

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Rough Controller Synthesis

Carlos Pinheiro, Ulisses Camatta and Angelo Rezek
*Federal University of Itajuba,
 Brazil*

1. Introduction

A new method to design rule-based controllers using concepts about rough sets is proposed. The method provides an efficient alternative for the design of rule-based controllers to compensate complex dynamic systems (nonlinear, with variable parameters, etc.). A systematic methodology to synthesize control rules is proposed. This approach serves to design fuzzy controllers and to define a new class of rule-based controllers, which will be called rough controllers. Numerical examples derived from computer simulations and a real application will be shown.

Rule-based models constitute an important tool in the representation of dynamic systems and controller models that use artificial intelligence techniques (fuzzy logic, neuro-fuzzy system, etc.). In general, the rules encapsulate the relationships between the model variables and provide mechanisms to connect the representations of the same with its computational procedures (Pedrycz & Gomide, 2007). There are two main schemes to construct rule-based models, those based on expert knowledge and those that are data-driven. There are several hybrid schemes that could be somewhere in between. In applications where the extraction of knowledge by experts is difficult due to the amount of data involved, data-driven methods are more efficient.

The Rough Set Theory (Pawlak, 1982) has been successfully applied in various areas such as data mining, decision systems, expert systems and other fields (Pawlak & Skowron, 2007). One of the main advantages of this approach is that it does not need for details in terms of probability distributions, belief intervals or possibilities values (Pawlak, 1991).

Few papers have addressed applications with rough sets related to control systems that use continuous and sampled variables. Most papers deal with mostly pure binary or symbolic variables (Ziarko & Katzberg, 1993; Kusiak & Shah, 2006).

This paper proposes a new approach to design rule-based controllers, aimed at applications in control systems of complex processes that utilize concepts about rough sets.

This chapter is organized as follows: a review of basic concepts about rough sets; the methodology proposed to design rule-based controllers; application examples; and final conclusions.

2. Background

An information system (*IS*) may be defined by $S = (U,A)$, where U is a set of objects or observations (o_i) called universe and A is a set of conditional attributes (a_j). The generic tabular representation of an information systems is illustrated in Table 1, where decision attribute values are defined in the last column of the table for a given decision attribute (d_i) and its corresponding classification $f(o_i,d_i)$. Generally rough sets deal with nominal values. For numerical attributes a discretization process is necessary, converting the values in nominal data. Some approaches may be utilized to minimize eventual effects of data quantization (Skowron and Son, 1995).

| | a_1 | ... | a_j | ... | a_n | d |
|----------|--------------|-----|--------------|-----|--------------|--------------|
| o_1 | $f(o_1,a_1)$ | | $f(o_1,a_j)$ | | $f(o_1,a_n)$ | $f(o_1,d_1)$ |
| \vdots | \vdots | | \vdots | | \vdots | \vdots |
| o_i | $f(o_i,a_1)$ | ... | $f(o_i,a_j)$ | ... | $f(o_i,a_n)$ | $f(o_i,d_i)$ |
| \vdots | \vdots | | \vdots | | \vdots | \vdots |
| o_m | $f(o_m,a_1)$ | ... | $f(o_m,a_j)$ | ... | $f(o_m,a_n)$ | $f(o_m,d_m)$ |

Table 1. Generic tabular representation of an *IS*

Consider an equivalence relation over U called indiscernibility relation (1). The set of all the equivalence classes determined by $IND(B)$ is represented by the notation $U / IND(B)$.

$$IND(B) = \{(o_i,o_j) \in U^2 \mid \forall a_k \in B, f(o_i,a_k) = f(o_j,a_k)\}$$
 (1)

Consider a set of all the elements from an equivalence class. Given $O \subseteq U$, it is important to know how many elements of O are defined by the elementary sets of S . To achieve this purpose, the lower approximation (B_*) and the upper approximation (B^*) are defined (2). A set O is called precise (*crisp*) if $B^*(O) = B_*(O)$, otherwise it is imprecise, rough or approximated.

$$B_*(O) = \{o \in U \mid U / IND(B) \subseteq O\};$$
 (2)

$$B^*(O) = \{o \in U \mid U / IND(B) \cap O \neq \emptyset\}.$$

A discernibility matrix is defined in (3), whose elements are given in (4).

$$M_D(B) = [m_D(i,j)]_{n \times n}, \quad i \geq 1, j \leq card(U / IND(B))$$
 (3)

$$m_D(i,j) = \{a_k \in B \mid f(o_i,a_k) \neq f(o_j,a_k)\}$$
 (4)

A discernibility function is defined in (5), where the set formed by the minimum term of $F(B)$ determines the reducts of B , which is defined as a set of minimum attributes necessary to maintain the same properties of an *IS* that utilizes all the original attributes of the system. There may be more than one reduct for the same set of attributes. For a large *IS*, the calculus of minimal reducts can consist a problem of complex computation, which rises with the amount of data of the process. Some approaches are utilized to deal with this kind of

problem in reduct processing, for example, through similarity relations (Huang et al., 2007). In information systems with data in numerical values, it usually is not necessary to calculate the reducts, because all the variables of the condition attributes are the reducts themselves.

$$F(B)=\wedge\{\vee\overline{m}_D(i,j)\};\overline{m}_D(i,j)=\{\overline{a}_k|a_k\in m_D(i,j)\}.$$

(5)

To transform a reduct into a decision rule, the values of the conditional attributes from the object class from which the reduct was originated are added to the corresponding attributes, and then the rule is completed with the decision attributes. For a determined reduct, an example of decision rule is illustrated in (6). The use of the rough set theory enables systematically that the decision rules have concise informations concerning the original information system, adequately treating eventual redundant, uncertain, or imprecise information in the data.

$$\begin{aligned} &\text{IF } a_1 = f(o_1,a_1) \text{ AND...AND } a_k = f(o_m,a_k) \text{ THEN} \\ &\quad d_1 = f(o_1,d_1) \text{ OR...OR } d_i = f(o_i,d_i) \end{aligned}$$

(6)

2.1 Example 1

As examples of the concepts expressed in this section and the following examples consider Table 2 below, where $U = \{o_1, o_2, o_3, o_4\}$ and $B = \{a_1, a_2\}$. For this information system, we have $U / IND(B) = \{\{o_1\}, \{o_2\}, \{o_3\}, \{o_4\}\}$. The discenibility matrix is illustrated in Table 3. The resulting discernibility function is $F(B) = a_2 \wedge a_1 \wedge (a_1 \vee a_2) \wedge (a_1 \vee a_2) \wedge a_1 \wedge a_2 = a_1 \wedge a_2$. Thus, the reduct obtained is $R = \{a_1, a_2\}$. Therefore, the resulting decision rules are the expressions given in (7).

| | a ₁ | a ₂ | d |
|----------------|----------------|----------------|----------------|
| o ₁ | b | b | δ ₁ |
| o ₂ | b | c | δ ₂ |
| o ₃ | c | b | δ ₃ |
| o ₄ | c | c | δ ₄ |

Table 2. Data referring to Example 1.

| | o ₁ | o ₂ | o ₃ | o ₄ |
|----------------|--------------------------------|--------------------------------|----------------|----------------|
| o ₁ | - | | | |
| o ₂ | a ₂ | - | | |
| o ₃ | a ₁ | a ₁ ,a ₂ | - | |
| o ₄ | a ₁ ,a ₂ | a ₁ | a ₂ | - |

Table 3. Discernibility matrix referring to Example 1.

$$\begin{aligned} &\text{IF } a_1 = b \text{ AND } a_2 = b \text{ THEN } d = \delta_1; \\ &\text{IF } a_1 = b \text{ AND } a_2 = c \text{ THEN } d = \delta_2; \\ &\text{IF } a_1 = c \text{ AND } a_2 = b \text{ THEN } d = \delta_3; \\ &\text{IF } a_1 = c \text{ AND } a_2 = c \text{ THEN } d = \delta_4. \end{aligned}$$

(7)

3. Methodology

For a more adequate representation of the numerical applications, the illustrated form in Table 4 will be adopted for the information systems employed in this paper. The condition attributes are x_i and their data are $x_N^{(k)}$. The decision attribute is y and their values are $y^{(k)}$.

| | | | | | |
|-------------|-------------|-------------|-----|-------------|-----------|
| x_1 | x_2 | x_3 | ... | x_N | y |
| $x_1^{(1)}$ | $x_2^{(1)}$ | $x_3^{(1)}$ | ... | $x_N^{(1)}$ | $y^{(1)}$ |
| $x_1^{(2)}$ | $x_2^{(2)}$ | $x_3^{(2)}$ | ... | $x_N^{(2)}$ | $y^{(2)}$ |
| ... | ... | ... | ... | ... | ... |
| $x_1^{(k)}$ | $x_2^{(k)}$ | $x_3^{(k)}$ | ... | $x_N^{(k)}$ | $y^{(k)}$ |
| ... | ... | ... | ... | ... | ... |
| $x_1^{(m)}$ | $x_2^{(m)}$ | $x_3^{(m)}$ | ... | $x_N^{(m)}$ | $y^{(m)}$ |
| ... | ... | ... | ... | ... | ... |
| $x_1^{(v)}$ | $x_2^{(v)}$ | $x_3^{(v)}$ | ... | $x_N^{(v)}$ | $y^{(v)}$ |

Table 4. Numerical Tabular Representation of an IS.

