A Study of TSK Inference Approaches for Control Problems

Jie Li[1], Fei Chao[2], and Longzhi Yang[3]

[1] School of Computing & Digital Technologies, Teesside University, Middlesbrough, U.K., jie.li@tees.ac.uk
[2] Department of Cognitive Science, Xiamen University, Xiamen, U.K., fchao@xmu.edu.cn
[3] Department of Computer and Information Sciences, Northumbria University, Newcastle upon Tyne, U.K., longzhi.yang@northumbria.ac.uk

Abstract. Fuzzy inference systems provide a simple yet powerful solution to complex non-linear problems, which have been widely and successfully applied in the control field. The TSK-based fuzzy inference approaches, such as the convention TSK, interval type 2 (IT2) TSK and their extensions TSK+ and IT2 TSK+ approaches, are more convenient to be employed in the control field, as they directly produce crisp outputs. This paper systematically reviews those four TSK-based inference approaches, and evaluates them empirically by applying them to a well-known cart centering control problem. The experimental results confirm the power of TSK+ and IT2 TSK+ approaches in enhancing the inference using either dense or sparse rule bases.

Keywords: TSK, TSK+, Fuzzy Control, Fuzzy Inference, Sparse Rule Base

1 Introduction

Fuzzy inference is a mechanism that uses fuzzy logic and fuzzy set theory to map input domains to output domains. A typical fuzzy inference system consists of two main parts, a rule base and an inference engine. Several inference engines have been developed, with the Mamdani inference [1] and the TSK inference [2] being most widely applied. Compared with the Mamdani inference approach, which takes human linguistic variables as inputs to produce fuzzy outputs and thus requires a defuzzification process to convert the fuzzy outputs to crisp values, the TSK inference approach uses polynomials as the rule consequences to directly generate crisp outputs. For better uncertainty management and performance, these fuzzy inference approaches have been extended to support interval type-2 (IT2) fuzzy sets. Generally speaking, an IT2 fuzzy set represents the membership of a given member as a crisp interval in the range of $[0, 1]$ [3]. Nevertheless, a complete knowledge base (also termed as a dense rule base), which covers the entire input domains, is always required by both conventional type-1 and IT2 fuzzy inference approaches; otherwise, no rule can be fired and no results can be consequently produced if a given input does not overlap with any rule antecedent in the rule base.

Fuzzy interpolation, firstly proposed in [4], relaxes the requirement of dense rule bases, thus to alleviates the problem of lack of knowledge in the rule base. Fundamentally, fuzzy interpolation considers the neighbouring rules in the rule base by means of fuzzified polynomial (usually linear) interpolation to produce the inference results. Therefore, when given inputs do not overlap with any rule antecedent, a certain conclusion can still be obtained. Various fuzzy interpolation methods and extensions have been developed in the literature, including the works reported in [4–11] using Mamdani style inference, and [3, 12] using TSK style inference. Due to the simplicity and effectiveness in representing and reasoning on human natural language, fuzzy inference and fuzzy interpolation technologies have been successfully applied to not only the control problems, such as the train operation system in Japan [13], intelligent home heating controller [14], and manufacturing scheduling and planning [15, 16], but also other decision-making problems, such as cybersecurity [17–19], business [20], computer network [21], and healthcare [22].

This paper systematically reviews different types of TSK fuzzy inference approaches and their corresponding extensions, including the TSK, IT2 TSK approach, TSK+ inference approach, IT2 TSK+ inference approach. Briefly, the convention TSK and IT2 TSK approaches are only applicable to problems with dense rule bases, but TSK+ and IT2 TSK+ can work with either dense or sparse rule bases. These approaches are then applied to a well-known control problem, the cart centering problem with various sizes of rule bases for empirical evaluation. The experimental results show that the TSK+ and IT2 TSK+ inference approaches enhance the convention TSK and IT2 TSK approaches by means of broader applicability.

The rest of this paper is structured as follows: Section 2 introduces the relevant background theory. Section 3 reviews the four different types of TSK inference approaches. Section 4 reports the experimentation of the TSK approaches on a cart centering problem; and Section 5 concludes the paper and suggests probable future work.

2 Background

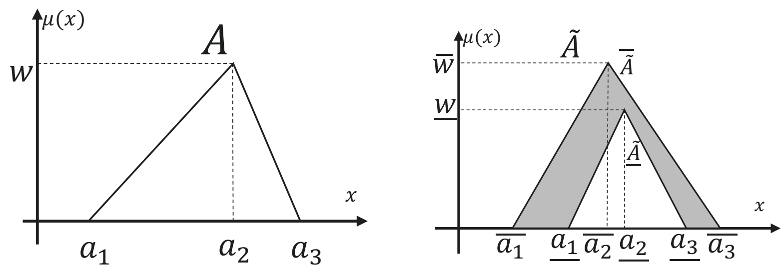
The relevant background theories, including fuzzy sets and interval type-2 fuzzy sets, are introduced in this section.

