

Article

Not peer-reviewed version

Fuzzy Logic, Neural Network and ANFIS in Delegation of Standard Concrete Beam Calculations

[Abtin Baghdadi](#)*, [Neira Babovic](#), [Harald Kloft](#)

Posted Date: 7 November 2023

doi: 10.20944/preprints202311.0469.v1

Keywords: Concrete; Beam Section; Machine Learning; Neural Network; Fuzzy Logic; ANFIS; ACI



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Fuzzy Logic, Neural Network and ANFIS in Delegation of Standard Concrete Beam Calculations

Abtin Baghdadi *, Neira Babovic  and Harald Kloft 

Institut für Tragwerksentwurf (ITE), Technische Universität Braunschweig, 38106 Braunschweig, Germany

* Correspondence: a.baghdadi@tu-braunschweig.de

Abstract: Machine Learning (ML) has proved its capabilities in different scientific and industrial fields, but it needs to be further investigated in the construction industry for practical utilization. One of the use cases of ML is delegating the structural calculation process. In this study, to discuss ML's capabilities in performing the work of a structural designer, calculations of concrete sections based on ACI (American Concrete Institute) as a case study were selected. At first, all manual design steps and standard considerations for a concrete beam section were coded parametrically in MATLAB. After comparing with structural design references to prove the accuracy of codes in calculating shear and bending capacities, the parametric results were used as initial data (*Look – up* table) for training the ML operators. Regarding different types of ML techniques and as a comparison between them, in the next steps, all essential codes for Fuzzy Logic (FL), Neural Network (NN) and Adaptive Neuro-Fuzzy Inference System (ANFIS) were coded in the same platform. The performance of the three coded ML operators to replace (delegate) standard calculations compared to direct calculations was individually investigated and displayed through parametric examples. After initial examples, the influences of the number of parameters and size of the *Look – up* table on the accuracy of each operator were discussed. The study concluded that although all three operators can delegate the standard calculation, the precision of the results differs considerably. In case of the desirable size of the *Look – up* table, ANFIS operators can represent the standard calculations with a different number of parameters and entirely high precision.

Keywords: concrete; beam section; machine learning; Neural Network; Fuzzy Logic; ANFIS; ACI

1. Introduction

Artificial intelligence (AI) and Machine Learning (ML) are two related terms in computer science and technology. Although they are related, their definitions are not interchangeable. While AI refers to creating machines that perform tasks that typically require human intelligence, such as perception, reasoning, learning, and decision-making, ML is only a subset of AI that involves training algorithms to learn patterns in data and make predictions or decisions based on them. AI is, therefore, a much broader concept encompassing various fields of study, such as computer vision, natural language processing, robotics, and more. Additionally, Machine Learning focuses on developing algorithms and statistical models that can analyze data and predict, interpret or decide [1]. AI typically uses a rule-based approach, where systems are programmed with a set of rules and instructions to follow. Machine Learning, on the other hand, uses a data-driven approach, where algorithms learn from data to identify patterns and make predictions. AI systems are typically designed and programmed by humans, while Machine Learning algorithms are trained on data [2]. Machine Learning algorithms require large amounts of data to be trained effectively, whereas AI systems can be designed with a smaller set of rules and instructions [1].

There are three main types of Machine Learning: supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, the algorithm is trained on a labelled dataset to predict an output given an input [3]. In unsupervised learning, the algorithm identifies patterns and structures in an unlabeled dataset. In semi-supervised learning, the algorithm is trained on a combination of labelled and unlabeled data. Finally, in reinforcement learning, the algorithm learns to make decisions

by interacting with an environment and receiving feedback in the form of rewards or penalties [4]. Each type of Machine Learning has its strengths and weaknesses, and the choice of which type to use depends on the nature of the problem and the available data [5]. Neural Networks (NN) and Adaptive Neuro-Fuzzy Inference Systems (ANFIS) are two types of supervised ML. A Fuzzy approach can be applied in supervised and unsupervised classifications as a mathematical framework [6], [7].

Fuzzy Logic (FL), developed by Zadeh [8], is an approach for dealing with imprecise or uncertain data often encountered in real-world applications. FL provides a way to handle linguistic variables and Fuzzy sets, delegating and manipulating imprecise or ambiguous data. It provides a set of rules for making decisions or predictions based on this data and allows for more flexible and nuanced reasoning than traditional Boolean logic [9]. FL has been used in various fields, including control systems, pattern recognition, and decision-making. For example, FL has been used to control the speed of electric motors in industrial processes, to diagnose medical conditions based on symptoms, and to make decisions in financial forecasting [10].

FL has been used in various fields, including structural studies. FL provides a framework for dealing with imprecise or uncertain data, often encountered in structural studies. FL allows for more flexible and nuanced reasoning, making it an attractive tool for analyzing complex structural systems. FL has been applied in structural studies in assessing seismic hazards and estimating earthquake probability and potential consequences. FL combined data from multiple sources, such as seismic data, geological information, and historical earthquake data, to generate a comprehensive seismic hazard assessment [11]. FL was also utilized to design and analyze structural systems to model the performance of complex systems, such as buildings, bridges, and other infrastructure, including optimizing the design of reinforced concrete structures [12]. FL can be used to analyze data from multiple sensors, such as strain gauges and accelerometers, to detect changes in the structural performance that may indicate damage or deterioration [13]. Another type of utilization was for parametric evaluation of dry concrete joints to continuously evaluate the joints with different parameter ratios. At the same time, the limited number of the tests can be generally managed, which will, without ML, end to discrete data [14].

Neural Networks (NN), a supervised technique, has been increasingly used in various fields, including structural studies. Using Neural Networks in structural studies can lead to more accurate and reliable analyses and help ensure the safety and reliability of critical infrastructure. Structural health assessment is one area where Neural Networks have been applied in structural studies. Structural health assessment involves monitoring the condition of structures, such as bridges and buildings, to detect any defects or damage. Neural Networks can be used to analyze data from multiple sensors, such as strain gauges and accelerometers, to detect changes in the structural performance that may indicate damage or deterioration [15]. Neural Networks have also been used to design and optimize structural systems. It was utilized to model the performance of complex systems, such as buildings, bridges, and other infrastructure, and to optimize their design based on performance criteria. For example, Neural Networks were utilized to optimize the design of steel-concrete composite structures [16]. Another application of NN in structural studies is in predicting structural performance under various loading conditions, trained on data from physical experiments or numerical simulations to predict the response of structural systems to different loads. Likewise, NN to predict the deflection and stress of steel-concrete composite [17].

Adaptive Neuro-Fuzzy Inference System (ANFIS) is another supervised ML approach, which is a type of ML algorithm that combines the strengths of Neural Networks and FL used in various fields, including structural studies such as in predicting the dynamic performance of structures under seismic loads. ANFIS can be trained on data from physical experiments or numerical simulations to predict the response of structural systems to seismic loads and the displacement and acceleration response of steel frame structures [18]. ANFIS was also employed in the assessment of structural health. It can be used to analyze data from sensors, such as accelerometers and strain gauges, to detect structural defects or damage. For example, ANFIS has been used to detect damage in reinforced concrete structures based

on data from acoustic emissions [19]. Another application in structural studies is in the design and optimization of structural systems. ANFIS can model the performance of complex systems, such as buildings and bridges, and optimize their design based on performance criteria. For example, ANFIS was used to optimize the design of steel-concrete composite beams [20].

Reinforced concrete is a widely used construction material in structural engineering due to its strength and durability. However, predicting the performance of reinforced concrete structures is a complex task, as it depends on various factors such as material properties, geometry, and loading conditions. FL, Neural Network, and ANFIS are three Machine Learning techniques that can be used in reinforced concrete research to accurately predict these structures' behaviour. For instance - FL is utilized to model the mechanical behaviour of concrete structures under different loading conditions and predict the deflection of reinforced concrete beams [21]. - Neural Networks can be trained on large datasets to identify complex patterns and relationships between input and output variables, predicting the compressive strength of concrete based on various input parameters [22]. - ANFIS has been used in reinforced concrete research to predict various properties of concrete structures, such as the compressive strength, flexural performance, and shear strength, along with predicting the performance of reinforced concrete beams with different reinforcement [23],[24].

Despite their potential, several challenges must be addressed to apply the ML approach practically. Some primary challenges can be mentioned: 1. Lack of engineers' familiarity with this method and difficulty in modelling complex relationships; 2- Lack of awareness about the differences in performance of different ML techniques to a wise tool selection. 3- Sensitivity of these techniques to the size of a *Look – up* table, number of parameters and type of relation between the input and outputs (linear and nonlinear). 4- Machine Learning models are often called "black boxes" because they can produce complex and non-intuitive results. This makes it challenging for engineers to interpret and explain the outputs of the models; in other words, the designers do not know up to which stand these methods are trustworthy. 5- the studies in which the ML approaches were used are mainly based on the researches and experimental data, while direct comparisons between the standard known calculations and ML's results are needed.

