A Fuzzy Clustering Algorithm for Developing Predictive Models in Construction Applications

Nima Gerami Seresht[1], Ph.D., Rodolfo Lourenzutti[2], Ph.D., and
Aminah Robinson Fayek[3], Ph.D., P Eng., (corresponding author)

Abstract

Fuzzy inference systems (FISs) are a predictive modeling technique based on fuzzy sets that utilize approximate reasoning to mimic the decision-making process of human experts. There are several expert- and data-driven methods for developing FISs, among which fuzzy clustering algorithms are the most frequently used data-driven methods. This paper introduces a new fuzzy clustering algorithm for developing FISs in construction applications that addresses two limitations of existing fuzzy clustering algorithms: the lack of capacity to determine the number of clusters automatically from the characteristics of the data, and the poor performance in predictive modeling of highly dimensional problems. Existing fuzzy clustering algorithms are limited in construction applications since determining the number of clusters based on subjective expert judgement reduces the accuracy of the resulting FIS, and construction systems are often highly dimensional with a large number of inputs affecting the system outputs. The fuzzy clustering algorithm proposed in this paper determines the number of clusters automatically based on the characteristics of the data, specifically the non-linearity observed within clusters, and assigns weights to the rules

[1] Postdoctoral fellow, Department of Civil & Environmental Engineering, 7-203 Donadeo Innovation Centre for Engineering, 9211 116 St NW, University of Alberta, Edmonton AB T6G 1H9, Canada.

[2] Postdoctoral fellow, Department of Civil & Environmental Engineering, 7-203 Donadeo Innovation Centre for Engineering, 9211 116 St NW, University of Alberta, Edmonton AB T6G 1H9, Canada.

[3] Director of the Construction Innovation Centre, Tier 1 Canada Research Chair in Fuzzy Hybrid Decision Support Systems for Construction, NSERC Industrial Research Chair in Strategic Construction Modeling and Delivery, Ledcor Professor in Construction Engineering, Professor, Department of Civil & Environmental Engineering, 7-287 Donadeo Innovation Centre for Engineering, 9211 116 St NW, University of Alberta, Edmonton AB T6G 1H9, Canada.

of FISs to improve their accuracy in highly dimensional problems. This paper advances the state-of-the-art of fuzzy clustering and contributes to construction modeling by providing a new data-driven technique for developing FISs that suits the characteristics of construction problems.

Keywords: Predictive modeling; fuzzy inference systems; fuzzy clustering; machine learning; construction modeling

1. Introduction

The introduction of fuzzy sets [1] provided an alternative way to address uncertainties originating from the subjectivity, imprecision, or linguistic expression of information (i.e., non-probabilistic uncertainties). Moreover, the introduction of fuzzy sets offered a means of addressing two challenges involved with modeling construction systems. First, many variables influencing construction systems exhibit non-probabilistic uncertainty (e.g., construction projects, construction operations, etc.), including those variables assessed by linguistic terms (e.g., high crew motivation). Second, the modeling of construction systems is hindered by limited data availability in construction contexts. The application of fuzzy sets can address these challenges by enabling modelers to acquire and process expert-knowledge where historical data is not available for the system. One of the applications of fuzzy sets in construction problems is in the development of fuzzy inference systems (FIS), which are a type of predictive modeling technique that map inputs to outputs using a fuzzy rule-based system.

There are several expert- and data-driven techniques for developing FISs. Expert-driven techniques rely on experts' knowledge regarding the interactions between the input and output variables, and data-driven techniques rely on historical data to map the input variables to the outputs. Since experts need to understand the structure of the system prior to defining the interactions between input and output variables by a set of rules (i.e., rule base), expert-driven methods can only be applied to those problems that have a small number of input and output variables (i.e., low dimensionality) [2]. According to Zadeh's principle of incompatibility [2], the dimensionality of a system has an inverse relationship with experts' understanding of system structure. Therefore, in highly dimensional construction systems, where system outputs (e.g., productivity) are predicted by a large number of input variables (e.g., factors influencing productivity), FISs developed by expert-driven methods may have poor predictive performance [3,4]. In contrast, data-driven methods rely on historical data to identify a system's structure, which makes them preferable to expert-driven methods for developing FISs in highly dimensional problems if historical data are available.

Among the different data-driven methods introduced in the literature for developing FISs [5–8], fuzzy clustering algorithms are the most commonly used techniques in engineering applications [9–13]. Various fuzzy clustering algorithms have been proposed in the literature for data partitioning and developing FISs, including fuzzy *c*-means (FCM) clustering [14], Gustafson-Kessel's algorithm (GK algorithm) [15], and subtractive clustering [16]. Existing fuzzy clustering algorithms have two limitations for developing FISs in construction applications. First, the majority of these algorithms do not have the capacity to determine the number of clusters automatically based on the characteristics of the data [4]. Accordingly, modelers need to decide on the number of clusters subjectively, though information about the appropriate number of clusters needed to represent a subjective variable may not be available in many applications of FISs. FCM clustering and GK algorithm both rely on the subjective knowledge of the modeler to specify the number of clusters and therefore disregard the characteristics of the data. Moreover, subtractive clustering determines the number of clusters based on the subjective judgment of the modeler regarding the minimum distance between two given cluster centers. Thus, subtractive clustering ignores the existence of non-linearity within each cluster, despite the fact that it is an important characteristic of the data, which can reduce the accuracy of the resulting FIS. Second, fuzzy clustering algorithms lack the capacity to assign weights to the rules of FISs, such that all input and output variables are equally weighted in the resulting FISs [17]. This issue decreases the accuracy of the resulting FISs in highly dimensional problems [17]. These two limitations are addressed in the present work through the development of a new fuzzy clustering algorithm. The proposed algorithm advances the state of the art of fuzzy clustering by determining the number of clusters automatically based on the non-linearity observed within clusters and assigning weights to the rules of FISs to improve their accuracy in highly dimensional problems. In the proposed fuzzy clustering algorithm, the number of clusters is determined by a novel iterative algorithm that increases the number of clusters by one in each iteration to reduce the amount of non-linearity within each cluster below a prespecified threshold. Moreover, Adam optimization, a gradient-based optimization algorithm with several applications in machine learning [18], is used to assign weights to the rules of the FISs. This paper also contributes to the existing body of knowledge on construction modeling by introducing a new data-driven method for developing FISs that is appropriate for modeling highly dimensional construction systems.

The remainder of the paper is organized as follows: Section 2 presents a brief review of the applications of fuzzy clustering algorithms in the construction domain, in addition to discussing FCM clustering and GK algorithm. Section 3 presents the proposed fuzzy clustering algorithm for developing predictive models in construction applications. Section 4 presents a numerical example to illustrate the proposed fuzzy clustering algorithm, and Section 5 tests the applicability of the proposed algorithm to construction problems by modeling construction labor productivity (CLP).

Finally, Section 6 presents final remarks and explores areas for future research. To improve the readability of this paper, the nomenclature of symbols used in the paper are presented in Table 1.

Table 1. Nomenclature of symbols.

Symbol	Description
$U = [u_{ij}]$	Partition matrix
$v_i^T = (v_{i1}, \dots, v_{ip})$	The centroid of cluster *i*
$A_i = \left[a_{kl}^{(i)}\right]_{n\times n}$	Norm-inducing matrix for cluster *i*
ρ_i	Volume constraints for norm-inducing matrix of cluster *i*
$\mu_{v,A}^{(i)}(z_j)$	Membership degree of a point z_j in cluster *i*
$h_i(.)$	The link function of rule *i*
$f(.)$	The link function of FIS
λ	The percentage of the data points located in cluster tails
$\lceil . \rceil$	Ceiling function
β_i	First parameter for linear state function of cluster *i*
α	Angle between two linear state functions
δ	Intra-cluster non-linearity threshold
$\omega \epsilon (0,1)$	Step size
$\varphi_1, \varphi_2 \in (0,1]$	Exponential decay rates for momentum estimates
ϵ	Numerical stabilization constant
m, v	Momentum vectors

2. Related Work

There are two main types of FISs, Mamdani FISs (M-FIS) [19,20] and Takagi-Sugeno FISs (TS-FIS) [21]. M-FISs use fuzzy membership functions to represent the input and the output variables of the system, which results in prediction of system outputs as fuzzy sets rather than crisp numbers. In the case of M-FISs, a defuzzification step is often required to determine the system output as a crisp number, since decisions are often made in practice based on crisp numbers rather than fuzzy sets. Since there are a wide range of defuzzification operators, the selection of a proper operator can significantly affect the accuracy of the system. In contrast, TS-FISs use a set of crisp functions of inputs (i.e., state functions) to predict the output of the system. In the case of TS-FISs, it was first assumed that the state functions were linear in nature [21]. Later, a general form of TS-FISs was introduced, in which the state functions can be any non-linear local model. In current applications of TS-FISs, state functions are often k-order polynomial functions [22]. By using a set of local models, TS-FISs are able to capture the complexity of construction systems with high accuracy and predict their behavior with robust calculation efficiency [3]. Moreover, since the state functions of TS-FISs are crisp functions, their outputs are predicted as crisp numbers and no defuzzification is required. Fuzzy clustering algorithms are capable of developing both M-FISs and TK-FISs using historical data. Given the aforementioned advantages of TS-FISs over M-FISs, the new fuzzy clustering algorithm introduced in this paper is focused on developing TS-FISs.

Clustering algorithms are traditionally used to create classes of data based on their similarities [23]. Unlike crisp clustering algorithms, fuzzy clustering algorithms can also be used for developing predictive models (i.e., FISs) by projecting fuzzy clusters into the input and output spaces [24,25]. Fuzzy clustering algorithms for predictive modeling have been applied in a number of engineering contexts, including civil engineering and construction engineering and management. Examples include FISs for controlling pendulum cranes [26], aircraft motion control models [27], stock trading forecasts [28], predicting the progress rate of road headers (i.e., an automated tunneling machinery) in tunneling projects [3], predicting the penetration index of tunnel boring machines (TBM) [29], and predicting CLP [4]. The use of fuzzy clustering algorithms in these applications allows the modeler to capture the non-probabilistic uncertainty of the system variables (i.e., input and output variables). These algorithms also enable the modeling of complex and non-linear relationships between the inputs and outputs effectively and accurately using a number of local functions [3,4,29].

Despite the extended use of fuzzy clustering algorithms in engineering applications, the reliance of these algorithms on user-defined parameters (e.g., number of clusters) can limit their applications, since the optimum values for such parameters may not be known by the modeler. To address this limitation, efforts have been made to determine the optimum value of user-defined parameters, either manually [3,4,29–31], or automatically, by combining fuzzy clustering algorithms with evolutionary optimization algorithms [32,33]. However, the manual optimization of such parameters may not always lead to optimum value, since the optimization process relies on the subjective judgment of modelers. Combining fuzzy clustering algorithms with evolutionary algorithms can also add significant computational costs to the modeling process [25]. These algorithms also have another shortcoming, in that fuzzy clustering algorithms often weight all input and output variables of the system equally. As the dimensionality of the system increases, the accuracy of the FISs developed by these algorithms decreases [17]. To remedy this issue, weights need to be assigned to the rules of FISs [32]. Since the introduction of traditional fuzzy clustering algorithms, such as FCM clustering in 1984 [14], GK algorithm in 1979 [15], and subtractive clustering in 1994 [16], efforts have been made to remedy the limitations of these algorithms and improve their accuracy. Some of the recent efforts are discussed in this section. For a more comprehensive review of common fuzzy clustering algorithms, the reader may refer to [34,35].