2.1 Type-1 Fuzzy Sets

Fuzzy logic defines the concept of the fuzzy sets that use membership functions to represent the relationships between elements and their degrees of membership, expressed in the range of $[0, 1]$ [23]. Given a type-1 fuzzy set, denoted as A, it can be expressed as:

$$A = \{(x, \mu_A(x)) | \forall x \in X, \forall \mu_A(x) \in [0, 1]\}, \tag{1}$$

where X is the domain of universe, $\mu_A(x)$ represents the membership for a given x. Assume that the conventional triangle membership is used to represent fuzzy set A as: $A = (a_1, a_2, a_3, w)$, as illustrated in Figure 1(a), where $w, w \in (0, 1]$, is the degree of confidence for fuzzy set A. Apprently $w = 1$, if A is a normal fuzzy set.



(a) A triangular fuzzy set

(b) LMF $\underline{\tilde{A}}$ and UMF $\overline{\tilde{A}}$ of a triangle IT2 fuzzy set $\tilde{A}$

Fig. 1: Triangle Fuzzy Sets

2.2 Interval Type-2 Fuzzy Sets

A type-2 fuzzy set, denoted as $\tilde{A}$, can be represented as:

$$\tilde{A} = \{((x, u), \mu_{\tilde{A}}(x, u)) | \forall x \in X, \forall u \in J_x \subseteq [0, 1], \quad \mu_{\tilde{A}}(x, u) \in [0, 1]\}\ , \qquad (2)$$

where X is the primary domain, J_x is the primary membership for a given element x, and $\mu_{\tilde{A}}(x, u)$ denotes the secondary membership. Taken the triangle IT2 fuzzy set $\tilde{A}$ as an example, as illustrated in Figure 1(b), It can be represented by a lower membership function (LMF), $\underline{\tilde{A}} = (\underline{a_1}, \underline{a_2}, \underline{a_3}, \underline{w})$, and a upper membership function (UMF), $\overline{\tilde{A}} = (\overline{a_1}, \overline{a_2}, \overline{a_3}, \overline{w})$. In this case, $\tilde{A} =< \underline{\tilde{A}}, \overline{\tilde{A}} >$, where $(\underline{a_1}, \underline{a_2}, \underline{a_3})$ and $(\overline{a_1}, \overline{a_2}, \overline{a_3})$ are respectively the three odd points of the LMF and UMF, and $\underline{w}$ and $\overline{w}$ denote respectively the degrees of confidence for $\underline{\tilde{A}}$ and $\overline{\tilde{A}}$, with $0 < \underline{w} \leqslant \overline{w} = 1$. The area between LMF and UMF, illustrated in grey in Figure 1(b), thus denotes the footprint of uncertainty (FOU), which represents the uncertainty of the fuzzy set $\tilde{A}$. Obviously, a larger FOU area implies a higher level of uncertainty; and the IT2 fuzzy set degenerates to a type-1 fuzzy set when $\underline{\tilde{A}}$ coincides with $\overline{\tilde{A}}$ (i.e., the area of $FOU(\tilde{A})$ is 0).

3 TSK Fuzzy Control

Four different TSK-style fuzzy models are expressed in this section.

3.1 Conventional Type-1 TSK Fuzzy Model

Suppose that a TSK-style fuzzy rule base comprises of n rules each with m antecedents:

$$\begin{aligned}
R_1 : &\textbf{IF } x_1 \text{ is } A_1^1 \text{ and } \cdots \text{and } x_m \text{ is } A_m^1 \\
&\textbf{THEN } y = f_1(x_1^1, \cdots, x_m^1) = p_0^1 + p_1^1 x_1^1 + \cdots + p_m^1 x_m^1, \\
\cdots\cdots & \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad (3) \\
R_n : &\textbf{IF } x_1 \text{ is } A_1^n \text{ and } \cdots \text{and } x_m \text{ is } A_m^n \\
&\textbf{THEN } y = f_n(x_1^n, \cdots, x_m^n) = p_0^n + p_1^n x_1^n + \cdots + p_m^n x_m^n,
\end{aligned}$$

where p_0^k and p_s^k, ($k \in \{1, 2, \cdots, n\}$ and $s \in \{1, 2, \cdots, m\}$) are constant parameters of the linear functions of rule consequences. The consequence polynomials deteriorate to constant numbers p_0^k when the outputs are discrete crisp numbers (usually to represent symbolic values). Given an input vector $(A_1^*, \cdots, A_m^*)$, the TSK engine performs inference in the following steps:

Step 1: Calculate the firing strength of each rule R_k ($k \in \{1, 2, \cdots, n\}$) by integrating the matching degrees between its antecedents and the given inputs:

$$\alpha_k = \mu(A_1^*, A_1^k) \wedge \cdots \wedge \mu(A_m^*, A_m^k), \tag{4}$$

where $\wedge$ is a t-norm usually implemented as a minimum operator, and $\mu(A_s^*, A_s^k)$ ($s \in \{1, 2, \cdots, m\}$) is the matching degree between fuzzy sets A_s^* and A_s^k:

$$\mu(A_s^*, A_s^k) = \max\{\min\{\mu_{A_s^*}(x), \mu_{A_s^k}(x)\}\}, \tag{5}$$

where $\mu_{A_s^*}(x)$ and $\mu_{A_s^k}(x)$ are the degrees of membership for a given value x within the domain. Note that $\alpha_k = 0$ if there is no overlap between the given inputs and any rule antecedent; in this case, rule $R - K$ will not be fired.

