International Journal of Innovative Computing, Information and Control
Volume **18**, Number **1**, February **2022**

ICIC International ©2022 ISSN 1349-4198
pp. **29–39**

COMPARISON STUDIES OF ITERATIVE ALGORITHMS FOR CENTER-OF-SETS TYPE-REDUCTION OF INTERVAL TYPE-2 FUZZY LOGIC SYSTEMS

Yang Chen and Jiaxiu Yang

College of Science
Liaoning University of Technology
No. 169, Shiying Street, Guta District, Jinzhou 121001, P. R. China
lxychenyang@lnut.edu.cn; y1290042383@163.com

Received July 2021; revised November 2021

Abstract. *Type-reduction is the kernel block of type-2 fuzzy logic systems. Studying the center-of-sets type-reduction is helpful for real applications of systems. This paper introduces the fuzzy reasoning, center-of-sets type-reduction and defuzzification of interval type-2 fuzzy logic systems. Based on the Karnik-Mendel algorithms, enhanced Karnik-Mendel algorithms and enhanced iterative algorithms with stopping condition for computing the centroids of interval type-2 fuzzy sets, this paper extends them for investigating the center-of-sets type-reduction of Mamdani type interval type-2 fuzzy logic systems. Several simulation experiments analyze the computational costs of three types of iterative algorithms for obtaining the output of interval type-2 fuzzy logic systems, which provides the potential reference values for designing and applying interval type-2 fuzzy logic systems.*

Keywords: Interval type-2 fuzzy logic systems, Center-of-sets type-reduction, Three types of iterative algorithms, Computational cost, Simulation

1. **Introduction.** As we all know, both the type-1 (T1) and interval type-2 (IT2) fuzzy sets (FSs) can be considered as the parametric models. The footprint of uncertainty (FOU) of an IT2 FS makes it own more design degrees of freedom compared with a T1 FS. Because the membership grades of IT2 FSs are themselves T1 FSs, IT2 FLSs based on IT2 FSs [1-3] increase the capability to cope with uncertainties. As the computational complexity of general T2 FLSs (GT2 FLSs [4]) is too high, this makes them difficult to apply in real applications. However, the computational relative simple IT2 FLSs [5-8] are the most widely used expert systems in current years.

The only structure difference between IT2 FLSs (see Figure 1) and T1 FLSs is that the block defuzzification for the latter is composed of both the type-reduction and defuzzification in the former. Furthermore, the major differences between two types fuzzy logic systems can be concluded as: 1) at least one IT2 FS can be adopted in antecedents or consequents of IT2 FLSs; 2) the inference operations in the former are much more complicated than in the latter; 3) the necessary block of type-reduction in the former transforms the IT2 FS to the T1 FS. The above three points make the computations in IT2 FLSs more challenged.

Currently, the centroid type-reduction is most popular theoretical research approach for T2 FLSs. Karnik and Mendel first developed the famous Karnik-Mendel algorithms [9,10] for computing the centroids of IT2 FSs or completing the centroid type-reduction

DOI: 10.24507/ijicic.18.01.29

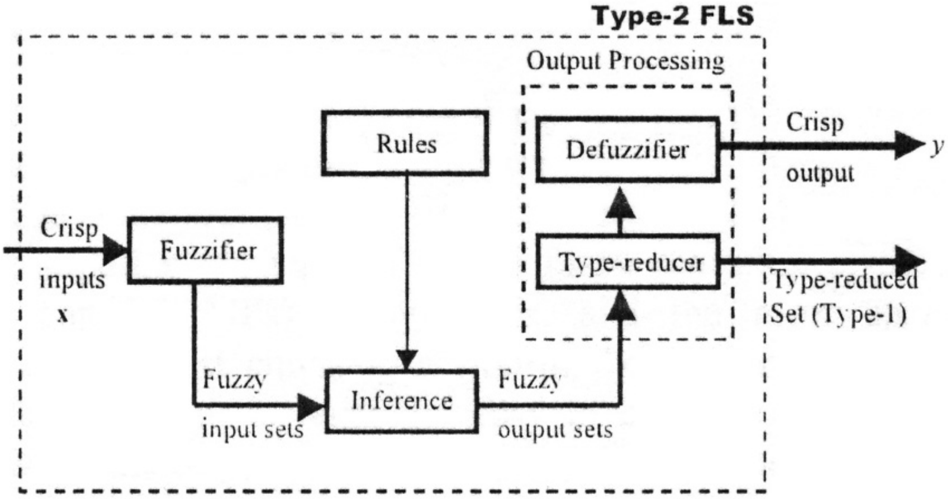


FIGURE 1. The blocks of an IT2 FLS

of IT2 FLSs. Although the KM algorithms are monotonic and convergent with super-exponentially speed [11], it usually needs two to six iterations for this type of computationally intensive algorithm. In order to reduce the computation time, Wu and Mendel proposed the enhanced KM (EKM) algorithms [12] for computing the centroids. Extensive simulation experiments show that the EKM algorithms can save about two iterations on average. Another type of iterative algorithm was also put forward, which was called as the enhanced iterative algorithms with stopping condition (EIASC [13]). It was proved in [13] that, when the number of sampling primary variables of IT2 FSs was less than 100, this type of iterative algorithm was significantly superior than the KM algorithm. Moreover, Mendel stated that these three types of iterative algorithms [4] are well to perform the centroid TR for T2 FLSs.

