Reconfigurable Universal Fuzzy Flip-Flop: Applications to Neuro-Fuzzy Systems

Essam A. Koshak

Problem Report submitted to the
Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements
for the degree of

Master of Science
in
Electrical Engineering

Afzel Noore, Ph.D., Chair
James D. Mooney, Ph.D.
Hany H. Ammar, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2012

Keywords: Reconfigurable Fuzzy Flip-Flop; Function Approximation

ABSTRACT

Reconfigurable Universal Fuzzy Flip-Flop: Applications to Neuro-Fuzzy Systems

Essam A. Koshak

This research proposes a universal fuzzy flip-flop (UFFF) that can be reconfigured as a fuzzy SR, D, JK, or T flip-flop. This structure is implemented in two different neuro-fuzzy frameworks. First, when the fuzzy flip-flop was integrated in a multi-layer neural network, the resulting reconfigurable fuzzy-neural structure showed significant learning ability. For hardware implementation, the sigmoid activation function of neurons in the hidden layers of a multilayer neural network was replaced by the quasi-sigmoidal transfer characteristics of the universal fuzzy flip-flop in the reconfigurable fuzzy-neural structure. Also, the activation function of the output layer of the neural network was replicated with a saturating linear transfer function by choosing appropriate parameters for the proposed reconfigurable universal fuzzy flip-flop. Experimental results show that the reconfigurable fuzzy-neural structure can be effectively trained using either a large or sparse set of data points to closely approximate nonlinear input functions. In the second context, the universal fuzzy flip-flop was used for learning and predicting the responses of a fuzzy state machine. The proposed design uses a structure of the universal fuzzy flip-flop and a logic processor of fuzzy logic neurons (OR and AND neurons). The experimental results illustrate that the performance of the fuzzy state machine using the proposed universal fuzzy flip-flop is comparable to that of the traditional neural network.

ACKNOWLEDGMENTS

I would like to thank the people whose appreciation, help, and support have resulted into this problem report. Foremost, I would like to thank my advisor Dr. Afzel Noore for providing me with the opportunity to research under his supervision. He has been a source of constant support, inspiration, and guidance for me during the course of my degree. His feedback and constructive criticism has played a vital role in my intellectual development. I am also deeply indebted to my problem report committee members Dr. James D. Mooney, and Dr. Hany H. Ammar. This problem report would not have been possible without their valuable guidance and feedback. I also want to thank Dr. Rita Lovassy for her valuable feedback on the topic. I also want to express my gratitude for my friends who have helped me maintain a balance between study and life.

Finally, I want to thank my family for their unflinching support and love. I find myself short of words as I begin to thank my mother Fawziya, my siblings: Emad, Hanan, Eman and Nabeel. Furthermore, I would thank my wife Razan, and my children: Anas, Leen, Bassel, Ammar and Lamar. Whatever little that I am today and whatever I will ever achieve in my life will be because of my family's relentless efforts. I truly believe that all my accomplishments have been possible because of my family's prayers, their guidance, and wisdom. They have always provided me with emotional support and reality checks during the hardest moments of my life. My family has always been my pillar of support. It is my source of strength and has taught me how to be a strong, motivated, and driven individual.

Table of Contents

List of Figures

List of Tables

ABBREVIATIONS

UFFF	Universal Fuzzy Flip-Flop
MSE	Mean Squared Error

1 Introduction

Fuzzy logic applications have evolved in hardware at the chip level for many embedded systems. Fuzzy logic gates and fuzzy flip-flops proposed in research have been designed for different fuzzy logic systems such as applications of fuzzy flip-flops to circuit design using fuzzy FPGA [1]-[7]. A fuzzy flip-flop is an extension of a binary flip-flop where the values in the truth table is superset of binary logic and includes all real values from 0 to 1. The binary AND, OR and NOT operations are substituted by their fuzzy equivalents, known as t-norm, co-norm, and fuzzy negation respectively. There are many fuzzy logic applications applied to real-world modeling and control of robotics, power systems, and antilock braking system [12]. Using the fuzzy flip-flop as a basic building block, some researchers have also combined it with neural networks to design intelligent fuzzy state machines [8]-[9]; other researchers have extended it to approximate nonlinear functions [10]-[13]. The unknown function to be approximated may resemble a non-linear control system, where a neural network can be used to implement the controller [14]. The integration of fuzzy logic and neural networks produces intelligent machine-learning hardware with the ability to learn from its input data. In the literature, most applications have focused on individual fuzzy flip-flops.

In this research, the design of a reconfigurable universal fuzzy flip-flop (UFFF) is proposed [13]. It can be configured as a fuzzy SR flip-flop, fuzzy D flip-flop, fuzzy JK flip-flop or fuzzy T flip-flop. Such a building block is useful for rapid prototyping and designing complex fuzzy systems. The reconfigurable UFFF is integrated in two applications: neural network function approximation and fuzzy-neural finite state machine.

First, the UFFF is integrated with a neural network to form a fuzzy-neural structure that has the benefits of both a neural network and a fuzzy system. The ability of the resulting fuzzy-neural structure to learn any nonlinear input function and generate an output that closely approximates the input is studied. Next, the UFFF is integrated with fuzzy-neural state machine structure that can form an embedded agent in the framework of granular computing. The learning ability of both proposed reconfigurable fuzzy-neural structures is studied.

2 Proposed Universal Fuzzy Flip-Flop

Fuzzy flip-flops designed as individualized fuzzy logic device such as fuzzy JK flip-flop. In this report, a universal fuzzy flip-flop is proposed to form a fuzzy flip-flop building block that can be reconfigured as a specific fuzzy memory element based on specific application [13]. The universal fuzzy flip-flop can be reconfigured to (a) meet design specification, (b) alter the fuzzy memory computing structure in the event of component failure, or (c) incorporate online design adaptations based on new features. Figure 1 illustrates the proposed universal flip-flop that can be reconfigured by signals X and Y into four flip-flop modes: SR, JK, D, and T. Here, A and B are the flip-flop binary inputs and the characteristic equation of the proposed universal flip-flop is defined as:

$$Q+ = (A+Q)(Y+A+\bar{B})(X+\bar{Y}+A)(\bar{X}+\bar{A}+\bar{B}+\bar{Q})(\bar{X}+\bar{Y}+\bar{A}+\bar{Q}) \tag{1}$$

Control Inputs		Flip-Flop Inputs		Flip-Flop Mode
X	Y	A	B	
0	0	S	R	SR
0	1	D	-	D
1	0	J	K	JK
1	1	T	-	T

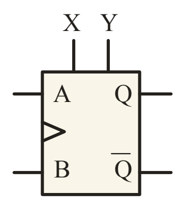

Figure 1 Universal Flip-Flop

The characteristic equation Q+ of the proposed universal fuzzy flip-flop can be realized by transforming the binary operators of the logical product (intersection), logical sum (union), and complement (negation) in Equation 1 to the corresponding fuzzy logic operators, described in fuzzy set theory. The logical product is transformed to fuzzy intersection operator referred to as t-norm ($\mathcal{T}$) or triangular norm operator. The logical sum is transformed to a fuzzy union operator referred to as the s-norm ($\mathcal{S}$) or t-conorm operator and the logical complement is replaced by the fuzzy negation operator ($\mathcal{N}$). The two operations t-norm and s-norm have a completely dual axiomatic skeleton and can be defined independently. The axiomatic skeleton for fuzzy set intersections and unions satisfy boundary conditions; they are commutative, monotonic, and associative. In the literature, there are several definitions of t-norms and s-norms. For example, there are algebraic norms, Dombi norms, and Yager norms that represent triangular norms of fuzzy system [11]-[12]. Appendix A summarizes selected t-norm and s-norm that are commonly used. In this research, the algebraic norms are first used and are defined as:

$a \,\mathcal{T}\, b = a.b$

$a \,\mathcal{S}\, b = a + b - ab$

$\mathcal{N}(a) = 1 - a$

Using the algebraic fuzzy norms, the binary characteristic equation of the reconfigurable universal flip-flop defined in Equation (1) can be transformed to the fuzzy characteristic equation given by [13]:

$$\begin{aligned} Q+ = \; & (1-X)(1-Y)[(A + Q - AQ)(AB - B + 1)] + \\ & (1-X)Y[A(A + Q - AQ)] + \\ & X(1-Y)[(1-ABQ)(A + Q - AQ)(AB - B + 1)] + \\ & XY[(AQ - 1)(AQ - Q - A)] \end{aligned} \tag{2}$$

Control inputs X and Y are binary while inputs A and B of the universal fuzzy flip-flop take any value from 0 to 1. Each input combination yields a large number of output sequences for the present state Q, and next state Q+. The following figures represent the fuzzy dynamic output characteristics between Q and Q+ for different values of A and B, and for different modes of the universal fuzzy flip-flop. Figure 2 shows the universal fuzzy flip-flop dynamic characteristics in JK-mode.