New fuzzy clustering algorithms to consider the data characteristics for developing FISs and reducing the reliance of the algorithms on user-defined parameters have been recently proposed in the literature [25,31,36–39]. Askari [25] introduced a fuzzy clustering algorithm based on FCM clustering technique that visualizes the data structure prior to clustering and locates the cluster centers inside the dense areas of the input space, thus improving the interpretability of FISs and avoiding redundancy in the rule base. Since the projections of clusters on each input/output space are normal but non-convex and irregularly shaped fuzzy membership functions, Askari [25] transforms the resulting membership functions to a Gaussian shape for improved interpretability of the FISs. There are also a variety of fuzzy clustering algorithms based on FCM clustering and possibilistic theory, which combine the concepts of entropy, typicality, and belongingness with traditional FCM clustering to avoid the high impact of noisy data on the results. Generally, in possibilistic fuzzy clustering algorithms, the impact of the noisy data on developing clusters is reduced by changing the distance function in FCM clustering to help identify the noisy data points and reduce their membership values in all clusters [37]. Examples of such fuzzy clustering algorithms are possibilistic *c*-means (PCM) clustering [39], fuzzy possibilistic *c*-means (FPCM) clustering [40], possibilistic fuzzy *c*-means (PFCM) clustering [38], and the generalized possibilistic fuzzy *c*-means (GPFCM) clustering introduced by Askari et al [37]. Previous studies show that considering the different characteristics of the data, such as the density of the data on the input space or noise in the data, can improve the performance of fuzzy clustering algorithms. In a similar manner, this paper introduces the use of another data characteristic, the non-linearity of data, to improve the accuracy and efficiency of fuzzy clustering algorithms by automatically detecting the number of clusters. The algorithm proposed in this paper can improve the accuracy of fuzzy clustering algorithms, since the non-linearity of data is an important characteristic when predicting the complex and non-linear behavior of systems using a set of linear state functions in TS-FISs. Moreover, since fuzzy clustering algorithms are computationally expensive, especially in highly dimensional problems, determining the number of clusters automatically avoids multiple runs of the algorithm and improves its efficiency.

Previous research has attempted to address the second limitation of fuzzy clustering algorithms, assigning equal weights to all the rules of the resulting FISs, which leads to low accuracy of these algorithms for predicting the behavior of highly dimensional systems. Various optimization techniques have been used to determine the optimum rule weights for FISs, including heuristic search methods [41], evolutionary and artificial swarm optimization algorithms [42], and gradient-decent algorithms [43]. In this paper, Adam optimization is used to determine the optimum rule weights of FISs. Due to its high computational efficiency, Adam optimization is well-suited for handling highly dimensional problems (e.g., modeling construction systems) or problems with a large number of data points [18]. Accordingly, the use of Adam optimization improves the accuracy of FISs in highly dimensional problems and with less computational cost, as compared to the use of evolutionary or artificial swarm optimization techniques [42].

Though there are a number of fuzzy clustering algorithms that have been introduced in the literature, FCM clustering is among the most commonly used in engineering applications [44]. Similar to other fuzzy clustering algorithms, FCM clustering allows a point to simultaneously belong to different clusters at different degrees of membership; therefore, clusters might have non-sharp boundaries. The sharpness of the boundaries of clusters is determined by the modeler using a parameter called the fuzzification coefficient $m \in (1, \infty)$, where the value of m has an inverse correlation with the sharpness of the boundaries of a cluster [14]. Moreover, the modeler must specify the number of clusters as an integer number C, where $2 \leq C < n$ and n stands for the sample size of the input data. Once the two aforementioned parameters are determined for the sample of $x = \{x_1, \ldots, x_n\}$, where $x_i^T = (x_{i1}, \ldots, x_{ip})$, $i = 1, \ldots, n$, where n represents the sample size and p stands for the dimension of the input data, the FCM clustering algorithm seeks to minimize the objective function, which is presented below in Equation 1.

$$\min J_m(U,\nu) = \sum_{i=1}^{C}\sum_{j=1}^{n} u_{ij}^{m}\left(x_j - \nu_i\right)^T B\left(x_j - \nu_i\right) \qquad 1$$

where B stands for a positive-definite matrix, which is fixed for all clusters; $\nu^T = (\nu_1, \dots, \nu_C)$ stands for the vector of the centroids; and $\nu_i^T = \left(\nu_{i1}, \dots, \nu_{ip}\right)$ is the centroid of cluster i. Finally, $U = [u_{ij}]$ is the partition matrix satisfying the following three conditions.

$u_{ij} \in [0,1], \quad \forall i = 1, \dots, C$ and $j = 1, \dots, n;$

$$\sum_{i=1}^{C} u_{ij} = 1, \quad \forall j = 1, \dots, n;$$

$0 < \sum_{j=1}^{n} u_{ij} < n, \ \forall i = 1, \dots, C.$

By using a fixed positive-definite matrix for all clusters (referring to B in Equation 1), FCM clustering ignores the fact that different clusters may have different structures (e.g., dispersion of data within each cluster). In order to capture the variant of the structures of different clusters, the GK algorithm, can be used. GK algorithm is the generalized form of the FCM clustering algorithm, and it allows for the use of different norm-inducing matrices A_i for different clusters. The objective function of GK algorithm is defined as shown in Equation 2.

$$\min J_m(U,\nu,A) = \sum_{i=1}^{C}\sum_{j=1}^{n} u_{ij}^{m}\left(x_j - \nu_i\right)^T A_i\left(x_j - \nu_i\right) \qquad 2$$

where $A = (A_1, \dots, A_C)$ is the vector of the norm-inducing matrices, and $A_i = \left[a_{kl}^{(i)}\right]_{n\times n}$ is a positive-definite matrix associated with cluster i. The fuzzy clustering algorithm proposed in this paper is based on GK algorithm, where the initial form of the clusters and the FIS is determined by this algorithm. The cluster number and rule weights are determined using the algorithms presented in Section 3.

3. Proposed Fuzzy Clustering Algorithm for Predictive Modeling

This section discusses the proposed fuzzy clustering algorithm for developing FISs in construction applications. Figure 1 presents the flowchart of the proposed fuzzy clustering algorithm, which consists of three major steps: (1) generating the FIS from data with GK algorithm; (2) determining the number of clusters; and (3) assigning rule weights using Adam optimization. The three steps of the methodology are further discussed in the following subsections.

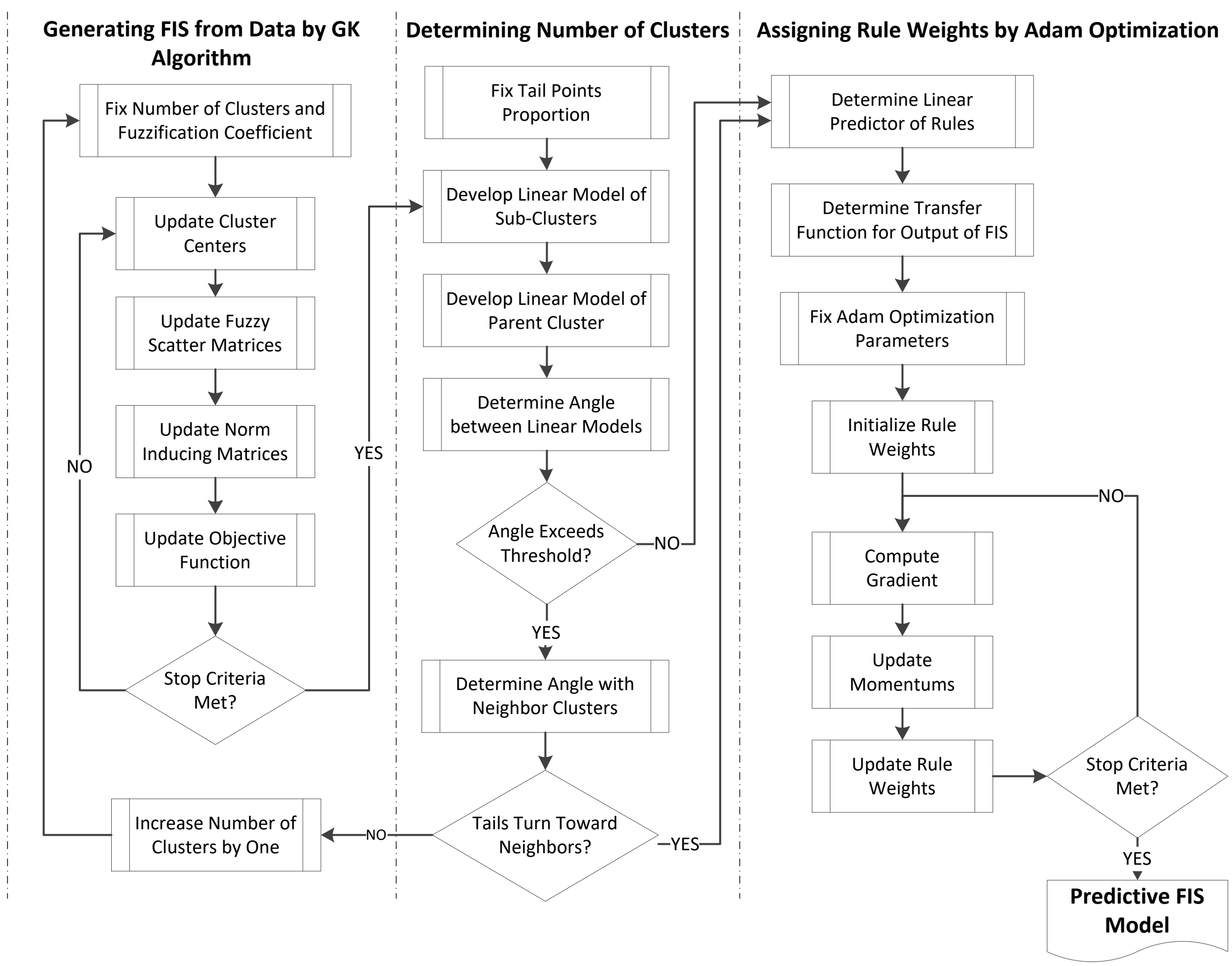

Figure 1. Flowchart of the proposed methodology for developing the FIS.

3.1. *Generating Fuzzy Inference System from Data Using Gustafson-Kessel's Algorithm*

The initial FIS was developed by clustering the sample data using GK algorithm [15], where the number of clusters is set to its minimum, $C = 2$, and the fuzzification coefficient $m \in (1, \infty)$ is specified by the modeler. It is notable that for modeling a linear system, in which the relationships between the inputs and outputs are perfectly linear, the number of clusters needs to be set to $C = 1$. However, such linear systems can be modeled more efficiently using statistical regression. In contrast, the fuzzy clustering algorithm proposed in this paper is suitable for modeling complex systems with non-linear relationships between the inputs and outputs. Accordingly, the minimum number of clusters is set to $C = 2$. Next, the clusters were projected into the input and output spaces, and the initial FIS was developed.