Step 2: Obtain the sub-output led by each rule R_k based on the given observation $(A_1^*, \cdots, A_m^*)$:

$$f_k(x_1^*, \cdots, x_m^*) = p_0^k + p_1^k Rep(A_1^*) + \cdots + p_m^r Rep(A_m^*), \tag{6}$$

where $Rep(A_s^*)$ is the representative value or defuzzified value of fuzzy set fuzzy set A_s^*, which is often calculated as the centre of gravity of the membership function.

Step 3: Determine the final output by integrating all the sub-outputs from all the rules:

$$y = \frac{\sum_{k=1}^{n} \alpha_k f_k(x_1^*, \cdots, x_m^*)}{\sum_{k=1}^{n} \alpha_k}. \tag{7}$$

It is clear from Eq. 5 that the firing strength will be 0 if a given input vector does not overlap with any rule antecedent. In this case, no rule will be fired and the conventional TSK approach will fail.

3.2 Type-1 TSK+ Fuzzy Model

TSK+ fuzzy inference approach [3, 12] is an extension of the conventional TSK fuzzy inference, which allows the TSK fuzzy inference to still be performed over a sparse rule base. Note that in a sparse rule base, the given observation may not overlap with any rule antecedent. In order to enable this extension, Eq. 5, which is used to obtain the firing strength for overlapped rules, is replaced by

$$\mu(A, A^*) = \left(1 - \frac{\sum_{i=1}^{3} |a_i - a_i^*|}{3}\right) \cdot d \cdot \frac{\min(w, w^*)}{\max(w, w^*)}, \tag{8}$$

where w and w^* denote the degrees of confidence for fuzzy sets A and A^*, respectively, and d, termed as $distance\ factor$, is a function of the distance between the two concerned fuzzy sets:

$$d = \begin{cases} 1 & ; \begin{array}{c} a_1 = a_2 = a_3 \\ \& \ a_1^* = a_2^* = a_3^* \end{array} \\ 1 - \frac{1}{1+e^{(-s \cdot \|A, A^*\| + 5)}} & ; \quad \text{otherwise,} \end{cases} \tag{9}$$

where $\|A, A^*\|$ represents the distance between the two fuzzy sets usually defined as the Euclidean distance of their representative values, and s ($s > 0$) is an adjustable sensitivity factor. Smaller value of s leads to a similarity degree which is more sensitive to the distance of the two fuzzy sets. Given a rule base as specified in Eq. 3 and an input vector $(A_1^*, \cdots, A_m^*)$, the TSK+ performs inferences using the same steps as those detailed in Section 3.1 except that Eq. 5 is replaced by Eq. 8.

In the TSK+ inference model, every rule in the rule base contributes to the final inference result to a certain degree. Therefore, even if the given observation does not overlap with any rule antecedent in the rule base, certain inference result can still be generated, which significantly improves the applicability of the conventional TSK inference system.

3.3 Conventional Interval Type-2 TSK Fuzzy Model

Generally speaking, in an IT2 TSK fuzzy model, the inputs and all the fuzzy sets in the rule antecedents are but not necessarily be IT2 fuzzy sets; and the consequence of IT2 TSK rules are zero or first order of polynomial functions, where the parameters can be either crisp values or a crisp interval. Assume that an IT2 TSK rule base is comprised of n rules as:

$$\begin{aligned} &R_1 : \textbf{IF } x_1 \text{ is } \tilde{A}_1^1 \text{ and } \ldots \text{ and } x_m \text{ is } \tilde{A}_m^1 \\ &\qquad \textbf{THEN } y = f_1(x_1^1, \cdots, x_m^1) = \tilde{p}_0^1 + \tilde{p}_1^1 x_1^1 + \cdots + \tilde{p}_m^1 x_m^1, \\ &\ldots \\ &R_n : \textbf{IF } x_1 \text{ is } \tilde{A}_1^n \text{ and } \ldots \text{ and } x_m \text{ is } \tilde{A}_m^n \\ &\qquad \textbf{THEN } y = f_n(x_1^n, \cdots, x_m^n) = \tilde{p}_0^n + \tilde{p}_1^n x_1^n + \cdots + \tilde{p}_m^n x_m^n, \end{aligned} \tag{10}$$

where $\tilde{A}_j^k, (j \in \{1, \ldots, m\}, k \in \{1, \ldots, n\})$ is an IT2 fuzzy set regarding input variable x_j in the k^{th} rule, ad discussed in Section 2.2. The consequence is a crisp polynomial function $y = f_k(x_1, \ldots, x_m) = \tilde{p}_0^k + \tilde{p}_1^k x_1^k + \cdots + \tilde{p}_m^k x_m^k$, where $\tilde{p}_j^k$ are parameters usually being crisp intervals, represented as $[\underline{\tilde{p}_j^k}, \overline{\tilde{p}_j^k}]$. For a given input $O(\tilde{A}_1^*, \cdots, \tilde{A}_m^*)$, the steps for calculating the final inference output can be summarised as follows:

