XIII International SAUM Conference
on Systems, Automatic Control and Measurements
Niš, Serbia, November 09th-11th, 2016

Hybrid-Fuzzy Controllers Applied in DC Servo Drive

S. S. Nikolić +, D. Antić +, S. Perić +, N. Danković +, M. Spasić +, and M. Milovanović +

+ University of Niš, Faculty of Electronic Engineering, Department of Control Systems, Aleksandra Medvedeva 14, 18000 Niš, Republic of Serbia
Phone: 00381529-363, Fax: 00381588-399,

E-Mails: {sasa.s.nikolic, dragan.antic, stanisa.peric, nikola.dankovic, miodrag.spasic, miroslav.b.milovanovic}@elfak.ni.ac.rs

Abstract* - The main idea of this paper is to present a possibility of application of hybrid-fuzzy controllers in control systems theory. In this paper, we have described a new method of using orthogonal functions in control of dynamical systems. We have used the existing orthogonal functions of Legendre type. These functions generate genarilzed quasi-orthogonal filter, which are used in the concluding phase of the fuzzy controllers. Proposed hybrid-fuzzy controllers of Takagi-Sugeno type has been applied to a DC servo drive system and performed experiments have verified efficiency and improvements of a new control method.

***Key words:* hybrid-fuzzy controller, quasi-orthogonal filter, quasi-orthogonal functions, modular DC servo drive.**

I. INTRODUCTION

Fuzzy control was introduced in early 1970s as an attempt to control systems which is too complex and highly nonlinear so their modelling is very difficult. Fuzzy logic has become a developing area after its introduction by Zadeh, especially over the last twenty years.

In the cases when the system parameters or complete model of a system are unknown, such as we have qualitative, inaccurate or uncertain information about the system, use of fuzzy controllers has verified as very efficient.

In papers [1], [2] transformation matrices (orthogonal algorithm of the least squares) were used in process of fuzzy modelling. These transformations were used for simplifying of fuzzy rules in design of the appropriate fuzzy controllers. Orthogonal estimator was applied for fuzzy controller parameter identification in [3]. Selection of optimal values of fuzzy rules based on orthogonal functions is introduced in [4]. Fuzzy controllers described in this way present generalization of existing traditional fuzzy controllers. In this paper generalization of Takagi-Sugeno fuzzy controllers [5-8] will be performed. Namely, existing generalized quasi-orthogonal functions will be used in consequential part of fuzzy controller rules.

II. GENERALIZED QUASI-ORTHOGONAL POLYNOMIALS

Orthogonal polynomials were in the focus of the research for the last two centuries. One of the most important applications of orthogonal polynomials is designing orthogonal filters. Today, these filters become an efficient tool for identification, modelling and control of dynamical systems [9-12].

Quasi-orthogonal functions and especially quasi-orthogonal polynomials as well as their numerous applications are discussed in many papers [11-13]. It is important to notice that classical orthogonal filters and orthogonal signal generators have transfer functions with the order of numerator polynomial for one less then denominator. In practice there is often need for filters of more general type i.e., filters with difference in orders of polynomials in transfer functions higher than one. This can be accomplished by using quasi-orthogonal filters.

On the other hand, many systems in practice are imperfect where we neglect some dynamics which can represents with constants ε and δ [9, 12, 14] in transfer function. This is a reason of formulation of theory about almost orthogonal and quazi-orthogonal functions and polynomials. Polynomials defined in this way are actually combination of two classes of polynomials and they are their generalization suitable for further applications.

Now, we define a combination of almost [9] and quasi-orthogonal polynomials [13] $P_n^{(k,\delta)}(x)$:

$$P_n^{(k,\delta)}(x)=\sum_{i=0}^{n}A_{n,i}^{k,\delta}x^i\,, \qquad (1)$$

where:

$$A_{n,i}^{k,\delta}=(-1)^{n+i+k}\frac{\prod_{j=1}^{n-k}(i+j\delta)}{i!(n-i)!}\,. \qquad (2)$$

In previous relation, k represents the order of quasi orthogonality, and δ represents constants, which is defined as $\delta=1+\varepsilon\approx 1$. Now, we can define quasi-orthogonality via inner product over interval (0, 1) with weight function $w(x)=1$ as:

$$\int_0^1 P_m^{(k,\delta)}(x)P_n^{(k,\delta)}(x)\,dx=\begin{cases}Q_k(\delta), & m\neq n,\\ N_n^{k,\delta}, & m=n,\end{cases}, \qquad (3)$$

where $Q_k(\delta)=q_0+q_1\delta+q_2\delta^2+\cdots+q_k\delta^k$ represents polynomial.