In this regard, the current paper aims to discuss the three types of Machine Learning, including Fuzzy, *NN* and ANFIS, to re-introduce and compare their performances in one example. In the case study, the Standard calculation of a concrete beam was selected, which is known to the engineer. This discussion discussed the sensitivity of the methods to the number of parameters and the size of the *Look – up* table. Additionally, the accuracy of all methods with each other and routine calculations was compared.

2. Discussion:

2.1. Initialization:

Coding, Standard Beam Section Calculation:

This paper aims to evaluate the performance of ML operators in delegating the standard structural computing process. As the case study, a wide range of structural subjects and Standards can be selected for analyzing or designing steel and concrete elements. Here, the known designing process of concrete beams (Figure 1) was chosen as the subject of interest, and in particular, the calculations of the beams' bending and shear capacity were defined as the output (referring to ACI318 – 14 [25] standard), whereas features of the section were the adjustable parameters (Figure 1) in the codes. In these calculations, after initialization and some basic calculations such as section area (A), amount of compressive and tensile re-bars, A_s , A'_s , and their percentages ρ_c , ρ_t , along with beam re-bar balance percentage ρ_b , were calculated.

Simple algorithmic codes were developed in MATLAB to calculate ACI and parametrically calculate the capacities. The shear calculation regarding steel (stirrups) and concrete capacities was

managed in one step. The shear capacity (V_c) is 75% of the steel and concrete capacities summation. The bending performance of the section was divided into three different steps. Figure 1 shows that the bending performances based on the comparison between ρ_b , ρ_t and availability of compressive re-bar get divided into three different performance types (P_1 , P_2 , P_3).

In the following process, comparison with available and minimum allowed re-bar categorizes the section to other subcategories $P_{(1,2,3),(1,2)}$. This category indicates the type of Standard formulas for calculating neutral lines and capacities of each type of section. Finally, a comparison between the amounts of the calculated strains indicates some coefficients which, based on the standard, should be multiplied by the bending results. Numbers showed this last step of calculations as ($P_{(1,2,3),(1,2),(1,2)}$). Finally, the following discussions interfaced the standard codes as a MATLAB function to all ML operations.

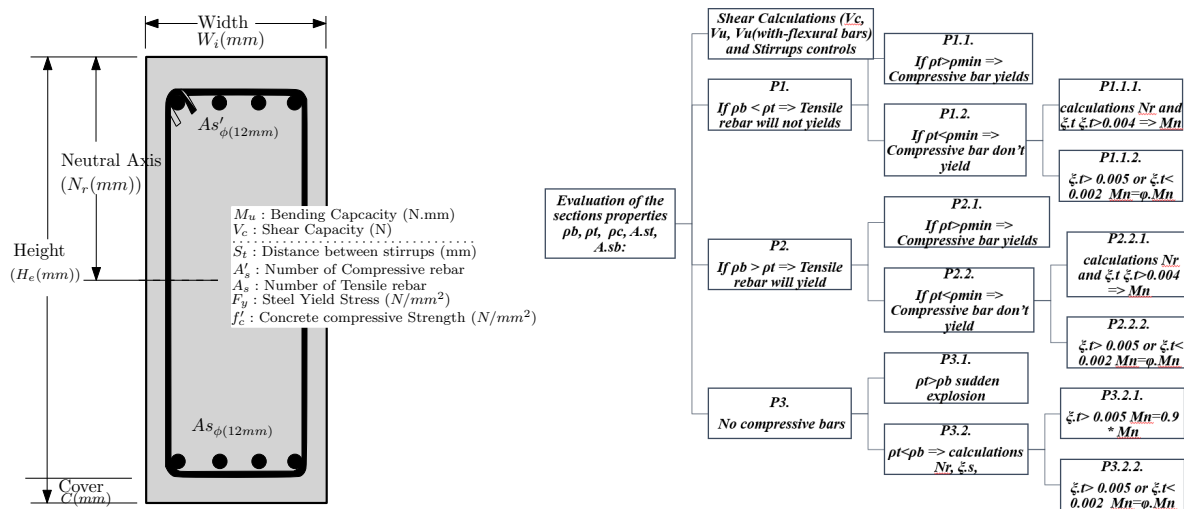


Figure 1. Section's Calculation Chart

The results of several series of parametric calculations will produce different databases (Look – up table) to be inferred with operators, and the operators' results will be compared to the direct ACI coding. The aim of the study is not to discuss the performance of the beams, and the accuracy of the parametric calculations results is not decisive. Nonetheless, to have a practical example, the accuracy of the coded ACI function was compared to other references. Table 1 compares the codes' results and calculated amounts in references. Although not all of the references use ACI, the results displayed similar results for shear and bending capacities, proving the accuracy of the coded ACI and the similarity between the referenced standards.

Table 1. Verification of the accuracy of the coded beam calculations in comparison to the references

	H_e	W_t	f_c	f_y	A_s'	A_s	S_t	$Perf$	N_r	M_u	V_u	Ref	
No	mm	mm	N/mm ²	N/mm ²	mm	mm	mm	name	mm	N.mm	N	Name	Amount
1	500	350	30	400	942	4825	-	2.1.2	174-174	7.1e08	-	11-5 [26]	Mu:7.1e08
2	300	350	21	400	942	3217	-	1.1.2	149-145	2.1e08	-	12-5 [26]	Mu:1.9e08
3	500	400	35	400	-	1357	-	3.2.1	47	2.6e08	-	5-5 [26]	Mu:2.4e08
4	350	200	32	500	-	804.2	-	3.2.1	75-74	1.3e08	-	[27]	Mu:1.2e08
5	3500	2000	32	500	-	8040	-	3.2.1	75-74	1.4e10	-	[27]	Mu:1.4e10
6	500	300	35	400	-	-	$\phi 10@80$	-	-	-	3.53e05	1-7 [1]	Vu:3.5e05
7	520	350	21	300	-	-	$\phi 12@125$	-	-	-	2.92e05	3-7 [26]	Vu:2.8e05
8	500	300	30	400	1231	2412	-	2.2.1	72	-	3.95e8	3-8 [28]	Mu:3.9e08
9	500	450	25	400	-	-	$\phi 12@100$	-	-	-	3.97e05	[29]	Vu:4.1e05
10	650	300	21	275	-	3220	-	3.1.1	-	5.0e08	-	[27]	Mu:4.8e08

The selection properties, including the Height (H_e) and Width (W_i), besides concrete and steel properties along with the steel mounts, are shown in the first row of Table 1. The N_r in this Table shows the depth of the neutral line, calculated by codes and references for bending calculations. In selection, examples with different performance types were selected. For instance, the bending capacity of the first section (No.1 Table, 1), by the codes and in the reference, are $7.1e08N.mm$ and $7.08e08N.mm$. Likewise, the comparison of shear capacities in example 6 shows $3.53N$ and $3.5N$. Generally, it was shown that the maximum difference between the results of the codes and references is 2%, proving the accuracy of the codes.

Coding, the Machine Learning operators:

Despite the different parameters for tuning, Machine Learning operators can be selected due to the limitation of the size of the paper, and the focus of the current study does not intensely discuss other possible variants of Coding. The codes were initially developed by other researchers/engineers and, alone as a possibility, were selected to prepare an overview for the structural engineers. This study aims to give an outline to the structural engineer for finding the solution for the desirable interpretation of their numerical and experimental results and does not aim to propose the best commands; the developed codes in this study can further researchers be replaced by other programming languages or can be optimized for promoting their performances. Alone, some parts of the developed MATLAB codes to give an overview will be briefly mentioned in each section.

2.2. Problem Definitions (Preparation of Look – up Table):

For discussing different ML techniques, all following sections use the proved codes as a MATLAB function called *ParsSolution.m*. This function can para-metrically calculate different ranges of the variables based on ACI. It imports all of the section parameters, including Height($H_e(mm)$), Width ($W_i(mm)$), concrete compressive strength ($f_c(n/mm^2)$), steel yield strength ($f_y(n/mm^2)$), Compressive re-bar ($A'_s(mm^2)$), tensile re-bar ($A_s(mm^2)$) and calculates the shear ($V_c(N.m/m)$) and bending capacity ($M_u(n.mm)$).

ParsSolution :

$[V_c, M_u, P] = \text{BeamParsSolution1}(LT_{Si}, Par_{No}, As, A'_s, ST, Fc, Fy, He, Wi);$

Figure 2 shows a series of analyses, including the properties of 50 different beam sections. The input was produced randomly (*randi*) but in a specific range for each variable (Table 2). The influences of these variables on the bending and shear capacities can be found in the last blue diagrams. The type of performance in the bar chart (P) is shown in blue, red and orange. For instance, the first section displayed 2 for the Height of blue and red also 1 for the orange part on top of the bar, which means the performance of it belongs to category (2.2.1). It displays the tensile re-bar yields, compressive does not yield, and the strain is higher than 0.004 (Chart in Figure 1).