To introduce the clustering steps, let z denote the sample data with n data points, which consist of input data $x \in \mathbb{R}^p$ and output data $y \in \mathbb{R}^q$, i.e., $z^T = ((x_1, y_1), \dots, (x_n, y_n))$. Let $z_1 = (x_1, y_1), \dots, z_n = (x_n, y_n)$ be n sample data points with the dimension of $p + q$, where p is the dimension of the inputs and q is the dimension of the outputs. The clustering of the sample data is accomplished through the six following steps:

Step 1. Initialize the partition matrix $U^{(0)}$.

$$U = [u_{ij}], \qquad u_{ij} \in [0,1], \qquad \forall i = 1, \dots, C \text{ and } j = 1, \dots, n;$$

Step 2. Update the cluster centers.

$$v_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} z_j}{\sum_{j=1}^{n} u_{ij}^{m}}, \qquad i = 1, \dots, C$$

Step 3. Update the fuzzy scatter matrices.

$$S_i = \sum_{j=1}^{n} u_{ij}^{m} \left(z_j - v_i\right)\left(z_j - v_i\right)^T$$

Step 4. Update the norm-inducing matrices.

$$A_i = \left[\rho_i \det(S_i)\right]^{\frac{1}{p+q}} S_i^{-1}$$

where ρ_i is the volume constraint for the norm-inducing matrix for cluster i. The volume constraints for the norm-inducing matrices constrain the determinants to constant real numbers (i.e., $\det(A_i) = \rho_i, \rho_i \in \mathbb{R}$).

Step 5. Update U, as shown in Equation 3:

$$u_{ij} = \left[\sum_{c=1}^{C} \left(\frac{\left(z_j - v_i\right)^T A_i \left(z_j - v_i\right)}{\left(z_j - v_c\right)^T A_c \left(z_j - v_c\right)}\right)^{\frac{1}{m-1}}\right]^{-1} \tag{3}$$

Step 6. Repeat Steps 2 through 5 until the stopping criteria are met. Examples of stopping criteria are a maximum number of iterations and small changes in the objective function *J*.

Next, let $d_{ij}^2 = \left(z_j - v_i\right)^T A_i \left(z_j - v_i\right)$. If $\exists i, j \colon d_{ij}^2 = 0$, then Equation 3 is undefined. In such cases, an alternative approach is necessary to obtain the membership degrees, which must satisfy the requirements presented in Equation 4 [14]:

$$\begin{gathered} u_{ij} = 0, \qquad \forall i \colon d_{ij}^2 \neq 0 \\ \sum_{i : d_{ij}^2 = 0} u_{ij} = 1 \end{gathered} \tag{4}$$

The two conditions presented in Equation 4 imply that if there is a point that perfectly matches one or more cluster prototypes, the membership degree of this point is fully shared among these clusters, resulting in zero membership of the point in other clusters. Once GK algorithm converges, the centroids of the final cluster v and the norm-inducing matrices A are used to calculate the membership degree of any data point in the universe of discourse. For a given vector of centroid v and norm-inducing matrices A, the membership degree of a point z_j in cluster i is determined by Equation 5.

$$\mu_{v,A}^{(i)}(z_j) = \left[\sum_{c=1}^{C} \left(\frac{(z_j - v_i)^T A_i (z_j - v_i)}{(z_j - v_c)^T A_c (z_j - v_c)} \right)^{\frac{1}{m-1}} \right]^{-1} \qquad 5$$

By implementing GK algorithm on the sample data z (i.e., considering both inputs and outputs), C clusters will be developed, each of which represent the membership function of one rule. Considering the rule represented by cluster i for a fixed input x^*, the surface $\mu_{v,A}^{(i)}(\mathrm{x}^*, \mathrm{y}) = \mu_{v,A}^{(\mathrm{i})}(z^*)$ as a function of y only, denoted by $\mu_{v,A}^{(i)}(y|x^*)$, is the membership function of the output for rule i and input x^*, where $z^* = (x_1^*, \dots, x_p^*, y_1, \dots, y_q)$. In existing fuzzy clustering algorithms, relationships between the output variables are often ignored, since during the development of rule i, cluster i is projected onto each output axis independently. In the proposed algorithm, such relationships between the outputs were considered, since cluster i is projected onto the whole output space (i.e., all output axes) at once. Next, a monotone differentiable link function is applied to the linear state function of each rule to obtain the rule output.

Once the output of the GK algorithm is produced, the state function value of rule i for an input x, y is defined, such that the point (x, y) is the closest to the centroid of the i^{th} cluster, according to the norm-inducing matrix A_i, where the centroids and the norm-inducing matrices are provided by the GK algorithm. Consider the partitioning of A_i and v_i as presented in Equations 6 and 7, respectively.

$$\begin{bmatrix} \begin{bmatrix} a_{11}^i & a_{12}^i & \cdots & a_{1p}^i \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1}^i & a_{p2}^i & \cdots & a_{pp}^i \end{bmatrix} & \begin{bmatrix} a_{1(p+1)}^i & \cdots & a_{1(p+q)}^i \\ \vdots & \ddots & \vdots \\ a_{p(p+1)}^i & \cdots & a_{p(p+q)}^i \end{bmatrix} \\ \begin{bmatrix} a_{(p+1)1}^i & a_{(p+1)2}^i & \cdots & a_{(p+1)p}^i \\ \vdots & \vdots & \ddots & \vdots \\ a_{(p+q)1}^i & a_{(p+q)2}^i & \cdots & a_{(p+q)p}^i \end{bmatrix} & \begin{bmatrix} a_{(p+1)(p+1)}^i & \cdots & a_{(p+1)(p+q)}^i \\ \vdots & \ddots & \vdots \\ a_{(p+q)(p+1)}^i & \cdots & a_{(p+q)(p+q)}^i \end{bmatrix} \end{bmatrix} = \begin{bmatrix} A_x^i & A_{x,y}^i \\ (A_{x,y}^i)^T & A_y^i \end{bmatrix} \tag{6}$$

$$v_i = \begin{bmatrix} v_{i1} \\ \vdots \\ v_{ip} \\ v_{i(p+1)} \\ \vdots \\ v_{i(p+q)} \end{bmatrix} = \begin{bmatrix} v_x^i \\ v_y^i \end{bmatrix}, \qquad z = \begin{bmatrix} x_1 \\ \vdots \\ x_p \\ y_1 \\ \vdots \\ y_q \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} \tag{7}$$

Then,

$\theta_i(x) = \arg_{\mathrm{y}} \min\ (z - v_i)^T A_i\ (z - v_i)$

$= \arg_y \min \left[\left(d_x^{(i)}\right)^T A_x^{(i)} d_x + \left(d_y^{(i)}\right)^T \left(A_{x,y}^{(i)}\right)^T d_x + \left(d_x^{(i)}\right)^T \left(A_{x,y}^{(i)}\right) d_y + \left(d_y^{(i)}\right)^T \left(A_y^{(i)}\right) d_y \right]$

$= v_y^{(i)} - \left(A_y^{(i)}\right)^{-1} \left(A_{x,y}^{(i)}\right)^T \left(x - v_x^{(i)}\right)$

where $d_x^{(i)} = \left(x - v_x^{(i)}\right)$; $d_y^{(i)} = \left(y - v_y^{(i)}\right)$, and $\theta_i(x)$ is the value of the linear state function of rule *i* for input x. The outputs of rule *i* are given by Equation 8.

$$\hat{y}_x^{(i)} = h_i\big(\theta_i(x)\big) \tag{8}$$

where $h_i: \mathbb{R}^q \to \mathbb{R}^q$, $i = 1, \ldots, C$, is a monotone differentiable function of any desired form. Once the output of all rules for an input *x* are determined, the output of the FIS is calculated, as shown in Equation 9.

$$\hat{y}_x = f\left(\frac{1}{\sum_{i=1}^{C} w_i \mu_{v_x,A_x}^{(i)}(x)} \sum_{i=1}^{C} \hat{y}_x^{(i)} w_i \mu_{v_x,A_x}^{(i)}(x)\right) \quad 9$$

where $\mu_{v_x,A_x}^{(i)}(x)$ is the membership value of point x in cluster i, which is determined based on its projection onto the input space (i.e., removing the coordinates of the output), and $w_i > 0$ is the weight of rule *I,* such that $\sum_{i=1}^{C} w_i = 1$. The clustering of the sample data and projection of the clusters onto the input and output spaces can be accomplished through the steps discussed in this section. However, determining the optimum number of clusters C is still challenging, as the optimum number of clusters need to be determined based on the characteristics of the sample data. A methodology is presented in the next section, which will help to address this challenge.

3.2. Determining the Number of Clusters

The proposed fuzzy clustering algorithm uses hierarchical clustering in order to determine the optimum number of clusters [45]. Fuzzy clustering is initiated with the minimum number of clusters $C = 1$, then those clusters where non-linearity is observed are divided into subclusters. Hierarchical clustering [45] refers to those fuzzy clustering algorithms where clusters are subdivided in order to improve a certain performance index. Let $\boldsymbol{z}_1, \boldsymbol{z}_2, \dots, \boldsymbol{z}_n$ be a set of training data and let C be the number of clusters. For a specific cluster c, $c \in \{1, \dots, C\}$, let $\boldsymbol{z}_1^{(c)}$, $\boldsymbol{z}_2^{(c)}, \dots, \boldsymbol{z}_{n_c}^{(c)}$ be all the training points that belong to cluster c, (i.e., $\max_j \mu^{(j)}\left(\boldsymbol{z}_i^{(c)}\right) = \mu^{(c)}\left(\boldsymbol{z}_i^{(c)}\right)$, $i = 1, \dots, n_c$). Next, the points are ordered based on their distance to the cluster centroid. Let $\boldsymbol{z}_{(1)}^{(c)}, \boldsymbol{z}_{(2)}^{(c)}, \dots, \boldsymbol{z}_{(n_c)}^{(c)}$ denote the training data ordered in such a way that $\left\|\boldsymbol{z}_{(1)}^{(c)} - \boldsymbol{v}_c\right\| > \left\|\boldsymbol{z}_{(2)}^{(c)} - \boldsymbol{v}_c\right\| > \cdots > \left\|\boldsymbol{z}_{(n_c)}^{(c)} - \boldsymbol{v}_c\right\|$. The sample is then restricted by discarding the points that are closest to the centroid of cluster c, such that the points in the tails of the cluster are retained. Let

$\boldsymbol{z}_{(1)}^{(c)}, \boldsymbol{z}_{(2)}^{(c)}, \ldots, \boldsymbol{z}_{(\lceil n_c \rceil)}^{(c)}$ be the restricted sample for the cluster c with $\lceil \lambda n_c \rceil$ points that have the lowest membership degrees, where $\lambda \in (0,1)$ is the percentage of the data points located in the tails of the cluster, the value of which is specified by the user, and $\lceil . \rceil$ is the ceiling function. Next, GK algorithm is applied on the restricted sample data, where the number of clusters is fixed to $C = 2$ to obtain the clusters of the tails. The process of developing the restricted sample data and the clustering of the data points located on the two tails of each cluster is further illustrated using a numerical example presented by Ren & Irwin [9]. Let $y = h(x) = sin(1.6x^3 - 4x^2 + 1)$ be the function to be approximated, and suppose a sample size of 41 is given as $z_1 = \big(-1, h(-1)\big)$, $z_2 = \big(-0.95, h(-0.95)\big), \ldots, z_{41} = (1, h(1))$. Next, using the GK algorithm, a TS-FIS is developed to approximate the function $h(x) = sin(1.6x^3 - 4x^2 + 1)$, where $C = 4$ and $m = 2$. Figure 2 presents the scatter plot of the sample data and the resulting FIS developed by the GK algorithm.