The first few members of the first order (k=1) quasi-orthogonal polynomial over interval (0, 1) with weight function $w(x)$=1 sequence are [12]:

$$P_1^{(1,\delta)}(x) = -x+1,$$
$$P_2^{(1,\delta)}(x) = -\frac{(\delta+2)}{2}x^2 + (\delta+1)x - \frac{\delta}{2},$$
$$P_3^{(1,\delta)}(x) = -\frac{(\delta+3)(2\delta+3)}{6}x^3 + (\delta+1)(\delta+2)x^2 - \quad (4)$$
$$-\frac{(\delta+1)(2\delta+1)}{2}x + \frac{\delta^2}{3},$$

After applying the substitution $x = e^{-t}$ to (4), we have the first few members of the first order (k=1) quasi-orthogonal polynomials in time domain:

$$\varphi_1^{(1,\delta)}(t) = -e^{-t}+1,$$
$$\varphi_2^{(1,\delta)}(t) = -\frac{(\delta+2)}{2}e^{-2t} + (\delta+1)e^{-t} - \frac{\delta}{2},$$
$$\varphi_3^{(1,\delta)}(t) = -\frac{(\delta+3)(2\delta+3)}{6}e^{-3t} + \quad (5)$$
$$+(\delta+1)(\delta+2)e^{-2t} - \frac{(\delta+1)(2\delta+1)}{2}e^{-t} + \frac{\delta^2}{3}.$$

In similar manner, quasi-orthogonal filters of the second order (k=2) can be generated.

Complete mathematical background for designing generalized quasi-orthogonal filters based on polynomials given by (4) can be found in [12, 15].

III. HYBRID-FUZZY CONTROLLERS

In order to generalize Takagi-Sugeno fuzzy controllers we will start with orthogonal base given by (5). Functions (5) will be used in consequential part of fuzzy controller rules. Range of the parameter δ can be determined by performing several experiments based on designers' experience [9, 12, 14, 15]. Membership functions $f_i(x)$ of normalized inputs [15], can be represented as follows:

$$f(x) = \sum_{j=1}^{M} p_j(x)\theta_j, \quad (6)$$

where function $p_j(x)$ is defined in the following way:

$$p_j(x) = \frac{\prod_{i=1}^{n}\mu_{A_i^j}(x_i)}{\sum_{j=1}^{M}\left[\prod_{i=1}^{n}\mu_{A_i^j}(x_i)\right]}, \; j = 1,2,\ldots,M, \quad (7)$$

whereby functions $\mu_{A_i^j}(x_i)$ are triangular membership functions, i.e.:

$$\mu_{A_i^j}(x_i) = \begin{cases} 0, & x \le a,\; c \le x, \\ \frac{x-a}{b-a}, & a \le x \le b, \\ \frac{c-x}{c-b}, & c \le x, \end{cases} \quad (8)$$

where m_i^j and σ_i^j are centers and scaled parameters of the membership functions, respectively, wherein: $a < b < c$, M is number of fuzzy rules, and n is number of inputs of fuzzy controller. Parameters θ_j are outputs centers of membership functions. It is clear that other types of functions can be chosen for membership functions. Rules derivation is the next step in designing fuzzy controller. In the case of type of fuzzy rules database "*multiple inputs – one output – multiple rules*", the most common in fuzzy control in practice, the following consideration is valid [15]. In general case, if we choose inputs, i.e., linguistic variables are e_i, $i = 1,2,...,r$ with the appropriate values E_{ij_i}, $i = 1,2,...,r$, $j_i = 1,2,...,n_i$, where n_i is a number of linguistic values defined for linguistic variable e_i, $\mu_{E_{ij_i}}$ represents membership function E_{ij_i}. Due to simplify the problem, let us suppose to number of linguistic variables is same for each input, i.e., $n_i = n$, $i = 1,2,...,r$. Total number of linguistic values for r inputs is $n \times r$. If output is control signal u with linguistic values $\varphi_k^{(1,\delta)}$, $k = 1,2,\ldots,m$ and appropriate membership functions $\mu_{\varphi_k^{(1,\delta)}}$, then rules database has the following form:

$$\begin{aligned} &\text{Rule}_1: \; IF\; (e_1 = E_{11}\; AND \ldots e_r = E_{r1})\; THEN\; \left(u = \varphi_1^{(1,\delta)}\right)\; OR, \\ &\vdots \\ &\text{Rule}_m: \; IF\; (e_1 = E_{1m}\; AND \ldots e_r = E_{rm})\; THEN\; \left(u = \varphi_m^{(1,\delta)}\right). \end{aligned} \quad (9)$$

When defined values $e_{10}, e_{20}, \ldots, e_{r0}$ are at inputs, conclusions are as follows:

$$\begin{aligned} &\text{Rule}_1: \; \mu^*_{\varphi_1^{(1,\delta)}}(u) = MIN\left(\mu_{E11}(e_{10}), \ldots, \mu_{Er1}(e_{r0}), \mu_{\varphi_1^{(1,\delta)}}(u)\right), \; \forall u \in \varphi^{(1,\delta)}, \\ &\vdots \\ &\text{Rule}_m: \; \mu^*_{\varphi_m^{(1,\delta)}}(u) = MIN\left(\mu_{E1m}(e_{m0}), \ldots, \mu_{Erm}(e_{r0}), \mu_{\varphi_m^{(1,\delta)}}(u)\right), \; \forall u \in \varphi^{(1,\delta)}. \end{aligned} \quad (10)$$

$$\mu^*_{\varphi^{(1,\delta)}}(u) = MAX\left(\mu^*_{\varphi_1^{(1,\delta)}}(u), \mu^*_{\varphi_2^{(1,\delta)}}(u), \ldots, \mu^*_{\varphi_k^{(1,\delta)}}(u), \ldots, \mu^*_{\varphi_m^{(1,\delta)}}(u)\right), \; \forall u \in \varphi^{(1,\delta)}, \quad (11)$$

where $\varphi^{(1,\delta)}$ is output set. Rules determined with (6)-(11) are derived whether based on process model or experts knowledge about controlled process. If we design PD fuzzy controller, for example, rules have the following form:

$$if\ e\ is \left\langle lin.value \right\rangle and\ \dot{e}\ is \left\langle lin.value \right\rangle then\ u\ is \left\langle \varphi_i^{(1,\delta)} \right\rangle . \quad (12)$$

When we define all rules, and if number of linguistic variables is not too large, we can form rules table for easy review. Fuzzy controller defined by generaralized quazi-orthogonal filter of the first order (*k*=1), i.e. by using relations (6)-(11), is actually generalized hybrid-fuzzy controller of T-S-1 type. In the similar way, we will define generalized hybrid-fuzzy controller of T-S-2 type. Unlike T-S-1 type, in T-S-2 type controller we use singleton functions in consequential part of fuzzy rules instead generalized quasi-orthogonal functions i.e., $\varphi_i^{(1,\delta)}$. These functions are defined by inner product $\left(\varphi_m^{(1,\delta)}(t), \varphi_n^{(1,\delta)}(t)\right)$ via quazi-orthogonal functions $\varphi_m^{(1,\delta)}$ on the interval (0, ∞) with weight *w*(*x*)=1 in the following way:

$$\int_0^\infty \varphi_m^{(1,\delta)}(t)\varphi_n^{(1,\delta)}(t)\,dt = \begin{cases} Q_{mn}^{1,\delta}(\delta), & m \neq n, \\ N_n^{1,\delta}, & m = n, \end{cases} \quad (13)$$

or via inner product of polynomials $\left(P_m^{(1,\delta)}(x), P_n^{(1,\delta)}(x)\right)$ on the interval (0, 1) with weight *w*(*x*)=1:

$$\int_0^1 P_m^{(1,\delta)}(x)P_n^{(1,\delta)}(x)\,dx = \begin{cases} Q_{mn}^{1,\delta}(\delta), & m \neq n, \\ N_n^{1,\delta}, & m = n, \end{cases} \quad (14)$$