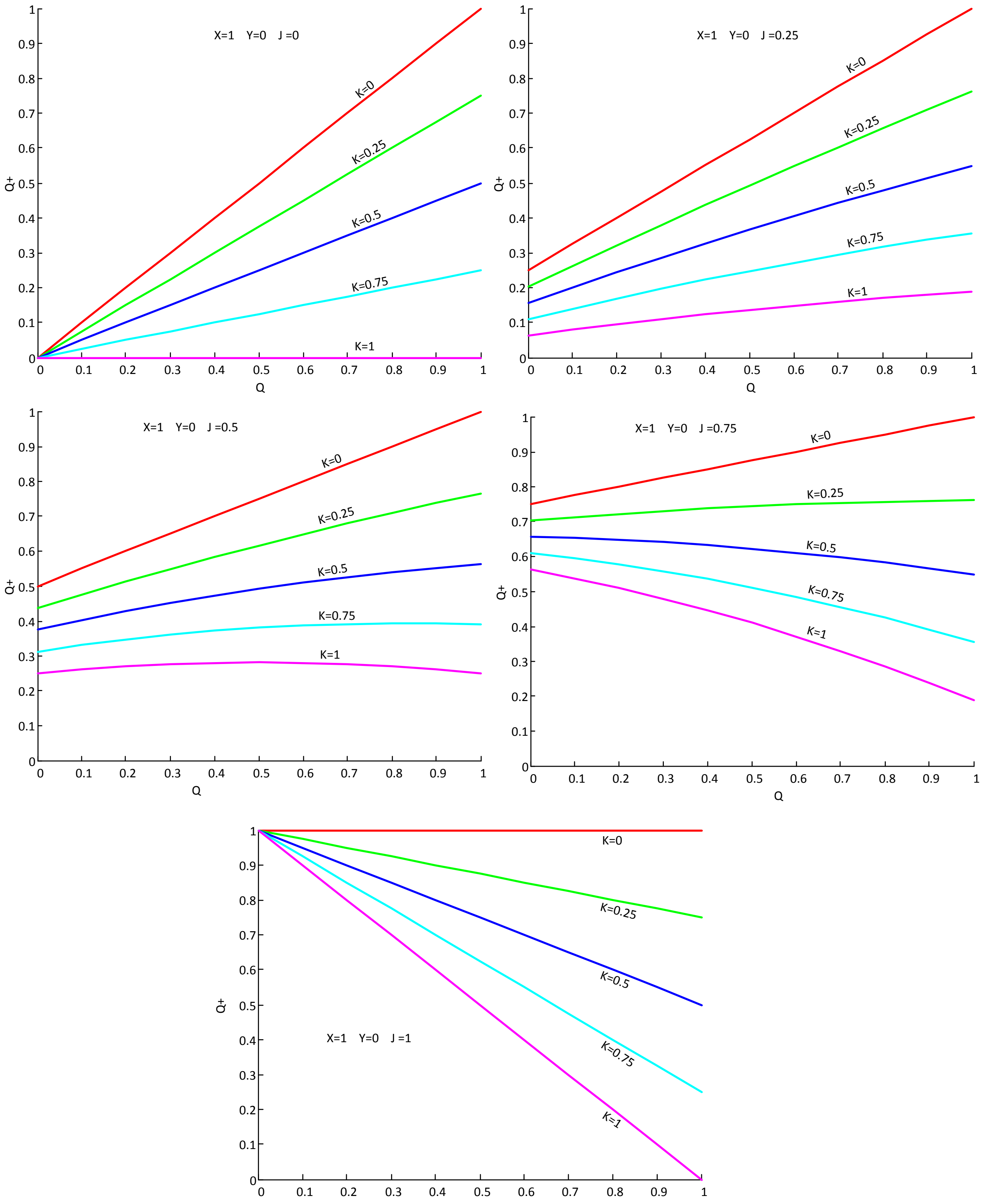

Figure 2 JK-mode of the universal fuzzy flip-flop characteristics using algebraic norms

As shown in the five graphs of Figure 2, where X=1, Y=0, and K=0, curves (lines) start from the same initial state Q=0 and converge to the same final state equal to 1 for any nonzero value of the J input. The speed of change as denoted by the slope depends on the value of the J input. The higher the value of J, the faster the change occurs in the next state value Q+ of the fuzzy flip-flop. Figure 2 also illustrates the pattern of changes for different configurations of the J and K inputs. These patterns are more complex and depend on which signal prevails [9]. Similarly, Figures 3, 4 and 5 show the universal fuzzy flip-flop dynamic characteristics in D-mode, T-mode, and SR mode respectively.

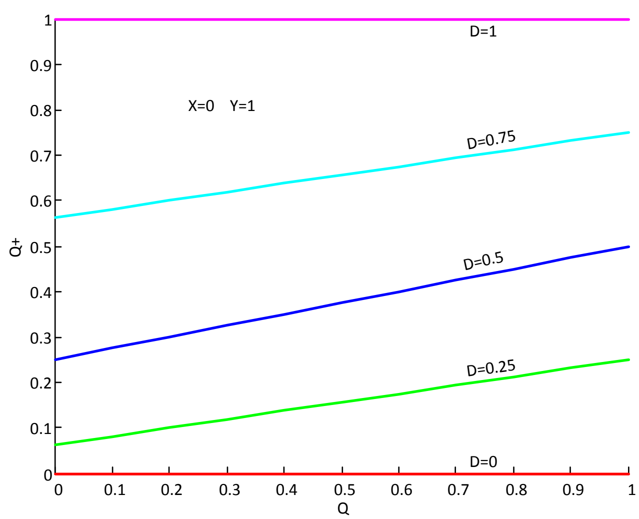

Figure 3 D-mode of the universal fuzzy flip-flop characteristics using algebraic norms

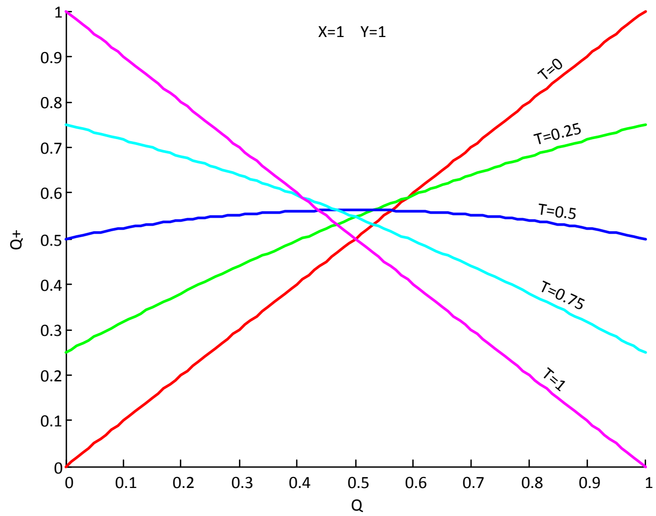

Figure 4 T-mode of the universal fuzzy flip-flop characteristics using algebraic norms