Step 1: Compute the firing strength α_k of the k^{th} rule by

$$\begin{aligned} \tilde{\alpha}_k =& [\underline{\tilde{\alpha}_k}, \overline{\tilde{\alpha}_k}] = \sqcap_{j=1}^m \tilde{\mu}(\tilde{A}_j^k, \tilde{A}_j^*) \\ =& [\min(\min(\tilde{\mu}_{\tilde{A}_1^k}(x), \tilde{\mu}_{\tilde{A}_1^*}(x)), ..., \min(\tilde{\mu}_{\tilde{A}_m^k}(x), \tilde{\mu}_{\tilde{A}_m^*}(x))), \\ & \min(\max(\tilde{\mu}_{\tilde{A}_1^k}(x), \tilde{\mu}_{\tilde{A}_1^*}(x)), ..., \max(\tilde{\mu}_{\tilde{A}_m^k}(x), \tilde{\mu}_{\tilde{A}_m^*}(x)))], \end{aligned} \tag{11}$$

where $\sqcap$ is the meet operation. It is clear for the Eq. 11, $\tilde{\alpha} = [0, 0]$ if the given input does not overlap with any rule antecedent.

Step 2: Determine the intermediate results from individual rule based on the given input O by:

$$\begin{aligned}\tilde{c}^k =& \tilde{p}_0^k + \tilde{p}_1^k x_1^k + \cdots + \tilde{p}_m^k x_m^k \\ =& [\underline{\tilde{p}_0^k} + \underline{\tilde{p}_1^k} x_1 + \cdots + \underline{\tilde{p}_m^k} x_m \ , \overline{\tilde{p}_0^k} + \overline{\tilde{p}_1^k} x_1 + \cdots + \overline{\tilde{p}_m^k} x_m],\end{aligned} \tag{12}$$

where $\tilde{c}^k$ is a crisp interval that indicates the intermediate result led by rule R_k. $\underline{\tilde{p}_j^i}$ and $\overline{\tilde{p}_j^i}, (j \in \{0, 1, \cdots, k\})$, denote the minimum and maximum values of crisp interval $\tilde{p}_j^i$, respectively.

Step 3: Generate the final output $\tilde{c}$ by:

$$\begin{aligned}\tilde{c} =& [\underline{\tilde{c}}, \overline{\tilde{c}}] \\ =& \int_{\tilde{c}^1 \in [\underline{\tilde{c}^1}, \overline{\tilde{c}^1}]} \cdots \int_{\tilde{c}^n \in [\underline{\tilde{c}^n}, \overline{\tilde{c}^n}]} \int_{\tilde{\alpha}^1 \in [\underline{\tilde{\alpha}^1}, \overline{\tilde{\alpha}^1}]} \cdots \int_{\tilde{\alpha}^n \in [\underline{\tilde{\alpha}^n}, \overline{\tilde{\alpha}^n}]} 1 \Big/ \frac{\sum_{i=1}^n \tilde{\alpha}^i \cdot \tilde{c}^i}{\sum_{i=1}^n \tilde{\alpha}^i} \ .\end{aligned} \tag{13}$$

This equation can be practically implemented by computing the two extreme values of the crisp interval, minimum $\underline{\tilde{c}}$ and maximum $\overline{\tilde{c}}$, separately:

$$\left[\underline{\tilde{c}} = \frac{\sum_{i=1}^{L} \overline{\tilde{\alpha}^i} \underline{\tilde{c}^i} + \sum_{j=L+1}^{n} \underline{\tilde{\alpha}^i} \underline{\tilde{c}^i}}{\sum_{i=1}^{L} \overline{\alpha}_i + \sum_{j=L+1}^{n} \underline{\alpha}_j}, \ \overline{\tilde{c}} = \frac{\sum_{i=1}^{R} \overline{\tilde{\alpha}^i} \overline{\tilde{c}^i} + \sum_{j=R+1}^{n} \underline{\tilde{\alpha}^i} \overline{\tilde{c}^i}}{\sum_{i=1}^{R} \overline{\alpha}_i + \sum_{j=R+1}^{n} \underline{\alpha}_j}, \right] \tag{14}$$

where L and R are the *switch points* that used to make sure $\underline{\tilde{c}}$ is minimized and $\overline{\tilde{c}}$ is maximized, which can be obtained by an iterative procedure. A number of implementations on such problem have been proposed in the literature and widely used in the real world, such as the Karnik-Mendel (KM) algorithms, enhanced Karnik-Mendel algorithms (EKMA), an iterative algorithm with stop condition (ISAC), and enhanced ISAC [24]. In particular, the Karnik-Mendel (KM) algorithm is adapted in this work due to its efficiency, and the details of this approach is omitted here as this is beyond the focus of this paper.