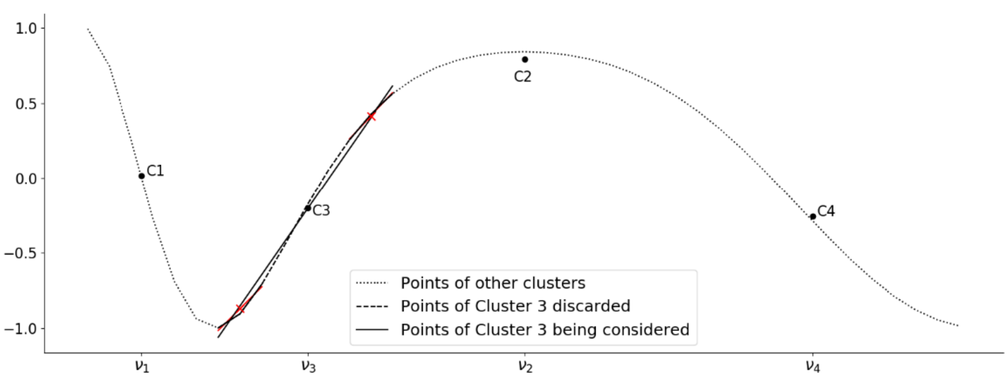

Figure 2. Subdivision of cluster 3 for non-linearity (centroids of subclusters are shown in red).

Next, to determine if cluster C_3 needed to be divided into two clusters, the non-linearity within this cluster (i.e., C_3) was tested by comparing the linear model of each subcluster to the linear model of the parent cluster. Let $\beta_1^{(c)}$ and $\beta_2^{(c)}$ be the parameters of the linear models of the first and second subcluster of cluster c, and let $\beta^{(c)}$ be the parameters of the linear model of the parent cluster,

cluster c. The angle between the linear models of the parent clusters and the subclusters 1 and 2 ($\alpha_i^{(c)}$) can be calculated using Equation 10.

$$\cos\left(\alpha_i^{(c)}\right) = \frac{\left(\beta_i^{(c)}\right)^T \beta^{(c)}}{\left\|\beta_i^{(c)}\right\| \left\|\beta^{(c)}\right\|}, \qquad i = 1, 2 \tag{10}$$

A large angle (e.g., $\alpha_i^{(c)} > 45^\circ$) between the linear models of the subclusters and the linear model of the parent cluster indicates the presence of non-linearity within that cluster. Accordingly, the parent cluster must be divided into two clusters by increasing the number of clusters C by one and re-implementing GK algorithm (refer to Section 3.1). However, small angles between linear models indicate that a low degree of non-linearity exists within the cluster, which can be modelled by the smooth transition between the clusters. Therefore, if $\alpha_1^{(c)} > \delta$ and/or $\alpha_2^{(c)} > \delta$, where δ is the threshold, there is evidence of non-linearity, and the cluster must be divided into two. In special cases, if the parameter vector of the subcluster is rotating towards the model in the closest neighboring cluster, the cluster will not be divided, even if $\alpha_1^{(c)} > \delta$, since the FIS is able to smoothly transition between rules and is in turn able to model the non-linear region well. It is worth noting that if the number of data points in a cluster is too small, the approach of dividing clusters into subclusters becomes unstable, since the linear models are not representative of the data points. Therefore, the number of points in each cluster should be monitored before the subdivision to make sure that the two linear models of the subclusters are truly representative of the data points in the tails of the cluster. Once the optimum number of clusters is determined using the methodology presented in this section, rule weights must be adjusted.

3.3. Assigning Rule Weights by Adam Optimization

The adjustment of rule weights is critical to the development of FISs in construction applications, since it can improve the accuracy of the system in highly dimensional problems. Fuzzy clustering algorithms naturally weight all the input and output variables equally. In the case of multi-dimensional problems, this equal weighting variables can decrease the accuracy of FISs, due to the underweighting of the output variables. This problem can be remedied by assigning weights to the rules of the FISs. Moreover, assigning weights to rules becomes especially critical in applications of GK algorithm, due to the use of different norm-inducing matrices. The use of different norm-inducing matrices for clusters allows the development of clusters with distinctive structures that capture the characteristics of the sample data more accurately. However, these structures result in issues when they are projected onto input and output spaces for the development of FISs, as those clusters with the highest dispersion of data on the input space will dominate the output function of the model. To illustrate this problem, consider the example discussed by Ren & Irwin [9]. Let $y = h(x) = sin(1.6x^3 - 4x^2 + 1)$ be the function to be approximated, and suppose we have a sample size of 41 given as $z_1 = (-1, h(-1))$, $z_2 = (-0.95, h(-0.95)), \ldots, z_{41} = (1, h(1))$. The results of GK algorithm for $C = 4$ and $m = 2$ are shown below in Tables 2 and 3. Figure 3 shows the scatter plot of the sample and the clusters' centers.

Table 2. Cluster centers obtained using the GK algorithm.

Cluster	x	y
$\boldsymbol{v_1}$	-0.8751	0.0168
$\boldsymbol{v_2}$	0.0031	0.7939
$\boldsymbol{v_3}$	-0.4945	-0.2009
$\boldsymbol{v_4}$	0.6628	-0.2565

Table 3. Norm-inducing matrices obtained using the GK algorithm.

A_1		A_2	
73.3151	8.7370	0.3572	-0.0128
8.7370	1.0548	-0.0128	2.7996
A_3		A_4	
40.5957	-9.5978	25.5036	9.9792
-9.5978	2.2938	9.9792	3.9440

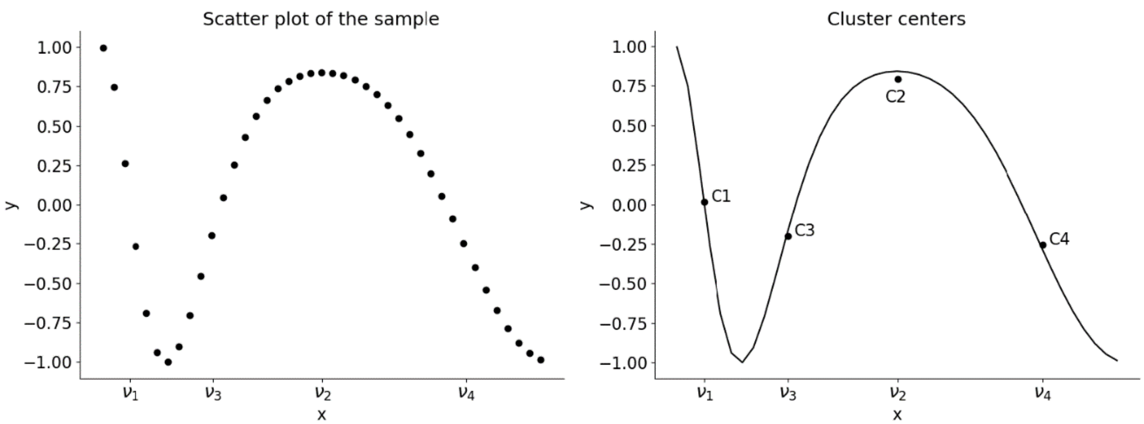

Figure 3. Scatter plot of the sample (left) and centers of the clusters (right).

Figure 3 illustrates how the second cluster (i.e., C_2) is more dispersed on the input space (x axis), as compared to the other three clusters. Next, consider the sample point $z_7 = (-0.7, -0.9981)$, where the Euclidean distances of point z_7 to the four cluster centers C_1 to C_4 are 0.2292, 9.1341, 0.0274, and 69.7022. Since the sample point z_7 has the smallest distance to the cluster center C_3, it also has the highest membership degree to cluster C_3, where $u_{3,7} = 0.8905$. Although the norm-inducing matrices A_i capture the behavior of the variables locally (i.e., the variation among variables and the relation between variables in a region of the function), the level of activation of each rule is solely determined by the input. Each rule is activated to the level equal to the membership value of the data point in the input space. As shown in Table 2, when projecting the cluster centers onto the input space (i.e., x-axis), cluster C_1 is penalized for its small dispersion of data. On the other hand, Cluster C_2 presents a higher variation of x, so the difference between the dispersion of data in these two clusters is reflected in the distance measure (note that the coefficient of x in A_2 is only 0.3572). When considering only the input spaces between clusters, cluster C_2 presents a smaller distance to most of the data points in the universe of discourse; as a result, the rule that corresponds to cluster C_2 will dominate the output function, except for those data points that are very close to the other cluster centers. For example, consider $x_7 = -0.7$, where the distance of x_7 to the four cluster centers C_1 to C_4 are 2.2483, 0.1766, 1.7139, and 47.36. Thus, the rule that corresponds to C_2 will be fired with the highest degree, as compared to the other rules. In order to further clarify the dominance of C_2 on the input space, Figure 4 shows the membership of cluster C_2 for any given value of x in the universe of discourse.

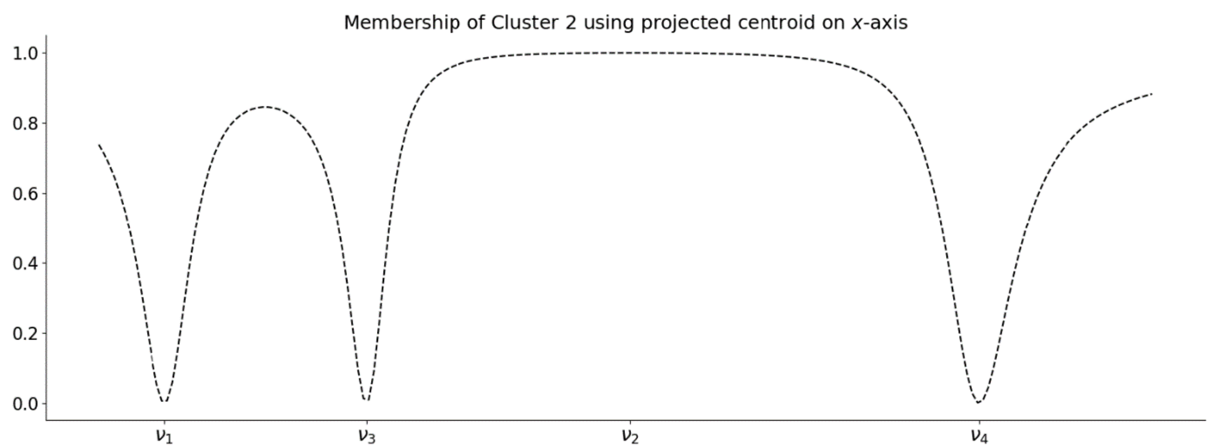

Figure 4. Membership function of cluster C_2.

Figure 4 further illustrates this phenomenon, where the cluster with the highest dispersion of data on the input space dominates the output of the FIS. As shown in Figure 4, even a small deviation from the other cluster centers (C_1, C_3, and C_4) causes a rapid increment in the membership function of cluster C_2. Accordingly, the weights shown in Equation 9 determine the relative importance of the rules and play a crucial role in counterbalancing the dominance of C_2. These rules also help to improve the accuracy of the FIS as a predictive model.

The fuzzy clustering algorithm proposed in this paper is integrated with the Adam optimization algorithm [18] in order to tune the link functions (referring to f in Equation 9) of the FISs by assigning weights to their rules. Adam optimization is a first-order, stochastic, gradient-based optimization algorithm with a wide application in different machine learning techniques [46–51]. The objective of optimization is to minimize the stochastic error of predictions made by the FIS, where the stochastic nature of the error arises from the random selection of data points for training, or from the inherent noise of the error of the outputs [18]. The naming of the "Adam" optimization algorithm refers to "adaptive momentum", indicating that the momentum parameters used in the Adam algorithm are updated during the process of optimization [52]. The momentum parameters were introduced to the iterative learning algorithms by Polyak [53] in order to increase the speed of convergence. The adjustment of the rule weights $W = [w_i]$ for improving the accuracy of the FIS using Adam optimization was completed through the following steps:

Step 1. Fix the optimization parameters, including step size $\omega \epsilon (0,1)$, exponential decay rates for momentum estimates $\varphi_1, \varphi_2 \in (0,1]$, and the numerical stabilization constant ϵ. In this paper, the values of the optimization parameters are set using the suggested default setting proposed by Kingma and Ba [18] as follows: $\omega = 0.001$, $\varphi_1 = 0.9$, $\varphi_2 = 0.999$, and $\epsilon = 10^{-8}$.