Table 2. Selected Range of Changes in the Variables

	H_e	W_i	f_c	f_y	A'_s	A_s	S_t
No	mm	mm	N/mm^2	N/mm^2	mm^2	mm^2	mm
Range	[150-1000]	[100-300]	[20-70]	[200-600]	[0-393]	[158-1570]	[500-50]

In addition to the parameter and the ranges, *ParsSolution* demands two other criteria. The first criterion informs *ParsSolution* about the number of parameters selected in each calculation series (Par_{No}). It means all different features of the section can be selected parametrically (i.e. $A_s, A'_s, S_T, f_c, F_y, H_e, W_i(Par_{No} = 7)$) or just some (e.g. $A_s, A'_s, S_T(Par_{No} = 3)$). The second criterion is the size of the Look – up table (LT_{Si}) or, in other words, the number of calculated sections. The number of parameters (Par_{No}) and the size of the Look – up table (LT_{Si}) as the primary measure was regarded for dissecting the performances of ML Operator (MLO) in the following sections. An MLO,

which can delegate the results with higher accuracy and a lower amount of initial data (low LT_{Si}), has a more desirable performance. It would show its considerable importance when the difficulties in data production are considered. Indifferent filed, the data can be collected by quaternary (asking from experts), numerical analysis or experimental test, while in structural engineering, increasing the LT_{Si} , especially in experimental studies, is complex and costly.

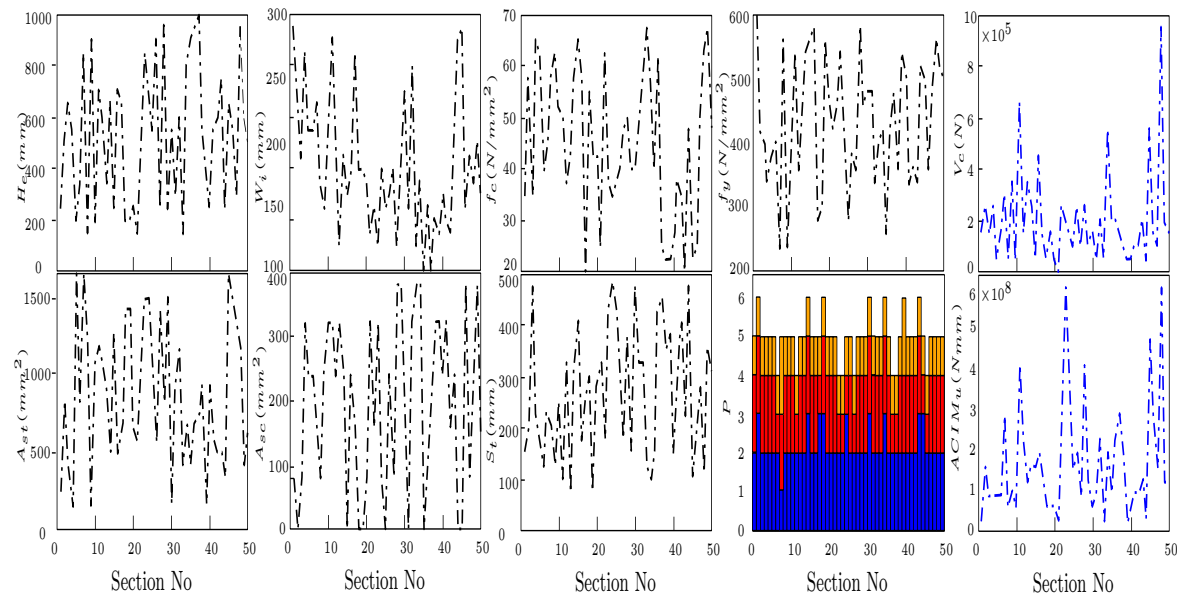


Figure 2. Random features of the 50 beam sections and the capacity based on ACI

2.3. Parametric Evaluation of the beams:

In this study, beam capacity was selected as one of the easy structural calculations to evaluate the MLO. Likewise, *ParsSoltion* can be used for independent parametric evaluations. It could also be connected to an optimization algorithm. Such a function, similar to software (e.g. SAP2000 Section Designer), can easily calculate the capacity of sections with different parameters. Hence, different operators can use it for multi-analysis to regard different aspects of the section's performances and perform statistical evaluations and sensitivity analysis by MATLAB Simulink.

One simple evaluation without coding Sensitivity Analysis is evaluating the influence of each parameter on M_u & V_c . Table 3, shows the influence of increasing the parameters twice (2.00) on this range's bending and shear capacities. This evaluation showed that increasing the amount of all parameters twice increases the shear and bending capacity 7.18 times and 8.14 times. It means that bending capacity is more sensitive (No.1). That is, if the influences of stirrups are ignored despite no changes in increasing of M_u , the changes in V_c were reduced to 4.74 (No.2). It means in the selected range (Table 2), distances between the stirrups influence the shear capacity up to 34%. The distance between the stirrups can not be due to the standard's limitations (e.g. for keeping the ductility of beams) limitless changed, but the range was accepted as a parametric study.

In this range, Height has a similar influence on shear and bending and has a linear relation (No.3). Width has a higher effect on shear rather than bending (No.4). The changes in bending are not linear, and the width effect on bending in one step gets higher (due to changes in P). F_c also shows a similar low influence on the V_c and M_u ; it means increasing the concrete properties regarding the *Costs* in comparison to the changes in other parameters, such as high is not economical (No.5). Generally, the lowest influence was shown by compressive re-bar (No.7) and higher by Height and tensile re-bars (No.8).

Table 3. influence of parameters changes on the beam capacities (all numbers are percentages)

No	H_e (%)	W_i (%)	f_c (%)	f_y (%)	$A_{s'}$ (%)	A_s (%)	S_t (%)	M_u (%)	V_c (%)
1.	2,00	2,00	2,00	2,00	2,00	2,00	2,00	8,15	7,18
2.	2,00	2,00	2,00	2,00	2,00	2,00	1,00	8,15	4,74
3.	2,00	1,00	1,00	1,00	1,00	1,00	1,00	2,03	2,07
4.	1,00	2,00	1,00	1,00	1,00	1,00	1,00	1,08	1,32
5.	1,00	1,00	2,00	1,00	1,00	1,00	1,00	1,07	1,14
6.	1,00	1,00	1,00	2,00	1,00	1,00	1,00	1,95	1,54
7.	1,00	1,00	1,00	1,00	2,00	1,00	1,00	1,00	1,00
8.	1,00	1,00	1,00	1,00	1,00	2,00	1,00	1,95	1,00
9.	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00

3. Machine Learning Operatos (MLO)

To discuss the performance of the following MLO, the specific range is based on the expected dimensions and properties of the material used in all following investigations (Table 2). In the prepared *Look – up* table, the parameters linearly increased. Based on the ACI calculation, the first section selects Fuzzy Logic for delegating, interpreting, and using data (*Look – up* table). Likewise, the Neural Network and ANFIS will repeat the same process. Additionally, their sensitivity to the size of LT_{Si} and Par_{No} will be investigated. Finally, their performances regarding the different sizes of the *Look – up* table and the number of parameters will be compared.

3.1. Fuzzy Log (FL):

Since MATLAB GUI canNot generate codes to accept different '*input*' and '*output*' parameters, codes for making the interpreter of the *Look – up* table and *MembershipFounctions* were developed. In the membership function, two types of functions were regarded, including ('*trimf*' and '*gaussmf*') to be adjusted by the operator ($Fac \in 1, 2$). Assigning the inputs to outputs for such a range of data was not effortless for making the rules. Hence, a '*for*' loop based on the *Look – up* table was developed to develop adaptable codes for all essential parameters in designing the beams. In which the '*1*' as the Variables and Weight for all rows of the Rule table was regarded. Then, the lookup table and '*mamdani*' was used to make the '*fis*' ($fis = newfis('LookupTableFIS', 'mamdani')$). Furthermore, '*input*', '*output*' etc., by *addvar* assigned to the '*fis*', and finalized by adding the rules.

Figure 3 displays an example of the series of section calculations. In this example, H_e , f_y , A_s as the variables were selected, and 250 sections were calculated to make the *Look – up* table ($LT_{Si} = 250$). This Figure shows how the parameters change linearly to increase the bending and shear capacity based on ACI. The same input in Fuzzy, the *Look – up* table, is used, and the two diagrams show the Fuzzy delegating results. In these diagrams, the flat parameters indicate the not chosen variable, which means the $Par_{No} = 3$. The shown parameter in the last two diagrams, in blue, is the shear and bending capacity based on ACI. Likewise, red diagrams display the results of the Fuzzy operator. In these diagrams, the yellow bars are the difference between the targets (diversity between input from ACI codes and Fuzzy outputs). A similar example was made, utilizing randomized input and, accordingly, chaotic results of Fuzzy and ACI. This example shows the capacity of FL to interpret the not-sorted input. This capability shows its importance when several results of the studies against the current study regarding the type of the input parameters canNot be sorted, Figure 4.