Sentences (8) derive from the IS in question. For example, for $x_1 = x_1^{(k)}$, $x_2 = x_2^{(k)}$, $x_3 = x_3^{(k)}$, and $x_N = x_N^{(k)}$ we have $y = y^{(m)}$ expressed by s_k . And for $x_1 = x_1^{(m)}$, $x_2 = x_2^{(m)}$, $x_3 = x_3^{(m)}$, and $x_N = x_N^{(m)}$ we have $y = y^{(m)}$ defined by s_m .

$$\begin{aligned} s_1: & \text{ IF } x_1 = x_1^{(1)} \text{ AND } x_2 = x_2^{(1)} \text{ AND... AND } x_N = x_N^{(1)} \text{ THEN } y = y^{(1)} \\ s_2: & \text{ IF } x_1 = x_1^{(2)} \text{ AND } x_2 = x_2^{(2)} \text{ AND... AND } x_N = x_N^{(2)} \text{ THEN } y = y^{(2)} \\ s_k: & \text{ IF } x_1 = x_1^{(k)} \text{ AND } x_2 = x_2^{(k)} \text{ AND... AND } x_N = x_N^{(k)} \text{ THEN } y = y^{(k)} \\ s_m: & \text{ IF } x_1 = x_1^{(m)} \text{ AND } x_2 = x_2^{(m)} \text{ AND...AND } x_N = x_N^{(m)} \text{ THEN } y = y^{(m)} \\ s_v: & \text{ IF } x_1 = x_1^{(v)} \text{ AND } x_2 = x_1^{(v)} \text{ AND...AND } x_N = x_N^{(v)} \text{ THEN } y = y^{(v)} \end{aligned}$$

(8)

For numeric values in ranges defined in the table, that is, $x_1^{(k)} \leq x_1 \leq x_1^{(m)}$, $x_2^{(k)} \leq x_2 \leq x_2^{(m)}$, $x_3^{(k)} \leq x_3 \leq x_3^{(m)}$ and $x_N^{(k)} \leq x_N \leq x_N^{(m)}$, the sentences s_k and s_m defined in (8) may be redefined by generic rule (9), or through the simplified form (10), where $\alpha^{(g)} = [x_1^{(k)}, x_1^{(m)}]$, $\beta^{(g)} = [x_2^{(k)}, x_2^{(m)}]$, $\gamma^{(g)} = [x_N^{(k)}, x_N^{(m)}]$ and $\delta^{(g)} = [y^{(k)}, y^{(m)}]$, considering that $y^{(k)} < y^{(m)}$.

$$r_g: \text{ IF } x_1^{(k)} \leq x_1 \leq x_1^{(m)} \text{ AND } x_2^{(k)} \leq x_2 \leq x_2^{(m)} \text{ AND ... AND } x_N^{(k)} \leq x_N \leq x_N^{(m)} \text{ THEN}$$

$$\min\{y^{(k)}, \dots, y^{(m)}\} \leq y \leq \max\{y^{(k)}, \dots, y^{(m)}\}$$

(9)

$$r_g: \text{ IF } x_1 = \alpha^{(g)} \text{ AND } x_2 = \beta^{(g)} \text{ AND...AND } x_N = \gamma^{(g)} \text{ THEN } y = \delta^{(g)}$$

(10)

To estimate numerical values in ranges of the data obtained in the rules, formula (11) will be used for numerical interpolations (Pinheiro, et al., 2010).

$$y = (x_n, x_n^{(i)}, y^{(i)})_{i=k,m \atop n=1,N} = y^{(k)} + \frac{(y^{(m)} - y^{(k)})}{N} \sum_{n=1}^N \frac{(x_n - x_n^{(k)})}{(x_n^{(m)} - x_n^{(k)})}$$

(11)

3.1 Example 2

In order to illustrate the concepts of this section and of those to follow, Table 5 will illustrate a simple example defined by the function $y = x_1 + x_2$ with x_1 and $x_2 \in [0, 1]$. This table is the

same as Table 2 from Example 1. The *IS* associated has two condition attributes (x_1 and x_2) of numerical values. Consequently, the reduct is defined by $\{x_1, x_2\}$, resulting in the same decision rules as those in (7), which can be written as (10), as proposed in the methodology presented in this section, and resulting in (12).

| x_1 | x_2 | y |
|-------|-------|-----|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 2 |

Table 5. Data referring to Example 2.

r_1 : IF $x_1 = [0, 1]$ AND $x_2 = [0, 1]$ THEN $y = [0, 2]$

(12)

Intermediate values in the data range $[0, 1]$ of the general rule in question can be estimated by (13), constituting a specific case of (11) for $n = 2$.

$$y = y^{(k)} + \frac{(y^{(m)} - y^{(k)})}{2} \left(\frac{(x_1 - x_1^{(k)})}{(x_1^{(m)} - x_1^{(k)})} + \frac{(x_2 - x_2^{(k)})}{(x_2^{(m)} - x_2^{(k)})} \right)$$

(13)

3.2 Fuzzy models

With the information of decision rules in form (12), it is simple to obtain the parameters of a corresponding fuzzy model. For modeling in linguistic (Mamdani) rules (14), two membership functions (Fig. 1), triangular and equally spaced, can be defined in the interval $[0, 1]$ for the input variables (x_1 and x_2), and another three functions (Fig. 2) defined in interval $[0, 2]$ for the output variable (y). Therefore, the resulting fuzzy rules are expressed by (15).

r_n : IF $x_1 = A_n$ AND $x_2 = B_n$ THEN $y = C_n$

(14)

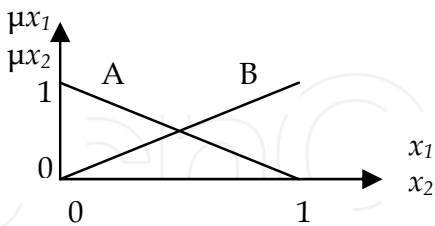


Fig. 1. Membership Functions.

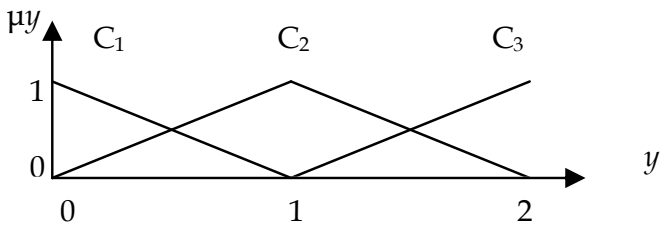


Fig. 2. Membership Functions.

$$\begin{aligned}
r_1: & \text{ IF } x_1 = A \text{ AND } x_2 = A \text{ THEN } y = C_1; \\
r_2: & \text{ IF } x_1 = A \text{ AND } x_2 = B \text{ THEN } y = C_2; \\
r_3: & \text{ IF } x_1 = B \text{ AND } x_2 = A \text{ THEN } y = C_2; \\
r_4: & \text{ IF } x_1 = B \text{ AND } x_2 = B \text{ THEN } y = C_3.
\end{aligned} \tag{15}$$

For modeling with functional (Takagi-Sugeno) rules (16), the membership functions can be the same as those in Figure 1 for the input variables. For the polynomial function coefficients of the information from the output variable, the same one can be calculated by (13), resulting in the rules expressed by (17). As an example of the calculation of the polynomial coefficient functions, using the decision rule in the form (12) with $x_1^{(k)} = 0$, $x_1^{(m)} = 1$, $x_2^{(k)} = 0$, $x_2^{(m)} = 1$, $y^{(k)} = 0$ and $y^{(m)} = 2$, where using (13) we have $y = ((2 - 0)/2)((x_1 - 0)/(1 - 0) + (x_2 - 0)/(1 - 0)) = x_1 + x_2$ which defines the coefficients of (16). Other examples of fuzzy models obtained with this methodology are illustrated in Pinheiro et al., 2010.

$$r_n: \text{ IF } x_1 = A_n \text{ AND } x_2 = B_n \text{ THEN } y_n = c_{0n} + c_{1n}x_1 + c_{2n}x_2 \tag{16}$$

$$\begin{aligned}
r_1: & \text{ IF } x_1 = A \text{ AND } x_2 = A \text{ THEN } y_1 = x_1 + x_2; \\
r_2: & \text{ IF } x_1 = A \text{ AND } x_2 = B \text{ THEN } y_2 = x_1 + x_2; \\
r_3: & \text{ IF } x_1 = B \text{ AND } x_2 = A \text{ THEN } y_3 = x_1 + x_2; \\
r_4: & \text{ IF } x_1 = B \text{ AND } x_2 = B \text{ THEN } y_4 = x_1 + x_2.
\end{aligned} \tag{17}$$

3.3 Rough models

Another simpler modeling option, called rough modeling, directly concerns the representation given in (12), where the data can be interpolated by (13). The advantage of this modeling in relation to the fuzzy models is that it does not require numerical fuzzification and defuzzification procedures, which can be advantageous in real-time applications in control systems, for example. The advantage of fuzzy models is its greater ability to function approximation, which is usually related to the possible intersections between the membership functions of associated fuzzy sets.

In order to illustrate the rough model, we have (12) where $x_1^{(k)} = 0$, $x_1^{(m)} = 1$, $x_2^{(k)} = 0$, $x_2^{(m)} = 1$, $y^{(k)} = 0$ and $y^{(m)} = 2$. For specific values of variables $x_1 = 0.25$ and $x_2 = 0.5$, the corresponding value of y is desired to be estimated. By using expression (13) comes $y = 0 + (2 - 0)/2((0.25 - 0)/(1 - 0) + (0.5 - 0)/(1 - 0)) = 0.75$, which consists of the same numerical value given by the original function of Example 2, where y is exactly given by $x_1 + x_2$.

3.4 Example 3

With the purpose of illustrating situations where data applications have fractional values, Table 6 illustrates an example defined by the nonlinear function $y = \sin(x_1)$, with $x_1 \in [0, \pi/2]$.

The condition attribute (x_1) has fractional values that will be quantized in this example in three equally-spaced intervals: $\alpha^{(1)} = [0.0000, 0.5236]$; $\alpha^{(2)} = [0.5236, 1.0472]$; $\alpha^{(3)} = [1.0472, 1.5708]$. Therefore, the decision rules are expressed by (18).

$$\begin{aligned}
r_1: & \text{ IF } x_1 = \alpha^{(1)} \text{ THEN } y = y^{(a)} \text{ OR } y = y^{(b)} \text{ OR } y = y^{(c)}; \\
r_2: & \text{ IF } x_1 = \alpha^{(2)} \text{ THEN } y = y^{(c)} \text{ OR } y = y^{(d)} \text{ OR } y = y^{(e)}; \\
r_3: & \text{ IF } x_1 = \alpha^{(3)} \text{ THEN } y = y^{(e)} \text{ OR } y = y^{(f)} \text{ OR } y = y^{(g)}.
\end{aligned} \tag{18}$$

| x_1 | y |
|--------|--------------------|
| 0.0000 | $y^{(a)} = 0.0000$ |
| 0.2618 | $y^{(b)} = 0.2588$ |
| 0.5236 | $y^{(c)} = 0.5000$ |
| 0.7854 | $y^{(d)} = 0.7071$ |
| 1.0472 | $y^{(e)} = 0.8660$ |
| 1.3090 | $y^{(f)} = 0.9659$ |
| 1.5708 | $y^{(g)} = 1.0000$ |

Table 6. Data of Example 3.