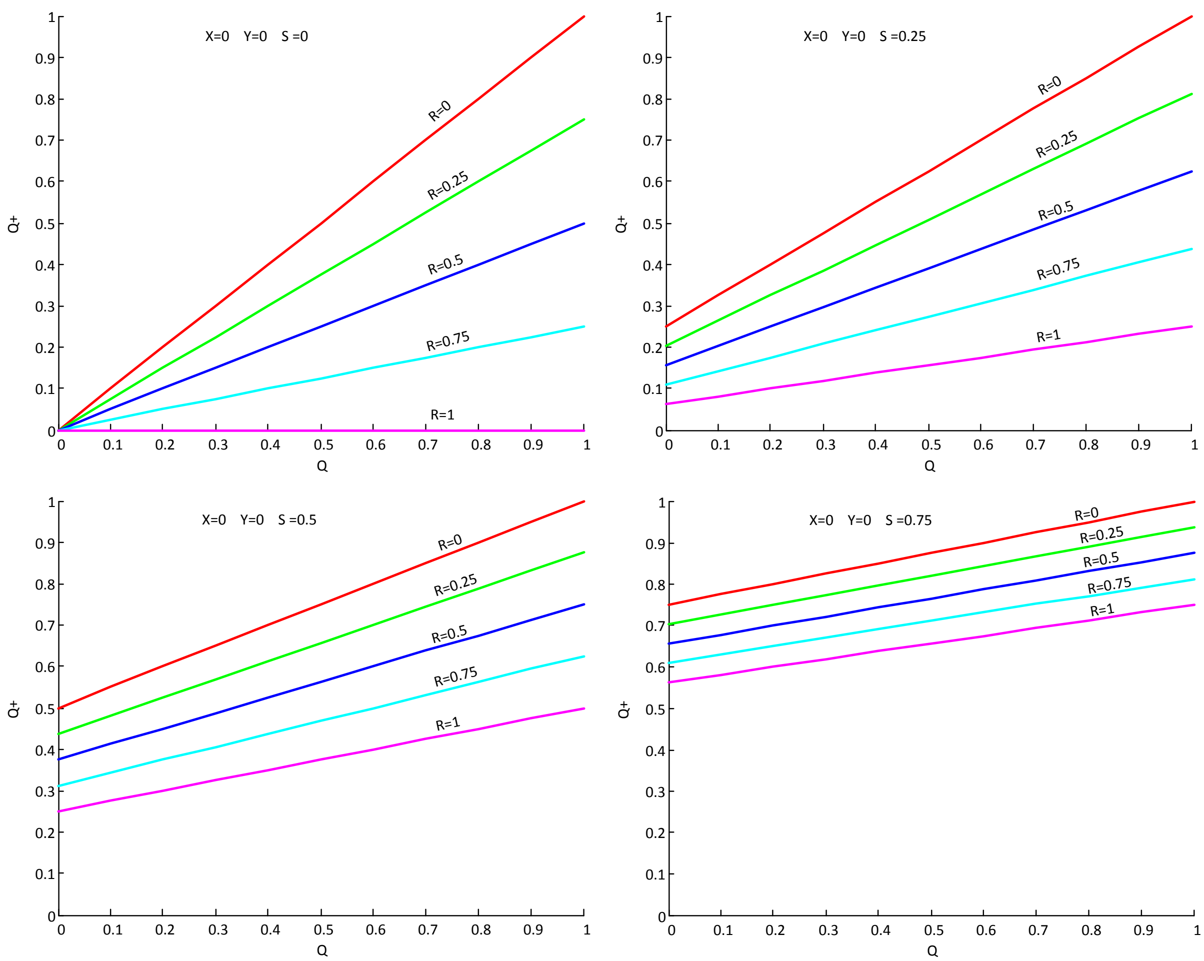

Figure 5 SR-mode of the universal fuzzy flip-flop characteristics using algebraic norms

Dombi norms are another example of triangular norms that have been discussed in fuzzy logic literature [11]-[12] and are defined as:

$$\mathrm{a\,T\,b} = \frac{1}{1+\left[\left(\frac{1}{a}-1\right)^{\alpha}+\left(\frac{1}{b}-1\right)^{\alpha}\right]^{\frac{1}{\alpha}}} \tag{3}$$

$$\mathrm{a\,S\,b} = \frac{1}{1+\left[\left(\frac{1}{a}-1\right)^{-\alpha}+\left(\frac{1}{b}-1\right)^{-\alpha}\right]^{\frac{-1}{\alpha}}} \tag{4}$$

Another example, Yager norms, are defined in literature [11]-[12] as:

$$\mathrm{a\,T\,b} = 1-\min\left[1,\left((1-a)^{w}+(1-b)^{w}\right)^{\frac{1}{w}}\right] \tag{5}$$

$$\mathrm{a\,S\,b} = \min\left[1,\left(a^{w}+b^{w}\right)^{\frac{1}{w}}\right] \tag{6}$$

where α and *w* are optimization parameters whose values are within the open interval $(0,\infty)$. If α=1 in (3) and (4), they are called Hamacher norms. If w=1 in (5) and (6), they are called Łukasiewicz norms. Using the Dombi and Yager fuzzy norms, the binary characteristic equation of the reconfigurable universal flip-flop defined in Equation (1) can be transformed to the fuzzy characteristic equation. Control inputs X and Y are binary values while the flip-flop inputs A and B of the universal fuzzy flip-flop take any value from 0 to 1. Each input combination yields a large number of output sequences for the present state Q, and next state Q+. Figures 12 to 19 in Appendix B and Appendix C represent the fuzzy dynamic output characteristics between the present state Q and the next state Q+ for different values of flip-flop inputs A and B, and for different modes of the universal fuzzy flip-flop using Dombi fuzzy norms and Yager fuzzy norms, respectively. These unlimited output response patterns make the universal fuzzy flip-flop a powerful building block for fuzzy logic system design.

3 Reconfigurable Fuzzy-Neural based Function Approximation

In the previous section the relationship between the current state Q and the next state Q+ of the UFFF was shown. However, if the input A is plotted versus the output Q+ of the UFFF in all four flip-flop modes in different norms, a quasi-sigmoid transfer characteristic is observed. For example, Figure 6 illustrates such curves for different values of Q using the UFFF in JK-mode. In fact, the different norms of various modes of the UFFF show the quasi-sigmoid transfer characteristic [12].

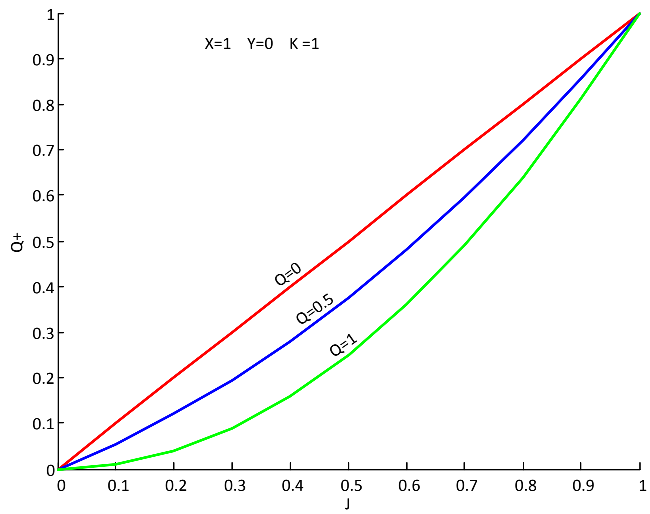

Figure 6 Input J versus output Q+ of the UFFF in JK-mode

Usually symmetrical and continuous differentiable activation functions are used in neural network neurons. Therefore, the UFFF modes can form a powerful, smoothed, and differentiable threshold unit in a multilayer perceptron. In this section, the proposed reconfigurable universal fuzzy flip-flop is used to learn any nonlinear input function to generate an approximate function at the output. It is well known that feedforward multilayer neural networks can uniformly approximate any nonlinear continuous function. Figure 7 shows a multilayer feedforward fuzzy-neural network with two hidden layers. In general, the output F is expressed as a function of the input ***x*** and interconnection weights ***w***.

$$F(x,w) = g_0(\sum_{h_2} w_{y_i h_2} g_2(\sum_{h_1} w_{h_2 h_1} g_1(\sum_{i} w_{h_{1i}} x_i + b_{h_1}) + b_{h_2}) + b_{y_i}) \tag{7}$$

where $i=1,2,\ldots,n;$ $w_{h_{1i}}$ is the synaptic weight between input x_i and a neuron in the first hidden layer; $w_{h_2 h_1}$ is the synaptic weight between a neuron h_1 in the first hidden layer and a neuron h_2 in the second hidden layer; and $w_{h_{yi} h_2}$ is the synaptic weight between a neuron h_2 in the second hidden layer and a neuron y_i in the output layer. Here, b_{h_1}, b_{h_2} and b_{y_i} are the bias vectors of the first hidden layer, the second hidden layer and the output layer, respectively.