In another example in which just H_e as the variable and ($LT_{Si} = 500$) as the table size was selected, the maximum Error (Differences between Fuzzy and Standard happened divided by the ACI capacities in each section), was limited to 16&17%. In this example, with three parameters, Error raised to '50%' and '59%' for the calculation of shear and bending, respectively, displaying a considerably low accuracy. In Figure 3, the wave format of these Fuzzy diagrams and Errors shows that the error amount is not the same in different sections. The changes in Error are not related to beam section performance but come from the nature of Fuzzy. In another example with '*1*' parameter and '*100*' *Look – up* table size, the Error in shear and bending capacity was '17&19%', compared to the previous example with

'500' Look – up table size is just '1%' percent less accurate. It means up to some limits; despite its positive influence, increasing the size considerably raises the analyzing *Costs*, canNot significantly improve the accuracy. Nonetheless, adjusting the Fuzzy codes and changes in several parameters or the Look – up table size can influence the results more. The influences of these adjusting components should be evaluated para-metrically.

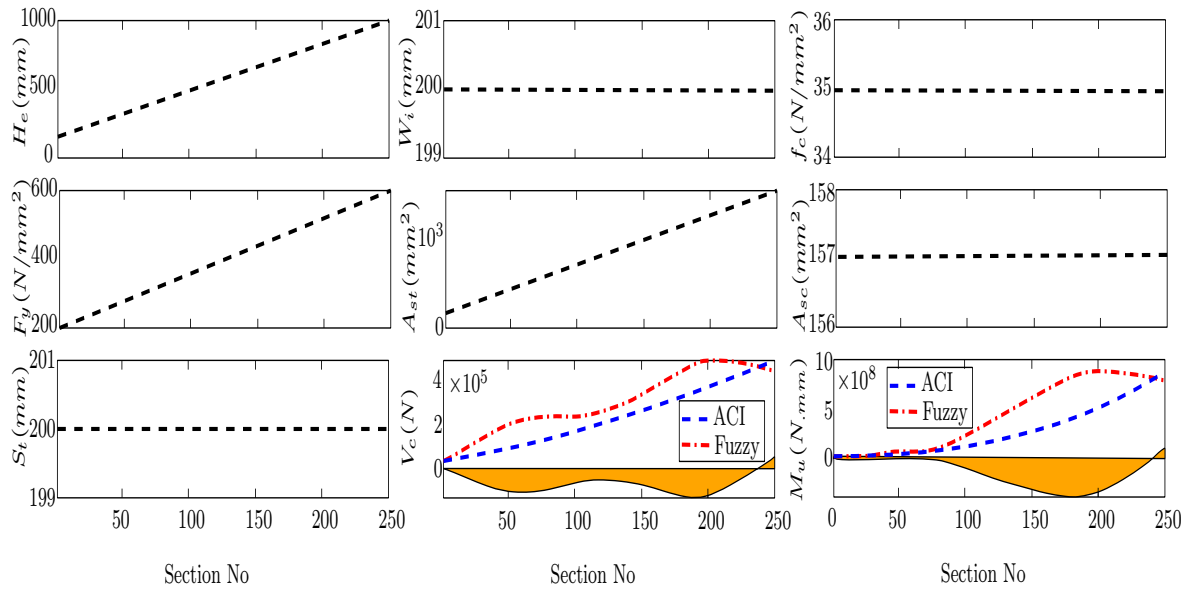


Figure 3. Results of Fuzzy operation with 250, table size and three variables and 'gaussmf'

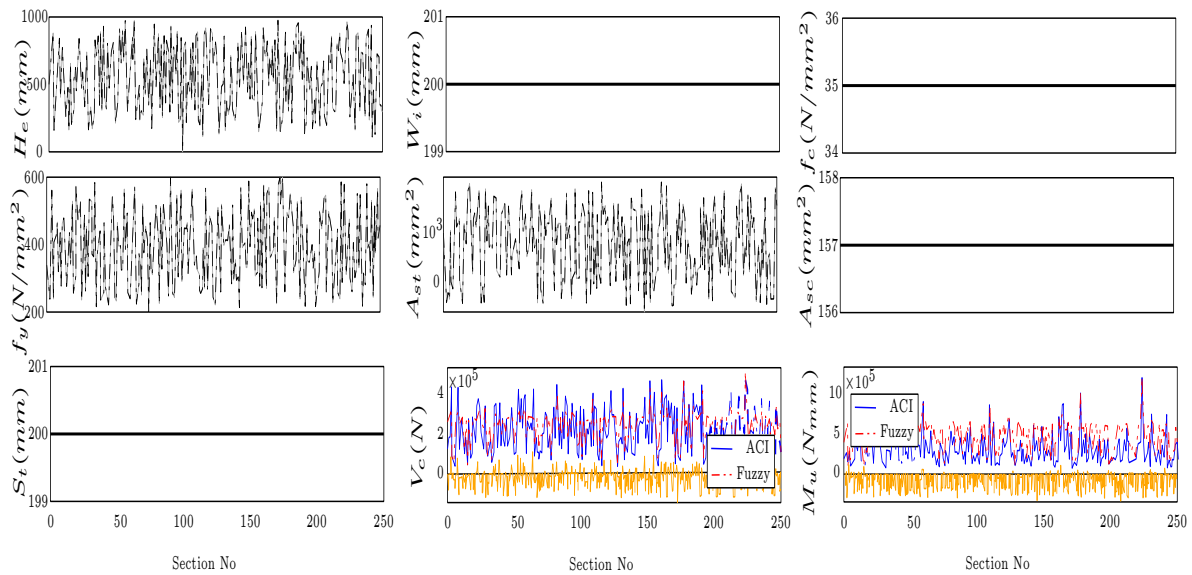


Figure 4. Results of Fuzzy operation with '250', table size and '3' variables and 'gaussmf'

Two more loops were added to the codes to evaluate the capability of adjusting the MLO. The first 'for' loop operates the MLO by a different range of initial data in the Look – up table. It means if the LT_{Si} is for instance 1208, ParsolutionSolution function, in the specific range calculates 1208, beam sections to be referenced by Fuzzy operator $LT_{Si} \in [20, 416, 812, 1208, 1604, 2000]$. This evaluation shows how much data (including all parameters) is needed for the selected beam example. The other parameters, such as a nest (loop), were also regarded. This loop, in five different steps, changes the number of parameters. This additional 'for' loop (PARA) in five different steps selects different

parameters, decided based on the results of parametric evaluations (efficient of the parameter on the capacity), Table 3. It was written by a *switch case* or function, demanding $5 \times 6 = 30$, general operations, while each one may have up to 2000, beam calculations.

$$PARA \in [1, 2, 3, 5, 7] \equiv [(H_e), (H_e, A_s), (H_e, A_s, F_y), (H_e, A_s, F_y, W_i, S_T), (H_e, A_s, F_y, W_i, S_T, F_c, A_s')].$$

The shear and bending Errors and the Standard deviation results are shown in Figure 5. Since the Costs of analysis is an important factor for evaluating the desirability of an operator's performance, the duration of the process was also measured and documented in these diagrams. In Figure 5, all analyses with any ' Par'_{No} ' reduced the Error by increasing the LT_{Si} , indicating the capability of them in case of suitable size of *Look-up* Table. The sensitivity of Error to the amount of LT_{Si} , by increasing that reduced. In all of the Errors and standard deviation, disregarding the Par_{No} , the accuracy is considerably low when the $LT_{Si} < 250$. Increasing the ' Par'_{No} ' increases the analysis Costs, but the influence of increasing the Par_{No} on the Errors is not linear. For instance, $Par_{No} = 4$ has higher accuracy than $Par_{No} = 3$. This influence might be related to the value of Par_{No} and the type of the membership functions. Despite the high probability of low accuracy in using FL in delegating the ACI codes, it can operate more stable (with less scatterly results) with suitable adjustment and input.