Using the proposed form (10), the rough model (19) can be written, where $\delta^{(1)} = [0.0000, 0.5000]$, $\delta^{(2)} = [0.5000, 0.8660]$ and $\delta^{(3)} = [0.8660, 1.0000]$.

$$\begin{aligned} r_1: & \text{ IF } x_1 = [0.0000, 0.5236] \text{ THEN } y = [0.0000, 0.5000]; \\ r_2: & \text{ IF } x_1 = [0.5236, 1.0472] \text{ THEN } y = [0.5000, 0.8660]; \\ r_3: & \text{ IF } x_1 = [1.0472, 1.5708] \text{ THEN } y = [0.8660, 1.0000]. \end{aligned}$$

(19)

To estimate the intermediate values of this model, the linear interpolation formula (20) can be used, which is the specific case of (11) for $n = 1$.

$$y = y^{(k)} + (y^{(m)} - y^{(k)})(x_1 - x_1^{(k)}) / (x_1^{(m)} - x_1^{(k)})$$

(20)

For instance, for $x_1 = 0.3927$ we have $y = 0 + (0.5 - 0)(0.3927 - 0) / (0.5236 - 0) = 0.375$, and for $x_1 = 1.1781$, we have $y = 0.866 + (1 - 0.866)(1.1781 - 1.0472) / (1.5708 - 1.0472) = 0.8995$. The average error value in relation to the original function is about 2.3%. A greater degree of quantization relative to the data from the example often leads to better precision in the interpolations, but with an increase in the number of modeling rules.

If eventually more than one rule results in estimated values (for example, for data at the ends of the condition attributes), the resulting value is given by the arithmetic average of the same.

3.5 Software

There are free access computational tools developed specifically for the processing of rough sets, such as RSL (Rough Sets Library), Rough Enough, CI (Column Importance facility), Rosetta, etc. These tools allow the processing of data of generic information systems, providing decision rules in a format similar to (6), for example. Data with fractional numeric values can be properly quantized through some established techniques. The reducts that determine the decision rules can be manually selected or determined by some known methods from the data processing of the *IS* used.

The methodology proposed in this paper allows the use of decision rules derived from processing of information system, aimed at building fuzzy models or rough models in order to design rule-based controllers.

4. Rule-based controllers

Figure 3 illustrates the typical structure of a ruled-based controller with PI action (Proportional plus Integral). The variable “*e*” represents the input error information of the

controller, variable “ u ” symbolizes the output of the same, and “ T ” denotes the sample time. Equation (21) expresses the discrete mathematical model of a PI controller with the respective proportional (K_p) and integral (K_i) gains. Many articles show the computational accomplishments of rule-based controllers, especially those that employ fuzzy logic. The actions of the fuzzy controllers can be PI, PD (proportional plus derivative), PID or Lead/Lad (Pinheiro & Gomide, 2000), depending on the context of their applications. The gains (proportional, integral, etc.) of fuzzy controllers are generally represented by scale factors that multiply the membership functions of the same, or are already fully incorporated in the expressions of their membership functions. Many control problems can be solved using a PI-controller (Astrom & Wittenmark, 1990) due to their applicability and easy tuning.

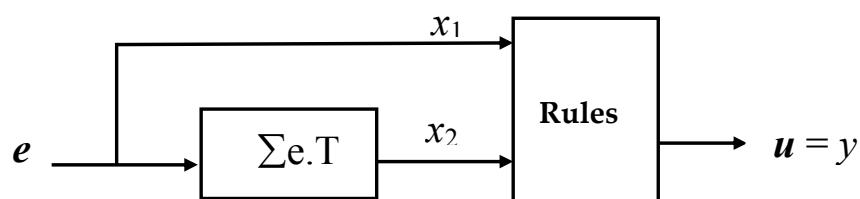


Fig. 3. Typical structure of a rule-based controller with PI action.

$$y = u(t) = K_p e(t) + K_i \sum e(t)T; \quad (21)$$

$$x_1 = K_p e(t); \quad x_2 = K_i \sum e(t)T.$$

4.1 Example 4

With relation to Figure 3, if the rules are the same as those exemplified in items 3.2 and 3.3 (where the simple data of Example 2 was used), Figure 4 shows the response (u) of the respective fuzzy controllers (linguistic and functional) or of the rough controllers for a step change in the error (e). The sample time (T) used was one tenth of a second. The points on the graph illustrate the discrete values resulting from the rule-based controllers (being practically identical to each other). And for the purpose of exemplification, the solid line represents the response of a conventional controller continuous in time with unit gains (proportional and integral). Comparing the results, it is possible to note that the design of the rule-based controllers was well fit.

The next section of this article will deal with more complex problems and practical contexts. Application examples like those of control systems with adaptive gains, active suspension systems, and speed regulator and current control for electric motors will be shown.

Questions regarding stability analysis resulting from the application of rough controllers can be performed by harmonic balance techniques, for example, in the same way that these techniques are used in stability analysis of fuzzy controls (Pinheiro & Gomide, 1997; Rezek et al., 2010).

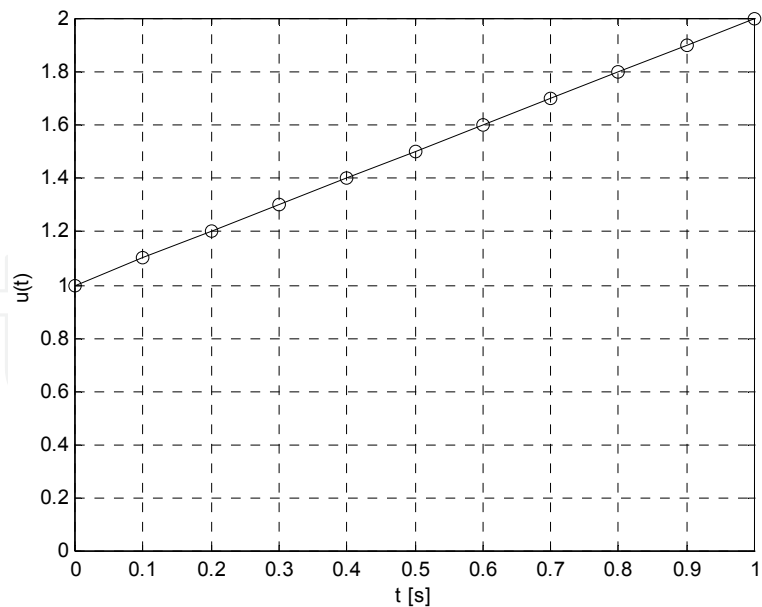


Fig. 4. Responses of rule-based controllers (for a step change in the error).

5. Application examples

This section provides some examples of applications of the methodology proposed to synthesize rule-based controllers, whose objective is to accomplish control loops appropriate for systems with nonlinear behavior, etc.

5.1 Example 5

This example includes a speed control loop of a system that operates in low rotations, which requires a controller with characteristics of adaptive gains due to the nonlinear effects of the controlled process. The block diagram illustrated in Figure 5 represents the controlled process with a transfer function (22) and two nonlinearities. The second nonlinearity, indicated by block (b), defines a dead-zone effect related to gear gaps of the system. The transfer function $P(s)$, shapes an electric motor that drives the system. The poles of the same are related to the electrical part associated with resistance and inductance of the motor. The mechanical part is related to moments of inertia and friction of the machine with its mechanical charge. The nominal values of the parameters are: $K = 2.55$; $c_0 = 0.73$; $c_1 = 1.74$; $d_0 = 0.73$. The saturation levels are ± 12 , the range of the dead-zone is ± 1 . Figure 6 illustrates a typical control loop to regulate the speed of the process, which works within a specific rotation range.

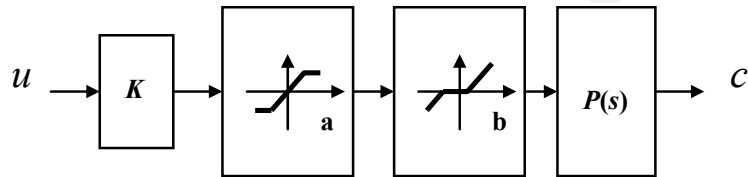


Fig. 5. Non-linear process

$$P(s) = \frac{d_0}{(s^2 + c_1s + c_0)} \tag{22}$$

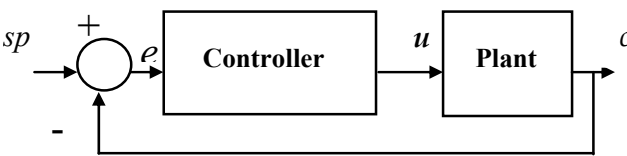


Fig. 6. Control Loop.

Figure 7 shows the responses of the control loop in question for a conventional PI controller with gains $K_p = 12$ and $K_i = 1$. The same were adjusted to meet the specifications of overshoot around 20% and settling time around seven seconds for a reference value or set point (sp) at 2.8 [rd/s]. The response values were normalized (c/sp) and are related to the following reference values $sp = [1.5; 2; 2.8]$. Due to the nonlinear characteristics of the plant, the dynamic responses of the control loop change according to the set-point values. Alterations in the control gains in function with the intensity of the error in the control loop, maintain the system dynamic within the desired specifications. The mapping of these gains by artificial neural networks or by fuzzy logic for example, allows for the accomplishment of controllers with characteristics of adaptive gains. Table 7 illustrates some suitable gain values in function with the intensity (x_1) of the error (e) of the control loop and its integral (x_2), in order to properly compensate the process. The mapping (or scheduling) of the gains can be defined as $u = y = K_p(x_1)x_1 + K_i(x_2)x_2$. Figure 8 illustrates the values of this mapping, where the data relative to the information on the input variables are at the top part of the figure, with x_1 in black and x_2 in gray. The output information (u) of the controller is found below the graphic.