The above three types of iterative algorithms for the centroid type-reduction belong to the pure theoretical studies, actually, the center-of-sets (COS) type-reduction [14-17] is a more favorable method for applying IT2 FLSs. As we all know, the inference is the key block for IT2 FLSs. This paper proposes the fuzzy reasoning of Mamdani type IT2 FLSs, and expands the above three types of iterative algorithms to perform the COS type-reduction and defuzzification of IT2 FLSs. In addition, several simulation examples are constructed to compare the performances of three types of iterative algorithms for obtaining the output of IT2 FLSs.

The rest of this paper is organized as follows. Section 2 introduces the Mamdani type IT2 FLSs. Section 3 proposes how to use three types of iterative algorithms to perform the COS TR of IT2 FLSs. Then six simulation examples are provided to show the performances of three types of iterative algorithms in Section 4. Finally the conclusions are given in Section 5.

2. **IT2 FLSs.** Generally speaking, IT2 FLSs can be divided into Mamdani type [6,18] and Takagi Sugeno Kang (TSK) type [17] from the aspect of inference construction. Without loss of generality, consider a Mamdani type IT2 FLS with p inputs and one output, in which the lth fuzzy rule can be represented as $\widetilde{R}^l$, i.e.,

$$\widetilde{R}^l : \text{If } x_1 \text{ is } \widetilde{F}_1^l \text{ and } \dots \text{and } x_p \text{ is } \widetilde{F}_p^l, \text{ then } y \text{ is } \widetilde{G}^l \ (l = 1, 2, \dots, M) \tag{1}$$

where $\widetilde{F}_i^l$ $(i = 1, 2, \dots, p)$ is the antecedent IT2 FS, and $\widetilde{G}^l$ is the consequent IT2 FS, let Y^l be the centroid interval for $\widetilde{G}^l$, and $Y^l = \left[\underline{y}^l, \overline{y}^l\right]$.

Here we let the membership functions (MFs) of antecedent, consequent and input measurement of IT2 be as

$$\mu_{\widetilde{F}_i^l}(x_i) = \left[\underline{\mu}_{\widetilde{F}_i^l}(x_i), \overline{\mu}_{\widetilde{F}_i^l}(x_i)\right] \tag{2}$$

$$\mu_{\widetilde{G}^l}(y) = \left[\underline{\mu}_{\widetilde{G}^l}(y), \overline{\mu}_{\widetilde{G}^l}(y)\right] \tag{3}$$

$$\mu_{\widetilde{X}_i}(x_i) = \left[\underline{\mu}_{\widetilde{X}_i}(x_i), \overline{\mu}_{\widetilde{X}_i}(x_i)\right] \tag{4}$$

The firing interval of the lth fuzzy rule is a T1 FS, i.e., $F^l = \left[\underline{f}^l, \overline{f}^l\right]$, when the non-singleton fuzzifier is adopted, i.e.,

$$\underline{f}^l(x') = T_{i=1}^p \underline{f}_i^l(x_i) = T_{i=1}^p \underline{\mu}_{\widetilde{Q}_i^l}\left(\underline{x}_{i,\max}^l\right) = T_{i=1}^p\left[\underline{\mu}_{\widetilde{X}_i}\left(\underline{x}_{i,\max}^l\right) \wedge \underline{\mu}_{\widetilde{F}_i^l}\left(\underline{x}_{i,\max}^l\right)\right] \tag{5}$$

$$\overline{f}^l(x') = T_{i=1}^p \overline{f}_i^l(x_i) = T_{i=1}^p \overline{\mu}_{\widetilde{Q}_i^l}\left(\overline{x}_{i,\max}^l\right) = T_{i=1}^p\left[\overline{\mu}_{\widetilde{X}_i}\left(\overline{x}_{i,\max}^l\right) \wedge \overline{\mu}_{\widetilde{F}_i^l}\left(\overline{x}_{i,\max}^l\right)\right] \tag{6}$$

in which $\underline{x}_{i,\max}^s$, and $\overline{x}_{i,\max}^s$ denote the corresponding x_i values for $\sup_{x_i}\underline{\mu}_{\widetilde{Q}_i^s}(x_i)$ and $\sup_{x_i}\overline{\mu}_{\widetilde{Q}_i^s}(x_i)$, respectively, T represents the product or minimum t-norm, and sup denotes the supremum.

Aggregating the M fuzzy rules, the output interval set for IT2 FLSs is as

$$Y = [y_l, y_r] = \int_{y^1\in\left[\underline{y}^1,\overline{y}^1\right]} \cdots \int_{y^M\in\left[\underline{y}^M,\overline{y}^M\right]} \times \int_{f^1\in\left[\underline{f}^1,\overline{f}^1\right]} \cdots \int_{f^M\in\left[\underline{f}^M,\overline{f}^M\right]} 1 \Bigg/ \frac{\sum_{l=1}^M f^l y^l}{\sum_{l=1}^M f^l} \tag{7}$$