Once the output interval or special IT2 fuzzy set is generated, type reduction or defuzzification needs to be applied. This can be readily implemented by applying a simple average operation:

$$c = \frac{\underline{\tilde{c}} + \overline{\tilde{c}}}{2}. \tag{15}$$

The same as the convention type-1 TSK fuzzy inference model, the convention IT2 TSK fuzzy inference approach is only able to work with dense rule bases; otherwise, no rule can be fired and consequently, no result can be generated.

3.4 Interval Type-2 TSK+ Fuzzy Model

In order to address the requirement of a dense rule base, the conventional IT2 TSK fuzzy inference approach has also been extended to IT2 TSK+ approach to work with the sparse rule base [25]. The working procedure of this extension follows the processes introduced in Section 3.2, which uses Eq. 8 to obtain the firing strength of each rule instead of the overlapped matching degree. As a result, Eq. 11 can be rewrite as:

$$\begin{aligned}\tilde{\alpha}_k =&[\underline{\tilde{\alpha}_k}, \overline{\tilde{\alpha}_k}] = \sqcap_{j=1}^{m} \tilde{\mu}(\tilde{A}_j^k, \tilde{A}_j^*) \\ =&[\mu(\underline{\tilde{A}_1^k}, \underline{\tilde{A}_1^*}) \wedge \cdots \wedge \mu(\underline{\tilde{A}_m^k}, \underline{\tilde{A}_m^*}), \mu(\overline{\tilde{A}_1^k}, \overline{\tilde{A}_1^*}) \wedge \cdots \wedge \mu(\overline{\tilde{A}_m^*}, \overline{\tilde{A}_m^*})].\end{aligned} \tag{16}$$

From here, the final crisp output for the given input can be calculated using the same steps as detailed in Section 3.3.

4 Experimentation

A well-known cart centering problem, which has been considered by a conventional IT2 TSK fuzzy model with a dense rule base in [26], is re-considered in this section for evaluation and comparison purpose.

4.1 Cart Centering Control Problem

A cart can only move along a line on a plane, which assumes the plane is frictionless, and the goal of this control problem is to drive and keep the cart to the central position of this line from a given initial position, as illustrated in Fig. 2. The inputs of the controller for this problem are the current position coordinates of cart x and the current velocity of cart v, and the output of this fuzzy model is force F that should be applied on the cart. In [26], the domain of the cart position x was restricted from $-0.75m$ to $0.75m$; the range of cart velocity v was restricted from $-0.75m/s$ to $0.75m/s$; the output force F was defined between $-0.18m/s$ and $0.18m/s$; and the sampling time used was $t = 0.1s$. This set of parameters and constraints were also utilised in this experiment, reported herein.

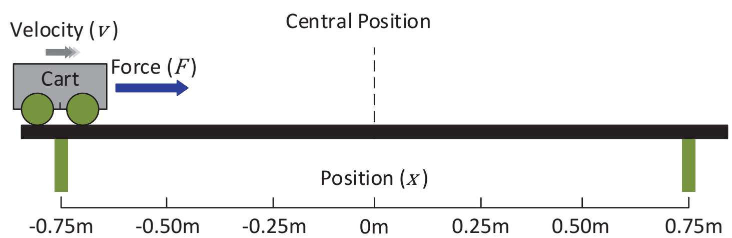
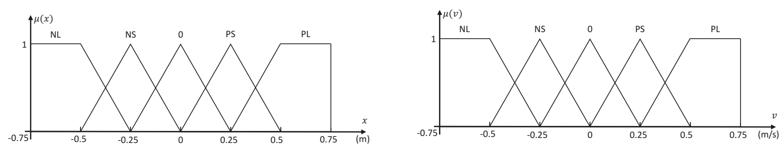


Fig. 2: The cart centering problem

4.2 Type-1 TSK Fuzzy Control

The work in [25] designed and demonstrated a 0-order IT2 TSK fuzzy model to solve this control problem, which used five linguistic values, denoted as IT2 fuzzy sets, to cover every domain of input variables x and v. In order to enable a direct comparison, the design reported in the work of [25] was adopted in this experimentation. Therefore, five type-1 fuzzy sets were created to cover the entire input domain x and v, which are negative large (NL), negative small (NS), zero (0), positive small (PS), and positive large (PL), as illustrated in Figure 3. And five crisp values were used as the output, which are also represented as NL, NS, 0, PS and PL, which are shown in Table 1. Consequently, a dense rule base, which contains in total 25 0-order TSK fuzzy rules were generated, as listed in Table 2.