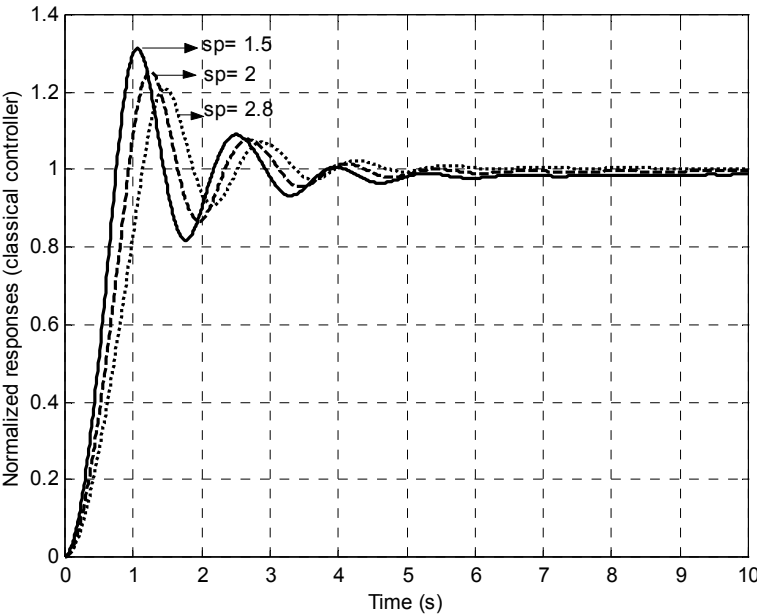


Fig. 7. Responses relative to Example 5 for a classic controller.

The information in Figure 8 represent the table of the information system of the problem in question, where it is desired to design a rule-based controller that incorporates the scaling gains, aiming for an effective compensation of the controlled process. This paper will employ the Rosetta (Øhrn & Komorowski, 1997), a software for processing of data related to information systems in general. This is a simple use freely accessed tool (<http://www.idi.ntnu.no/~aleks/rosetta/>). The following procedures were performed in

| x_1 | K_p | x_2 | K_i |
|-------|-------|-------|-------|
| 0.00 | 20.0 | 0.00 | 1.40 |
| 0.08 | 20.0 | 0.07 | 1.40 |
| 0.16 | 11.5 | 0.15 | 1.00 |
| 0.31 | 6.37 | 0.30 | 1.00 |
| 0.74 | 3.23 | 0.72 | 1.00 |
| 1.19 | 2.35 | 1.17 | 0.89 |
| 1.62 | 2.02 | 1.60 | 0.76 |
| 2.00 | 2.00 | 1.99 | 0.75 |
| 2.50 | 2.00 | 2.48 | 0.75 |
| 3.01 | 2.00 | 3.00 | 0.75 |

Table 7. Adaptive gains in function of the error and its integral.

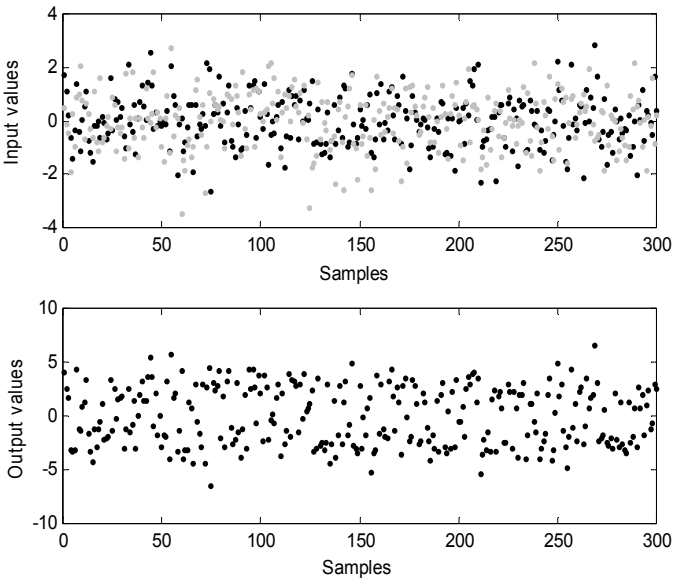


Fig. 8. Mapping of the gains.

the tool: *Import IS; Discretization* \rightarrow *Equal frequency binning*; \rightarrow *Intervals* = 5; *Reduction* \rightarrow *Exhaustive calculation*; *Rule generator*. The decision rules (the first three and the last two) that resulted from processing the data done by the software are shown below (23). The “*” symbol denotes the inferior and superior values of the data of the IS correspondent, that in this example are -2.6759 and 2.8149 for x_1 and -3.5027 and 2.7042 for x_2 .

$x_1(0.6875,*)$ AND $x_2(0.2942,0.8800) \Rightarrow y(4.0889)$ OR $y(4.2937)$... OR $y(2.5230)$...OR $y(3.4186)$
 $x_1(0.6875,*)$ AND $x_2(-0.2123,0.2942) \Rightarrow y(2.4749)$ OR $y(3.6601)$ OR $y(5.4837)$... OR $y(1.8793)$
 $x_1(0.1744,0.6875)$ AND $x_2(-0.9279,-0.2123) \Rightarrow y(1.7301)$ OR ... OR $y(2.0570)$ OR $y(1.2289)$...
...
 $x_1(0.1744,0.6875)$ AND $x_2(0.2942,0.8800)) \Rightarrow y(2.8625)$ OR $y(3.0640)$ OR ... OR $y(2.2344)$...
 $x_1(*,-0.8340)$ AND $x_2(0.2942, 0.8800) \Rightarrow y(-2.4899)$ OR $y(-1.8370)$ OR ... OR $y(-3.2713)$... (23)

By using the methodology proposed, the rules above can be written as (24), whose parameter values are $x_1^{(a)} = -2.6759$; $x_1^{(b)} = -0.834$; $x_1^{(c)} = -0.2338$; $x_1^{(d)} = 0.1744$; $x_1^{(e)} = 0.6875$; $x_1^{(f)} = 2.8149$; $x_2^{(a)} = -3.5027$; $x_2^{(b)} = -0.9279$; $x_2^{(c)} = -0.2123$; $x_2^{(d)} = 0.2942$; $x_2^{(e)} = 0.88$; $x_2^{(f)} = 2.7042$.

$$\begin{aligned}
r_1: & \text{ IF } x_1 = [x_1^{(e)}, x_1^{(f)}] \text{ AND } x_2 = [x_2^{(d)}, x_2^{(e)}] \text{ THEN } y = [2.5230, 4.2937]; \\
r_2: & \text{ IF } x_1 = [x_1^{(e)}, x_1^{(f)}] \text{ AND } x_2 = [x_2^{(c)}, x_2^{(d)}] \text{ THEN } y = [1.8793, 5.4837]; \\
r_3: & \text{ IF } x_1 = [x_1^{(d)}, x_1^{(e)}] \text{ AND } x_2 = [x_2^{(b)}, x_2^{(c)}] \text{ THEN } y = [1.2289, 2.0570]; \\
r_4: & \text{ IF } x_1 = [x_1^{(b)}, x_1^{(c)}] \text{ AND } x_2 = [x_2^{(a)}, x_2^{(b)}] \text{ THEN } y = [-3.4899, -2.8470]; \\
r_5: & \text{ IF } x_1 = [x_1^{(a)}, x_1^{(b)}] \text{ AND } x_2 = [x_2^{(c)}, x_2^{(d)}] \text{ THEN } y = [-6.5810, -1.9420]; \\
r_6: & \text{ IF } x_1 = [x_1^{(b)}, x_1^{(c)}] \text{ AND } x_2 = [x_2^{(e)}, x_2^{(f)}] \text{ THEN } y = [-1.2319, -0.4610]; \\
r_7: & \text{ IF } x_1 = [x_1^{(a)}, x_1^{(b)}] \text{ AND } x_2 = [x_2^{(e)}, x_2^{(f)}] \text{ THEN } y = [-2.7080, 1.1847]; \\
r_8: & \text{ IF } x_1 = [x_1^{(c)}, x_1^{(d)}] \text{ AND } x_2 = [x_2^{(d)}, x_2^{(e)}] \text{ THEN } y = [-1.8116, 2.4170]; \\
r_9: & \text{ IF } x_1 = [x_1^{(b)}, x_1^{(c)}] \text{ AND } x_2 = [x_2^{(c)}, x_2^{(d)}] \text{ THEN } y = [-2.4210, -1.62330]; \\
r_{10}: & \text{ IF } x_1 = [x_1^{(a)}, x_1^{(b)}] \text{ AND } x_2 = [x_2^{(b)}, x_2^{(c)}] \text{ THEN } y = [-4.2604, -2.4360]; \\
r_{11}: & \text{ IF } x_1 = [x_1^{(c)}, x_1^{(d)}] \text{ AND } x_2 = [x_2^{(e)}, x_2^{(f)}] \text{ THEN } y = [-1.2340, 2.2624]; \\
r_{12}: & \text{ IF } x_1 = [x_1^{(c)}, x_1^{(d)}] \text{ AND } x_2 = [x_2^{(b)}, x_2^{(c)}] \text{ THEN } y = [-3.0277, 1.6000]; \\
r_{13}: & \text{ IF } x_1 = [x_1^{(c)}, x_1^{(d)}] \text{ AND } x_2 = [x_2^{(a)}, x_2^{(b)}] \text{ THEN } y = [-4.4430, 4.1896]; \\
r_{14}: & \text{ IF } x_1 = [x_1^{(b)}, x_1^{(c)}] \text{ AND } x_2 = [x_2^{(b)}, x_2^{(c)}] \text{ THEN } y = [-3.2030, -2.0760]. \\
r_{15}: & \text{ IF } x_1 = [x_1^{(e)}, x_1^{(f)}] \text{ AND } x_2 = [x_2^{(b)}, x_2^{(c)}] \text{ THEN } y = [1.1753, 6.4760]; \\
r_{16}: & \text{ IF } x_1 = [x_1^{(b)}, x_1^{(c)}] \text{ AND } x_2 = [x_2^{(d)}, x_2^{(e)}] \text{ THEN } y = [-1.8250, -1.0780]; \\
r_{17}: & \text{ IF } x_1 = [x_1^{(e)}, x_1^{(f)}] \text{ AND } x_2 = [x_2^{(a)}, x_2^{(b)}] \text{ THEN } y = [0.6120, 2.7360]; \\
r_{18}: & \text{ IF } x_1 = [x_1^{(c)}, x_1^{(d)}] \text{ AND } x_2 = [x_2^{(c)}, x_2^{(d)}] \text{ THEN } y = [-2.0980, 2.1297]; \\
r_{19}: & \text{ IF } x_1 = [x_1^{(d)}, x_1^{(e)}] \text{ AND } x_2 = [x_2^{(c)}, x_2^{(d)}] \text{ THEN } y = [1.7996, 2.4580]; \\
r_{20}: & \text{ IF } x_1 = [x_1^{(d)}, x_1^{(e)}] \text{ AND } x_2 = [x_2^{(e)}, x_2^{(f)}] \text{ THEN } y = [2.9106, 3.6160]; \\
r_{21}: & \text{ IF } x_1 = [x_1^{(a)}, x_1^{(b)}] \text{ AND } x_2 = [x_2^{(a)}, x_2^{(b)}] \text{ THEN } y = [-5.4544, -3.0290]; \\
r_{22}: & \text{ IF } x_1 = [x_1^{(e)}, x_1^{(f)}] \text{ AND } x_2 = [x_2^{(e)}, x_2^{(f)}] \text{ THEN } y = [2.8684, 5.6692]; \\
r_{23}: & \text{ IF } x_1 = [x_1^{(d)}, x_1^{(e)}] \text{ AND } x_2 = [x_2^{(a)}, x_2^{(b)}] \text{ THEN } y = [0.6848, 1.2190]; \\
r_{24}: & \text{ IF } x_1 = [x_1^{(d)}, x_1^{(e)}] \text{ AND } x_2 = [x_2^{(d)}, x_2^{(e)}] \text{ THEN } y = [2.2344, 3.0640]; \\
r_{25}: & \text{ IF } x_1 = [x_1^{(a)}, x_1^{(b)}] \text{ AND } x_2 = [x_2^{(d)}, x_2^{(e)}] \text{ THEN } y = [-3.2713, -1.8400].
\end{aligned} \tag{24}$$