Combine the centroids of M consequent IT2 FSs with the corresponding firing intervals to obtain the interval FS $Y_{\mathrm{COS}}(x')$, i.e.,

$$Y_{\mathrm{COS}}: \begin{cases} Y_{\mathrm{COS}}(x') = 1/\left[y_l(x'), y_r(x')\right] \\ y_l(x') = \min\limits_{\forall w_l\in\left[\underline{f}^l(x'),\overline{f}^l(x')\right]} \sum\limits_{l=1}^M \underline{y}^l w_l \Bigg/ \sum\limits_{l=1}^M w_l \\ y_r(x') = \max\limits_{\forall w_l\in\left[\underline{f}^l(x'),\overline{f}^l(x')\right]} \sum\limits_{l=1}^M \overline{y}^l w_l \Bigg/ \sum\limits_{l=1}^M w_l \end{cases} \tag{8}$$

where $\underline{y}^l$, and $\overline{y}^l$ are left and right centroid end points for IT2 FS $\widetilde{G}^l$; here the y_l and y_r can be computed by many different types of type-reduction algorithms.

The defuzzified (crisp) output can be achieved by taking the arithmetic average of y_l and y_r, i.e.,

$$y = \frac{y_l + y_r}{2} \tag{9}$$

3. **Three Types of Iterative Algorithms for COS Type-Reduction.**

At early times, KM algorithms, EKM algorithms, and EIASC are proposed according to the centroid type-reduction. This section provides the COS type-reduction and defuzzification for Mamdani IT2 FLSs based on the above three types of algorithms. Let the output set be $Y_{\mathrm{Mamdani,IT2}}(x) = [y_l, y_r]$, and then the crisp out is as $(y_l + y_r)/2$.

3.1. **KM algorithms.**

As shown in Equation (8), KM algorithms cannot compute y_l and y_r in the closed form; however, they calculate $y_l(L)$ and $y_r(R)$ in the approximation way, i.e.,

$$y_l(L) = \frac{\sum_{l=1}^{L} \underline{y}^l \overline{f}^l + \sum_{l=L+1}^{M} \underline{y}^l \underline{f}^l}{\sum_{l=1}^{L} \overline{f}^l + \sum_{l=L+1}^{M} \underline{f}^l} \approx y_l \tag{10}$$

$$y_r(R) = \frac{\sum_{l=1}^{R} \overline{y}^l \underline{f}^l + \sum_{l=R+1}^{M} \overline{y}^l \overline{f}^l}{\sum_{l=1}^{R} \underline{f}^l + \sum_{l=R+1}^{M} \overline{f}^l} \approx y_r \tag{11}$$

where both L and R are the switch points, for Equation (10), when $l = L+1$, w_l starts to transform from the upper firing degree $\overline{f}^l$ to the lower firing degree $\underline{f}^l$; while for Equation (11), when $l = R+1$, w_l starts to transform from the upper firing degree $\underline{f}^l$ to the lower firing degree $\overline{f}^l$.

Table 1 shows the specific computation steps for KM algorithms to complete the COS type-reduction of IT2 FLSs.

Table 1. Computation steps for KM algorithms to complete the COS type-reduction of IT2 FLSs

Step	KM algorithms to compute y_l
1	Initialize θ_l, set $\theta_l = \left[\underline{f}^l + \overline{f}^l\right] \big/ 2$, $l = 1, \ldots, M$, compute $c' = \frac{\sum_{l=1}^{M} y^l \theta_l}{\sum_{l=1}^{M} \theta_l}$
2	Find L $(1 \leq L \leq M-1)$, which satisfies $\underline{y}^L \leq c' \leq \underline{y}^{L+1}$
3	When $l \leq L$, set $\theta_l = \overline{f}^l$; when $l \geq L+1$, set $\theta_l = \underline{f}^l$, compute $y_l(l) = \frac{\sum_{l=1}^{L} \underline{y}^l \overline{f}^l + \sum_{l=L+1}^{M} \underline{y}^l \underline{f}^l}{\sum_{l=1}^{L} \overline{f}^l + \sum_{l=L+1}^{M} \underline{f}^l}$
4	Check if $y_l(l) = c'$, if so, stop and set $y_l(l) = y_l$, $l = L$
5	Set $c' = y_l(l)$ and return to Step 2
Step	KM algorithms to compute y_r
1	The same as former in Step 1
2	Find R $(1 \leq R \leq M-1)$, which satisfies $\underline{y}^R \leq c' \leq \underline{y}^{R+1}$
3	When $l \leq R$, set $\theta_l = \underline{f}^l$; when $l \geq R+1$, set $\theta_l = \overline{f}^l$, compute $y_r(l) = \frac{\sum_{l=1}^{R} \overline{y}^l \underline{f}^l + \sum_{l=R+1}^{M} \overline{y}^l \overline{f}^l}{\sum_{l=1}^{R} \underline{f}^l + \sum_{l=R+1}^{M} \overline{f}^l}$
4	Check if $y_r(l) = c'$, if so, stop and set $y_r(l) = y_r$, $l = R$
5	Set $c' = y_r(l)$ and return to Step 2

3.2. **EKM algorithms.** EKM algorithms [10,12,19,20] are actually originated from KM algorithms. And the former improves the latter for three points: 1) a better initialization approach is adopted to reduce the iterations; 2) change the iteration stop condition to cancel the non-necessary iteration; 3) a smart calculation technique is set to decrease the computational cost.