(a) Fuzzy partition for the domain of Position

(b) Fuzzy partition for the domain of Velocity

Fig. 3: Fuzzy partition on input domain

Table 1: Output values

Output Label	**NL**	**NS**	**0**	**PS**	**PL**
Value	-0.16	-0.08	0	0.08	0.16

Table 2: Generated dense rule base for Type-1 TSK fuzzy model with 25 rules

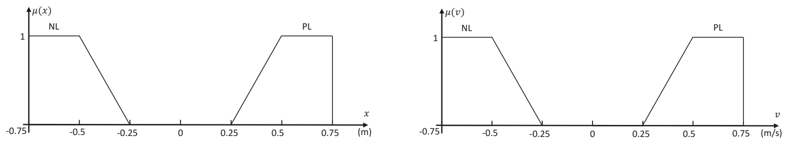
Velocity (v)	**Position (x)**				
	NL	**NS**	**0**	**PS**	**PL**
NL	PL	PL	PL	PS	0
NS	PL	PL	PS	0	NS
0	PL	PS	0	NS	NL
PS	PS	0	NS	NL	NL
PL	0	NS	NL	NL	NL

Table 3: Sparse rule base for Type-1 TSK fuzzy model with 4 rules

Velocity (v)	**Position (x)**	
	NL	**PL**
NL	PL	0
PL	0	NL

In order to evaluate the TSK+ fuzzy inference approach working with a sparse rule base, three fuzzy sets from each domain were artificiality removed from the above example to demonstrate an extremely sparse rule base, as shown in Figure 4. As a results, only two boundary fuzzy sets of each input domain were kept, which is composed to four rules, as listed in Table 3.

Given an initial state of the cart, $x = 0.5m$ and $v = 0.5m/s$, the conventional TSK and TSK+ fuzzy inference approaches were both employed on the dense and sparse rule base, if applicable, for system performance comparison. The results are shown in the fist and second column of Figure 6.



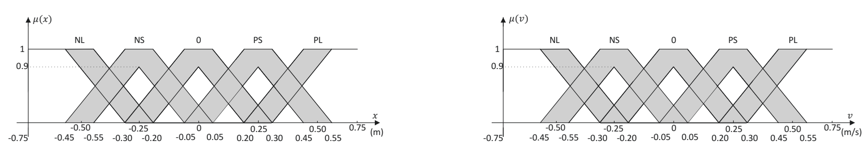
(a) Kept fuzzy values for Position (b) Kept fuzzy values for Velocity

Fig. 4: Fuzzy partition on input domain

In particular, Figure 5(a) and 5(b) demonstrated the results that produced by conventional TSK with dense rule base; the results generated by TSK+ with dense rule base are shown in Figure 5(e) and 5(f); and Figure 5(i) and 5(j) illustrated the results that obtained by TSK+ with only 4 boundary rules.

4.3 Type-2 Fuzzy Control

In this experiment, the IT2 TSK fuzzy model was implemented. The fuzzy rule bases detailed in Section 4.2 were used in this section, but all fuzzy sets were changed to IT2 fuzzy sets instead of the type-1 fuzzy sets, which are expressed in Figure 5. In addition, instead of using crisp values, five crisp interval values were employed as the output as listed in Table 4. From here, this cart centring problem can be solved by an IT2 0-order TSK fuzzy model with a dense rule base that composed of 25 rules, as shown in Table 2. Again, three fuzzy variables from each input domain were manually removed to simulate the situation of lack of information for comparison purpose. In the same the situation as described in Section 4.2, only two boundary fuzzy sets, NL and PL, were kept on each input domain to construct an extremely sparse IT2 0-order rule base, which only contains 4 rules, as shown in Table 3.



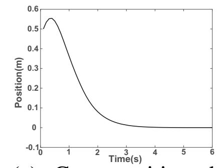
(a) IT2 membership for the domain of Position (b) IT2 membership function for the domain of Velocity

Fig. 5: Input membership functions

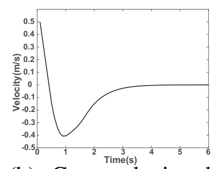
Table 4: Fuzzy partition of output domain

Output label	**Value**	**Linguistic value**
NL	[-0.18 -0.14]	NL
NS	[-0.10 -0.06]	NS
0	[-0.02 0.02]	0
PS	[0.06 0.10]	PS
PL	[0.14 0.18]	PL

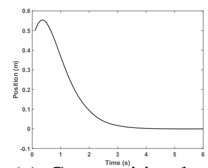
The simulated results of employing the conventional IT2 TSK and IT2 TSK+ approaches over the dense and sparse rule base, if applicable, for a given initial state of

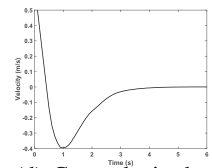


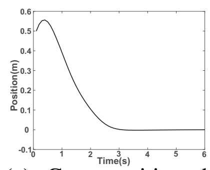
(a) Cart position by conventional TSK with dense rule base