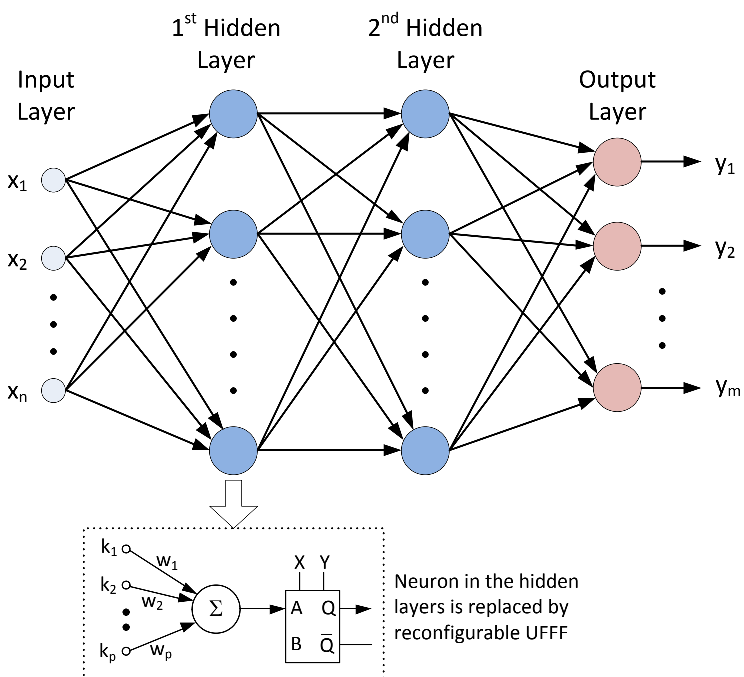

Figure 7 Fuzzy-neural network structure

In the proposed reconfigurable fuzzy-neural structure, the sigmoid activation function $g_1(.)$ and $g_2(.)$ are defined by the quasi-sigmoid transfer characteristics of the universal fuzzy flip-flop as delineated in the fuzzy characteristic Equation (2). The output activation function g_0 in the feedforward neural network is a linear transfer function. The UFFF was also incorporated in the neurons of the output layer to emulate a linear transfer function. Consequently, the neurons in the reconfigurable fuzzy-neural structure use the same building block of neurons with UFFF based transfer function for hardware implementation. The linear transfer function can be implemented by reconfiguring the UFFF in JK-mode where X=1, Y=0, K=1, and Q=0 and as seen in Figure 6. The proposed reconfigurable fuzzy-neural structure demonstrates the learning ability to perform nonlinear input-output mapping for function approximation.

3.1 Experiments on Fuzzy-Neural based Function Approximation

Two functions $y_1(x)$ and $y_2(x_1, x_2)$ are considered to examine the performance of the proposed reconfigurable fuzzy-neural structure when approximating complex nonlinear input functions [13]. The first complex function is represented by $y_1(x) = [sin(4x)cos(20x)/2.5]+ 0.5$. Using this function, a large number of data points (1000 data points) were generated. A second nonlinear complex function is represented by $y_2(x_1, x_2) = (1+ x_1^{-2} + x_2^{-1.5})^2$ where $1\leq (x_1, x_2) \leq 5$. Our experiment chose a sparse dataset (50 data points) identical to the data points used by other researchers for similar applications, providing a baseline for comparing the efficacy of the proposed reconfigurable fuzzy-neural network with previously published results [15]-[16]. In the case of the first nonlinear complex function $y_1(x)$, the performance of the proposed reconfigurable fuzzy-neural network was compared with the feedforward neural network using the hyperbolic tangent sigmoid activation function (*tansig*) which yielded the best performance. These two datasets show learning ability when the data points are large and when the data points are sparse. Furthermore, since the proposed UFFF can be reconfigured to a fuzzy SR flip-flop, fuzzy D flip-flop, fuzzy JK flip-flop or fuzzy T flip-flop, the performance of function approximation in each of these modes was further studied.

The data points and the corresponding values of $y_1(x)$ were sampled and evaluated when approximating the function $y_1(x)$ using the multilayer neural network. Pairs of data points are used to train the multilayer neural network using the Levenberg-Marquardt algorithm with a maximum of 120 epochs. In our experiment, each hidden layer had 20 neurons and each neuron in the first and second hidden layers had a *tansig* activation function. The neurons in the output layer had a linear transfer function. The initial weights were randomly assigned small values.

The approximated function generated by the multilayer neural network and the proposed reconfigurable fuzzy-neural structure for each mode of the UFFF are shown in Figure 8. The graphs show that the function approximation of the fuzzy-neural structure for each mode closely matched the performance of the feedforward neural network. The mean squared error (MSE) was calculated for all five cases representing the fuzzy-neural structure, based on the four modes of the reconfigurable universal flip-flop and the multilayer neural network with *tansig* activation function in the hidden layers. These results are shown in Figure 9. The average MSE values were obtained after running the experiment 150 times. The best approximation of the function $y_1(x)$ was obtained when the UFFF was configured in the fuzzy T flip-flop mode. The results are comparable to those obtained with the multilayer neural network using the *tansig* activation function.

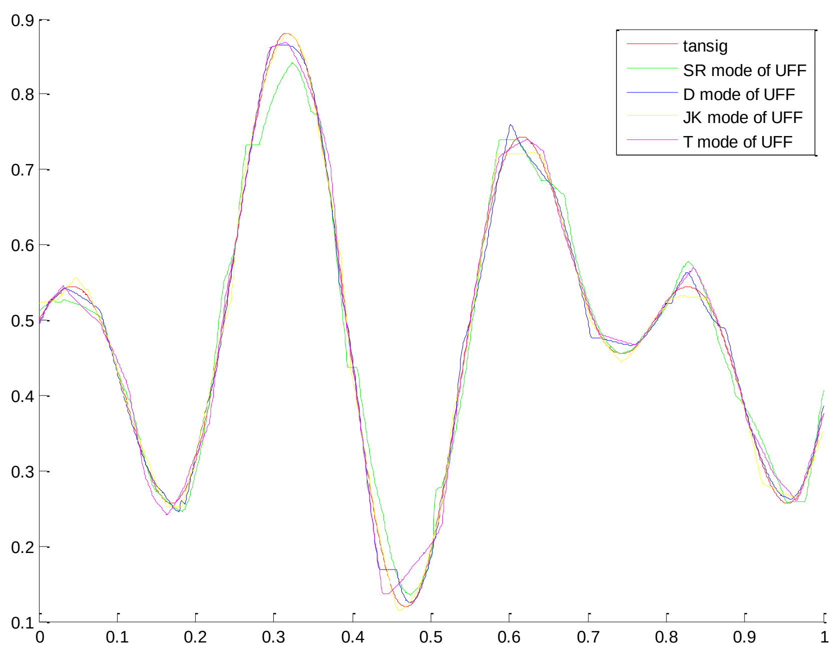

Figure 8 Approximated function using the reconfigurable universal fuzzy-neural structure

In the first hidden layer, a subset of neurons extracts the local features of the nonlinear function by partitioning the input space into regions. The remaining neurons in the first layer learn the characteristics of these individual regions. In the second hidden layer, each neuron learns the global features of each individual region in the first layer and is combined to generate the approximated function at the output. Higher accuracy is obtained by increasing the number of neurons in the hidden layers.

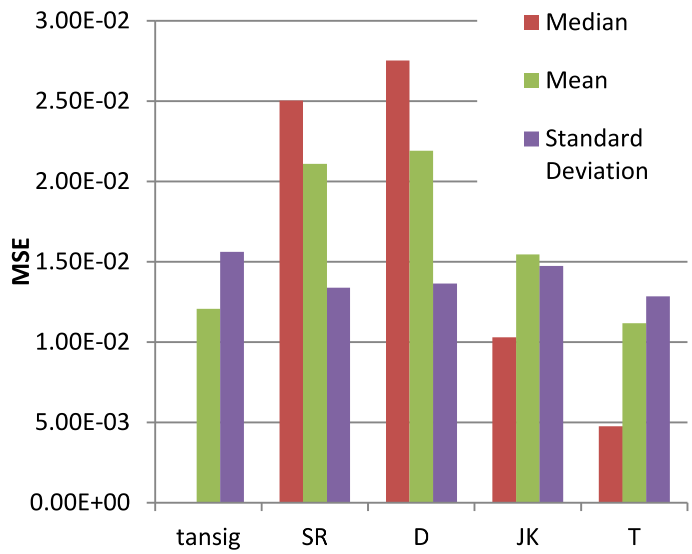

Figure 9 Performance of the reconfigurable universal fuzzy-neural structure

Next the function $y_2(x_1, x_2)$ was approximated with a sparse dataset of only 50 data points, as used in similar experiments by other researchers [15]-[16]. The UFFF in the fuzzy-neural structure was reconfigured in the fuzzy T flip-flop mode to obtain the best performance. The tests were performed 30,000 times to be consistent with the experimental design of previously published work. Table 1 compares the results of MSE obtained by the proposed fuzzy-neural structure when the universal flip-flop is configured in the fuzzy T flip-flop mode with the results obtained by Scherer [15] and Sugeno et al. [16]. The results show that even with the sparse dataset, the reconfigurable neuro-fuzzy structure has good learning ability and performed better as a function approximator compared with the recently proposed relational neuro-fuzzy system [15] and the results reported in [16].