Table 2 provides the specific computation steps for EKM algorithms to complete the COS type-reduction of IT2 FLSs.

Table 2. Computation steps for EKM algorithms to complete the COS type-reduction of IT2 FLSs

Step	EKM algorithms to compute y_l
1	Set $k = [M/2.4]$ (the closest integer to $M/2.4$) and compute $a = \sum_{l=1}^{k} \underline{y}^l \overline{f}^l + \sum_{l=k+1}^{M} \underline{y}^l \underline{f}^l$, $b = \sum_{l=1}^{k} \overline{f}^l + \sum_{l=k+1}^{M} \underline{f}^l$, $c' = a/b$
2	Find $k' \in [1, M-1]$, which satisfies $\underline{y}^{k'} \leq c' \leq \underline{y}^{k'+1}$
3	Check if $k' = k$, if so, stop and set $c' = y_l$, $k = L$; otherwise, go to Step 4
4	Compute $s = sign(k'-k)$, $a' = a + s \sum_{l=\min(k,k')+1}^{mak(k,k')} \underline{y}^l \left(\overline{f}^l - \underline{f}^l\right)$, $b' = b + s \sum_{l=\min(k,k')+1}^{mak(k,k')} \left(\overline{f}^l - \underline{f}^l\right)$, and $c''(k') = a'/b'$
5	Set $c' = c''(k)$, $a = a'$, and $b = b'$ and return to Step 2
Step	EKM algorithms to compute y_r
1	Set $k = [M/1.7]$ (the closest integer to $M/1.7$) and compute $a = \sum_{l=1}^{k} \overline{y}^l \underline{f}^l + \sum_{l=k+1}^{M} \overline{y}^l \overline{f}^l$, $b = \sum_{l=1}^{k} \underline{f}^l + \sum_{l=k+1}^{M} \overline{f}^l$, $c' = a/b$
2	The same as the former in Step 2
3	The same as the former in Step 3, except for setting $c' = y_r$ and $k = R$
4	Compute $s = sign(k'-k)$, $a' = a - s \sum_{l=\min(k,k')+1}^{mak(k,k')} \overline{y}^l \left(\overline{f}^l - \underline{f}^l\right)$, $b' = b - s \sum_{l=\min(k,k')+1}^{mak(k,k')} \left(\overline{f}^l - \underline{f}^l\right)$, and $c''(k') = a'/b'$
5	The same as the former in Step 5

3.3. **EIASC.** EIASC is a type of non-KM iterative algorithm. This type of simple and fast type-reduction algorithms is comparatively easy to understand. The design idea of EIASC is based on the shape and monotoncity of IT2 FS. In Equation (10), $y_l(L)$ first decreases as L increases, then $y_l(L)$ increases as L increases. In Equation (11), $y_r(R)$ first increases as R increases, then $y_r(R)$ decreases as R increases.

Table 3 provides the specific computation steps for EIASC to complete the COS type-reduction of IT2 FLSs.

4. Simulation Experiments.

Six simulation examples are provided in this section to show how to use three types of iterative algorithms to perform the COS type-reduction and defuzzification for IT2 FLSs. In the first two examples, suppose that there are four antecedents and one consequent for each fuzzy rule, every antecedent is characterized by four IT2 FSs, so that, the total number of fuzzy rules is sixteen. Let the sth rule be as

$$\text{If } x_1 \text{ is } \widetilde{F}_1^s \text{ and} \ldots \text{and } x_4 \text{ is } \widetilde{F}_4^s, \text{ then } y \text{ is } \widetilde{G}^s \tag{12}$$

where $\widetilde{G}^s$ $(s = 1, 2, \ldots, 16)$ is the consequent IT2 FS, and $\widetilde{F}_i^s$ $(i = 1, 2, 3, 4; s = 1, 2, \ldots, 16)$ is the antecedent IT2 FS.

TABLE 3. Computation steps for EIASC to complete the COS type-reduction of IT2 FLSs

Step	EIASC to compute y_l	EIASC to compute y_r
1	Initialization:	Initialization:
	$a = \sum_{l=1}^{M} \underline{y}^l \underline{f}^l$, $b = \sum_{l=1}^{M} \underline{f}^l$, $L = 0$	$a = \sum_{l=1}^{M} \overline{y}^l \overline{f}^l$, $b = \sum_{l=1}^{M} \overline{f}^l$, $R = M$
2	Compute:	Compute:
	$L = L + 1$	$a = a + \overline{y}^R \left(\overline{f}^R - \underline{f}^R\right)$
	$a = a + \underline{y}^L \left(\overline{f}^L - \underline{f}^L\right)$	$b = b + \left(\overline{f}^R - \underline{f}^R\right)$
	$b = b + \left(\overline{f}^L - \underline{f}^L\right)$	$y_r = a/b$
	$y_l = a/b$	$R = R - 1$
3	If $y_l \le \underline{y}^{L+1}$, stop;	If $y_r \ge \overline{y}^R$, stop;
	otherwise, return to Step 2	otherwise, return to Step 2

Example A: For the IT2 FLSs, choose each antecedent as the Gaussian MF with uncertain standard derivation (see Figure 2), i.e.,

$$\mu_i^s(x_i) = \exp\left\{-\frac{1}{2}\left(\frac{x_i - m_i^s}{\sigma_i^s}\right)^2\right\} \quad (i = 1, 2, 3, 4; s = 1, 2, \ldots, 16) \tag{13}$$

where $\sigma_i^s \in [\sigma_{i1}^s, \sigma_{i2}^s]$, the centroid interval for consequent $C_{\widetilde{G}^s} = \left[l_{\widetilde{G}^s}, r_{\widetilde{G}^s}\right]$ $(s = 1, 2, \ldots, 16)$.