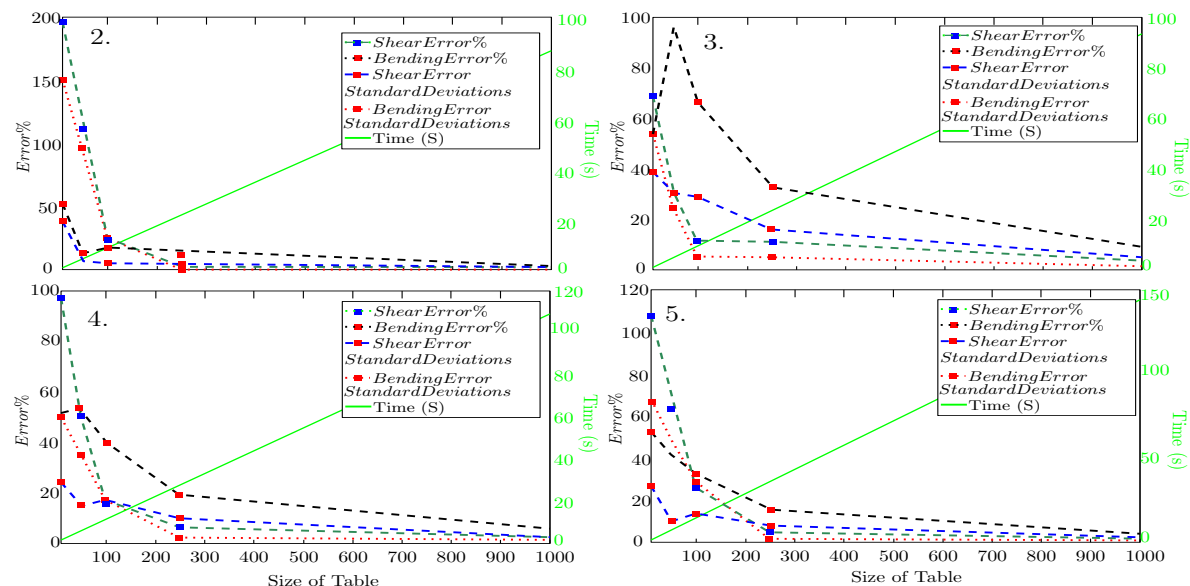


Figure 5. Results of parametric adjustment of Fuzzy operator, $Par_{No} = [2, 3, 4, 5]$, $LT_{Si} = [20 - 1000]$

3.2. Neural Network (NN):

To compare NN and FL, Figure 6 displays the same examples solved and delegated by Neural Network codes. The codes of Neural Networks were developed in MATLAB. In order to fit the Network, as the activation operators, Sigmoid activation ('*sigmoid*') and Hyperbolic tangent Sigmoid ('*tansig*'), along with Linear function ('*purelin*') for transferring the input were selected. In addition to inputs, targets and the number of the hidden layer (NHL), these operators were (by '*newff*' command) assigned to the Network. For choosing, 'in/out-put' and 're/post-processing' functions were coded. Likewise, operators were coded for removing rows with constant values ('*removeconstantrows*') and mapping rows with minimum and maximum values ('*mapminmax*'). Furthermore, '*dividerand*' for random division of data and every sample ('*sample*') was picked, in which 70% of the data for training and equal percent for validation and tests of the Network selected (30%). Additionally, for choosing and helping the train function, Levenberg-Marquardt ('*trainlm*') and Mean squared Error ('*mse*') were assigned to the Network performance function. In the codes, the number of the hidden layer (NHL) as an adjustable parameter was regarded (e.g. $NHL = 5$).

The same types of *Look – up* tables as the input imported by Fuzzy were prepared in this section to be interpreted by *NN*. Figure 6 illustrates the results of these two examples with sorted and randomized input types. In this example, the size of the Table and parameters' number are 250 & 3 respectively, ($LT_{Si} = 250, Par_{No} = 3$). Likewise, the red and blue diagrams show the *NN* and ACI results, while the differences between *NN* and ACI are shown in the yellow bar. The remarkably higher accuracy of *NN* compared to FL in both delegated results is visible. Generally, bending capacity (M_u), compared to shear (V_c), has a higher number of effective parameters, which causes the relation between feature and output to be more nonlinear. The Error results in these two examples show that *NN* can present a more exact bending capacity in sorted inputs. Despite this, randomized input bending diagrams show less exact. The *NN* operator also calculated several other examples based on and in compression to FL; it can be calculated that the capability of the Neural Network for delegating the ACI calculation is considerably higher.

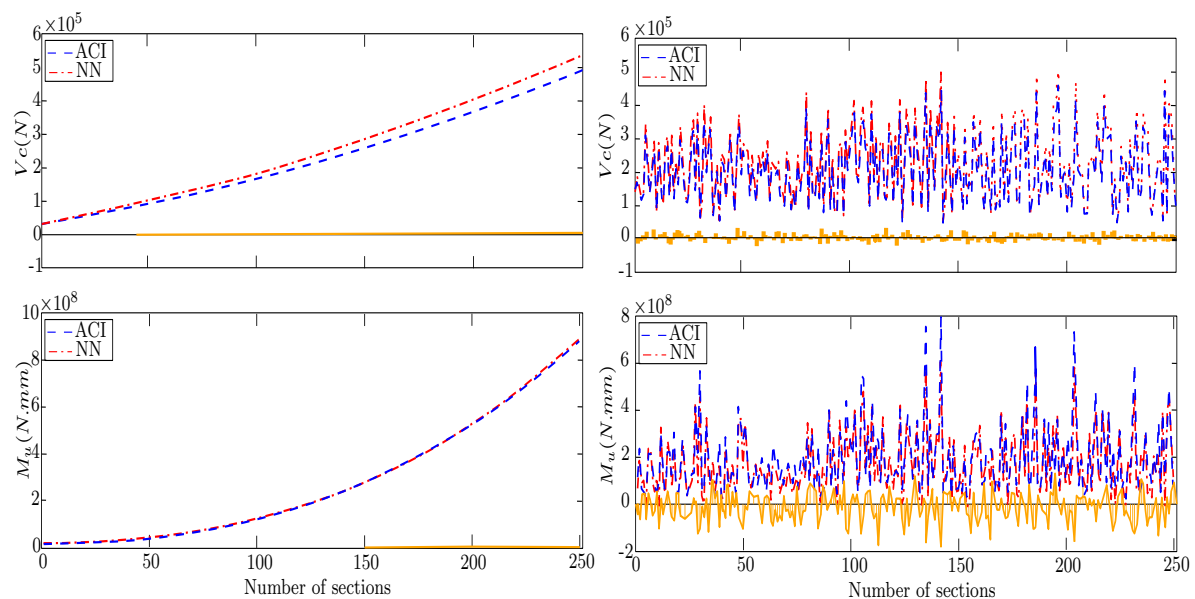


Figure 6. Neural Network Results of sorted (left) and randomize (right) inputs, $LT_{Si} = 250, Par_{No} = 3$

One of the main components of the Neural Network is the number of the hidden layer (NHL). In a series of example same problem ($LT_{Si} = 250, Par_{No} = 3$) with different number of hidden layer ($NHL = [1 - 10]$) was repeated, Figure 7. It can be seen that compared to the chaotic nature of the coded *NN*, the NHL cannot show a strong influence on the results of this example. Based on multiple examples tested by the current study, it can be concluded that increasing the number of layers for more than three layers has more influence on the analysis *Costs* than the amount of Errors.

Nonetheless, the main highlight of using *NN* is the chaotic amount of accuracy. In other words, despite the input parameters, in a series of examples were increased gradually, the Error amounts in some were almost zero, and in some, they were several times higher than the maximum amounts of bending and shear capacity. To illustrate, a repetition of the same problem was documented in Figure 7. In these examples ($LT_{Si} = 100, Par_{No} = 3, NHL = 10$), the Error of delegating the shear and bending beside the Errors Standard deviation was displayed. It can be seen that the Error in examples 1&6 is almost zero while, in example 2, shear is 30, times higher than ACI calculations. Despite FL showing, in general, lower accuracy, such an issue (chaotic results) was not detected. Some examples have acceptable performance in bending Error and are high in shear. Hence, in an operator like *NN* with low robustness, the Error amounts in objects may differ considerably. The high probability of facing low accuracy in the results of FL and, despite high capability, chaotic results of *NN* indicates the necessity of developing a method using the potentials of Fuzzy and *NN* while solving their issues.

It should be remembered that in all MLO in this study against other Multi-Objective optimization algorithms, the objects (here V_c & M_u) are calculated separately in two different parallel interpreting processes.

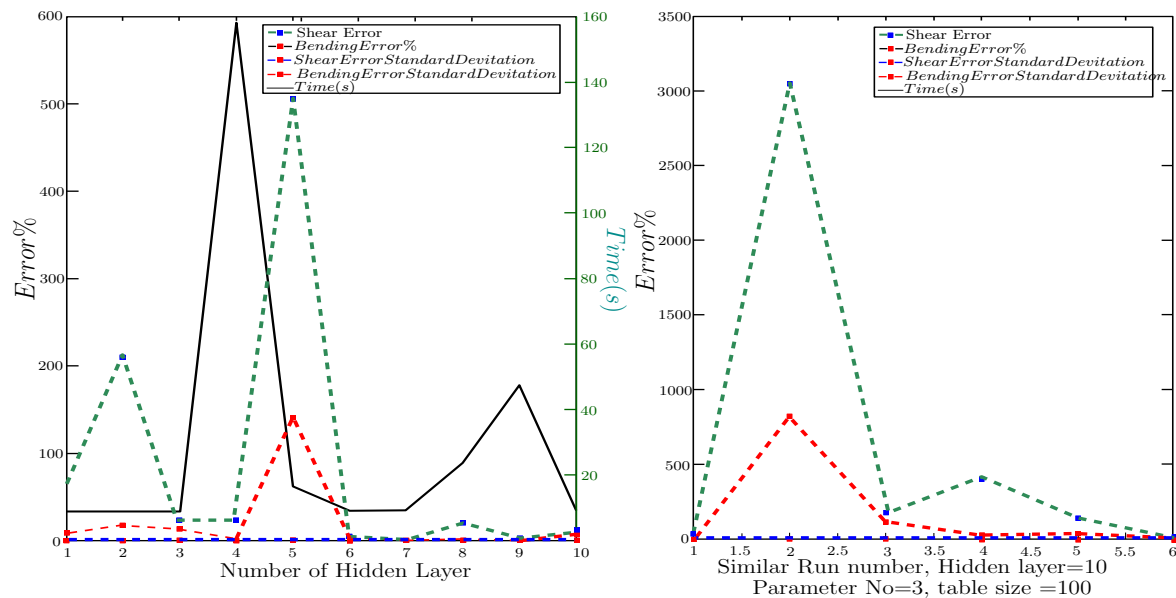


Figure 7. Neural Network Results, for (right) discussing the influence of NHL and (left) robustness of the results in repetition of an exact problem.