Step 2. Initialize the rule weights matrix $W = [w_i]$, the first and second momentum vectors m_0 and v_0, and the time step $t_0 = 0$.

Step 3. Create *m* samples from the training set randomly.

Step 4. Compute the gradient using Equation 11.

$$g_{t+1} \leftarrow \frac{1}{m} \nabla_w \sum_{i=1}^{m} L\left(f\left(\frac{1}{\sum_{i=1}^{C} w_i \mu_{\nu_x, A_x}^{(i)}(x)} \sum_{i=1}^{C} \hat{y}_x^{(i)} w_i \mu_{\nu_x, A_x}^{(i)}(x)\right), y_i\right) \quad 11$$

where ∇_w stands for the gradient of function f based on w; f stands for the transfer function that determines the output of the FIS (refer to Equation 9); y_i stands for the actual output for data point i; and L stands for any distance measure function.

Step 5. For the time step $t+1$, update the first and second momentum:

$$m_{t+1} \leftarrow \varphi_1 m_t + (1-\varphi_1^t) g_t$$

$$v_{t+1} \leftarrow \varphi_2 v_t + (1-\varphi_2^t) g_t^2$$

where φ_1^t and φ_2^t denote the values of φ_1 and φ_2 to the power of *t* respectively.

Step 6. Correct the bias in the first and second momentum:

$$\widehat{m}_t \leftarrow \frac{m_t}{1-\varphi_1^t}$$

$$\hat{v}_t \leftarrow \frac{v_t}{1-\varphi_2^t}$$

Step 7. Update the weights using Equation 12.

$$w_{t+1} \leftarrow w_t - \frac{\omega \widehat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \tag{12}$$

Step 8. Repeat Steps 3 through 7 until the stop criteria are met.

The use of the Adam optimization algorithm for adjusting the rule weights can counterbalance the dominance of those clusters with a high dispersion of data on the input space; it also helps to improve the accuracy of FISs in multi-dimensional problems. Once the rule weights are determined by the Adam optimization algorithm, the development of the FIS is complete, and the FIS can be used as a predictive model. This process is illustrated in the next section using a numerical example. The applicability of this algorithm for modeling construction systems is also tested through the modeling of CLP.

3.4. Numerical Example

To illustrate the process of developing FISs using the proposed algorithm, the numerical example introduced by Ren & Irwin [9] is solved in this sub-section. Let $y = h(x) = sin(1.6x^3 - 4x^2 + 1)$ be the function to be approximated, and suppose a sample size of 41 is given as $z_1 = (-1, h(-1))$, $z_2 = (-0.95, h(-0.95)), \ldots, z_{41} = (1, h(1))$. In the first step, the number of clusters are fixed to one ($C = 2$), and a two-rule TS-FIS is developed to predict the function $f(x)$. Figure 5 presents the scatter plot of the input data, as well as the prediction made by the FIS with one cluster and the cluster center. GK algorithm Figure 7 presents the scatter plot of the input data points and the results of fuzzy clustering, including the cluster centers and the predictions made by the TS-FIS for $C = 2$. The FIS developed at this stage has two rules, which are equally weighted ($w_1 = w_2 = 0.5$).

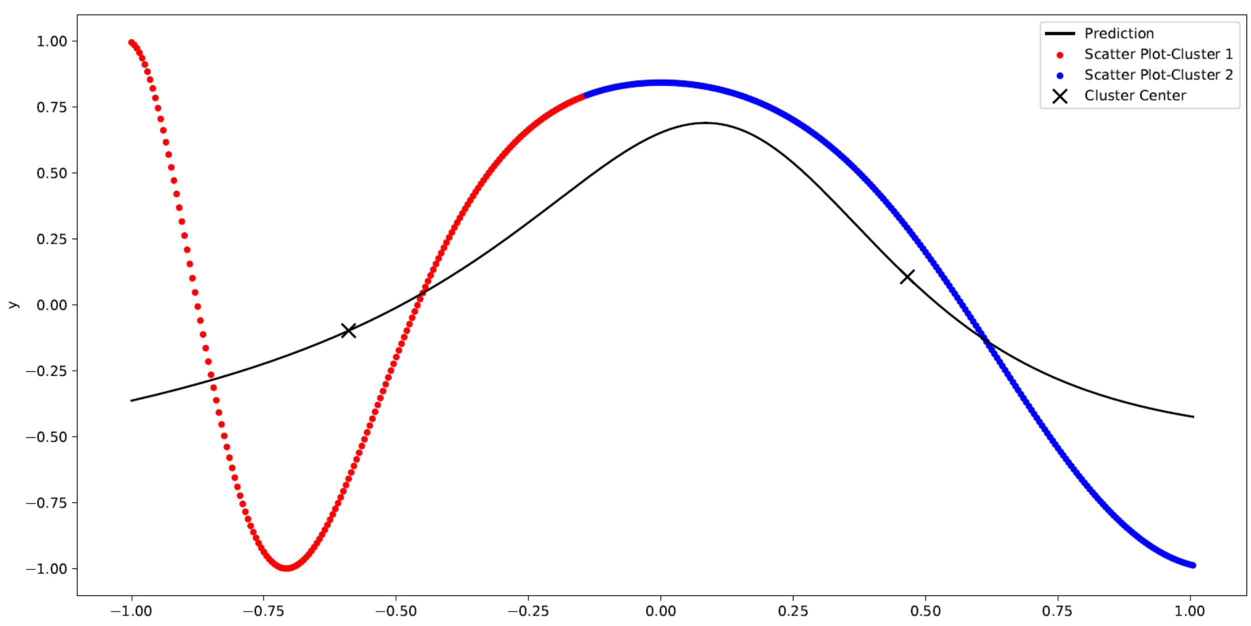

Figure 7. Scatter plot for input data and the two-cluster FIS predictive model.

The data points located on the tails of the two clusters are used to form the subclusters, and the angle between the linear predictors of the subclusters and their parent clusters are determined. This process is implemented for the two clusters shown in Figure 7, and the results show that non-linearity exists within cluster 1, as presented in Figure 8.

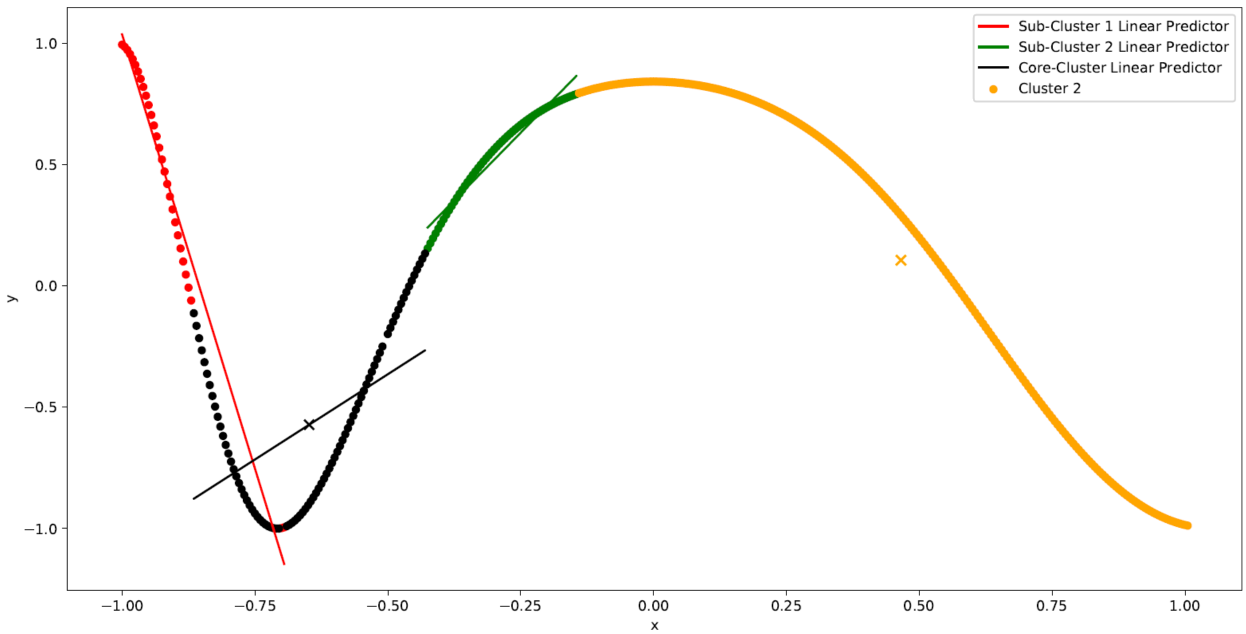

Figure 8. Determining the angle between the linear predictors of the subclusters and core clusters.

The angle between subcluster 1 and its parent cluster (C_1) is $\alpha_1 = 136.60°$, while the angle between subcluster 2 and its parent cluster (C_1) is $\alpha_2 = 11.31°$. Accordingly, non-linearity exists within cluster 1, where $\alpha_1 > 45°$. In this case, the number of clusters must be increased by one and GK algorithm is re-implemented. Figure 9 illustrates the resulting three-cluster FIS that has been developed.

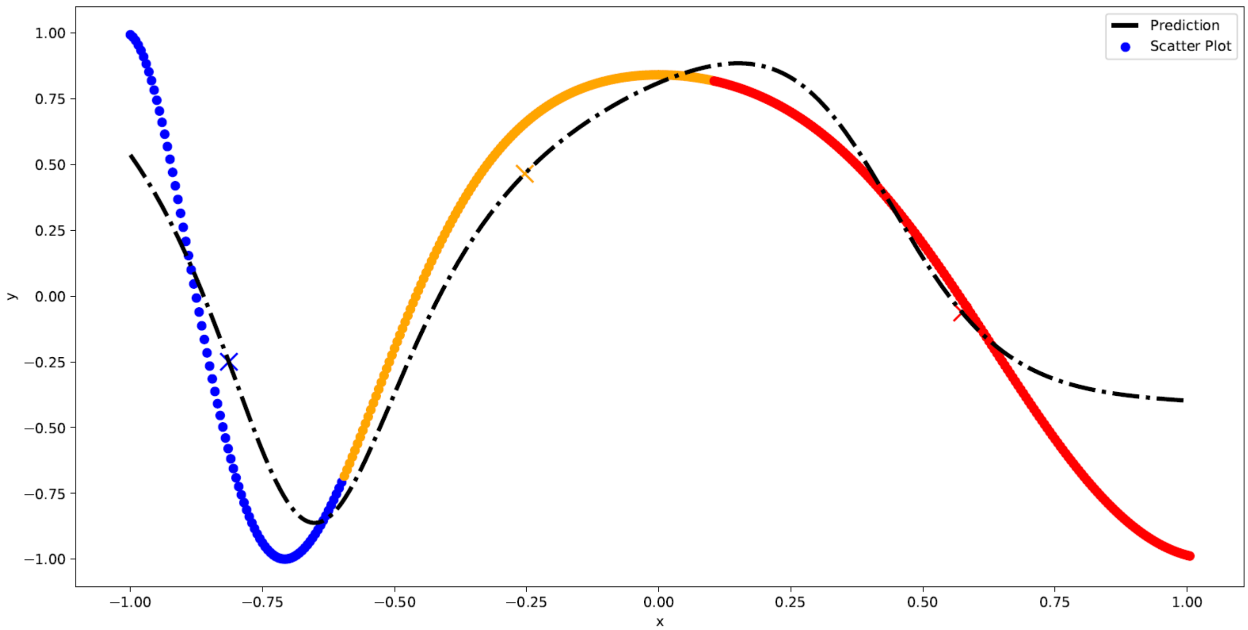

Figure 9. Scatter plot for input data and the three-cluster FIS predictive model.