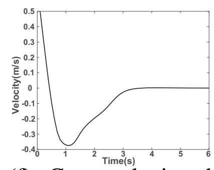
(b) Cart velocity by conventional TSK with dense rule base


(c) Cart position by conventional IT2 TSK with dense rule base

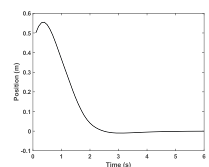

(d) Cart velocity by conventional IT2 TSK with dense rule base

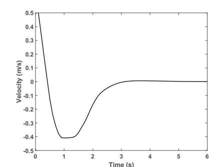


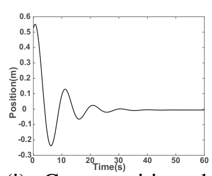
(e) Cart position by TSK+ with dense rule base



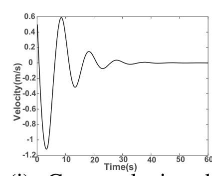
(f) Cart velocity by TSK+ with dense rule base


(g) Cart position by IT2 TSK+ with dense rule base

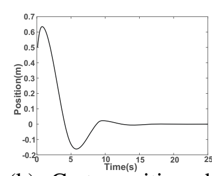

(h) Cart velocity by IT2 TSK+ with dense rule base



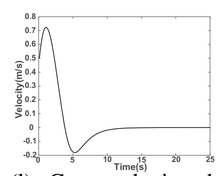
(i) Cart position by TSK+ with sparse rule base



(j) Cart velocity by TSK+ with sparse rule base



(k) Cart position by IT2 TSK+ with sparse rule base



(l) Cart velocity by ITS TSK+ with sparse rule base

Fig. 6: Performance comparison

the cart, $x = 0.5m$ and $v = 0.5m/s$, are illustrated in the third and fourth column of Figure. 6.

The experimental results show that all four approaches (conventional TSK, TSK+, conventional IT2 TSK and IT2 TSK+) performed well in controlling the cart to the target position from the given initial state. From Figure 5(g), it is clear that the IT2 TSK+ with the dense rule base took less time (around 2.5 seconds) to drive the cart to the target state from the initial state with relatively smooth control, compared with the performances produced by other three approaches (around 3 seconds), based on the dense rule base. In term of controlling over the sparse rule base, although the TSK+ and IT2 TSK+ approaches took much longer to drive the cart to the goal position, around 35 seconds and 17 seconds, respectively; however, considering only 4 boundary rules were applied instead of a dense rule base with 25 rules, such performance indicates the power of both TSK+ and IT2 TSK+ approaches in reasoning from incomplete knowledge and system complexity reduction.

5 Conclusion

This paper systematically reviews four different types of TSK-based fuzzy inference approaches, including the convention TSK, TSK+, IT2 TSK and IT2 TSK+, in terms of their effectiveness for control problems. Compared with the conventional TSK and IT2 TSK approaches, which are only workable with dense rule bases, the TSK+ and IT2 TSK+ are applicable to both dense and sparse rule bases significantly increasing the applicability of the TSK-based fuzzy inference systems. The experimental results demonstrate the power of the TSK+ and IT2 TSK+ in mobile cart control.

For future works, more real-world applications, such as truck backer-upper control [5], navigation of autonomous mobile robot control [27], powered exoskeleton control [28], and robotic control [29], will be considered for more thorough evaluation of the approach. And then, it is worthwhile to compare the performance between the conventional Mamdani inference approach and the Mamdani-based fuzzy interpolation approaches. In addition, it is interesting to investigate how the sparse rule base generation approaches, such as [30, 31], can be applied to help generate sparse rule bases, and thus more compact TSK fuzzy models.

References

1. E. H. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *Computers, IEEE Transactions on*, C-26(12):1182–1191, 1977.
2. T. Takagi and M. Sugeno. Fuzzy identification of systems and its applications to modeling and control. *Systems, Man and Cybernetics, IEEE Transactions on*, SMC-15(1):116–132, 1985.
3. J. Li, L. Yang, Y. Qu, and G. Sexton. An extended takagi–sugeno–kang inference system (tsk+) with fuzzy interpolation and its rule base generation. *Soft Computing*, 22(10):3155–3170, 2018.
4. L. T. Kóczy and K. Hirota. Approximate reasoning by linear rule interpolation and general approximation. *International Journal of Approximate Reasoning*, 9(3):197–225, 1993.
5. Z. Huang and Q. Shen. Fuzzy interpolation and extrapolation: A practical approach. *Fuzzy Systems, IEEE Transactions on*, 16(1):13–28, 2008.
6. L. Yang, F. Chao, and Q. Shen. Generalized adaptive fuzzy rule interpolation. *IEEE Transactions on Fuzzy Systems*, 25(4):839–853, Aug 2017.
7. L. Yang and Q. Shen. Adaptive fuzzy interpolation. *Fuzzy Systems, IEEE Transactions on*, 19(6):1107–1126, Dec 2011.
8. L. Yang and Q. Shen. Closed form fuzzy interpolation. *Fuzzy Sets and Systems*, 225:1–22, 2013. Theme: Fuzzy Systems.
9. N. Naik, R. Diao, and Q. Shen. Dynamic fuzzy rule interpolation and its application to intrusion detection. *IEEE Transactions on Fuzzy Systems*, PP(99):1–1, 2017.
10. S. M. Chen, X. Y. Zou, and D. Barman. Adaptive weighted fuzzy rule interpolation based on ranking values and similarity measures of rough-fuzzy sets. *Information Sciences*, 488:93 – 110, 2019.
11. S. M. Chen and S. I. Adam. Adaptive fuzzy interpolation based on ranking values of interval type-2 polygonal fuzzy sets. *Information Sciences*, 435:320 – 333, 2018.
12. J. Li, Y. Qu, H. P. H. Shum, and L. Yang. Tsk inference with sparse rule bases. In *Advances in Computational Intelligence Systems*, pages 107–123, Cham, 2017. Springer.