Table 1 Performance of sparse data experiment

Method	Testing MSE
Qualitative Modeling by Sugeno et al. [16]	0.2810
Neuro-Fuzzy Relational Systems by Scherer [15]	0.2730
Proposed T-mode of UFFF based Neural Network [13]	0.2672

4 UFFF Based Fuzzy State Machine

A fuzzy Moore state machine approach was proposed using fuzzy JK flip-flop [8]-[9]. The proposed structure consists of combinational processing module (logic processor) before the JK flip-flop. The structure was considered as an embedded agent in the framework of granular computing. The logic processor optimizes the inputs of the flip-flop by adjusting the network connections so that a specific performance index is minimized. In the following experiment, the proposed UFFF was incorporated into a such fuzzy-neural state machine structure as shown in Figure 10.

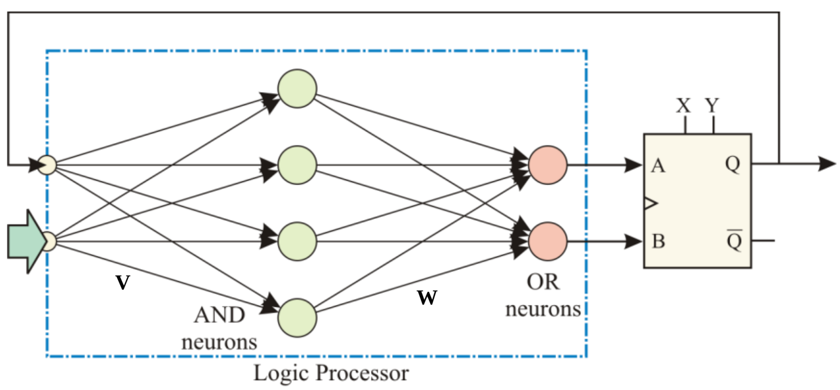

Figure 10 Reconfigurable universal fuzzy-neural state machine structure in learning mode

4.1 Experiment on UFFF Based Fuzzy State Machine

In this experiment, the reconfigurable universal fuzzy-neural state machine structure based on the algorithm discussed in [8] and [9] was applied to learn the input vector [0 0; 0 1; 1 1; 1 0; 0 0; 1 0; 1 1; 0 1] and the corresponding outputs = [0; 0; 0; 0; 1; 1; 1; 1]. Here, N is the number of elements to be trained; it is 8 in this example. There are two nodes of hidden layer and two nodes of the output layer in the logic processor. The performance index T to be minimized is a standard sum of squared errors: $T(W, V) = \Sigma[\text{target}(k+1) - Q^{+}(k)]^2$; where W and V are the arrays of the connections (weights) of the OR and AND neurons, respectively, and k = 1, 2, …, N. Figure 11 shows graphs of the performance index versus successive learning epochs for different modes of the UFFF. Table 2 lists the mean square errors (MSE) for this experiment using the proposed reconfigurable universal fuzzy-neural state machine in different modes. The results show significant performance and learning ability for the system for different UFFF modes. This experiment demonstrates that the fuzzy state machine can be implemented using the UFFF. Moreover, this UFFF based fuzzy state machine can form a reconfigurable agent that has been learned rather than being designed and can be embedded in the framework of granular computing. Based on the experimental results, the performance of the UFFF configured in the fuzzy T flip-flop mode performed best.

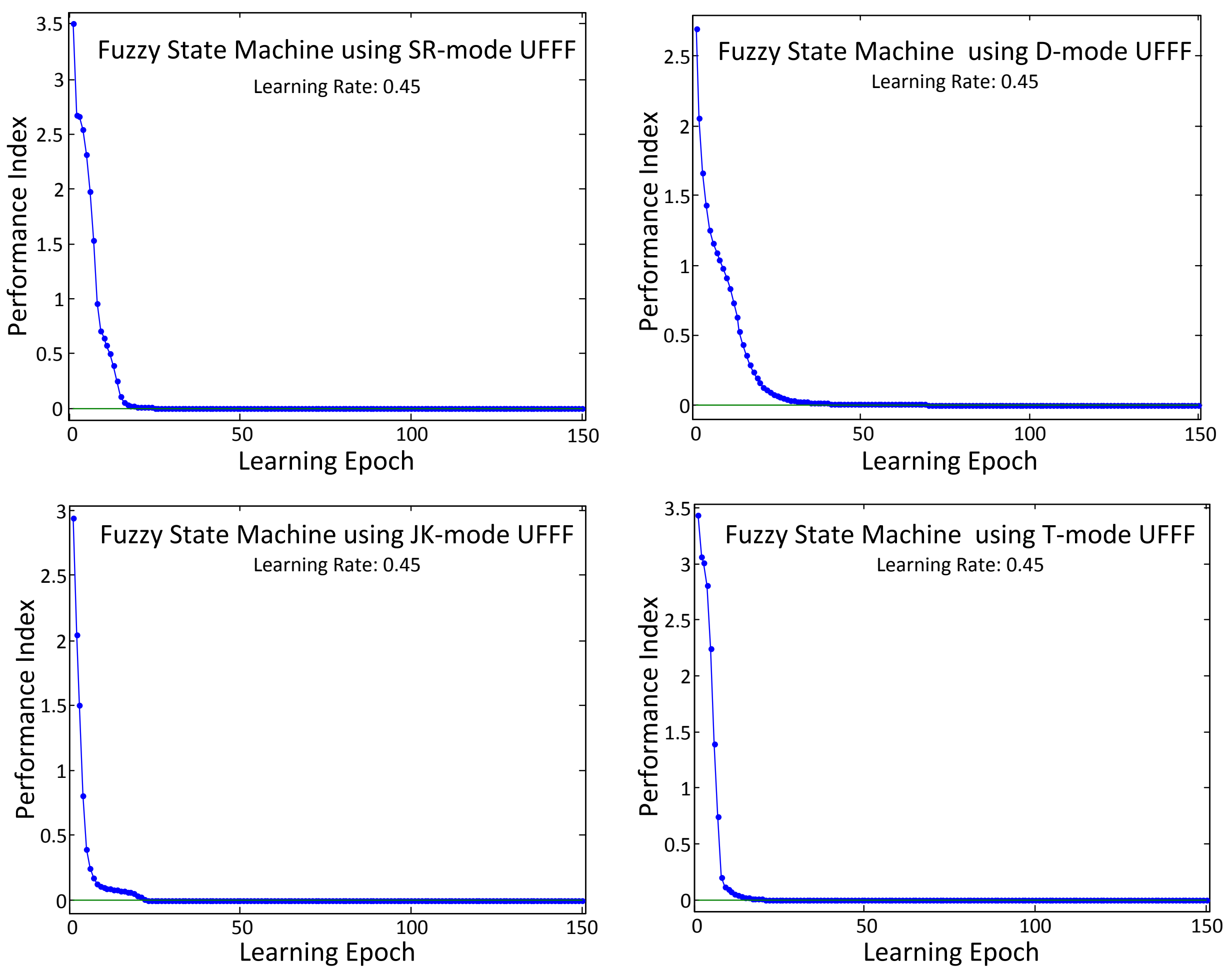

Figure 11 Performance index versus successive learning epochs

Table 2 MSE for reconfigurable universal fuzzy-neural state machine

UFFF mode	Minimum	Median	Mean	STD
SR	7.2040e-010	6.2095e-008	6.5833e-003	4.3750e-002
D	1.3968e-012	6.6698e-011	2.9425e-002	1.6824e-001
JK	0	1.0544e-007	2.0931e-003	8.5479e-003
T	0	1.2328e-013	1.6874e-007	1.6805e-006

5 Conclusion

The design of a reconfigurable universal fuzzy flip-flop is proposed to improve existing designs that have primarily focused on individual fuzzy flip-flops such as fuzzy JK, D or T flip-flops. The functionality of the reconfigurable UFFF is extended to produce a myriad of responses to optimize the performance for specific applications. It can be reconfigured to operate as a fuzzy SR flip-flop, fuzzy JK flip-flop, fuzzy D flip-flop or fuzzy T flip-flop. In this research, the reconfigurable UFFF was integrated in two applications. First, the UFFF was integrated in the hidden layer and the output layer of a multilayer neural network suitable for embedded hardware implementation. The sigmoid activation function of the neurons in the hidden layers was replaced by the quasi-sigmoid transfer characteristics of the UFFF. The activation function of the output layer was replicated by reconfiguring the UFFF to generate a saturating linear transfer function. The learning ability of the resulting reconfigurable fuzzy-neural structure was demonstrated by a non-linear function approximation application. The UFFF was configured in each of the four modes by selecting different values for the fuzzy inputs. The learning ability of the proposed reconfigurable fuzzy-neural structure was studied when (a) the available data points were large and (b) when the data points were sparse. The mean square error obtained by using two nonlinear complex functions shows that the fuzzy-neural structure has very good learning ability, as demonstrated in the function approximator application. In another application, the UFFF was integrated in a fuzzy-neural state machine structure and reconfigured in different operating modes. Experimental results demonstrate the ability of the embedded UFFF to learn and predict the next state and output of the Moore state machine with high accuracy.