Select the mean, and lower and upper standard derivations parameters of Gaussian antecedent MF as

$$m_i^l = 1.8 + rand(4, 16) \tag{14}$$

$$\sigma_{i1}^l = 2.2 + rand(4, 16) \tag{15}$$

$$\sigma_{i2}^l = \sigma_{i1}^l + 2 * rand(4, 16) \tag{16}$$

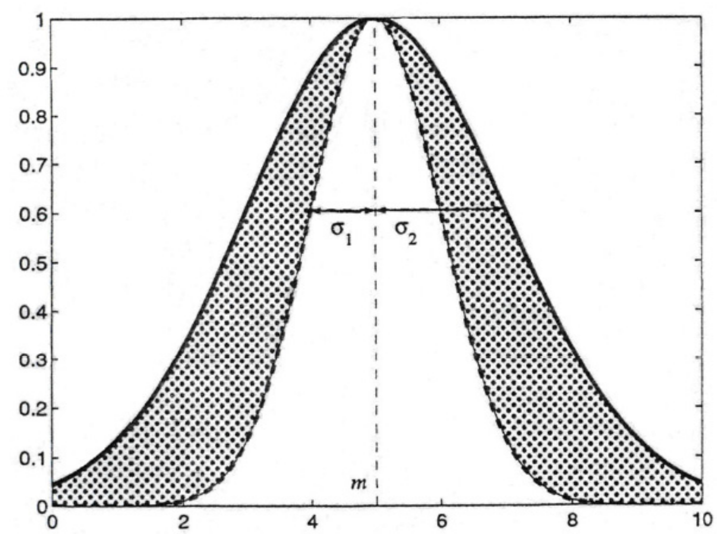


FIGURE 2. Gaussian MF with uncertain standard derivation [6,16]

In order to simplify the expressions, here we adopt the singleton fuzzifier, i.e., the input measurement is as

$$x = rand(16, 4) * 7 \tag{17}$$

Set $c_1 = 2 \cdot rand(1, 4)$, $c_2 = 2 \cdot rand(1, 4)$, and let

$$l_{\tilde{G}^s} = c_1 * x', \quad r_{\tilde{G}^s} = (c_1 + c_2) * x' \tag{18}$$

Example B: For the IT2 FLSs, choose each antecedent as the Gaussian MF with uncertain mean (see Figure 3), i.e.,

Choose the standard derivation, and left and right uncertain means parameters of Gaussian antecedent MF as

$$\sigma_i^l = 2.3 + rand(4, 16) \tag{19}$$

$$m_{i1}^l = 1.7 + rand(4, 16) \tag{20}$$

$$m_{i2}^l = m_{i1}^l + 2 * rand(4, 16) \tag{21}$$

Here the input measurement and the centroid interval for consequent are still chosen as the forms of (17) and (18), respectively. In Examples C and D, the parameters of IT2 FLSs are selected as forms in Examples A and B. However, the number of antecedents for each fuzzy rule is 5; therefore, there are thirty-two fuzzy rules for IT2 FLSs. For the last two examples, we choose the parameters of IT2 FLSs that are selected as forms in Examples A and B. The number of antecedents for each fuzzy rule is increased to 6, so that, the total number of fuzzy rules for IT2 FLSs is sixty-four.

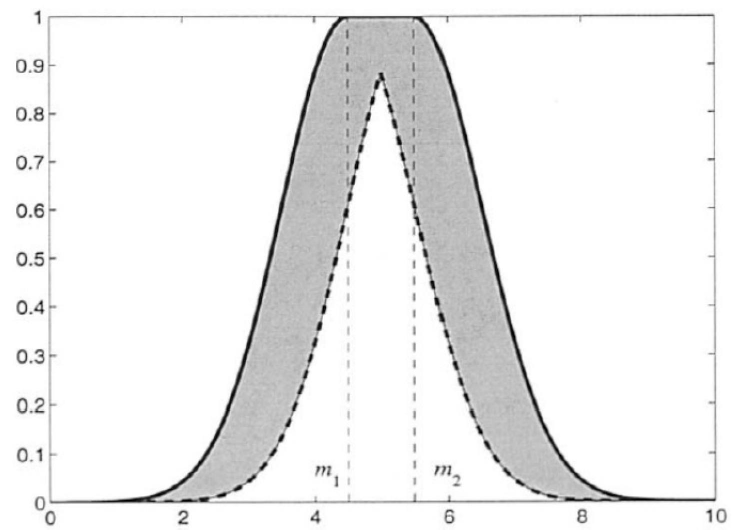


FIGURE 3. Gaussian MF with uncertain mean [15]

Next we do the quantitative studies for three types of COS type-reduction algorithms. For the above six examples, the COS type-reduction left endpoints, right endpoints, and defuzzified outputs computed by three types of iterative algorithms are shown in Table 4, Table 5, and Table 6, respectively.