where $Q_{mn}^{1,\delta}(\delta)$ is a polynomial which has real value for certain value *δ*. On the other hand, in order to design generalized hybrid T-S-2 fuzzy controller, we will give several values of inner product (14):

$$\begin{aligned} N_1^{1,\delta} &= \int_0^1 P_1^{(1,\delta)}(x)P_1^{(1,\delta)}(x)\,dx = \frac{1}{3}, \\ Q_{21}^{1,\delta} &= \int_0^1 P_2^{(1,\delta)}(x)P_1^{(1,\delta)}(x)\,dx = \frac{1}{24}(2-3\delta), \quad (15) \\ N_2^{1,\delta} &= \int_0^1 P_2^{(1,\delta)}(x)P_2^{(1,\delta)}(x)\,dx = \frac{1}{60}(2-3\delta+3\delta^2). \end{aligned}$$

For certain value of parameter *δ*, values of inner product is calculated (15). These values are used during forming rules database of hybrid T-S-2 fuzzy controller [15]. Fuzzy rules are determined in the similar way as the previously mentioned type of fuzzy controller. Using the same notation as for T-S-1 type, we can create the rules in the following way:

$$\text{Rule}_m: \ IF\ \left(e_1 = E_{1m}\ AND \ldots e_r = E_{rm}\right)\ THEN\ \left(u = \left(P_m^{(1,\delta)}(x), P_{m\pm1}^{(1,\delta)}(x)\right)\right),$$

$$\mu^*_{\left(P_i^{(1,\delta)}(x),P_j^{(1,\delta)}(x)\right)}(u) = MAX\begin{pmatrix} \mu^*_{\left(P_1^{(1,\delta)}(x),P_1^{(1,\delta)}(x)\right)}(u),\ \mu^*_{\left(P_2^{(1,\delta)}(x),P_1^{(1,\delta)}(x)\right)}(u),\ldots \\ \ldots,\mu^*_{\left(P_k^{(1,\delta)}(x),P_{k\pm1}^{(1,\delta)}(x)\right)}(u),\ldots,\ \mu^*_{\left(P_m^{(1,\delta)}(x),P_{m\pm1}^{(1,\delta)}(x)\right)}(u) \end{pmatrix}$$

where $\left(P_i^{(1,\delta)}(x), P_j^{(1,\delta)}(x)\right)$ is output set.

IV. EXPERIMENTAL RESULTS

We have used modular DC servo drive for validation of proposed hybrid fuzzy controllers. This servo system [15, 16], beside hardware, includes open-architecture software that extends the MATLAB environment for real-time control experiments. The servo system setup consists of several modules (DC motor, brass inertia, backlash, encoder, magnetic brake, gearbox with the output disc) arranged in the chain, mounted at the metal rail and coupled with the small clutches. The rotation angle of the shaft is measured via incremental encoder. RTDAC/USB acquisition board with A/D converters is used for all the measurements whereby all the functions of the board can be accessed from the Modular Servo Toolbox. The model of the system which is considered is linear because of ignoring effect of dry friction as well as effect of saturation [15, 16].

The goal of the positional servo drive control is to lead radial position to desired referent position by appropriate control signal (rotor voltage). For this purpose hybrid-fuzzy PD controller (T-S-1 type) is designed. Outputs of the controller are error $e = \theta_{ref} - \theta$ (difference between desired and actual value of rotor's angle) and its differential $\dot{e}$, with the following technical requirements: saturation time $t_s \leq 5\text{s}$ and maximal leap $\Pi_{\%} \leq 10$. By using MATLAB fuzzy toolbox parameters of hybrid fuzzy controller are determined. Selected linguistic variables for inputs are „*e*“ and „*de*“, and for the output is „*u*“. For variables „*e*“ and „*de*“ the same set of values is chosen {negative big, negative small, around zero, positive small, positive big}, i.e. {NB, NS, Z, PS, PB} and the same triangular type of membership functions shown in Fig. 1.