Appendix A: List of selected triangular norms and conorms

Table 3 Representative t-norms and s-norms [12]

Norm	t-norm (a T b)	s-norm (a S b)
Standard (min-max)	$\min(a,b)$	$\max(a,b)$
Algebraic	ab	$a+b-ab$
Drastic	a when $b=1$, b when $a=1$, 0 otherwise	a when $b=0$, b when $a=0$, 1 otherwise
Łukasiewicz	$\max(0, a+b-1)$	$\min(1, a+b)$
Yager	$1-\min\left[1,\left((1-a)^w+(1-b)^w\right)^{\frac{1}{w}}\right]$	$\min\left[1,\left(a^w+b^w\right)^{\frac{1}{w}}\right]$
Dombi	$\dfrac{1}{1+\left[\left(\frac{1}{a}-1\right)^{\alpha}+\left(\frac{1}{b}-1\right)^{\alpha}\right]^{\frac{1}{\alpha}}}$	$\dfrac{1}{1+\left[\left(\frac{1}{a}-1\right)^{-\alpha}+\left(\frac{1}{b}-1\right)^{-\alpha}\right]^{\frac{-1}{\alpha}}}$
Hamacher	$\dfrac{ab}{v+(1-v)(a+b-ab)}$	$\dfrac{a+b-(2-v)ab}{1-(1-v)ab}$
Frank	$\log_s\left[1+\dfrac{(s^a-1)(s^b-1)}{s-1}\right]$	$1-\log_s\left[1+\dfrac{(s^{1-a}-1)(s^{1-b}-1)}{s-1}\right]$
Dubois-Prade	$\dfrac{ab}{\max(a,b,d)}$	$\dfrac{a+b-ab-\min(a,b,1-d)}{\max(1-a,1-b,d)}$
Schweizer-Sklar	$\max\left(0,a^p+b^p-1\right)^{\frac{1}{p}}$	$1-\max\left(0,(1-a)^p+(1-b)^p-1\right)^{\frac{1}{p}}$

Appendix B: The UFFF characteristics using Dombi norms

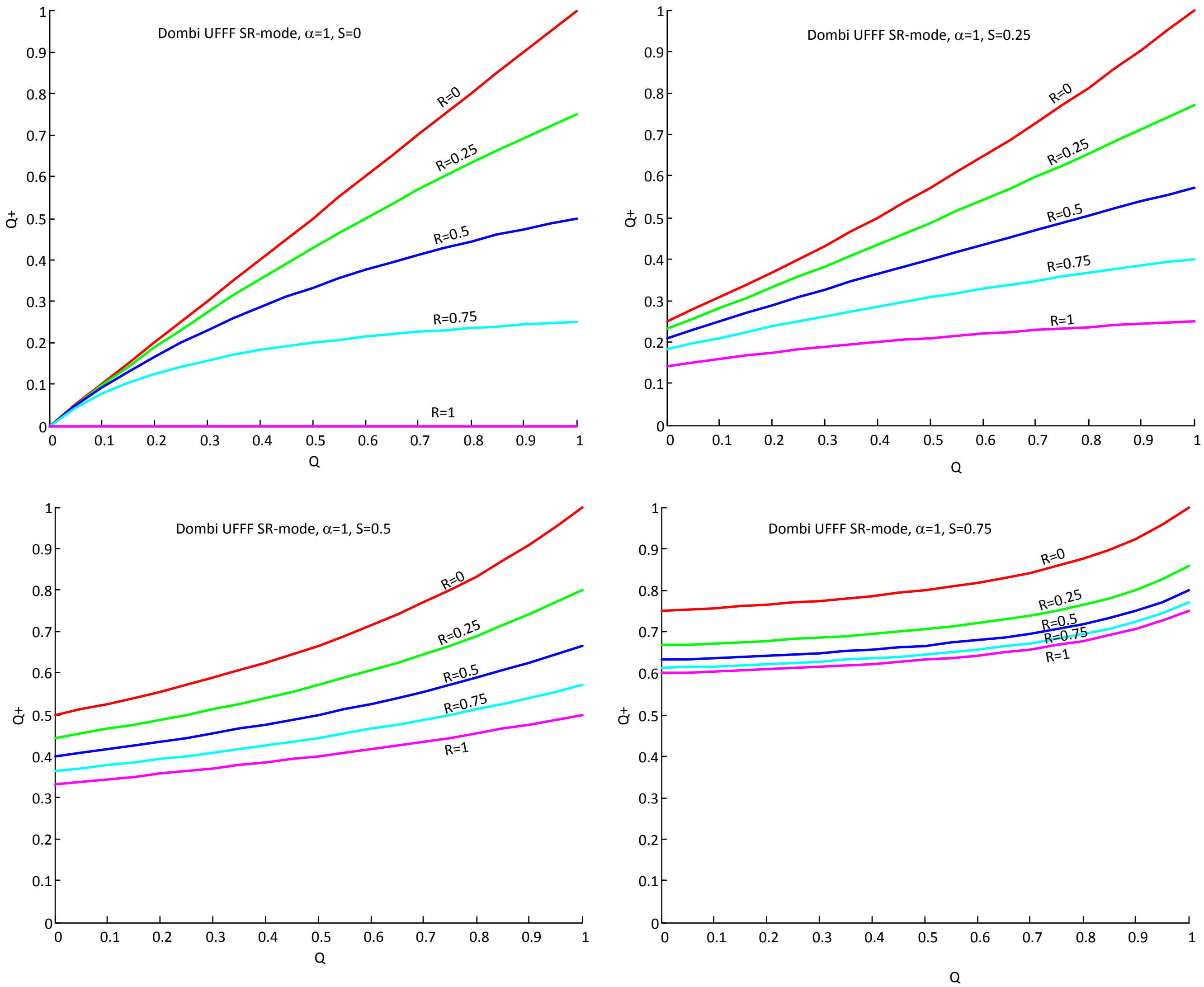

Figure 12 SR-mode of the universal fuzzy flip-flop characteristics using Dombi norms

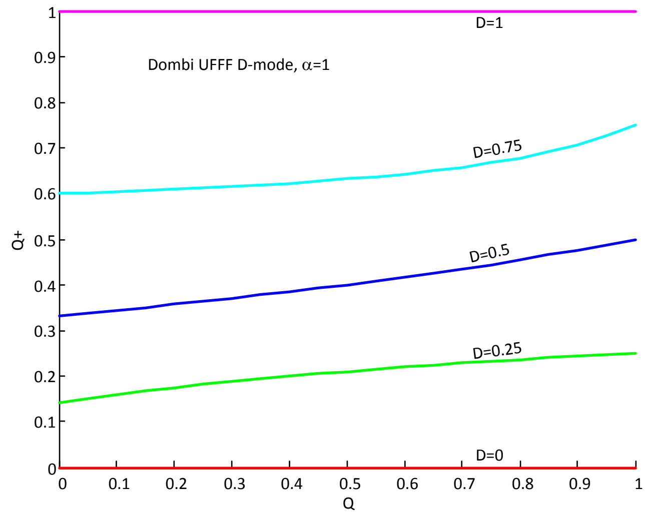

Figure 13 D-mode of the universal fuzzy flip-flop characteristics using Dombi norms

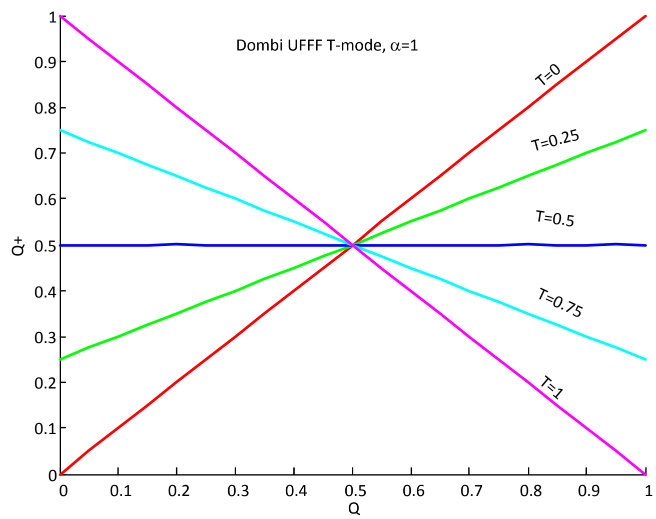

Figure 14 T-mode of the universal fuzzy flip-flop characteristics using Dombi norms