13. S. Yasunobu, S. Miyamoto, and H. Ihara. Fuzzy control for automatic train operation system. *IFAC Proceedings Volumes*, 16(4):33 – 39, 1983. 4th IFAC/IFIP/IFORS Conference on Control in Transportation Systems , Baden-Baden, FRG, 20-22 April 1983.
14. J. Li, L. Yang, H. P. H. Shum, G. Sexton, and Y. Tan. Intelligent home heating controller using fuzzy rule interpolation. In *15th UK Workship on Computational Intelligence, UKCI*, 2015.
15. L. Yang, J. Li, P. Hackney, F. Chao, and M. Flanagan. Manual task completion time estimation for job shop scheduling using a fuzzy inference system. In *10th IEEE International Conference on Cyber, Physical and Social Computing*, 2017.
16. L. Yang, J. Li, F. Chao, P. Hackney, and M. Flanagan. Job shop planning and scheduling for manufacturers with manual operations. *Expert Systems*, 2018.
17. N. Elisa, J. Li, Z. Zuo, and L. Yang. Dendritic cell algorithm with fuzzy inference system for input signal generation. In *UK workshop on computational intelligence*, pages 203–214. Springer, 2018.
18. Z. Zuo, J. Li, P. Anderson, L. Yang, and N. Naik. Grooming detection using fuzzy-rough feature selection and text classification. In *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE, 2018.
19. Z. Zuo, J. Li, B. Wei, L. Yang, F. Chao, and N. Naik. Adaptive activation function generation for artificial neural networks through fuzzy inference with application in grooming text categorisation. In *2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 2019.
20. X. Fu, X. J. Zeng, D. Wang, D. Xu, and L. Yang. Fuzzy system approaches to negotiation pricing decision support. *Journal of Intelligent & Fuzzy Systems*, 29(2):685–699, 2015.
21. J. Li, L. Yang, X. Fu, F. Chao, and Y. Qu. Dynamic qos solution for enterprise networks using tsk fuzzy interpolation. In *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–6. IEEE, 2017.
22. G. Gürsel. Healthcare, uncertainty, and fuzzy logic. *Digital Medicine*, 2(3):101, 2016.
23. L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338 – 353, 1965.
24. D. Wu and M. Nie. Comparison and practical implementation of type-reduction algorithms for type-2 fuzzy sets and systems. In *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, pages 2131–2138, June 2011.
25. J. Li, L. Yang, X. Fu, F. Chao, and Y. Qu. Interval type-2 tsk+ fuzzy inference system. In *2018 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 1–8. IEEE, 2018.
26. N. Enyinna, A. Karimoddini, D. Opoku, A. Homaifar, and S. Arnold. Developing an interval type-2 tsk fuzzy logic controller. In *2015 Annual Conference of the North American Fuzzy Information Processing Society and the 5th World Conference on Soft Computing*, 2015.
27. N. Baklouti, R. John, and A. M. Alimi. Interval type-2 fuzzy logic control of mobile robots. *Journal of Intelligent Learning Systems and Applications*, 4(04):291, 2012.
28. K. Yin, K. Xiang, M. Pang, J. Chen, P. Anderson, and L. Yang. Personalised control of robotic ankle exoskeleton through experience-based adaptive fuzzy inference. *IEEE Access*, 2019. doi: 10.1109/ACCESS.2019.2920134.
29. F. Chao, D. Zhou, C.M. Lin, L. Yang, C. Zhou, and C. Shang. Type-2 fuzzy hybrid controller network for robotic systems. *IEEE Transactions on Cybernetics*, 2019. doi: 10.1109/TCYB.2019.2919128.
30. J. Li, H. P. H. Shum, X. Fu, G. Sexton, and L. Yang. Experience-based rule base generation and adaptation for fuzzy interpolation. In *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 102–109. IEEE, 2016.
31. Y. Tan, J. Li, M. Wonders, F. Chao, H. P. H. Shum, and L. Yang. Towards sparse rule base generation for fuzzy rule interpolation. In *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 110–117. IEEE, 2016.