Figure 9 has the normalized responses of the control loop now using the rough controller designed by the rules (24). The responses tend to maintain the specified characteristics of overshoot and settling time for different set-point values, different from the conventional PI controller responses (whose responses are shown in Fig. 7). This shows that the rule-based controller incorporated the relationships (nonlinear) of the gains from Table 7 in function of the error and its integration. The performance of the controller has adaptive actions according to the intensity of the error information of the control loop.

The rules for a corresponding functional fuzzy controller are obtained by the form described in item 3.2 from the rules (24). The resulting coefficients of the polynomial functions of the fuzzy model in form (16) are: $c_{01} = 1.79$; $c_{11} = 0.42$; $c_{21} = 1.51$; $c_{02} = 2.05$; $c_{12} = 0.85$; $c_{22} = 3.56$; $c_{03} = 1.62$; $c_{13} = 0.81$; $c_{23} = 0.58$; $c_{04} = -2.60$; $c_{14} = 0.54$; $c_{24} = 0.12$; $c_{05} = -2.24$; $c_{15} = 1.26$; $c_{25} = 4.58$; $c_{06} = -0.88$; $c_{16} = 0.64$; $c_{26} = 0.21$; $c_{07} = -0.82$; $c_{17} = 1.06$; $c_{27} = 1.07$; $c_{08} = -1.66$; $c_{18} = 5.18$; $c_{28} = 3.61$; $c_{09} = -1.70$; $c_{19} = 0.66$; $c_{29} = 0.79$; $c_{010} = -1.75$; $c_{110} = 0.49$; $c_{210} = 1.27$; $c_{011} = -1.08$; $c_{111} = 4.28$; $c_{211} = 0.96$; $c_{012} = 1.30$; $c_{112} = 5.67$; $c_{212} = 3.23$; $c_{013} = 3.90$; $c_{113} = 10.57$; $c_{213} = 1.67$; $c_{014} = 3.89$; $c_{114} = 4.40$; $c_{214} = 3.69$; $c_{015} = 3.75$; $c_{115} = 1.24$; $c_{215} = 3.70$; $c_{016} = -1.49$; $c_{116} = 0.62$; $c_{216} = 0.64$; $c_{017} = 1.71$; $c_{117} = 0.50$; $c_{217} = 0.41$; $c_{018} = 0.00$; $c_{118} = 5.18$; $c_{218} = 4.17$; $c_{019} = 1.82$; $c_{119} = 0.64$; $c_{219} = 0.65$; $c_{020} = 2.62$; $c_{120} = 0.69$; $c_{220} = 0.19$; $c_{021} = -2.04$; $c_{121} = 0.66$; $c_{221} = 0.47$; $c_{022} = 1.74$; $c_{122} = 0.66$; $c_{222} = 0.77$; $c_{023} = 0.96$; $c_{123} = 0.52$; $c_{223} = 0.10$; $c_{024} = 1.88$; $c_{124} = 0.81$; $c_{224} = 0.71$; $c_{025} = -2.59$; $c_{125} = 0.39$; $c_{225} = 1.22$. The modal values for

the Gaussian membership functions are obtained by the arithmetic average of the parameter values of the antecedents of the rough rules (24), in other words: $m_{1ab} = (x_1^{(a)} + x_1^{(b)})/2 = -1.755$; $m_{1bc} = (x_1^{(b)} + x_1^{(c)})/2 = -0.5339$; $m_{1cd} = (x_1^{(c)} + x_1^{(d)})/2 = -0.0297$; $m_{1de} = (x_1^{(d)} + x_1^{(e)})/2 = 0.431$; $m_{1ef} = (x_1^{(e)} + x_1^{(f)})/2 = 1.7512$; $m_{2ab} = (x_2^{(a)} + x_2^{(b)})/2 = -2.2153$; $m_{2bc} = (x_2^{(b)} + x_2^{(c)})/2 = -0.5701$; $m_{2cd} = (x_2^{(c)} + x_2^{(d)})/2 = 0.041$; $m_{2de} = (x_2^{(d)} + x_2^{(e)})/2 = 0.5871$; $m_{2ef} = (x_2^{(e)} + x_2^{(f)})/2 = 1.7921$. The dispersion values of the membership functions (0.8 in this example) are chosen in order for the intersection of the same to remain in a membership degree around 0.5. The results obtained with the corresponding fuzzy controller are very similar to the responses illustrated in Figure 9.

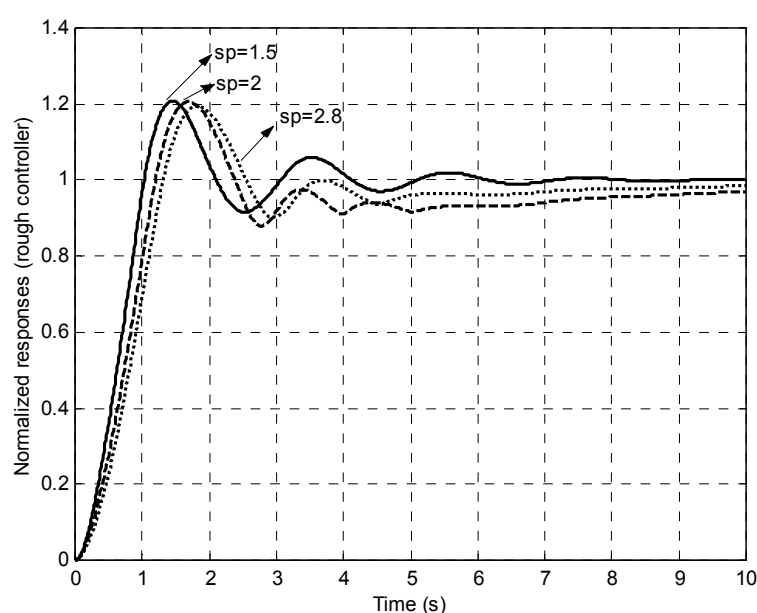


Fig. 9. Responses relative to Example 5 with rough controller.

5.2 Example 6

This example deals with an active suspension model used in automotive systems. Figure 10 illustrates a typical system known as $\frac{1}{4}$ model. The spring and damper of the structure are represented by coefficients K_f and B , respectively. The parameter M_s corresponds to the sprung mass of the vehicle. The M_r is the mass of the wheel and tire and K_p represents the elasticity of the same. d_p , d_r and d_s are vertical displacement of the tire, wheel and body of the vehicle, respectively. The force F_a represents the action exerted by an active damper aiming the imposition of determined dynamic characteristics in the suspension.

The system can be represented in state variables (25). Variable x_1 represents the vertical displacement of the suspended mass, x_2 represents the speed of the same, and its derivation is the corresponding acceleration. Variable x_3 represents the vertical displacement of the wheel, x_4 represents the speed of the same, and its derivation is the corresponding acceleration. Variable u_1 expresses a disturbance in the suspension, like the vertical displacement of the tire. The magnitude of u_2 represents the compensation force of the damper system.

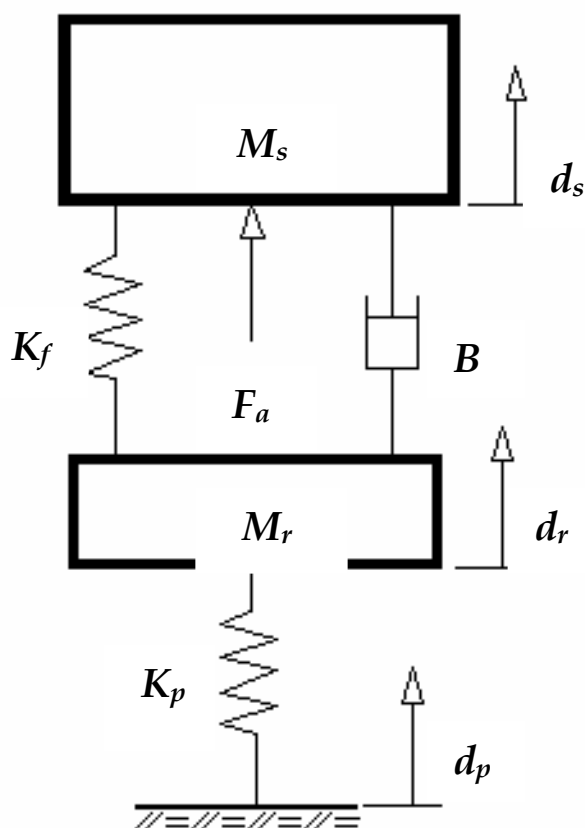


Fig. 10. Model of Example 6.