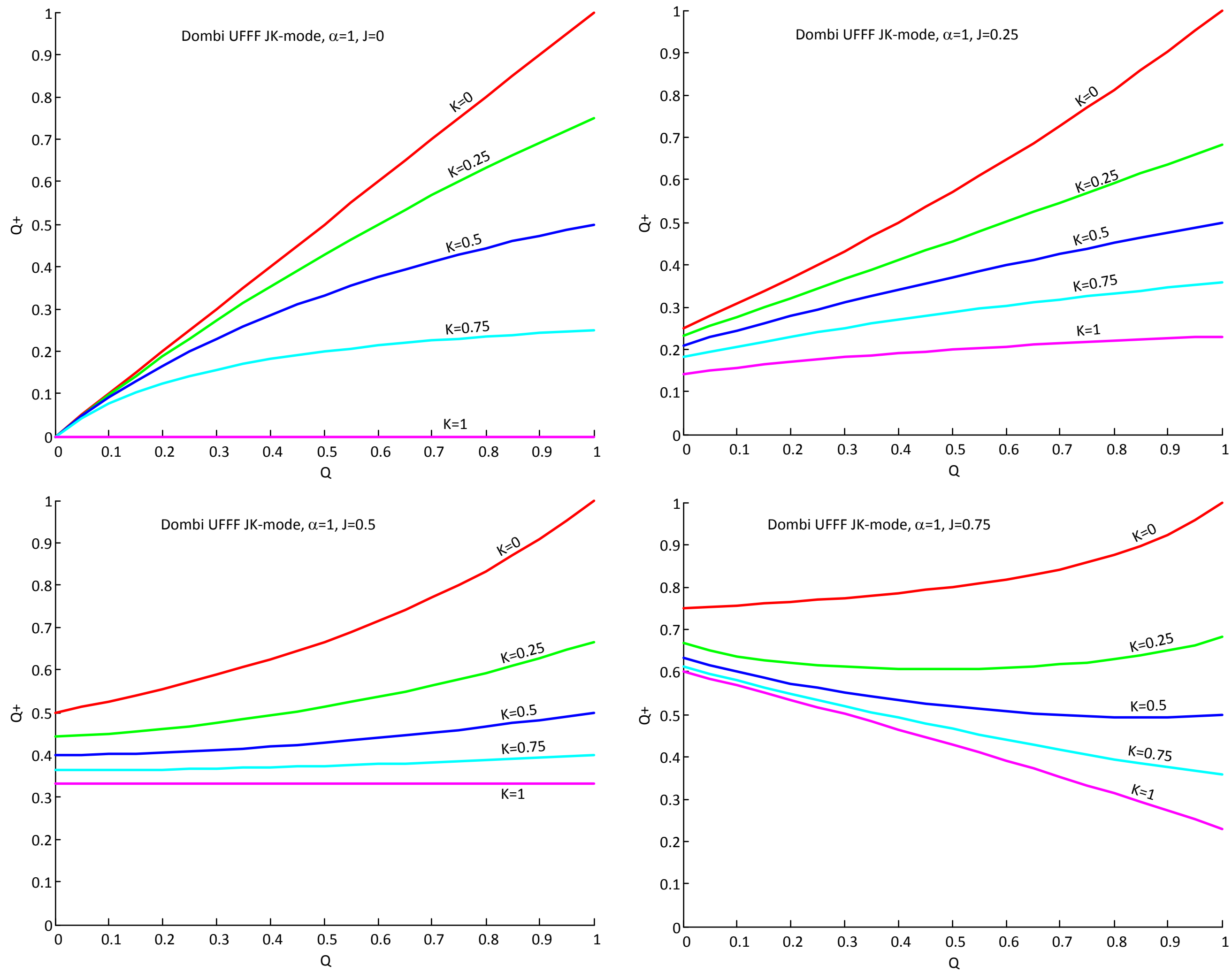

Figure 15 JK-mode of the universal fuzzy flip-flop characteristics using Dombi norms

Appendix C: The UFFF characteristics using Yager norms

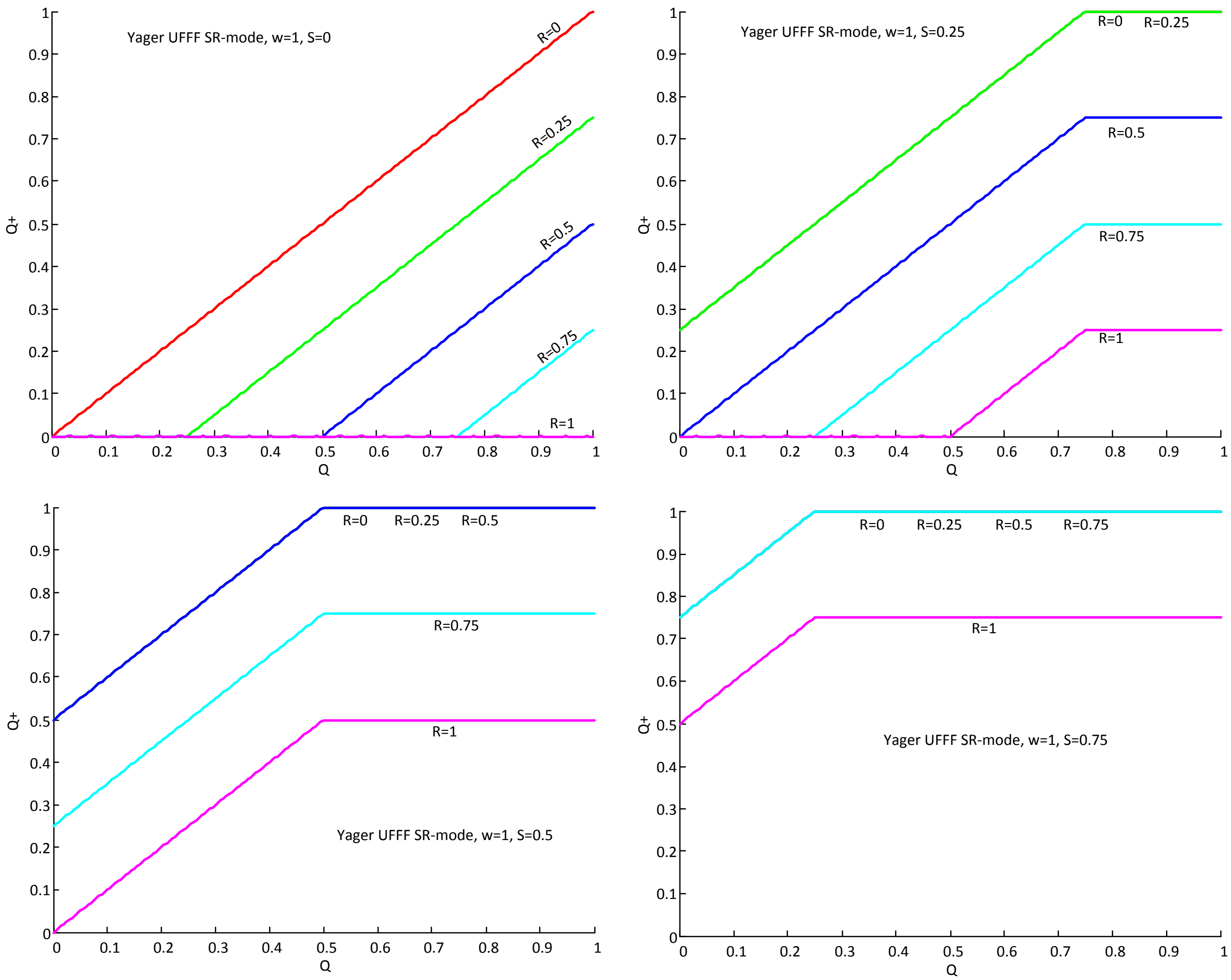

Figure 16 SR-mode of the universal fuzzy flip-flop characteristics using Yager norms

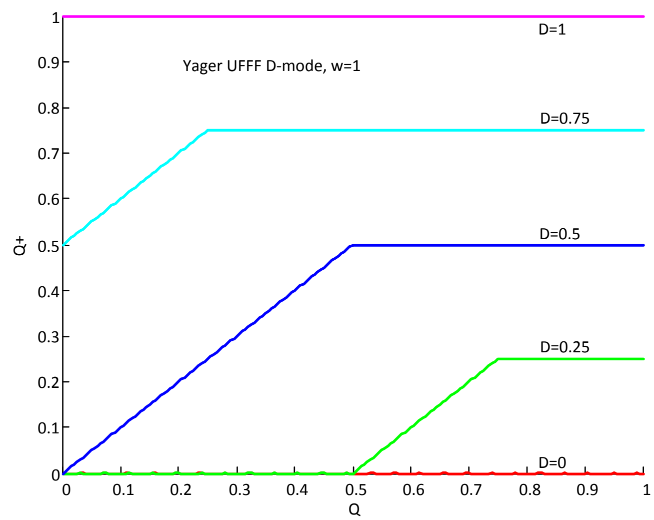

Figure 17 D-mode of the universal fuzzy flip-flop characteristics using Yager norms