3.3. Adaptive Neuro-Fuzzy-Inference system (ANFIS):

Adaptive Neuro-Fuzzy Inference System is a hybrid artificial intelligence model that combines the strengths of FL and Neural Networks. This operator was similarly developed in MATLAB, regarding the same feature coded for Fuzzy and NN. As the *TrainData*, features of the beam section and the capacity of the beams calculated by ACI codes were selected. Inputs by Gaussian membership function '*gaussmf*' and linear output types in '5' layers were selected ('*linear*'). To remove and generate the system, subtractive clustering was chosen (*Sub.Clustering*). The options by '*inputdlg*' were assigned to create dialogue box '*PARAMS*'. The data, including the inputs, outputs and training data, were assigned to the '*fis*', by '*genfis2*', selected FIS generation approach. During the initial investigations, '*Sub.Clustering*' with '*genfis2*' and '*genfis3*', both tried and due to this study's low influence and concentration, '*genfis3*' were in preparation for the results used. The cluster number was '10', and the Exponent matrix was partitioned to '2'. The number of maximum iterations and the number of Epochs was '100', and the minimum improvement was ' $1e - 5$ '. In these studies, the initial step was 0.01, and the step size was 0.9, while 1.1 was the step size increasing rate and 0 as the error target was assigned.

Like the previous section, as the first examples, three parameters and the sorted and randomized inputs were selected to compare directly. Figure 8 shows the results of ANFIS in the delegation of beam calculations. It can be seen that the Error in the calculation of bending and shear capacities is deficient (approaching zeros). Generally, ANFIS has a robust homogeneous performance. The range of examples made by ANFIS displayed higher accuracy in calculating the bending in all examples with different LT_{Si} and Par_{No} , was repeated in average the bending Error is less than shear Error. The shear Error increases, especially with a low number of LT_{Si} and Par_{No} . In this investigation, different example series were solved by ANFIS. Based on an example with $LT_{Si} = 2400$ when the number of parameters is 7, the calculation accuracy is $\approx 6 \times 10^4$ times lower than the same operation with a 1 parameter. Despite this, the Error range in both is less than 1.3% and ignorable, Figure 9.

Similar results were experienced for the Standards deviation V_c & M_u . It might be concluded that the capability of this operator is more suitable for complex structural issues. If the duration of the process as the analyzing *Costs* is regarded, increasing the LT_{Si} is the main factor for raising the

Costs. Likewise, increasing the Par_{No} , from 1 to 7, when $LT_{Si} = 800$ in average increased the analyzing Costs to 450%. ANFIS, in repeating the same problem, had the same results, proving its analytical stability. Additionally, increasing the size of LT_{Si} in sorted input type proved its low positive influence on reducing the Error, but randomized input increased the Error.

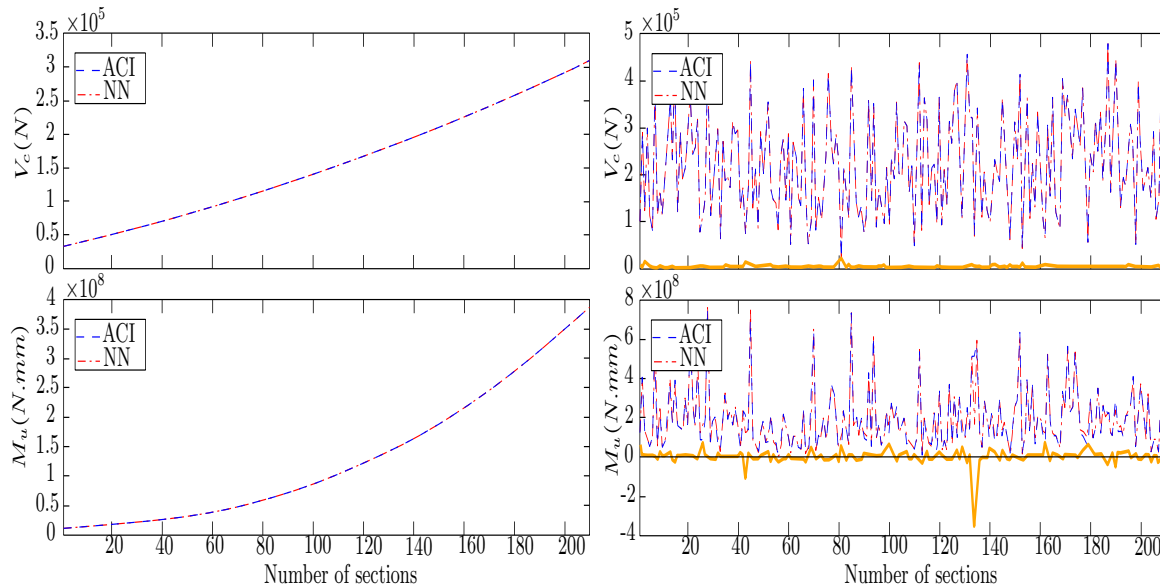


Figure 8. ANFIS results of sorted (left) and randomize (right) inputs, $LT_{Si} = 220, Par_{No} = 3$

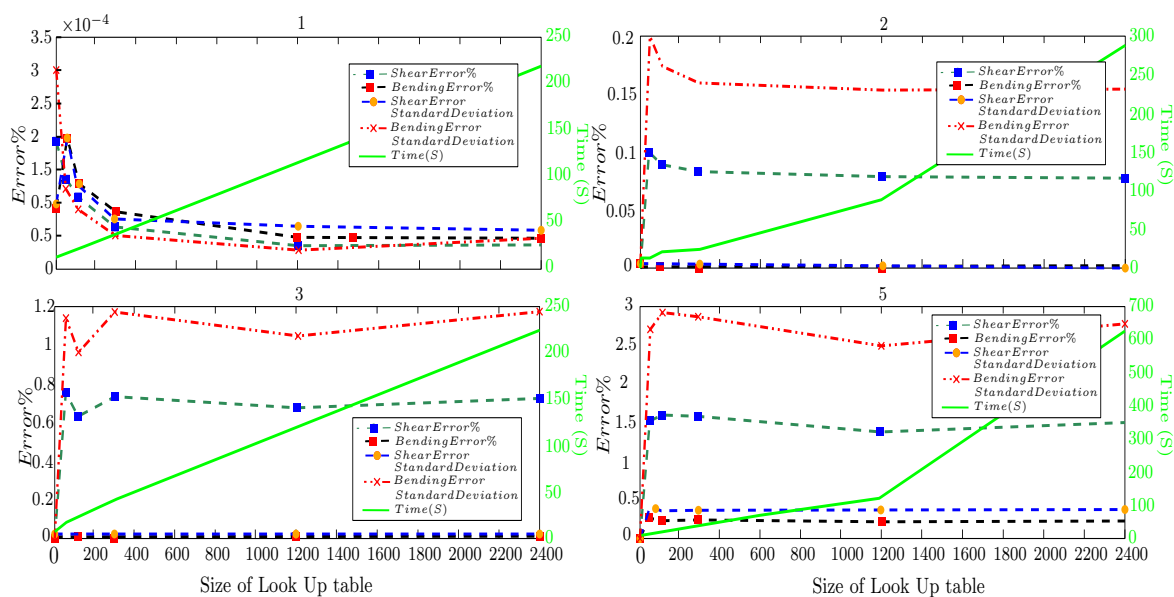


Figure 9. Performance of ANFIS $LT_{Si} = [15 - 2400], Par_{No} = [1 - 5]$

3.4. Comparison between the three MLO:

Figure 10 (Left) displays a series of examples based on a randomized *Look-up* table, in which LT_{Si} range increases up to 1000. The Error of V_C & M_u , in addition to the analyzing duration by increasing the LT_{Si} increases, is entirely visible in NN diagrams. ANFIS, in addition to having higher accuracy, has a robust performance; it did not indicate significant changes in V_C & M_u capacity representation of the ACI bending and shear capacity with less than 1% differences. In this range, the analyzing Costs of Fuzzy and Neural Networks are similar and for the same problems are almost 66% of ANFIS. The

other examples showed higher differences between the *Costs* of FL and NN compared to ANFIS by increasing the LT_{Si} , Par_{No} & NHL .