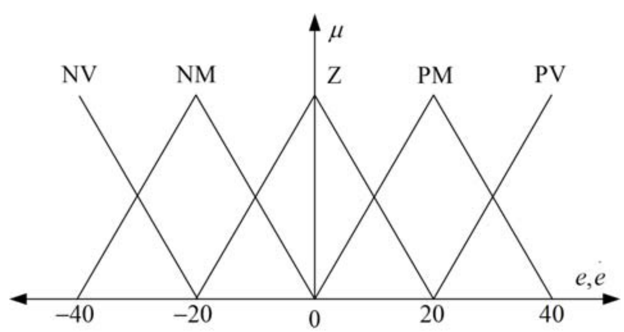

Fig 1 Membership functions

Generalized quazi-orthogonal type of membership functions described by (6) is chosen for the variable „*u*“. Adopted value for parameter *δ* is 1.02. Definition domain of the variables are *U(e)*=[-40 40]; *U($\dot{e}$)*=[-25 25]; *U(u)*=[0 1]. Rules database of the hybrid-fuzzy controller of T-S-1 type, with diagonal form [15] is designed using (9) and (12) and it has 25 rules. These rules are given in Table I.

TABLE I
Rules database of the hybrid-fuzzy controller of PD type

e / $\dot{e}$	NB	NS	Z	PS	PB
NB	$\varphi_5^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$	$\varphi_2^{(1,\delta)}$	$\varphi_2^{(1,\delta)}$
NS	$\varphi_3^{(1,\delta)}$	$\varphi_5^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$	$\varphi_2^{(1,\delta)}$
Z	$\varphi_3^{(1,\delta)}$	$\varphi_3^{(1,\delta)}$	$\varphi_5^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$
PS	$\varphi_1^{(1,\delta)}$	$\varphi_3^{(1,\delta)}$	$\varphi_3^{(1,\delta)}$	$\varphi_5^{(1,\delta)}$	$\varphi_4^{(1,\delta)}$
PB	$\varphi_1^{(1,\delta)}$	$\varphi_1^{(1,\delta)}$	$\varphi_3^{(1,\delta)}$	$\varphi_3^{(1,\delta)}$	$\varphi_5^{(1,\delta)}$

By using chosen parameters of the hybrid-fuzzy controller and considered system we obtained experimental results shown in Figs. 2 and 3. Due to verification, traditional PD controller is designed using well-known method for setting poles [15]. After applying of this method, we obtained the following values for PD controller's parameters: k_P=0.6 and k_D=0.03. From the figures we can see that the system with proposed fuzzy controller faster reach the steady state (158rad/s) with referent step input and also better tracks given referent signal.

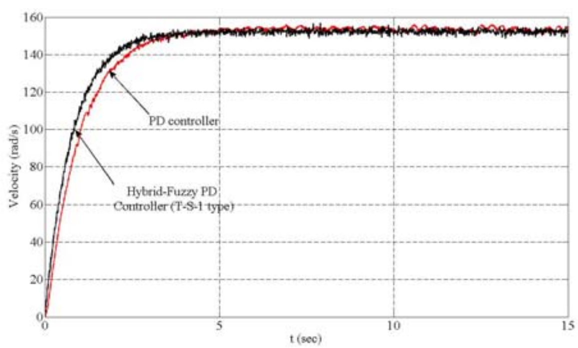

Fig 2 The response of the system with appropriate controllers

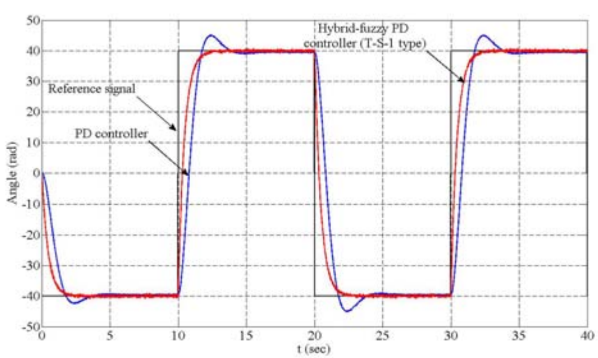

Fig 3 The response of the system with traditional and hybrid-fuzzy controller

ACKNOWLEDGMENT

This paper was realized as a part of the projects III 43007, and TR 35005, financed by the Ministry of Education, Science and Technological Development of the Republic of Serbia within the framework of integrated and interdisciplinary research for the period 2011-2016.