These results show that non-linearity is not present in any of the three clusters. The FIS presented in Figure 9 has three rules, which are equally weighted ($w_1 = w_2 = w_3 = 0.33$). Although the inclusion of three rules in the FIS enables the model to predict the non-linearity of the transitions between clusters, the accuracy of the FIS is not yet maximized, since the rules weights are not adjusted. In order to improve the accuracy of the FIS, the rule weights are adjusted using Adam optimization and the final FIS is developed, as presented graphically in Figure 10 and illustrated in Table 4.

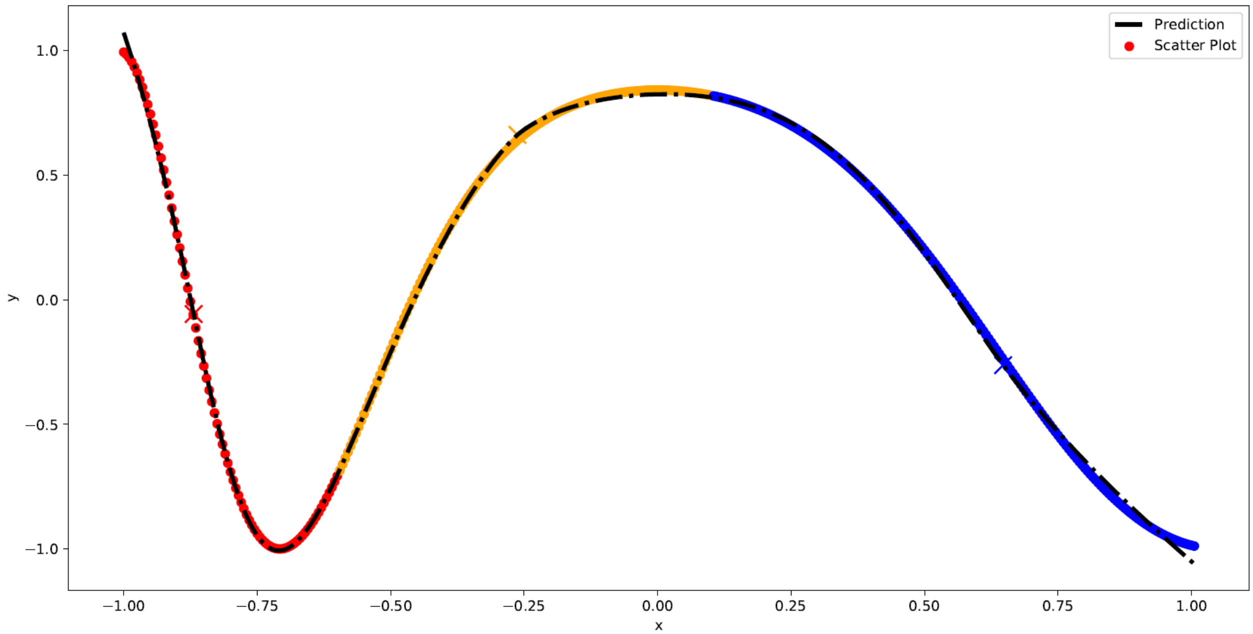

Figure 10. Scatter plot for input data and the final three-cluster FIS predictive model.

Table 4. Parameters of final FIS developed using the proposed algorithm.

Cluster	Cluster Center		Linear State Function	Rule Weight
	x	y		
1	-0.87	-0.02	$y = -10.42x - 9.10$	0.23
2	-0.27	0.66	$y = 1.90x + 1.16$	0.30
3	0.65	-0.26	$y = -2.91x + 1.62$	0.47

As show in Figure 10 and Table 4, implementing Adam optimization improves the accuracy of the FIS by decreasing the rule weights for clusters 1 and 2 and increasing the rule weight for cluster 3.

4. Predictive Model of Construction Labor Productivity

In this section, the applicability of the proposed algorithm in construction problems was tested by developing a TS-FIS to predict CLP. The accuracy of the FIS was then compared to predictive models developed using the FCM clustering technique, as proposed by Tsehayae and Fayek [4]. Modeling CLP is a highly dimensional problem, since there are numerous factors that affect its value. In addition, the optimum number of clusters for modeling CLP is unknown. Accordingly, the capacity of the proposed fuzzy clustering algorithm for determining the number of clusters automatically and assigning weights to the rules of FISs can be tested by modeling CLP. The accuracy of the results from the proposed algorithm can then be compared to that existing fuzzy clustering algorithms, which lack the aforementioned two capabilities.

Modeling CLP has been a major research interest within the construction domain for the last few decades. While the construction industry is labor intensive [54], CLP significantly impacts the cost and time of construction projects. Therefore, construction researchers and practitioners are constantly searching for ways to improve CLP outcomes. Predictive models can improve CLP by helping construction practitioners to identify the most critical factors and practices affecting it in order to facilitate improved project cost estimation, scheduling, and decision making [55,56]. A number of different models have been developed for the purpose of predicting CLP; for example, Tsehayae and Fayek [4] developed a M-FIS using FCM clustering algorithm to predict CLP for concrete placing activities. Similarly, Heravi and Eslamdoost [57] implemented the artificial neural network (ANN) algorithm to predict CLP in power plant construction projects, and El-Gohary et al. [58] applied the ANN algorithm to predict the CLP of carpenters.

In this paper, the proposed fuzzy clustering algorithm is used to develop a TS-FIS for predicting the CLP of concrete placing activities using the empirical data collected in a previous study by Tsehayae and Fayek [4]. Next, an M-FIS and a TS-FIS were developed using the FCM clustering algorithm. Three FISs were then compared based on their accuracy for predicting actual CLP field data and the results of extreme condition analysis. The empirical data for concrete placing activities were collected in Alberta, Canada on four different construction project contexts: industrial buildings, residential and commercial warehouse buildings, residential and commercial high-rise buildings, and institutional buildings. The data were collected by documenting the value of the factors influencing CLP and value of CLP on a daily basis at the construction site. A total of 93 data points are used for developing the predictive model of CLP in this paper; the details of the data collection intervals and proportions of data collected from each construction project context are provided in [4]. The input variables of the three FISs were selected based on the previous research conducted by Tsehayae and Fayek [59], where 169 factors influencing CLP were identified through an extensive literature review. Next, the number of input variables was reduced by feature selection in order to increase the accuracy of the predictive model [60]. Feature selection techniques search for a subset of input variables, which predict the output of the system with the highest accuracy. There are various techniques available for feature selection, such as correlation-based methods and wrapper methods. Ahmad and Pedrycz [60] propose the use of the wrapper method in applications where the predictive model is developed in the form of an FIS. In this problem, the wrapper method and the entire sample of 93 data points are used for feature selection and 20 input variables were selected out of the 169 initial input variables for developing the FIS. Table 5 presents the selected input variables.

Table 5. Input factors for the FIS of CLP.

Input Factor	Scale of Measure
Crew size	Integer (total number of crew members)
Craftsperson on job training	Real number (no. training sessions attended x duration of training, hrs)
Crew composition	Proportion (ratio journeyman to apprentice to helper)
Co-operation among craftspeople	1–5 predetermined rating
Craftsperson motivation	1–5 predetermined rating
Fairness of work assignment	1–5 predetermined rating
Location of work scope (distance)	Real number (distance, m)
Location of work scope (elevation)	Real number (elevation, m)
Congestion of work area	Real number (ratio of actual peak manpower to actual average manpower)

Input Factor	Scale of Measure
Fairness in performance review of crew by foreman	1–5 predetermined rating
Site congestion	Real number (ratio free site space to total site area)
Treatment of foremen by superintendent and project manager	1–5 predetermined rating
Uniformity of work rules by superintendent	1–5 predetermined rating
Out-of-sequence inspection or survey work	Real number (number of occurrences per week)
Safety training	Real number (no. training sessions attended x duration of training, hrs)
Oil price fluctuation	Real number (fluctuations of global oil price, $)
Natural gas price	Real number ($/GJ)
Concrete placement technique	Categorical: pump (1), crane and bucket (2), direct chute (3)
Structural element	Categorical: columns (1), footings (2), grade beams (3), pile caps (4), slabs (5), walls (6)
Safety inspections	Real number (number of inspections per month)

For the development of the two TS-FISs, the number of clusters is determined by the algorithm to be equal to 3 ($C = 3$), where the threshold for the angle between the two linear models is $\delta = 45$ degrees. The outputs of the rules of M-FISs are not modeled as linear state models; thus, the cluster number of the M-FIS cannot be determined automatically using the angle between such models. Tsehayae and Fayek [4] suggest that the number of clusters of the M-FIS should be optimized manually by changing the number of clusters within the range of $C \in [1, 10]$ and selecting the value of C that creates the M-FIS with the minimum RMSE. For the development of all FISs, the fuzzification coefficient m was considered to be equal to $m = 2$, as suggested by Pedrycz and Gomide [22]. Min and max fuzzy operators were used for AND and OR operations between the input and output variables. The center of area (COA) defuzzification technique was used to defuzzify the outputs of the M-FIS. Next, the accuracy of the three FISs was measured by comparing their predictions to the actual field data and calculating two error measures, which are commonly used for evaluating predictive models (i.e., mean absolute error (MAE) and root mean square error (RMSE)). For this purpose, the sample data are divided into training and testing sets using the same approach utilized by Tsehayae and Fayek [4], in which 70% of the data are used for training and 30% is used for testing. The results are presented below in Table 6.

Table 6. The accuracy for the three FISs for predicting CLP.

	M-FIS (FCM clustering)		TS-FIS (FCM clustering)		TS-FIS (Proposed Algorithm)	
Testing Error	MAE	1.94	MAE	0.79	MAE	0.07
	RMSE	2.21	RMSE	0.98	RMSE	0.22
Training Error	RMSE	2.00	RMSE	0.90	RMSE	0.06

According to these results, the TS-FIS developed by the proposed fuzzy clustering algorithm had the highest accuracy among the three FISs, with an MAE and RMSE of 0.07 and 0.22. The second most accurate algorithm was the TS-FIS developed by FCM clustering, with an MAE and RMSE of 0.79 and 0.98. Finally, the third most accurate algorithm was the M-FIS developed by FCM clustering, with an MAE and RMSE of 1.94 and 2.21. Accordingly, the results of comparison show that the proposed algorithm can create more accurate FISs, as compared to the FCM clustering technique.

Next, the performance of the three FISs for predicting the behavior of the model in extreme conditions (i.e., where CLP is extremely low or extremely high) was tested. Predicting the behavior of systems in extreme conditions is an important function of predictive models in engineering applications and control systems, since decisions in these contexts are often made when the outputs of a system surpass or fall below a pre-determined threshold. To visualize the behaviour of the three FISs in the extreme conditions, a scatter plot of the results was developed by showing the actual field data on the horizontal axis and the predictions of the three FISs on the vertical axis. As presented in Figure 11, the perfect prediction results (i.e., $MAE = RMSE = 0$) are located on the straight line of $y = x$; thus, a smaller distance between the predictions of each FIS to the straight line of $y = x$ indicate more accurate predictions of the CLP. Due to confidentiality considerations, the value of CLP was normalized using the following equation $x_n^i = \frac{x^i - \min(x)}{\max(x) - \min(x)}$, where x_n^i stands for the normalized value of CLP for data point i and x^i stands for the original value of CLP.