In order to measure the computational efficiencies, the specific computation times (whose unit is the second) of defuzzified outputs for three types of iterative algorithms are provided in Table 7, in which the last line represents the averages for six examples.

Here we define the time reducing rate (TRR) for the latter two types of algorithms compared with the first type of algorithms, i.e.,

$$\text{TRR}_{1,2} = (t_0 - t_{1,2})/t_0 \times 100\% \tag{22}$$

where t_0 denotes the computation time for KM algorithms, and $t_{1,2}$ represent the computation times for EKM, and EIASC, respectively.

For the above illustrated six examples, the following conclusions can be made by observing Tables 4-7.

1) The COS type-reduction left endpoints, right endpoints, and defuzzified outputs computed by three types of iterative algorithms are slightly different, among which, the results of first two types of algorithms are almost the same.

Table 4. COS type-reduction left endpoints computed by three types of iterative algorithms

Num	KM	EKM	EIASC
Example A	8.6478	8.6478	8.6478
Example B	13.8737	13.8737	13.8737
Example C	13.6523	13.6523	13.6523
Example D	5.5408	5.5635	5.5408
Example E	14.1426	14.1448	14.1426
Example F	14.1344	14.6382	14.1344

Table 5. COS type-reduction right endpoints computed by three types of iterative algorithms

Num	KM	EKM	EIASC
Example A	24.5338	24.5338	23.5777
Example B	25.9618	25.9618	24.4816
Example C	24.4915	24.4915	23.5358
Example D	31.8958	31.8773	26.9786
Example E	37.6797	37.6797	34.2817
Example F	45.2733	45.2432	37.4745

Table 6. Defuzzified outputs computed by three types of iterative algorithms

Num	KM	EKM	EIASC
Example A	16.5908	16.5908	16.1127
Example B	19.9177	19.9177	19.1776
Example C	19.0719	19.0719	18.5940
Example D	18.7183	18.7204	16.2597
Example E	25.9111	25.9117	24.2121
Example F	29.7038	29.9407	25.8044

Table 7. Computation time for three types of algorithms

Num	KM	EKM	EIASC	TRR_1 (%)	TRR_2 (%)
Example A	0.000095	0.000081	0.000071	14.74	25.26
Example B	0.000124	0.000084	0.000074	32.26	40.32
Example C	0.000127	0.000116	0.000092	8.66	27.56
Example D	0.000194	0.000124	0.000112	36.08	42.27
Example E	0.000146	0.000138	0.000117	5.48	19.86
Example F	0.000145	0.000135	0.000129	6.90	11.03
Average	0.000139	0.000113	0.000099	18.71	28.78

2) The computation times of EIASC are the shortest, while the computation times of KM algorithms are the longest. The computation times of EKM algorithms take the intermediate position.

3) Compared with the KM algorithms, EKM algorithms can obtain the maximum TRR as 36.08%, the minimum TRR as 5.48%, and average TRR as 18.71%. While EIASC can get the maximum TRR as 42.27%, the minimum TRR as 11.03%, and average TRR as 28.78% in contrast to KM algorithms.

These six simulation examples illustrate the effectiveness and feasibility about adopting three types of iterative algorithms for designing the COS type-reduction and defuzzification of IT2 FLSs. Under the condition that the defuzzified values only have a little change, the computation times of three types of algorithms decrease in order, i.e., the computational efficiencies of IT2 FLSs have been improved.

5. Conclusions.

This paper proposed a type of IT2 FLS based on three types of iterative algorithms. Moreover, the blocks of fuzzy reasoning, COS type-reduction and defuzzification are also discussed. Six simulation experiments show the crisp out value and specific computation time results of IT2 FLSs obtained by three types of iterative algorithms. As the results of crisp outputs have a little change, and the computation times of the proposed three types of algorithms decrease in order, this may provide the potential value for optimizing and applying T2 FLSs.

Next, the authors will go on studying the COS type-reduction of IT2 and GT2 FLSs based on noniterative algorithms [14,22,24-27], the influence of initialization, weighting, and search space [19-21,23,28-31] for completing the centroid type-reduction of T2 FLSs, and the parameter optimization of T2 FLSs [40-43] for fuzzy control [32-34], fuzzy identification [35,36], forecasting [37-39] and so on.

Acknowledgment. The paper is sponsored by the National Natural Science Foundation of China (61973146, 61773188), the Liaoning Province Natural Science Foundation (20180550056), the Doctoral Start-up Foundation of Liaoning Province (No. 2021-BS-258), and the Talent Fund Project of Liaoning University of Technology (xr2020002). The authors are grateful to Professor Jerry Mendel, for he has provided the authors some contributed valuable ideas.

REFERENCES

[1] J. M. Mendel, *Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2001.

[2] J. Huang, M. Ri, D. R. Wu and S. Ri, Interval type-2 fuzzy logic modeling and control of a mobile two-wheeled inverted pendulum, *IEEE Transactions on Fuzzy Systems*, vol.26, no.4, pp.2030-2038, 2018.

[3] D. R. Wu, Z. G. Zeng, H. Mo and F. Y. Wang, Interval type-2 fuzzy sets and systems: Overview and outlook, *Acta Automatica Sinica*, vol.46, no.8, pp.1539-1556, 2020.