Likewise, the duration of Fuzzy has an exponential relation with Par_{No} and LT_{Si} . It means that in representing the small problem, FL performs faster than NN and ANFIS. As mentioned, this study just selected MATLAB as a proper platform, and rewriting the codes in primary programming languages (e.g. C++) can considerably reduce the analyzing *Costs*. It should be mentioned that the time for preparing the *Look-up* table was not considered when discussing the analyzing *Costs*. Accuracy (Error), duration (time) and robustness of the operator are the three selected. Hence, Figure 10 (Right) contrasts the MLO operations' robustness by running the same problem with times to compare directly.

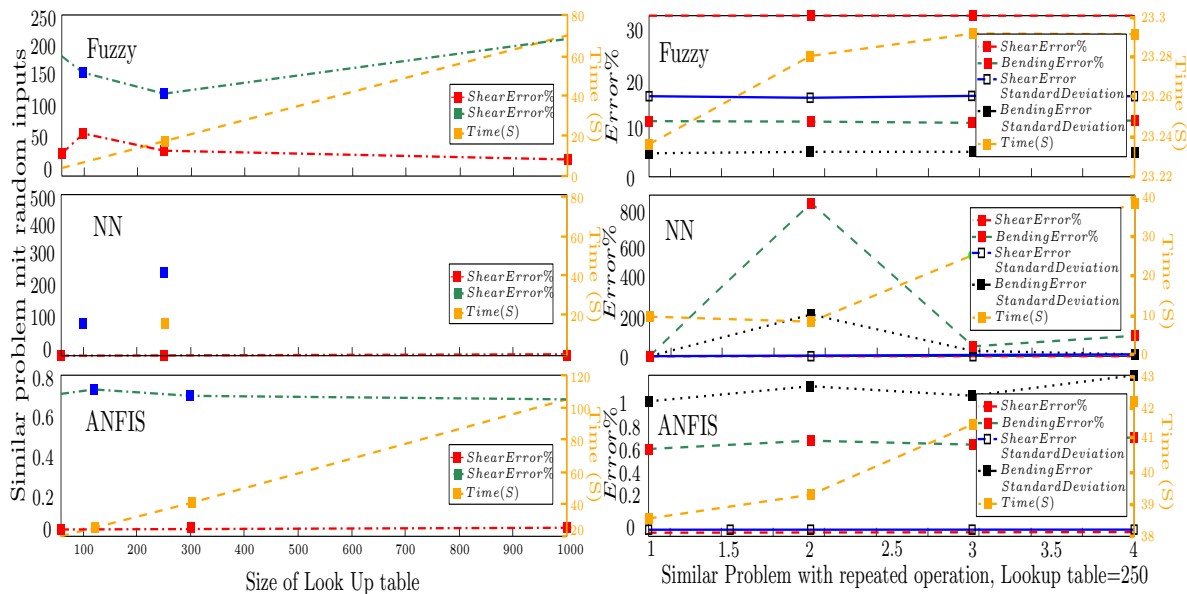


Figure 10. Time comparison as the *Costs* for regarding different MLO, number of parameters and size of the *Look-up* table, without the duration of time needed for the production of the *Look-up* table and stability of the results on repetition of the similar problems

A general contrast between the performance of the MLO was made by fitting 3D surfaces to the results. This surface is also fitting managed in MATLAB, while the size of the *Look-up* table (LT_{Si}) and parameter Number (Par_{No}), along with the average amount of shear and bending amounts, were separately selected. Figure 11 displays the accuracy of all three coded Machine Learning operators (including FL, NN and ANFIS). The top diagrams of Figure 11 illustrate the bending and shear errors in the down row. The results of various analyses were shown next to each other in these diagrams. The average amounts of shear and bending differences (Error between ACI and MLO), regarding the range of $LT_{Si} = [15 - 2400]$, $Par_{No} = [1 - 5]$, by fitting surfaces on them were plotted in MATLAB. Generally, one of the main usages of the MLO is curve fitting. Despite the higher power of MLO for fitting continuous surfaces to the discrete data and synergy between curve fitting and all discussed MLO, due to simpleness, in this diagram, simple codes in MATLAB were used.

FL has the lowest accuracy despite the highest stability in the calculations and results. The amount of Error in the calculation of the bending capacity gets lower by reducing the size of the *Look-up* Table and increasing the number of parameters. On average, the bending calculated by FL and ACI in this range maximum differ $\pm 53.8\%$, while for shear, this difference increases to $\pm 119.8\%$. Generally, FL initial coding starts with defining the Linguistic Variables. Linguistic Variables include 1- The name of the variable (e.g. A_s Height Width), 2- Amount of variables for each group (e.g. A_s in $[0.2, 0.5, 0]$), 3- Range of the changes in variables (e.g. Height of a beam), and 4-Fuzzy set. Hence, each Linguistic Variable can be defined. It enables the FL to convert the initial digits to Fuzzy parameters (called Fuzzification) and, after the operation, to digits (called Defuzzification). This will be continued by

developing the Rules for the logical implications needed. The conversion of the digits to logical and Linguistic variables enables the FL to import different types of data (e.g. Verbal information of a questionnaire form) but can be the reason for reducing the accuracy of the results (increasing the Error).

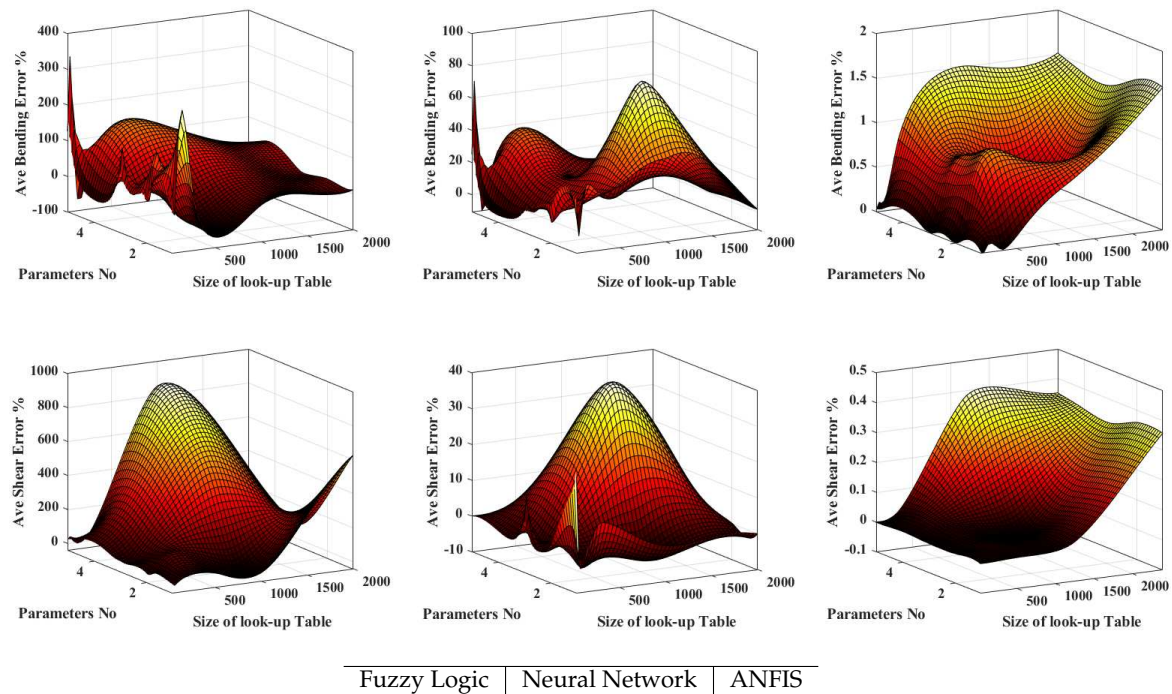


Figure 11. comparison between the performances of the discussed MLO, by fitting surfaces to shear and bending Errors, while $LT_{Si} = [15 - 2000]$, $Par_{No} = [1 - 5]$ as the parameter ranges were regarded.

The results of *NN* on the average show (± 15.7 , $\pm 7.52\%$) Error for shear and bending, respectively, which is considerably lower than FL. Likewise, increasing the size of the *Look-up Table* or, in other words, preparation of more initial data (e.g. from numerical or experimental structural analyses) can reduce the Error and scattered results faced more in calculating bending capacities.