$$\begin{aligned}
 \dot{x}_1 &= x_2 \\
 \dot{x}_2 &= -\frac{K_f}{M_s}x_1 - \frac{B_a}{M_s}x_2 + \frac{K_f}{M_s}x_3 + \frac{B_a}{M_s}x_4 + \frac{1}{M_s}u_2 \\
 \dot{x}_3 &= x_4 \\
 \dot{x}_4 &= \frac{K_f}{M_r}x_1 + \frac{B_a}{M_r}x_2 - \frac{K_f + K_p}{M_r}x_3 - \frac{B_a}{M_r}x_4 + \frac{K_p}{M_r}u_1 - \frac{1}{M_r}u_2
 \end{aligned} \tag{25}$$

There are some types of well-known strategies to control active suspension systems. Expression (26) defines a typical strategy. The magnitude of F_a corresponds to the force developed by the active damper in the system. The same depends on values C_{on} and C_{off} defined for the coefficient of the damper system (obtained by controlled leaking of fluid of the damper by an electrically controlled valve) or by variations of magnetic characteristics of the fluid by a current-controlled induction), along with information of the absolute speed (V_{abs}) and relative speed (V_{rel}) of the process. V_{abs} is the absolute speed of the sprung mass and V_{rel} is the relative speed between the sprung mass and the mass of the wheel-tire set.

$$F_a = \begin{cases} C_{on}V_{abs} + C_{off}V_{rel} & \text{if } V_{abs}V_{rel} \geq 0, \\ C_{off}V_{rel} & \text{if } V_{abs}V_{rel} < 0. \end{cases} \tag{26}$$

Some papers (Pinheiro et al., 2007; Dong, et al., 2010) show the application of fuzzy logic to control suspension systems. In the first reference cited, the fuzzy control rules were obtained by qualitative analyses of the logic expressed by (36). The results obtained with the use of fuzzy controller were better than those with the typical control. This explanation is that with the traditional algorithm, the command force of the system is only related to the two values (C_{on} and C_{off}) of the coefficient of the damper selected by the logic. The compensation force for the fuzzy controller can vary in a wider operation range in function of the membership functions adopted. Figure 11 shows the values of the variables of the suspension system under various operating conditions.

Now, the methodology proposed in this paper will be applied to generate a rule-based controller to control the suspension system in question. The data in Figure 11 constitutes the information system of the example, where x_1 is related to V_{abs} , x_2 with V_{rel} and y with F_a . Similar to the previous example, the *IS* in question was processed by Rosetta, and by using the proposed method the rules (27) were synthesized, where: $x_1^{(a)} = -2.385$; $x_1^{(b)} = -0.681$; $x_1^{(c)} = -0.184$; $x_1^{(d)} = 0.383$; $x_1^{(e)} = 0.90$; $x_1^{(f)} = 2.731$; $x_2^{(a)} = -0.3153$; $x_2^{(b)} = -0.078$; $x_2^{(c)} = -0.008$; $x_2^{(d)} = 0.04$; $x_2^{(e)} = 0.1$; $x_2^{(f)} = 0.368$.

- r_1 : IF $x_1^{(e)} \leq x_1 \leq x_1^{(f)}$ AND $x_2^{(e)} \leq x_2 \leq x_2^{(f)}$ THEN $-3.709 \leq y \leq 1.562$
- r_2 : IF $x_1^{(e)} \leq x_1 \leq x_1^{(f)}$ AND $x_2^{(d)} \leq x_2 \leq x_2^{(e)}$ THEN $-3.593 \leq y \leq 1.379$
- r_3 : IF $x_1^{(c)} \leq x_1 \leq x_1^{(d)}$ AND $x_2^{(d)} \leq x_2 \leq x_2^{(e)}$ THEN $0.621 \leq y \leq 0.226$
- r_4 : IF $x_1^{(e)} \leq x_1 \leq x_1^{(f)}$ AND $x_2^{(b)} \leq x_2 \leq x_2^{(c)}$ THEN $-2.385 \leq y \leq -1.23$
- r_5 : IF $x_1^{(e)} \leq x_1 \leq x_1^{(f)}$ AND $x_2^{(c)} \leq x_2 \leq x_2^{(d)}$ THEN $-2.279 \leq y \leq -1.218$
- r_6 : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(c)} \leq x_2 \leq x_2^{(d)}$ THEN $-1.092 \leq y \leq -0.597$
- r_7 : IF $x_1^{(a)} \leq x_1 \leq x_1^{(b)}$ AND $x_2^{(a)} \leq x_2 \leq x_2^{(b)}$ THEN $1.513 \leq y \leq 3.387$
- r_8 : IF $x_1^{(e)} \leq x_1 \leq x_1^{(f)}$ AND $x_2^{(a)} \leq x_2 \leq x_2^{(b)}$ THEN $-2.251 \leq y \leq -1.128$
- r_9 : IF $x_1^{(a)} \leq x_1 \leq x_1^{(b)}$ AND $x_2^{(d)} \leq x_2 \leq x_2^{(e)}$ THEN $0.967 \leq y \leq 3.443$
- r_{10} : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(b)} \leq x_2 \leq x_2^{(c)}$ THEN $-0.513 \leq y \leq -1.062$
- r_{11} : IF $x_1^{(a)} \leq x_1 \leq x_1^{(b)}$ AND $x_2^{(c)} \leq x_2 \leq x_2^{(d)}$ THEN $0.923 \leq y \leq 3.174$
- r_{12} : IF $x_1^{(c)} \leq x_1 \leq x_1^{(d)}$ AND $x_2^{(a)} \leq x_2 \leq x_2^{(b)}$ THEN $-0.437 \leq y \leq -0.074$
- r_{13} : IF $x_1^{(b)} \leq x_1 \leq x_1^{(c)}$ AND $x_2^{(a)} \leq x_2 \leq x_2^{(b)}$ THEN $0.783 \leq y \leq 1.547$
- r_{14} : IF $x_1^{(c)} \leq x_1 \leq x_1^{(d)}$ AND $x_2^{(c)} \leq x_2 \leq x_2^{(d)}$ THEN $-0.555 \leq y \leq 0.188$
- r_{15} : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(a)} \leq x_2 \leq x_2^{(b)}$ THEN $-1.088 \leq y \leq -0.580$
- r_{16} : IF $x_1^{(c)} \leq x_1 \leq x_1^{(d)}$ AND $x_2^{(e)} \leq x_2 \leq x_2^{(f)}$ THEN $-1.048 \leq y \leq 0.116$
- r_{17} : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(d)} \leq x_2 \leq x_2^{(e)}$ THEN $-1.361 \leq y \leq -0.773$
- r_{18} : IF $x_1^{(b)} \leq x_1 \leq x_1^{(c)}$ AND $x_2^{(d)} \leq x_2 \leq x_2^{(e)}$ THEN $0.282 \leq y \leq 0.800$
- r_{19} : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(c)} \leq x_2 \leq x_2^{(d)}$ THEN $0.300 \leq y \leq 0.810$
- r_{20} : IF $x_1^{(c)} \leq x_1 \leq x_1^{(d)}$ AND $x_2^{(b)} \leq x_2 \leq x_2^{(c)}$ THEN $-0.384 \leq y \leq 0.300$
- r_{21} : IF $x_1^{(a)} \leq x_1 \leq x_1^{(b)}$ AND $x_2^{(e)} \leq x_2 \leq x_2^{(f)}$ THEN $0.854 \leq y \leq 2.688$
- r_{22} : IF $x_1^{(d)} \leq x_1 \leq x_1^{(e)}$ AND $x_2^{(e)} \leq x_2 \leq x_2^{(f)}$ THEN $-2.169 \leq y \leq -0.992$
- r_{23} : IF $x_1^{(b)} \leq x_1 \leq x_1^{(c)}$ AND $x_2^{(e)} \leq x_2 \leq x_2^{(f)}$ THEN $0.235 \leq y \leq 0.848$
- r_{24} : IF $x_1^{(a)} \leq x_1 \leq x_1^{(b)}$ AND $x_2^{(b)} \leq x_2 \leq x_2^{(c)}$ THEN $1.073 \leq y \leq 2.998$
- r_{25} : IF $x_1^{(b)} \leq x_1 \leq x_1^{(c)}$ AND $x_2^{(b)} \leq x_2 \leq x_2^{(c)}$ THEN $0.408 \leq y \leq 0.991$

(27)

A suspension model with the parameters $M_s = 400$ [Kg], $M_r = 50$ [Kg], $B_a = 500$ [Ns/m], $K_f = 20000$ [N/m], $K_p = 250000$ [N/m], using a classical control with $C_{off} = 500$ [Ns/m], $C_{on} = 1400$ [Ns/m], and applying the strategy defined by rules (27), in Figure 12 we have responses of

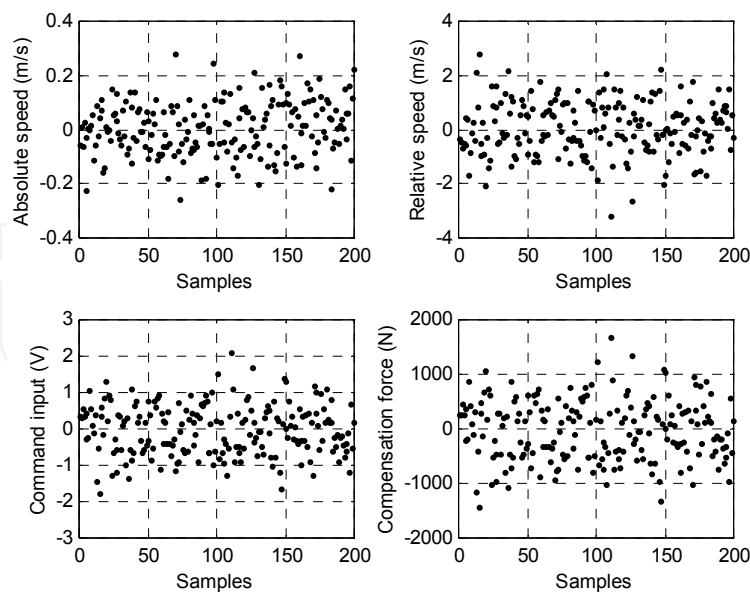


Fig. 11. Values of the variables of the suspension system under various operating conditions.

the acceleration of the sprung mass of the process for a sudden dislocation of 0.05 meters in the tire of the system. The results obtained indicate a better response (smaller acceleration) of the system using a rule-based controller in relation to the classical strategy. Therefore, just as in the fuzzy controller cited, the compensation force commanded by the rough controller can vary in wider operation ranges, since the rules incorporate the various operating conditions of the system (Fig. 11) in its generation procedure.