[4] J. M. Mendel, General type-2 fuzzy logic systems made simple: A tutorial, *IEEE Transactions on Fuzzy Systems*, vol.22, no.5, pp.1162-1182, 2014.

[5] J. M. Mendel, Type-2 fuzzy sets and systems: An overview, *IEEE Computational Intelligence Magazine*, vol.2, no.2, pp.20-29, 2007.

[6] Y. Chen, D. Z. Wang and S. C. Tong, Forecasting studies by designing Mamdani interval type-2 fuzzy logic systems: With combination of BP algorithms and KM algorithms, *Neurocomputing*, vol.174, no.b, pp.1133-1146, 2016.

[7] C. H. Hsu and C. F. Juang, Evolutionary robot wall-following control using type-2 fuzzy controller with species-de-activated continuous ACO, *IEEE Transactions on Fuzzy Systems*, vol.21, no.1, pp.100-112, 2013.

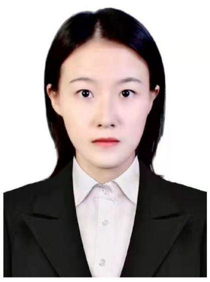
[8] C. S. Lee, M. H. Wang and H. Hagras, Type-2 fuzzy ontology and its application to personal diabetic-diet recommendation, *IEEE Transactions on Fuzzy Systems*, vol.18, no.2, pp.316-328, 2010.

[9] N. N. Karnik and J. M. Mendel, Centroid of a type-2 fuzzy set, *Information Sciences*, vol.132, no.1, pp.195-220, 2001.
[10] J. M. Mendel, On KM algorithms for solving type-2 fuzzy set problems, *IEEE Transactions on Fuzzy Systems*, vol.21, no.3, pp.426-446, 2013.
[11] J. M. Mendel and F. L. Liu, Super-exponential convergence of the Karnik-Mendel algorithms for computing the centroid of an interval type-2 fuzzy set, *IEEE Transactions on Fuzzy Systems*, vol.15, no.2, pp.309-320, 2007.
[12] D. Wu and J. M. Mendel, Enhanced Karnik-Mendel algorithms, *IEEE Transactions on Fuzzy Systems*, vol.17, no.4, pp.923-934, 2009.
[13] D. R. Wu and M. Nie, Comparison and practical implementations of type reduction algorithms for type-2 fuzzy sets and systems, *International Conference on Fuzzy Systems*, Taipei, Taiwan, pp.2131-2138, 2011.
[14] M. A. Khanesar, A. Jalalian and O. Kaynak, Improving the speed of center of set type-reduction in interval type-2 fuzzy systems by eliminating the need for sorting, *IEEE Transactions on Fuzzy Systems*, vol.25, no.5, pp.1193-1206, 2017.
[15] Y. Chen, Design of Takagi-Sugeno-Kang type interval type-2 fuzzy logic systems based on enhanced opposite direction searching algorithms, *International Journal of Innovative Computing, Information and Control*, vol.16, no.6, pp.2091-2101, 2020.
[16] D. Z. Wang and Y. Chen, Study on permanent magnetic drive forecasting by designing Takagi Sugeno Kang type interval type-2 fuzzy logic systems, *Transactions of the Institute of Measurement and Control*, vol.40, no.6, pp.2011-2023, 2018.
[17] A. Khosravi and S. Nahavandi, Load forecasting using interval type-2 fuzzy logic systems: Optimal type reduction, *IEEE Transactions on Industrial Informatics*, vol.10, no.2, pp.1055-1063, 2014.
[18] G. M. Méndez and M. D. L. A. Hernández, Hybrid learning for interval type-2 fuzzy systems based on orthogonal least-squares and back-propagation methods, *Information Sciences*, vol.179, no.13, pp.2146-2157, 2009.
[19] X. Liu, J. M. Mendel and D. Wu, Study on enhanced Karnik-Mendel algorithms: Initialization explanations and computation improvements, *Information Sciences*, vol.184, no.1, pp.75-91, 2012.
[20] Y. Chen and D. Z. Wang, Study on centroid type-reduction of general type-2 fuzzy logic systems with weighted enhanced Karnik-Mendel algorithms, *Soft Computing*, vol.22, no.4, pp.1361-1380, 2018.
[21] Y. Chen and D. Z. Wang, Study on centroid type-reduction of general type-2 fuzzy logic systems with weighted Nie-Tan algorithms, *Soft Computing*, vol.22, no.22, pp.7659-7678, 2018.
[22] J. M. Mendel and X. Liu, Simplified interval type-2 fuzzy logic systems, *IEEE Transactions on Fuzzy Systems*, vol.21, no.6, pp.1056-1069, 2013.
[23] Y. Chen, J. X. Wu and J. Lan, Study on weighted-based noniterative algorithms for centroid type-reduction of interval type-2 fuzzy logic systems, *AIMS Mathematics*, vol.5, no.6, pp.7719-7745, 2020.
[24] J. W. Li, R. John, S. Coupland and G. Kendall, On Nie-Tan operator and type-reduction of interval type-2 fuzzy sets, *IEEE Transactions on Fuzzy Systems*, vol.26, no.2, pp.1036-1039, 2018.
[25] E. Ontiveros-Robles, P. Melin and O. Castillo, New methodology to approximate type-reduction based on a continuous root-finding Karnik Mendel algorithm, *Algorithms*, vol.10, no.3, pp.77-96, 2017.
[26] M. Biglarbegian, W. Melek and J. Mendel, On the robustness of type-1 and interval type-2 fuzzy logic systems in modeling, *Information Sciences*, vol.181, no.7, pp.1325-1347, 2011.
[27] S. Greenfield and F. Chiclana, Accuracy and complexity evaluation of defuzzification strategies for the discretised interval type-2 fuzzy set, *International Journal of Approximate Reasoning*, vol.54, no.8, pp.1013-1033, 2013.
[28] Y. Chen, J. X. Wu and J. Lan, Study on reasonable initialization enhanced Karnik-Mendel algorithms for centroid type-reduction of interval type-2 fuzzy logic systems, *AIMS Mathematics*, vol.5, no.6, pp.6149-6168, 2020.
[29] Y. Chen, Study on centroid type-reduction of interval type-2 fuzzy logic systems based on noniterative algorithms, *Complexity*, vol.2019, Article ID 7325053, pp.1-12, 2019.
[30] A. M. El-Nagar and M. El-Bardini, Simplified interval type-2 fuzzy logic system based on new type-reduction, *Journal of Intelligent & Fuzzy Systems*, vol.27, no.4, pp.1999-2010, 2014.
[31] S. Greenfield, F. Chiclana, S. Coupland and R. John, The collapsing method of defuzzification for discretised interval type-2 fuzzy sets, *Information Sciences*, vol.179, no.13, pp.2055-2069, 2009.
[32] S. C. Tong and Y. M. Li, Observer-based adaptive fuzzy backstepping control of uncertain pure-feedback systems, *Science China Information Sciences*, vol.57, no.1, pp.1-14, 2014.

