing $\mathbf{\Omega}_{[1]}^{-1}$ and $\mathbf{\Omega}_{[2]}^{-1}$, one can easily check that using equations (2.54), (2.41), (2.55) and (2.56) for $\mathbf{Q}_{[i]}$, $\mathbf{\Omega}_{[i]}$, and $\mathbf{g}_{[i]}(\mathbf{z})$, $(i = 1, 2)$, we obtain

$$\mathbf{S}_{[1]}(\mathbf{z}) = \mathbf{S}_{[2]}(\mathbf{z}),$$

for all $\mathbf{z} \in D^3$. Thus, both systems of rules describe exactly the same system, indeed. □

In this way we can apply the results of Theorems 2.4 and 2.10 independently of the order of "If-then" rules provided that every element "$z_1^{i_1} \cdots z_{n-1}^{i_{n-1}} z_n^{i_n}$" of the generator corresponds both to the appropriate sequence of labels

$$\left(A_{i_1}, \ldots, A_{i_{n-1}}, A_{i_n}\right) \in \{N_1, P_1\} \times \ldots \times \{N_{n-1}, P_{n-1}\} \times \{N_n, P_n\},$$

which occurs in the antecedent of the rule, and to the appropriate element q_v in the consequents vector $\mathbf{q}$.

An extension of the above result for MIMO systems is straightforward and will be omitted.

2.8 Summary

The simplest TS system with one input and output and two fuzzy sets for the input variable is capable of expressing exactly any nonlinear, continuous and monotonic function of one variable. The system of fuzzy rules of such TS system has clear linguistic interpretation. If the TS system approximates a nonmonotonic function, the fuzzy sets may be very difficult for interpretation, even if they satisfy boundary conditions. Therefore in the fuzzy modeling we should rather avoid nonmonotonic membership functions.

By proving Theorem 2.4 we established an exact relationship between the P1-TS systems and a class of functions to which they are equivalent. It plays a crucial role in modeling, synthesis and analysis of many physical systems by using highly interpretable fuzzy rules. The notion of the *generator* and the *fundamental matrix* of the rule-based system belong to the most important ones, both for the theory and applications. We showed that the P1-TS system is nothing else than a *multi-linear* (or *multi-affine*) *polynomial* as stated in (2.26). It is worth adding that every *Boolean* (or *switching*) *function* $\{0,1\}^n \to \{0,1\}$ has a unique representation as a multi-linear polynomial. Such representation has been originally introduced by Zhegalkin [219] and was called *canonical polynomial form* of a Boolean function and plays an important role in many applications [6], [46], [48], [118].

The question arises: "What is the class of polynomials of the form (2.26)?" We can say informally that two multivariate polynomials are structurally the same if they differ in nonzero coefficients. Thus, the number of structurally different functions of n variables performed by the considered TS systems is 2^{2^n}. Observe that (2.26) is a part of the well-known *Kolmogorov-Gabor polynomial* (KGP for short) [49], [68]. More precisely, a *zero-order TS model* with *two linear* membership functions is equivalent to the KGP minus all the components of the type $z_i^m \cdots z_j^l \cdots z_k^r$ with the powers $\max\{m, l, \ldots, r\} \geq 2$ for $n > 1$. This observation seems to be worthy of discussion. The output of the zero-order TS system with n inputs will be denoted by S and the Kolmogorov-Gabor polynomial by KGP_n. We will say that two polynomials $p_1(\mathbf{z})$ and $p_2(\mathbf{z})$ are *equally powerful* and write $p_1 \equiv p_2$, if they are the same with the exception of nonzero coefficients, e.g. $1+2z_1+z_1^2 z_2 \equiv 3+5z_1-4z_1^2 z_2$. Furthermore, we will say that $p_1(\mathbf{z})$ is more *powerful* then $p_2(\mathbf{z})$, and write $p_1(\mathbf{z}) \supset p_2(\mathbf{z})$, if all monomials from $p_2(\mathbf{z})$ are included in $p_1(\mathbf{z})$ and at least one monomial (with nonzero coefficient) is included in $p_1(\mathbf{z})$, but not in $p_2(\mathbf{z})$, e.g. $1 + 2z_1 + z_1 z_2 + z_1^2 \supset 3 + 5z_1 + z_1 z_2$. One can prove that $KGP_n \supset S$ for all $n > 1$. For example, in the case of the system with $n = 4$ variables, the KGP has exactly 70 coefficients that uniquely define KGP_4, whereas a zero-order TS system has 16 coefficients only. A different situation occurs when we allow the rules in which the consequents are polynomials or the membership functions of fuzzy sets are polynomials of the degree $n > 1$.

One of the most important interpretations of Theorem 2.4 says that by formulating the consequents of the fuzzy rules which should express a given function f, the only information needed by an expert are the values of this function in all vertices of the hypercuboid D^n.

We introduced a compact matrix description of the MIMO P1-TS model. Observe that we can always set up a sequence of the antecedents of the rules e.g. by ordering the vertices of the hypercuboid D^n as shown in Section 2.3. In such case we can obtain an unambiguous model of the rule-based system in the matrix form (2.48) by establishing only the matrix of consequents of the rules. This fact can be used for the preservation of the computer memory needed to store the expert knowledge about the process modeled by a TS model.

Finally, we considered an equivalence problem of the rule-bases in the context of the matarules taking into account that in reality the rule-bases can be noncomplete and/or contradictory ones. The theorems proved in this chapter are valid independently of the sequences of the rules of a TS model.