5.3 Example 7

This example shows a real application of control loops in cascade for speed regulation and current control in a drive system with a DC motor. Figure 13 shows a block diagram of the process in question. The motor is activated by a driver (chopper), which uses power transistor. Electronic circuits generate firing pulses to command the chopper and are controlled by a computer that performs the control algorithms of the system, in other words, two regulation loops in cascaded (Fig. 14) for the variables speed and current. Hall sensors provide information on the stator current (I_a) of the motor and the rotation (W) of the same, whose information are acquired by a data acquisition system coupled with the control computer. A synchronous machine operating as a generator feeds a set of electrical resistors switched by relays, and this set works as variable loads for the DC motor. This system has nonlinearities, mainly due to saturation of the driver used (amplifier and chopper) and the nonlinear characteristics of the series excitation motor. Real results of the tests performed in this system will be shown. The results are derived from experiments that use conventional controllers with PI actions to regulate the speed and current of the system, and rough control algorithms also with proportional and integral actions for the same purposes. Discrete representations equal to (28) were used for the realizations of the control algorithms, where variable “ e ” represents the control loop error (of the speed and of the current), “ u ” symbolizes the output variable of the controller in question, and “ a_1 , b_0 and b_1 ” are the parameters for the classic PI controllers.

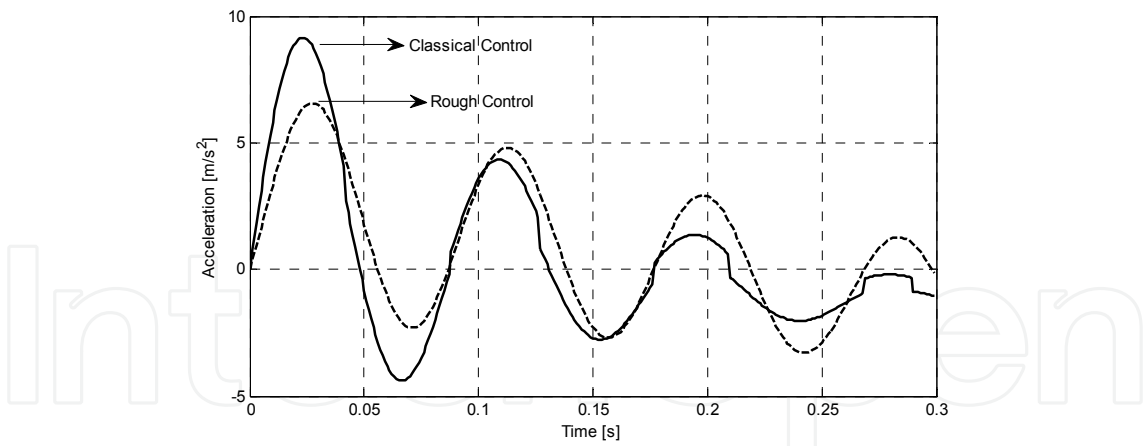


Fig. 12. Responses of the suspension system with classic and rough controls.

$$u(t) = b_oe(t) + b_1e(t - 1) + a_1u(t - 1)$$

(28)

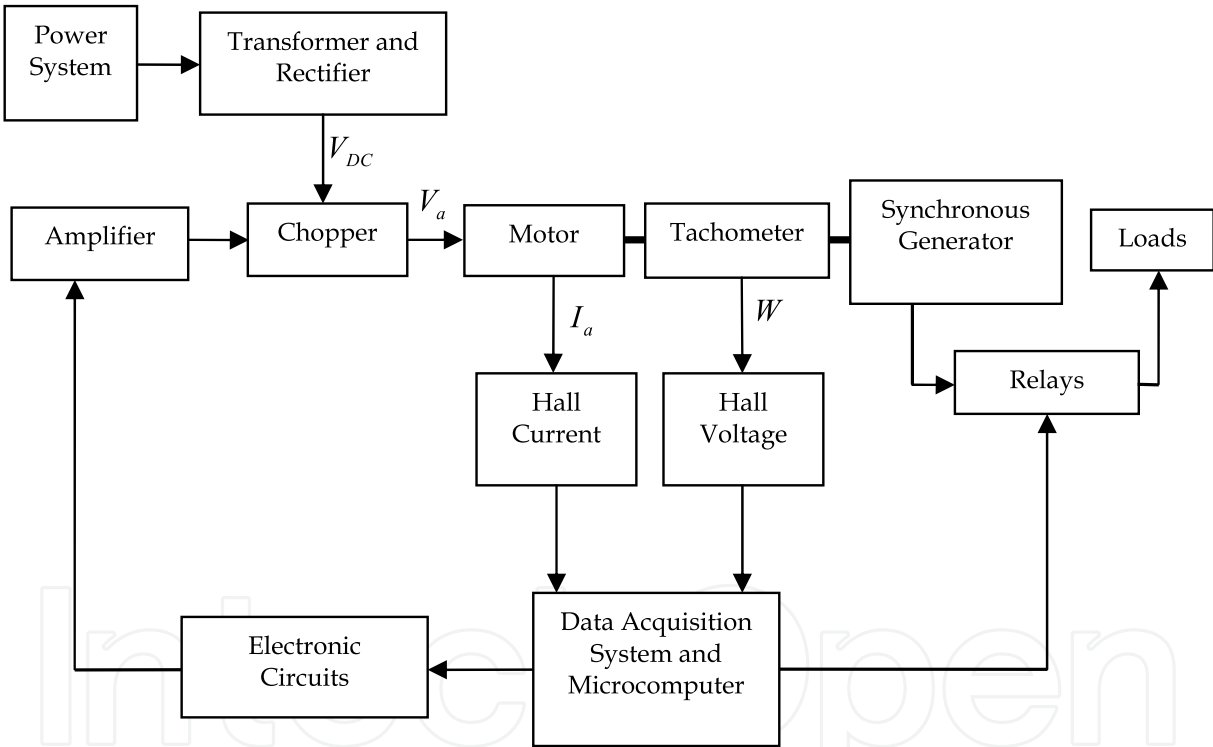


Fig. 13. Block diagram of the system in reference to Example 7.

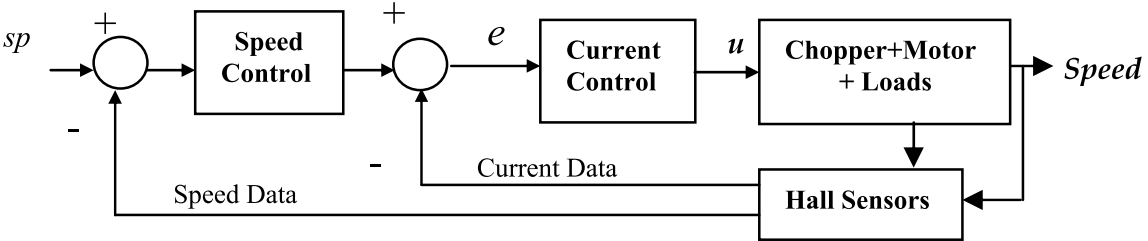


Fig. 14. Control Loops of Example 7

Figure 15 illustrates data from the practical tests with the conventional controllers, where the values of the current error and of the output command are normalized in p.u. The parameters are $a_1 = 1$, $b_0 = 0.5074$, $b_1 = -0.406$, and the sample time is 0.01 [s]. The variables $x_1 = e(t)$, $x_2 = e(t-1)$, $x_3 = u(t-1)$ and $y = u(t)$ will be used to generate the rules of a rough controller for the current loop.

Rosetta was used with the following procedures performed in the tool: *Import IS; Discretization → Equal frequency binning → Intervals = 3; Reduction → Manual Reducer; Rule generator*. The rules obtained are shown below, the first three and the last two.

$$\begin{aligned} r_1: & \quad x_1 = [0.3283, \quad 1.0000] \text{ AND } x_2 = [-0.3368, 0.3283] \text{ AND } x_3 = [-0.0628, 0.3362] \\ & \quad \quad \quad \text{THEN } y = [0.1261, \quad 0.9346]; \\ r_2: & \quad x_1 = [-0.3434, 0.3283] \text{ AND } x_2 = [0.3283, 1.0000] \text{ AND } x_3 = [0.3362, 1.0000] \\ & \quad \quad \quad \text{THEN } y = [-0.0640, \quad 0.8150]; \\ r_3: & \quad x_1 = [-0.3434, 0.3283] \text{ AND } x_2 = [-0.3368, 0.3283] \text{ AND } x_3 = [-0.0628, 0.3362] \\ & \quad \quad \quad \text{THEN } y = [-0.2517, \quad 0.4416]; \\ & \quad \quad \quad \dots \\ r_{26}: & \quad x_1 = [0.3283, \quad 1.0000] \text{ AND } x_2 = [-1.0000, -0.3283] \text{ AND } x_3 = [0.3362, 1.0000] \\ & \quad \quad \quad \text{THEN } y = [0.7209, \quad 1.2189]; \\ r_{27}: & \quad x_1 = [-1.0000, -0.3434] \text{ AND } x_2 = [-1.0000, -0.3283] \text{ AND } x_3 = [0.3362, 1.0000] \\ & \quad \quad \quad \text{THEN } y = [0.1696, 0.3973]. \end{aligned} \tag{29}$$

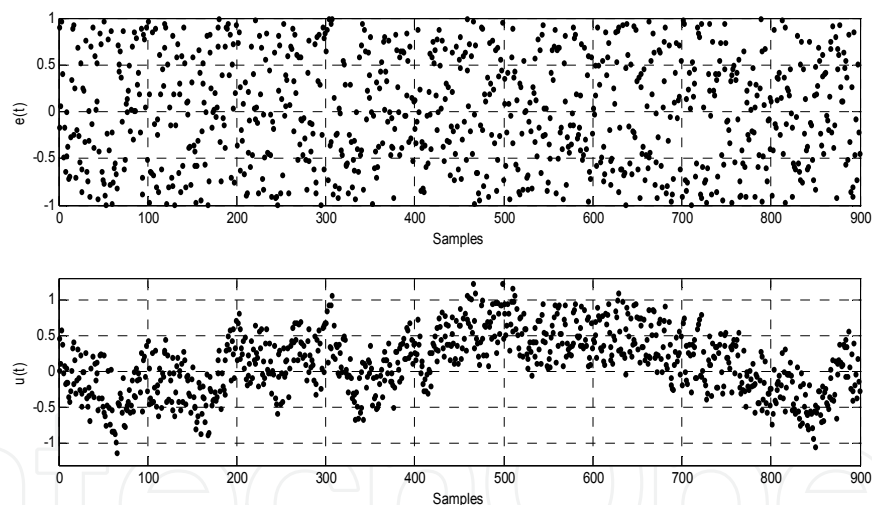


Fig. 15. Values of the variables for the system under various operating conditions.