[33] O. Castillo, L. Amador-Angulo, J. R. Castro et al., A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems, *Information Sciences*, vol.354, no.c, pp.257-274, 2016.
[34] T. Zhao and J. Xiao, State feedback control of interval type-2 Takagi-Sugeno fuzzy systems via interval type-2 regional switching fuzzy controllers, *International Journal of Systems Science*, vol.46, no.15, pp.2756-2769, 2015.
[35] O. Castillo, P. Melin, E. Ontiveros et al., A high-speed interval type 2 fuzzy system approach for dynamic parameter adaptation in metaheuristics, *Engineering Applications of Artificial Intelligence*, vol.85, pp.666-680, 2019.
[36] G. Ju and L. Chen, Fuzzy model identification and self-learning algorithm for dynamic systems, *Acta Automatica Sinica*, vol.21, no.6, pp.749-753, 1995.
[37] Y. Chen, D. Z. Wang and W. Ning, Forecasting by TSK general type-2 fuzzy logic systems optimized with genetic algorithms, *Optimal Control Applications & Methods*, vol.39, no.1, pp.393-409, 2018.
[38] D. Bernardo, H. Hagras and E. Tsang, A genetic type-2 fuzzy logic based system for the generation of summarized linguistic predictive models for financial applications, *Soft Computing*, vol.17, no.12, pp.2185-2201, 2013.
[39] M. H. F. Zarandi, B. Rezaee, I. B. Turksen and E. Neshat, A type-2 fuzzy rule-based expert system model for stock price analysis, *Expert Systems with Applications*, vol.36, no.1, pp.139-154, 2009.
[40] Sasmoko, W. Budiharto and H. Prabowo, Self-diagnostic using fuzzy logic for teaching learning quality improvement in universities, *ICIC Express Letters*, vol.13, no.3, pp.247-253, 2019.
[41] E. Ontiveros-Robles, P. Melin and O. Castillo, Comparative analysis of noise robustness of type 2 fuzzy logic controllers, *Kybernetika*, vol.54, no.1, pp.175-201, 2018.
[42] Y. Chen and J. X. Yang, Study on center-of-sets type-reduction of interval type-2 fuzzy logic systems with noniterative algorithms, *Journal of Intelligent & Fuzzy Systems*, vol.40, no.6, pp.11099-11106, 2021.
[43] D. R. Wu and J. M. Mendel, Recommendations on designing practical interval type-2 fuzzy logic systems, *Engineering Applications of Artificial Intelligence*, vol.85, pp.182-193, 2019.

Author Biography

Yang Chen received the B.S. degree and the M.S. degree in Information and Computing Science and Applied Mathematics from Liaoning University of Technology, Jinzhou, China, in 2004 and 2007, respectively. He received the Ph.D. degree in Control Theory and Control Engineering from Northeastern University, Shenyang, China in 2020. He is currently an associate professor in College of Science, Liaoning University of Technology. His current research interests include fuzzy reasoning and fuzzy control, type-2 fuzzy logic systems, intelligent computing, and permanent magnetic drive.



Jiaxiu Yang received the B.S. degree in Mathematics and Applied Mathematics from Dezhou University, Dezhou, China, in 2020. She is currently a Master candidate in College of Science, Liaoning University of Technology. Her current research interests include fuzzy reasoning and fuzzy control, intelligent computing, and type-2 fuzzy logic systems.