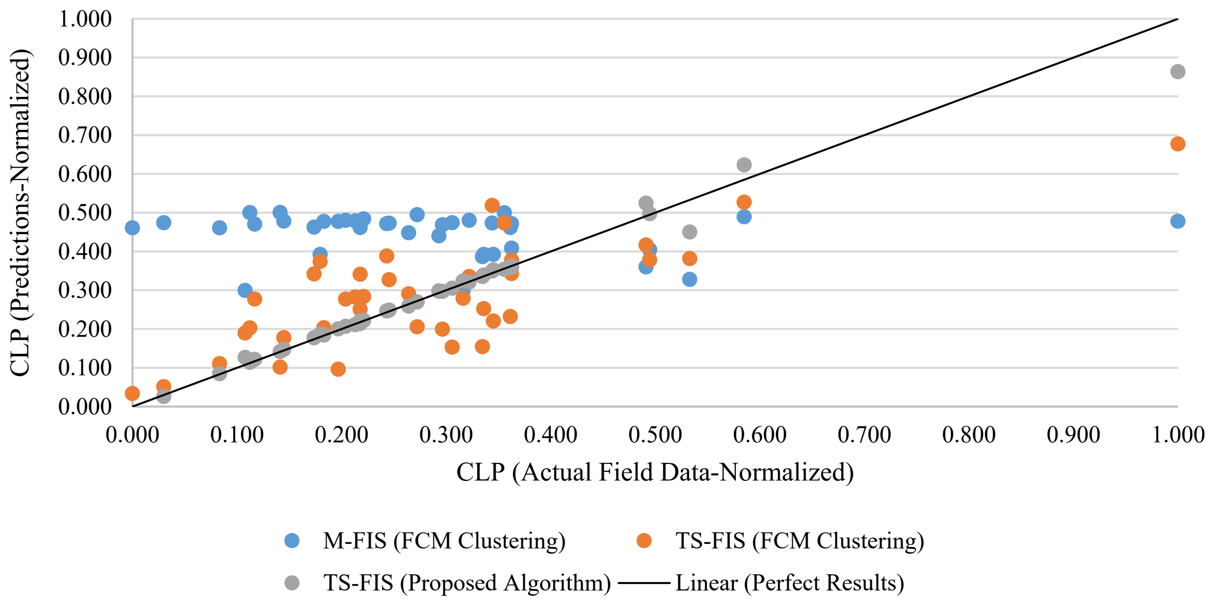

Figure 11. Scatter plot of prediction results vs. actual field data.

The results presented in Figure 11 shows that among the three FISs tested, the TS-FIS developed by the proposed algorithm has the closest predictions to the straight line of $y = x$ in extreme conditions. In order to further investigate the performance of the three FISs in predicting the value of CLP in extreme conditions, the empirical data points were divided into 10 categories, based on the region where the actual value of CLP was located on the x-axis in Figure 11. Next, the accuracy of the three FISs was determined by calculating the MAE and RMSE for each category of the results, as presented in Table 7.

Table 7. Results of the analysis of extreme conditions.

Category	Actual Data CLP Range	M-FIS (FCM Clustering)		TS-FIS (FCM Clustering)		TS-FIS (Proposed Algorithm)	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
C_1	[0,0.1)	0.427	0.428	0.027	0.027	0.009	0.012
C_2	[0.1,0.2)	0.300	0.306	0.099	0.115	0.005	0.007
C_3	[0.2,0.3)	0.226	0.229	0.071	0.081	0.003	0.003
C_4	[0.3,0.4)	0.093	0.105	0.095	0.114	0.003	0.004
C_5	[0.4,0.5)	0.111	0.113	0.096	0.098	0.017	0.023
C_6	[0.5,0.6)	0.151	0.160	0.105	0.115	0.060	0.065
C_7	[0.6,0.7)	NA	NA	NA	NA	NA	NA
C_8	[0.7,0.8)	NA	NA	NA	NA	NA	NA
C_9	[0.8,0.9)	NA	NA	NA	NA	NA	NA
C_{10}	[0.9,10)	0.523	0.523	0.323	0.323	0.137	0.137

Note: No data points are located in those categories, for which the value of error measures is NA.

As shown in Table 7, the TS-FIS developed by the proposed algorithm has the highest accuracy for predicting CLP in extreme conditions, with an MAE and RMSE of 0.009 and 0.012 for C_1, and 0.137 and 0.137 for C_{10}. However, the accuracy significantly decreases in extreme conditions (i.e., C_1 and C_{10}), as compared to conditions where CLP is closer to its median value (i.e., C_3, C_4). This phenomenon (i.e., reduction of accuracy in extreme conditions) can also be observed in the results produced by the TS-FIS and the M-FIS developed using the FCM clustering technique.

The results of comparison of the three FISs confirms that use of Adam optimization to assign weights to rules of FISs improves the accuracy of these models for predicting the behavior of highly dimensional systems. Moreover, the algorithm proposed for automatic determination of cluster numbers can improve the efficiency of the fuzzy clustering algorithms, where cluster numbers are typically optimized manually. The extreme conditions test shows that the TS-FIS developed using the proposed algorithm outperforms those FISs developed using the FCM clustering algorithm. However, in extreme conditions, the accuracy of all the three FISs decreases, as compared to conditions where the output is closer to its median value.

5. Conclusions and Future Research

Fuzzy clustering algorithms are one of the most common techniques for developing data-driven FISs in engineering applications. Despite their wide application in predictive modeling for engineering problems, fuzzy clustering algorithms have two limitations in this area. First, the majority of fuzzy clustering algorithms rely on the modeler's judgment for determining the appropriate number of clusters. Such knowledge may not be accessible to the modeler in many engineering applications. Second, fuzzy clustering algorithms equally weight all the input and output variables of the system being modeled; this approach can decrease the accuracy of these algorithms for developing FISs in highly dimensional problems. In this paper, a new fuzzy algorithm was introduced to address these limitations by integrating GK algorithm with Adam optimization. A novel approach was developed to determine the number of clusters, based on the non-linearity observed within each cluster. This new algorithm was then used to predict CLP for concrete placing activities, and the results were compared to those of two FISs developed by the FCM clustering algorithm. A comparison of the result showed that automatic determination of the number of clusters improves the efficiency of fuzzy clustering algorithms, and helps to avoid reliance on the subjective judgment of the modeler. Moreover, the use of Adam optimization for assigning weights to the rules of the FIS significantly improves their accuracy in highly dimensional problems.

Although the proposed algorithm outperformed the FCM clustering algorithm in predicting system behavior in extreme conditions, the accuracy of the FIS developed using this algorithm significantly decreased in extreme conditions, as compared to those conditions where the output of the model was close to its median value. In future research, efforts will be made to address this limitation by increasing the significance of the data points located on the two extremes of the universe of discourse for developing fuzzy clusters and/or for determining the rule weights. Moreover, the proposed fuzzy clustering algorithm may provide non-convex membership functions for rule activation, which makes it more difficult to interpret the reasoning process of the FIS. Converting the membership functions in the input space to one of the widely used convex shapes (e.g., trapezoidal, exponential, Gaussian) would increase the interpretability of the model. Finally, in future research the proposed method will be used to develop predictive models for different construction applications, such as modeling the production rate of construction equipment, modeling organizational competency, and predicting the performance of construction projects and organizations.

6. Acknowledgments

This work is supported by the Natural Sciences and Engineering Research Council of Canada Industrial Research Chair in Strategic Construction Modeling and Delivery (NSERC IRCPJ 428226–15), which is held by Dr. Aminah Robinson Fayek.

7. List of Abbreviations

FIS	Fuzzy inference system
FCM clustering	Fuzzy *c*-means clustering
GK algorithm	Gustafson-Kessel's algorithm
CLP	Construction labor productivity
M-FIS	Mamdani FIS
TS-FIS	Takagi-Sugeno FIS
TBM	Tunnel boring machines
PCM	Possibilistic *c*-means
FPCM	Fuzzy possibilistic *c*-means
PFCM	Possibilistic fuzzy *c*-means
GPFCM	Generalized possibilistic fuzzy *c*-means
ANN	Artificial neural network
COA	Center of area
MAE	Mean absolute error
RMSE	Root mean square error

8. References

[1] L.A. Zadeh, Fuzzy sets, Inf. Control. 8 (1965) 338–353. https://doi.org/10.1016/S0019-9958(65)90241-X.

[2] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning-I, Inf. Sci. (Ny). 8 (1975) 199–249. https://doi.org/10.1016/0020-0255(75)90036-5.

[3] A. Yazdani-Chamzini, S.H. Yakhchali, M. Mahmoodian, Risk ranking of tunnel construction projects by using the ELECTRE technique under a fuzzy environment, Int. J. Manag. Sci. Eng. Manag. 8 (2013) 1–14. https://doi.org/10.1080/17509653.2013.783185.

[4] A.A. Tsehayae, A.R. Fayek, Developing and Optimizing Context-Specific Fuzzy Inference System-Based Construction Labor Productivity Models, J. Constr. Eng. Manag. 142 (2016) 04016017. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001127.

[5] M. Tang, X. Chen, W. Hu, W. Yu, Generation of a probabilistic fuzzy rule base by learning from examples, Inf. Sci. (Ny). 217 (2012) 21–30. https://doi.org/10.1016/j.ins.2012.06.021.

[6] S. Rastegar, R. Araújo, J. Mendes, Online identification of Takagi–Sugeno fuzzy models based on self-adaptive hierarchical particle swarm optimization algorithm, Appl. Math. Model. 45 (2017) 606–620. https://doi.org/10.1016/j.apm.2017.01.019.

[7] L. Teng, Y. Wang, W. Cai, H. Li, Robust fuzzy model predictive control of discrete-time takagi-sugeno systems with nonlinear local models, IEEE Trans. Fuzzy Syst. 26 (2018) 2915–2925. https://doi.org/10.1109/TFUZZ.2018.2815521.

[8] M. Function, Short paper T – S Fuzzy Model Identification Based on a Novel Hyperplane-Shaped, 25 (2017) 1364–1370.

[9] L. Ren, G.W. Irwin, Robust fuzzy Gustafson-Kessel clustering for nonlinear system identification, Int. J. Syst. Sci. 34 (2003) 787–803. https://doi.org/10.1080/00207720310001655515.

[10] S.H. Tsai, Y.W. Chen, A novel identification method for Takagi–Sugeno fuzzy model, Fuzzy Sets Syst. 338 (2018) 117–135. https://doi.org/10.1016/j.fss.2017.10.012.

[11] J. Li, L. Yang, Y. Qu, G. Sexton, An extended Takagi–Sugeno–Kang inference system (TSK+) with fuzzy interpolation and its rule base generation, Soft Comput. 22 (2018) 3155–3170. https://doi.org/10.1007/s00500-017-2925-8.

[12] N. Vafamand, M.M. Arefi, A. Khayatian, Nonlinear system identification based on Takagi-Sugeno fuzzy modeling and unscented Kalman filter, ISA Trans. 74 (2018) 134–143. https://doi.org/10.1016/j.isatra.2018.02.005.