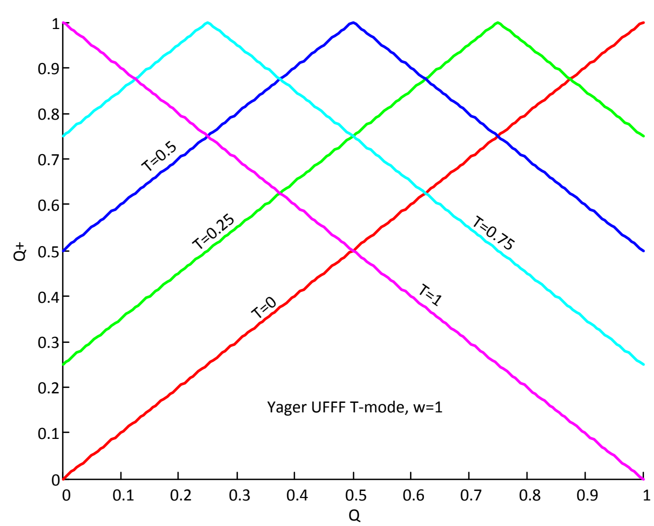

Figure 18 T-mode of the universal fuzzy flip-flop characteristics using Yager norms

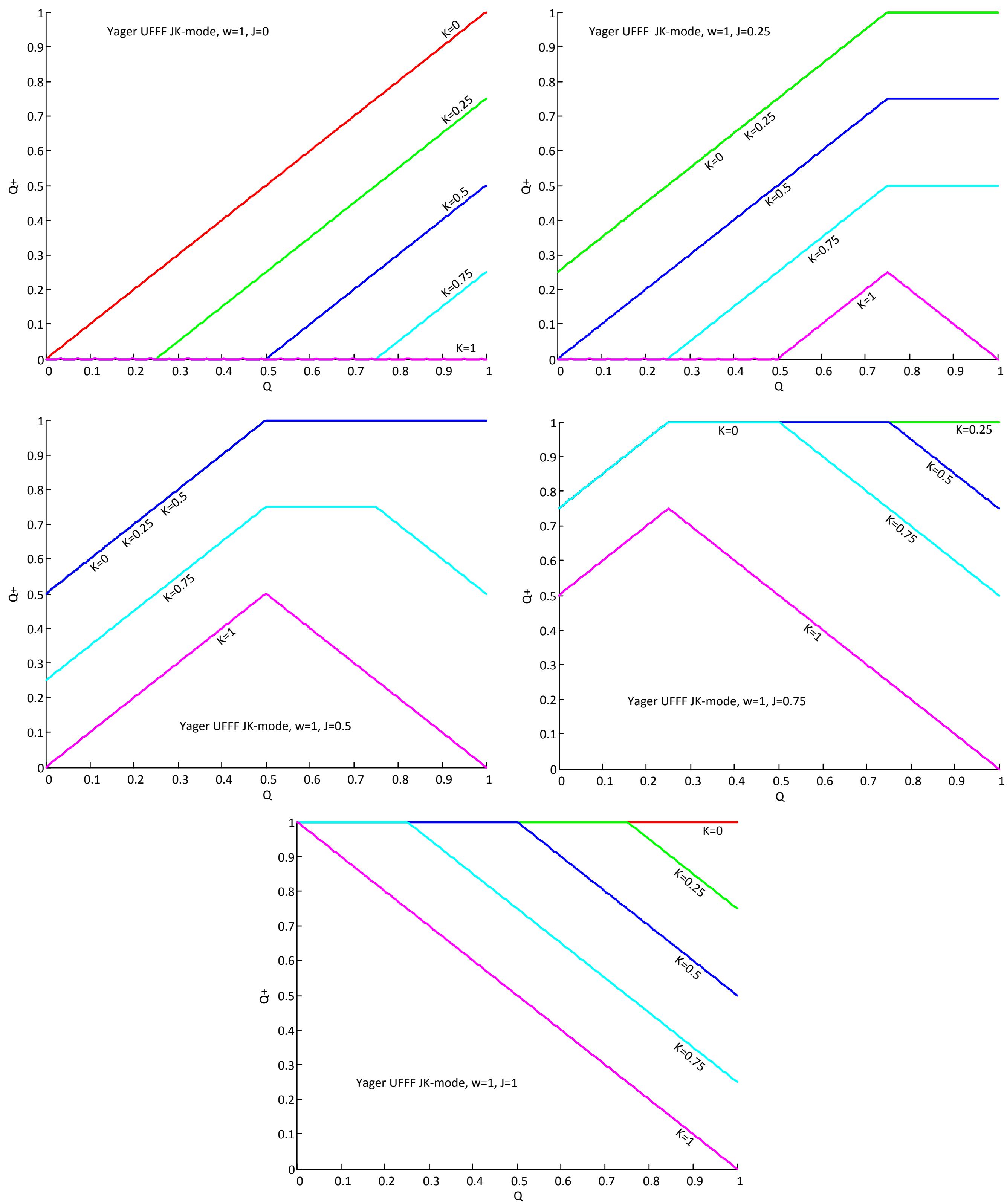

Figure 19 JK-mode of the universal fuzzy flip-flop characteristics using Yager norms

References

[1] S. Yoshida, Y. Takama and K. Hirota. Fuzzy flip-flops and memory element and their applications to circuit design using fuzzy FPGA. *Journal of Advanced Computational Intelligence and Intelligent Informatics,* 4(5), pp. 380-386, 2000.

[2] K. Hirota and K. Ozawa. The concept of fuzzy flip-flop. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5), pp. 980-997, 1989.

[3] K. Hirota and W. Pedrycz. Design of fuzzy systems with fuzzy flip-flops. *IEEE Transactions on Systems, Man and Cybernetics*, 25(1), pp. 169-176, 1995.

[4] J. Diamond, W. Pedrycz and D. McLeod. Fuzzy JK flip-flops as computational structures: Design and implementation. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 41(3), pp. 215-226, 1994.

[5] K. Ozawa, K. Hirota, L. T. Koczy and K. Omori. Algebraic fuzzy flip-flop circuits. *Fuzzy Sets and Systems*, 39(2), pp. 215-226, 1991.

[6] K. Ozawa, K. Hirota, L. T. Koczy, W. Pedrycz and N. Ikoma. Summary of fuzzy flip-flop. *International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and the Second International Fuzzy Engineering Symposium*, 3, pp. 1641-1648, 1995.

[7] J. Virant, N. Zimic and M. Mraz. T-type fuzzy memory cells. *Fuzzy Sets and Systems,* 102(2), pp. 175-183, 1999.

[8] W. Pedrycz and A. Gacek, Learning of fuzzy automata, *International Journal of Computational Intelligence and Applications,* 1, pp. 19-33, 2001.

[9] A. Bargiela and W. Pedrycz, "Intelligent agents and granular worlds," in *Granular Computing: An Introduction*, 1st ed., The Netherlands: Kluwer Academic, pp. 323-348, 2003.

[10] P. Liu and H. Li. Approximation analysis of feedforward regular fuzzy neural network with two hidden layers. *Fuzzy Sets and Systems,* 150(2), pp. 373-396. 2005.

[11] R. Lovassy, L. T. Koczy and L. Gal. Function approximation capability of a novel fuzzy flip-flop based neural network. *International Joint Conference on Neural Networks*, pp.1900-1907, 2009.

[12] R. Lovassy, Multilayer Perceptrons Based on Fuzzy Flip-Flops, Ph.D. dissertation, Széchenyi István University, 2010.

[13] E. Koshak, A. Noore and R. Lovassy. Intelligent reconfigurable universal fuzzy flip-flop. *IEICE Electronics Express*, 7(15), pp. 1119-1124, 2010.

[14] W. Miller III, R. Sutton, and P. Werbos. Neural Networks for Control. 5th print, *MIT Press*, 1996.

[15] R. Scherer. Neuro-fuzzy relational systems for nonlinear approximation and prediction. *Nonlinear Analysis: Theory, Methods & Applications*, 71(12), pp. e1420-e1425, 2009.

[16] M. Sugeno and T. Yasukawa. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1(1), pp. 7-31, 1993.