Now that the rough controller has information on three inputs, numerical values in ranges of the data obtained in the rules can be interpolated by means of (11) with $n = 3$. The acquisition of rules for the rough controller in the speed loop is performed similarly as described for the current loop. Figure 16 shows the real result of a test performed on the described system. The responses of the speed regulation and of the current became better with rough controllers than with classic controllers, as much in the starting of the motor as in the load alterations of the same. There are smaller peaks in the current and speed, both in speed variations (such increasing the input reference in the starting of the motor, for example), and in load variation (in this case a reduction that occurred between 7 and 8 seconds in the test). The explanation for these characteristics is due to the fact that the rule-

based controllers incorporate the various operating conditions of the system, generating rules to compensate suitably the nonlinearities of the system.

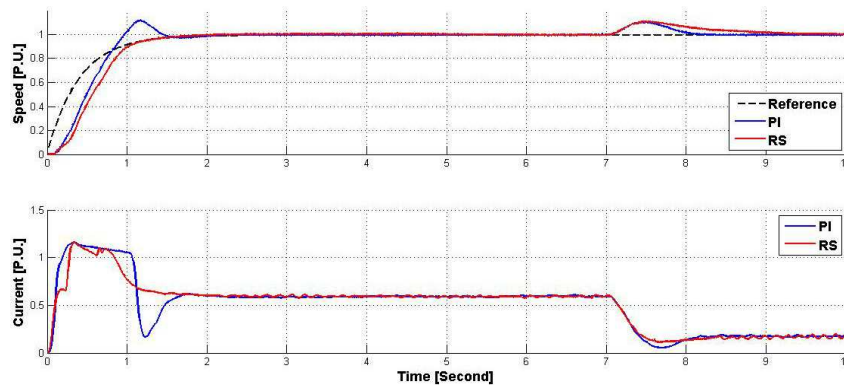


Fig. 16. Real responses of the system with classic and rough controllers.

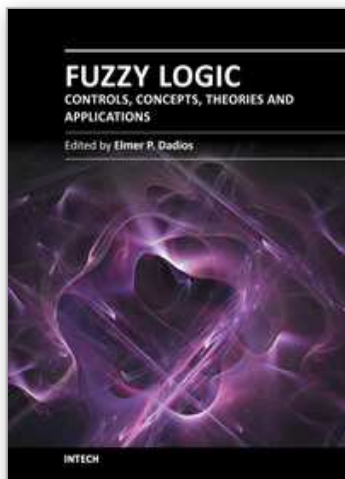
6. Conclusion

This paper has presented a new approach to design rule-based controllers using concepts about rough sets. The proposed methodology allows obtaining rule parameters in a systematic form and with simple computations, as much for fuzzy controllers as for rough controllers. Example 1 illustrates some basic concepts about rough sets. Using a simple linear function is shown in Example 2 how to apply the approach proposed in this chapter in the modeling of rule-based models. Example 3 shows how a rough model can estimate the values associated with a basic nonlinear function. The results obtained in Example 4 show the same values for a fuzzy model and a rough model, when the approach involves a linear function. In this example the linear function was associated with the function of a proportional-integral controller. These results can also be confronted with those obtained in the work referenced in Pinheiro et al., 2010. In Example 5 a practical context of adaptive gains is synthesized through a rough controller in the control of a nonlinear system. Example 6 deals with an active suspension model used in automotive systems. The methodology proposed in this paper was applied to generate a rule-based controller to control the suspension system in question. The results can be confronted with those obtained in the works referenced in Pinheiro et al., 2007 and Dong et al., 2010. The dynamic responses obtained were similar to the works mentioned. An experimental application was shown in Example 7, an example of control loops in cascade for speed regulation and current control in a drive system with a DC motor. Two rough controllers were synthesized to regulate the speed and current in the system. The results can be compared with those obtained in the work referenced in Rezek et al., 2010. The dynamic responses obtained were similar to the work mentioned, where was used two fuzzy controllers for the same purposes. The results obtained in this work indicate that the methodology proposed is adequate for applications in real control systems. The impact of the rough controllers in relation to the fuzzy controllers is that it does not require fuzzification and defuzzification procedures, which can be advantageous in real-time applications for control systems. The application of LMI (linear matrix inequalities) techniques and Lyapunov functions will also be investigated to design rough controllers and to analyze the stability in control loops, the same way that these methods are applied in control systems that use functional fuzzy

controllers (Wang et al., 1996; Tseng & Chen, 2009). Future papers will address issues with rough controllers aiming at applications in control systems with multiple inputs and multiple outputs (MIMO).

7. References

- Astrom, K. & Wittenmark, B. (1990). *Computer-Controlled Systems: Theory and Design*, Prentice Hall, ISBN , 0-13-168600-3, Englewood Cliffs, USA
- Dong, X.; Yu, M.; Liau, C. & Chen, W. (2010). Comparative research on semi-active control strategies for magneto-rheological suspension, *Nonlinear Dynamics*, Springer, Vol.59, pp. 433-453, Netherlands
- Huang, B.; Guo, L. & Zhou, X. (2007). Approximation Reduction Based on Similarity Relation, *IEEE Fourth International Conf. on Fuzzy Systems and Knowledge Discovery*, pp. 124-128
- Kusiak, A. & Shah, A. (2006). Data-Mining-Based System for Prediction of Water Chemistry Faults, *IEEE Transactions on Industrial Electronics*, No.2, pp. 593-596
- Øhrn, A. & Komorowski, J. (1997). ROSETTA: A Rough Set Toolkit for Analysis of Data. *Third International Joint Conference on Information Sciences*, pp. 403 – 407
- Pawlak, Z. (1982). Rough Sets, *International Journal of Computer and Inf. Sciences*, pp. 341-356,
- Pawlak, Z. (1991). *Rough Sets: Theoretical Aspects of Reasoning about Data*. Kluwer Academic Publishers, Dordrecht, Holland
- Pawlak, Z. & Skowron, A. (2007). Rudiments of rough sets, *Information Sciences*, 177; pp. 3-27
- Pedrycz, W. & Gomide, F. (2007). *Fuzzy Systems Engineering: Toward Human Centric Computing*, Wiley Interscience/IEEE, ISBN 978-0-471-78857-7, Hoboken, USA
- Pinheiro, C. & Gomide, F. (1997). Frequency Response Design of Fuzzy Controllers, *VII International Fuzzy Systems Association World Congress*, Vol.3, pp. 434-439, Praga
- Pinheiro, C. & Gomide, F. (2000). On Tuning Nonlinear Fuzzy Control Systems. In: *IEEE International Conference on Fuzzy Systems, IX IEEE International Conference on Fuzzy Systems* (in CD ROM), San Antonio, USA
- Pinheiro, C.; Machado, J.; Bombard, A.; Lima, J. & Dias, J. (2007), Fuzzy Logic Control For Magnetorheological Damper, *Electrorheological Fluids And Magnetorheological Suspensions*, World Scientific, pp. 603-609, Singapore
- Pinheiro, C.; Gomide, F.; Carpinteiro, O. & Lima, I. (2010). Granular Synthesis of Rule-Based Models and Function Approximation using Rough Sets, *Novel Developments in Granular Computing*, ed. JingTao Yao, Information Science, ISBN 978-1-60566-324-1, N. York, USA
- Rezek, A.; Pinheiro, C.; Darido, T.; Silva, V.; Vicentini, O. & Assis, W. (2010). Comparative Performance Analysis for Digital Regulators in Series DC Motor Controlled Drive, *International Journal of Power & Energy Systems*, Vol.30, No.1, pp. 15-22
- Skowron, A. & Son, N. H (1995). Quantization of Real value Attributes: Rough Set and Boolean reasoning approach, *International Joint Conf. on Information Sciences*, pp. 34- 37
- Tseng, C. & Chen, B. (2009), Robust Fuzzy Observer-Based Fuzzy Control Design for Nonlinear Discrete-Time Systems with Persistent Bounded Disturbances, *IEEE Transactions on Fuzzy Systems*, Vol.17, No.3, pp. 711-723
- Wang, H.; Tanaka, K. & Griffin, M. (1996), An Approach to Fuzzy Control of Nonlinear Systems: Stability and Design Issues, *IEEE Transactions on Fuzzy Systems*, Vol.4, No.1, pp. 14-23
- Ziarko, W. & Katzberg, J. (1993). Rough sets approach to system modeling and control algorithm acquisition. *IEEE Conference: Communications, Computers and Power in the Modern Environment*, pp. 154-164



Fuzzy Logic - Controls, Concepts, Theories and Applications

Edited by Prof. Elmer Dadios

ISBN 978-953-51-0396-7

Hard cover, 428 pages

Publisher InTech

Published online 28, March, 2012

Published in print edition March, 2012

This book introduces new concepts and theories of Fuzzy Logic Control for the application and development of robotics and intelligent machines. The book consists of nineteen chapters categorized into 1) Robotics and Electrical Machines 2) Intelligent Control Systems with various applications, and 3) New Fuzzy Logic Concepts and Theories. The intended readers of this book are engineers, researchers, and graduate students interested in fuzzy logic control systems.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Carlos Pinheiro, Ulisses Camatta and Angelo Rezek (2012). Rough Controller Synthesis, Fuzzy Logic - Controls, Concepts, Theories and Applications, Prof. Elmer Dadios (Ed.), ISBN: 978-953-51-0396-7, InTech, Available from: <http://www.intechopen.com/books/fuzzy-logic-controls-concepts-theories-and-applications/rough-controllers>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2012 The Author(s). Licensee IntechOpen. This is an open access article distributed under the terms of the [Creative Commons Attribution 3.0 License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IntechOpen

IntechOpen