[13] K. Chafaa, M. Ghanai, K. Benmahammed, Fuzzy modelling using Kalman filter K., IET Control Theory Appl. 1 (2007) 58–64. https://doi.org/10.1049/iet-cta.

[14] Bezdek, James C., Robert Ehrlich, William Full, FCM: The Fuzzy C-Means Clustering Algorithm, Comput. Geosci. 10 (1984) 191–203.

[15] D.E. Gustafson, W.C. Kessel, Fuzzy Clustering with a Fuzzy Covariance Matrix, in: IEEE Conf. Decis. Control Incl. 17th Symp. Adapt. Process., IEEE, 1979: pp. 761–766. https://doi.org/10.1109/CDC.1978.268028.

[16] R.R. Yager, P.F. Dimitar, Generation of Fuzzy Rules in Mountain Clustering, J. Intell. Fuzzy Syst. 2 (1994) 209–219. https://doi.org/10.3233/IFS-1994-2301.

[17] X. Wang, Y. Wang, L. Wang, Improving fuzzy c-means clustering based on feature-weight learning, Pattern Recognit. Lett. 25 (2004) 1123–1132.

https://doi.org/10.1016/j.patrec.2004.03.008.

[18] D.P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, (2014) 1–15. http://arxiv.org/abs/1412.6980.

[19] E.H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, Int. J. Man. Mach. Stud. 7 (1975) 1–13. https://doi.org/10.1016/S0020-7373(75)80002-2.

[20] E.H. Mamdani, APPLICATION OF FUZZY LOGIC TO APPROXIMATE REASONING USING LINGUISTIC SYNTHESIS, in: Proc. Sixth Int. Symp. Mult. Log., IEEE Computer Society Press, 1976: pp. 196–202. https://doi.org/10.1192/bjp.112.483.211-a.

[21] T. Takagi, M. Sugeno, Fuzzy Identification of Systems and Its Applications to Modeling and Control, IEEE Trans. Syst. Man Cybern. SMC-15 (1985) 116–132. https://doi.org/10.1109/TSMC.1985.6313399.

[22] W. Pedrycz, F. Gomide, Fuzzy Systems Engineering: Toward Human-Centric Computing, John Wiley & Sons, New Jersey, 2007.

[23] G. Attri, N. Arunima, D. Amit Kumar, G. Saptarsi, P. Mrityunjoy, A Short Review on Different Clustering Techniques and Their Applications, in: Emerg. Technol. Model. Graph., Springer, Singapore, 2020: pp. 69–83. https://doi.org/https://doi.org/10.1007/978-981-13-7403-6_9.

[24] N. Gerami Seresht, R. Lourenzutti, A. Salah, A.R. Fayek, Overview of Fuzzy Hybrid Techniques in Construction Engineering and Management, in: Fuzzy Hybrid Comput. Constr. Eng. Manag. Theory Appl., Emerald Publishing Limited, London, 2018: pp. 37–107.

[25] S. Askari, A novel and fast MIMO fuzzy inference system based on a class of fuzzy clustering algorithms with interpretability and complexity analysis, Expert Syst. Appl. 84 (2017) 301–322. https://doi.org/10.1016/j.eswa.2017.04.045.

[26] R.E. Precup, H.I. Filip, M.B. Rədac, E.M. Petriu, S. Preitl, C.A. Dragoş, Online identification of evolving Takagi-Sugeno-Kang fuzzy models for crane systems, Appl. Soft Comput. J. 24 (2014) 1155–1163. https://doi.org/10.1016/j.asoc.2014.01.013.

[27] P. Hušek, K. Narenathreyas, Aircraft longitudinal motion control based on Takagi–Sugeno fuzzy model, Appl. Soft Comput. J. 49 (2016) 269–278. https://doi.org/10.1016/j.asoc.2016.07.038.

[28] P.C. Chang, J.L. Wu, J.J. Lin, A Takagi-Sugeno fuzzy model combined with a support vector regression for stock trading forecasting, Appl. Soft Comput. J. 38 (2016) 831–842. https://doi.org/10.1016/j.asoc.2015.10.030.

[29] A.C. Adoko, S. Yagiz, Fuzzy Inference System-Based for TBM Field Penetration Index Estimation in Rock Mass, Geotech. Geol. Eng. 37 (2019) 1533–1553. https://doi.org/10.1007/s10706-018-0706-5.

[30] N.B. Siraj, A.R. Fayek, A.A. Tsehayae, Development and Optimization of Artificial Intelligence-Based Concrete Compressive Strength Predictive Models, Int. J. Struct. Civ. Eng. Res. 5 (2016) 156–167. https://doi.org/10.18178/ijscer.5.3.156-167.

[31] H. Saberi, A. Rahai, F. Hatami, A fast and efficient clustering based fuzzy time series algorithm (FEFTS) for regression and classification, Appl. Soft Comput. J. 61 (2017) 1088–1097. https://doi.org/10.1016/j.asoc.2017.09.023.

[32] K. V. Shihabudheen, G.N. Pillai, Recent advances in neuro-fuzzy system: A survey,

Knowledge-Based Syst. 152 (2018) 136–162. https://doi.org/10.1016/j.knosys.2018.04.014.

[33] N. Gerami Seresht, A.R. Fayek, Computational method for fuzzy arithmetic operations on triangular fuzzy numbers by extension principle, Int. J. Approx. Reason. 106 (2019) 172–193. https://doi.org/10.1016/j.ijar.2019.01.005.

[34] J. Nayak, B. Naik, H.S. Behera, Fuzzy C-Means (FCM) Clustering Algorithm: A Decade Review from 2000 to 2014, in: L.C. Jain, H.S. Behera, J.K. Mandal, D.P. Mohapatra (Eds.), Comput. Intell. Data Min. - Vol. 2, Springer India, New Delhi, 2015: pp. 133–149.

[35] A. Gosain, S. Dahiya, Performance Analysis of Various Fuzzy Clustering Algorithms: A Review, Procedia Comput. Sci. 79 (2016) 100–111. https://doi.org/10.1016/j.procs.2016.03.014.

[36] M. Askari, A.H.D. Markazi, A new evolving compact optimised Takagi-Sugeno fuzzy model and its application to nonlinear system identification, Int. J. Syst. Sci. 43 (2012) 776–785. https://doi.org/10.1080/00207721.2010.523798.

[37] S. Askari, N. Montazerin, M.H. Fazel Zarandi, Generalized Possibilistic Fuzzy C-Means with novel cluster validity indices for clustering noisy data, Appl. Soft Comput. J. 53 (2017) 262–283. https://doi.org/10.1016/j.asoc.2016.12.049.

[38] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, IEEE Trans. Fuzzy Syst. 13 (2005) 517–530. https://doi.org/10.1109/TFUZZ.2004.840099.

[39] R. Krishnapuram, J.M. Keller, A Possibilistic Approach to Clustering, IEEE Trans. Fuzzy Syst. 1 (1993) 98–110. https://doi.org/10.1109/91.227387.

[40] N.R. Pal, K. Pal, J.C. Bezdek, Mixed c-means clustering model, IEEE Int. Conf. Fuzzy Syst. 1 (1997) 11–21. https://doi.org/10.1109/fuzzy.1997.616338.

[41] H. Ishibuchi, T. Yamamoto, Rule weight specification in fuzzy rule-based classification systems, IEEE Trans. Fuzzy Syst. 13 (2005) 428–435. https://doi.org/10.1109/TFUZZ.2004.841738.

[42] A.K. Paul, P.C. Shill, M.R.I. Rabin, K. Murase, Adaptive weighted fuzzy rule-based system for the risk level assessment of heart disease, Appl. Intell. 48 (2018) 1739–1756. https://doi.org/10.1007/s10489-017-1037-6.

[43] M.Z. Jahromi, A New Rule-weight Learning Method based on Gradient Descent, Lect. Notes Eng. Comput. Sci. (2009).

[44] W. Pedrycz, Granular Computing: Analysis and Design of Intelligent Systems, CRC Press, Boca Raton, FL, 2016.

[45] A. Pedrycz, M. Reformat, Hierarchical FCM in a stepwise discovery of structure in data, Soft Comput. 10 (2006) 244–256. https://doi.org/10.1007/s00500-005-0478-8.

[46] T.P. Lillicrap, J.J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, D. Wierstra, Continuous control with deep reinforcement learning, (2015). http://arxiv.org/abs/1509.02971.

[47] R. Karthik, U. Gupta, A. Jha, R. Rajalakshmi, R. Menaka, A deep supervised approach for ischemic lesion segmentation from multimodal MRI using Fully Convolutional Network, Appl. Soft Comput. 84 (2019) 105685. https://doi.org/10.1016/j.asoc.2019.105685.

[48] S. Samui, I. Chakrabarti, S.K. Ghosh, Time–frequency masking based supervised speech enhancement framework using fuzzy deep belief network, Appl. Soft Comput. J. 74 (2019)

583–602. https://doi.org/10.1016/j.asoc.2018.10.031.

[49] A.M. Grachev, D.I. Ignatov, A. V. Savchenko, Compression of recurrent neural networks for efficient language modeling, Appl. Soft Comput. J. 79 (2019) 354–362. https://doi.org/10.1016/j.asoc.2019.03.057.

[50] N. Carlini, D. Wagner, Towards Evaluating the Robustness of Neural Networks, Proc. - IEEE Symp. Secur. Priv. (2017) 39–57. https://doi.org/10.1109/SP.2017.49.

[51] A. Daemi, H. Kodamana, B. Huang, Gaussian Process Modelling with Gaussian Mixture Likelihood, J. Process Control. 81 (2019) 209–220. https://doi.org/10.1016/j.jprocont.2019.06.007.

[52] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, Cambridge, MA, 2016.

[53] B.T. Polyak, Some methods of speeding up the convergence of iteration methods, USSR Comput. Math. Math. Phys. 4 (1964) 1–17. https://doi.org/10.1016/0041-5553(64)90137-5.

[54] A.M. Jarkas, Critical Investigation into the Applicability of the Learning Curve Theory to Rebar Fixing Labor Productivity, J. Constr. Eng. Manag. 136 (2010) 1279–1288. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000236.

[55] W. Yi, A.P.C. Chan, Critical Review of Labor Productivity Research in Construction Journals, J. Manag. Eng. 30 (2014) 214–225. https://doi.org/10.1061/(ASCE)ME.1943-5479.0000194.

[56] A.A. Tsehayae, A.R. Fayek, System model for analysing construction labour productivity, Constr. Innov. 16 (2016) 203–228. https://doi.org/10.1108/CI-07-2015-0040.

[57] G. Heravi, E. Eslamdoost, Applying Artificial Neural Networks for Measuring and Predicting Construction-Labor Productivity, J. Constr. Eng. Manag. 141 (2015) 04015032-

1-04015032–11. https://doi.org/10.1061/(ASCE)CO.1943-7862.

[58] K.M. El-Gohary, R.F. Aziz, H.A. Abdel-Khalek, Engineering Approach Using ANN to Improve and Predict Construction Labor Productivity under Different Influences, J. Constr. Eng. Manag. 143 (2017) 04017045. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001340.

[59] A.A. Tsehayae, A.R. Fayek, Identification and comparative analysis of key parameters influencing construction labour productivity in building and industrial projects, Can. J. Civ. Eng. 41 (2014) 878–891. https://doi.org/10.1139/cjce-2014-0031.

[60] S.S.S. Ahmad, W. Pedrycz, Data and Feature Reduction in Fuzzy Modeling through Particle Swarm Optimization, Appl. Comput. Intell. Soft Comput. 2012 (2012) 1–21. https://doi.org/10.1155/2012/347157.