CONCLUSION

In this paper generalized quasi-orthogonal functions of the first order (combination almost and quasi-orthogonal functions) in the consequential part of Takagi-Sugeno fuzzy controller (T-S-1 type) are used. Instead of linear functions new orthogonal base was used. All results and theoretically developed methods were verified by experiments performed on modular DC servo drive system.

REFERENCES

[1] H. H. Su, Z. J. Yao, "Fuzzy analysis method for multi-index orthogonal test", *Journal of Nanjing University of Aeronautics & Astronautics*, vol. 36, no. 1, pp. 29–33, 2004.
[2] J. Yen, L. Wang, "Simplifying fuzzy rule-based models using orthogonal transformation methods", *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 29, no. 1, pp. 13–24, 1999.
[3] L. Wang, R. Langari, "Building Sugeno-type models using fuzzy discretization and orthogonal parameter estimation techniques", *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 4, pp. 454–458, 1995.
[4] A. Lotfi, M. Howarth, J. B. Hull. "Orthogonal fuzzy rule-based systems: Selection of optimum rules", *Neural Computing & Applications*, vol. 9, no. 1, pp. 4–11, 2000.
[5] D. Antić, M. Milojković, Z. Jovanović, S. Nikolić, "Optimal design of the fuzzy sliding mode control for a DC servo drive", *Journal of Mechanical Engineering*, vol. 56, no. 7–8, pp. 455–463, 2010.
[6] D. Antić, M. Milojković, S. Nikolić, "Fuzzy sliding mode control with additional fuzzy control component", *FACTA UNIVERSITATIS Series: Automatic Control and Robotics*, vol. 8, no. 1, pp. 25–34, 2009.
[7] R.-E. Precup, E. M. Petriu, M.-B. Rădac, St. Preitl, L.-O. Fedorovici, C.-A. Dragoş, "Cascade control system-based cost effective combination of tensor product model transformation and fuzzy control", *Asian Journal of Control*, vol. 17, no. 2, pp. 381–391, 2014.
[8] R.-E. Precup, R.-C. David, E. M. Petriu, M.-B. Rădac, St. Preitl, J. Fodor, "Evolutionary optimization-based tuning of low-cost fuzzy controllers for servo systems", *Knowledge-Based Systems*, vol. 38, pp. 74–84, 2013.
[9] B. Danković, S. Nikolić, M. Milojković, Z. Jovanović, "A class of almost orthogonal filters", *Journal of Circuits, Systems, and Computers*, vol. 18, no. 5, pp. 923–931, 2009.
[10] S. Nikolić, D. Antić, B. Danković, M. Milojković, Z. Jovanović, S. Perić, "Orthogonal functions applied in antenna positioning", *Advances in Electrical and Computer Engineering*, vol. 10, no. 4, pp. 35–42, 2010.
[11] S. S. Nikolić, D. S. Antić, S. Lj. Perić, N. B. Danković, M. T. Milojković, "Design of generalised orthogonal filters: Application to the modelling of dynamical systems", *International Journal of Electronics*, vol. 103, no. 2, pp. 269–280, 2016.
[12] S. S. Nikolić, D. S. Antić, M. T. Milojković, M. B. Milovanović, S. Lj. Perić, D. B. Mitić, "Application of neural networks with orthogonal activation functions in control of dynamical systems", *International Journal of Electronics*, vol. 103, no. 4, pp. 667–685, 2016.
[13] M. T. Milojković, D. S. Antić, S. S. Nikolić, Z. D. Jovanović, S. Lj. Perić, "On a new class of quasi-orthogonal filters", *International Journal of Electronics*, vol. 100, no. 10, pp. 1361-1372, 2013.
[14] D. Antić, B. Danković, S. Nikolić, M. Milojković, Z. Jovanović, "Approximation based on orthogonal and almost orthogonal functions", *Journal of the Franklin Institute*, vol. 349, no. 1, pp. 323–336, 2012.
[15] S. S. Nikolić, Application of generalized classical filters in intelligent control systems, *PhD Dissertation*, University of Niš, Faculty of Electronic Engineering, 2014.
[16] M. Milojković, S. Nikolić, B. Danković, D. Antić, Z. Jovanović, "Modelling of dynamical systems based on almost orthogonal polynomials", *Mathematical and Computer Modelling of Dynamical Systems*, vol. 16, no. 2, pp. 133–144, 2010.