Due to the issues of FL and *NN* (i.e. low accuracy and analyzing stability), this study also developed ANFIS codes. The results of this operator, which has the advantages of FL and *NN*, are shown in the last column of Figure 11. Against FL and *NN*, enlarging the *Look-up table* can considerably increase the analysis *Costs* and does not enhance the accuracy. Increasing the number of parameters (beams sections feature) also reduces accuracy. Additionally, its performance in the calculation of shear compared to bending calculation is more accurate. ANFIS, with all sizes of inputs, should have entirely acceptable performance while, on average, the differences between the ACI direct calculation and ANFIS for bending and shear are (± 0.83 , $\pm 0.11\%$). The maximum experienced average Error, in general, is less than 1.5%, which indicates the capability of ANFIS for delegating standard designs.

4. Conclusion:

This study discussed the capability of different Machine Learning techniques in delegating conventional structural calculations. In other words, the study wants to investigate the possibility of replacing a design engineer with a Machine Learning operator. For such an evaluation and further practical utilization, different structural problems can be addressed, including structural analysis and design. Likewise, different Machine Learning methods can be selected. In the current study, as the structural problem, the Standard calculation of reinforced concrete beam section based on ACI was chosen. A wide range of computational operators was in the past decades developed (e.g. for

deep learning and optimizations). The selected operators by this study, which might be regarded as a part of Machine Learning techniques, are Fuzzy Logic (FL), Neural Network (NN) and (Adaptive Neuro-Fuzzy Inference System) ANFIS, which were separated and entirely coded in MATLAB. After proving the accuracy of the coded standard by comparing it to references, since the operators need training data, *Look – up* tables based on Standard parametric calculations were prepared. The relation between the Errors of each operator (i.e. comparison to direct ACI-based calculations), besides the number of parameters and size of the *Look – up* tables, was also discussed.

Some highlighted results:

- All coded three operators are adaptable and capable of delegating the standard parametric calculation.
- The performance of all three operators depends on the number of parameters and size of training data.
- Fuzzy Logic has the lowest accuracy and most stable operation, shown by repetition of the same problems.
- The accuracy of the NN compared to FL is considerably higher, and the speed of the operation (analyzing *Costs*) is slightly lower.
- Despite the high capacity of the Neural Network, the results of its operation on the same problem are less stable.
- Increasing the size of a *Look – up* table and reducing the number of parameters can significantly improve the accuracy of the FL and NN results. The scatter performance of NN in bending calculation is considerably higher.
- High probability of facing low accuracy in the results of FL and, despite high capability, chaotic results of NN indicates the necessity of developing a method using the potentials of Fuzzy and NN while solving their issues.
- Overall, ANFIS provides a robust framework for modelling systems for calculating beam capacities, indicating the ability to handle uncertainty and learn from structural studies data.
- ANFIS can delegate the ACI calculation, with, on average, less than $\pm 1\%$ difference.
- More complex subjects, such as larger input size or parameter number, can slightly reduce the precision of ANFIS.

Some highlighted Out Look:

- Developing Parsolutions, like what in this study was for preparation of the *Look – up* table coded for other structural subjects and proper parametric evaluation (e.g. by sensitivity analyses), assist the engineers in the industry to design optimally.
- To increase the accuracy and reduce the analyzing *Costs*, the developed FL, NN and ANFIS codes should be developed in other platforms and primary Coding languages for preparing software.
- The developed MLO was used with the most common adjustments and parameters. The Coding options and parameters should be tuned in the next step for more desirable operation.

Author Contributions: Conceptualization, AB and HK; methodology, AB and NB; software; writing original draft preparation, AB; writing review and editing, NB and AB; visualization, NB; supervision, HK, NB and AB; project administration, HK; All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable

Informed Consent Statement: Not applicable

Data Availability Statement: Not applicable

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FL	Fuzzy Logic
NN	Nural Network
$ANFIS$	Adaptive Neuro-Fuzzy Inference System
MLO	Machine Learning Operators
NHL	the number of the hidden layer
W_i	Width of the concrete Section
H_e	Height of the concrete Section
S_t	Distance between stirrups (mm)
M_u	Bending capacity (N.mm)
V_c	Shear capacity
F_y	Steel Yield Stress (N/mm ²)
f'_c	Concrete compressive Strength (N/mm ²)
A'_s	Number of Compressive re-bar
A_s	Number of Tensile re-bar
N_r	Neutral Axis of the concrete Section
$\rho_{t,b,min}$	tensile, balance and minimum re-bar percentage in the section
ξ_t	Tensile strain in the concrete section
Par_{No}	Number of the parameters used in each operation
LT_{Si}	Size of the <i>Look – up</i> table (number of the calculated section in each operation)

References

1. Burnham, K. Artificial Intelligence vs. Machine Learning: What's the Difference? *Northeastern University* **2020**.
2. Goldenberg, S.L.; Nir, G.; Salcudean, S.E. A new era: artificial intelligence and machine learning in prostate cancer. *Nature Reviews Urology* **2019**, *16*, 391–403.
3. Brownlee, J. A tour of machine learning algorithms. 2019. URL <https://machinelearningmastery.com/a-tour-of-machine-learning-algorithms> **2020**.
4. Weidman, S. *Deep Learning from Scratch: Building with Python from First Principles*; O'Reilly Media, 2019.
5. Makrynioti, N.; Vasiloglou, N.; Pasalic, E.; Vassalos, V. Modelling machine learning algorithms on relational data with datalog. Proceedings of the Second Workshop on Data Management for End-To-End Machine Learning, 2018, pp. 1–4.
6. Aksjonov, A.; Nedoma, P.; Vodovozov, V.; Petlenkov, E.; Herrmann, M. Detection and evaluation of driver distraction using machine learning and fuzzy logic. *IEEE Transactions on Intelligent Transportation Systems* **2018**, *20*, 2048–2059.
7. Thomas, A. An introduction to neural networks for beginners. Technical report, Technical report in Adventures in Machine Learning, 2017.
8. Zadeh, L.A. The role of fuzzy logic in modeling, identification and control. In *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Lotfi A Zadeh*; World Scientific, 1996; pp. 783–795.
9. Klir, G.; Yuan, B. *Fuzzy sets and fuzzy logic*; Vol. 4, Prentice hall New Jersey, 1995.
10. Ross, T.J. *Fuzzy logic with engineering applications*; John Wiley & Sons, 2009.
11. Frangopol, D.M.; Hong, K. A Probabilistic-Fuzzy Model for Seismic Hazard. Reliability and Safety Analyses under Fuzziness. Springer, 1995, pp. 302–325.
12. Khatibinia, M.; Gharehbagh, S.; Moustafa, A. Seismic reliability-based design optimization of reinforced concrete structures including soil-structure interaction effects. *Earthquake Engineering-From Engineering Seismology to Optimal Seismic Design of Engineering Structures* **2015**, pp. 267–304.
13. Sahu, S.; Kumar, P.B.; Parhi, D.R. Intelligent hybrid fuzzy logic system for damage detection of beam-like structural elements. *Journal of Theoretical and Applied Mechanics* **2017**, *55*, 509–521.
14. Baghdadi, A.; Meshkini, A.; Kloft, H. Parametric design of in-plane concrete dry joints by FE method and Fuzzy logic toward utilising additive manufacturing technique. Proceedings of IASS Annual Symposia. International Association for Shell and Spatial Structures (IASS), 2020, Vol. 2014, pp. 1–12.

15. Yu, Y.; Rashidi, M.; Samali, B.; Mohammadi, M.; Nguyen, T.N.; Zhou, X. Crack detection of concrete structures using deep convolutional neural networks optimized by enhanced chicken swarm algorithm. *Structural Health Monitoring* **2022**, *21*, 2244–2263.
16. Behnam, A.; Esfahani, M.R. Prediction of biaxial bending behavior of steel-concrete composite beam-columns by artificial neural network. *Int. J. Optim. Civil Eng* **2018**, *8*, 381–399.
17. İpek, S.; Güneyisi, E.M.; Mermerdaş, K.; Algin, Z. Optimization and modeling of axial strength of concrete-filled double skin steel tubular columns using response surface and neural-network methods. *Journal of Building Engineering* **2021**, *43*, 103128.
18. Fathali, M.; Asgarian, B. Seismic response prediction of steel frame structures using adaptive neuro-fuzzy inference system. *Structural Engineering and Mechanics* **2019**, *69*(5), 547–556.
19. Bastami, A., K.A.; P., F. Acoustic emission-based damage detection in reinforced concrete structures using adaptive neuro-fuzzy inference system. *Construction and Building Materials* **2020**, *262*, 120918.
20. Bazargan-Lari, R.; A., H. Optimization of steel-concrete composite beams using ANFIS algorithm. *Journal of Constructional Steel Research* **2019**, *155*, 47–60.
21. Darain, K.; Hossain, M.; Jumaat, M.; Arifuzzaman, M. Prediction of Deflection Behavior of NSM Strengthened Reinforced Concrete Beam Using Fuzzy Logic. *Selected Scientific Papers-Journal of Civil Engineering* **2022**, *17*, 1–10.
22. Chopra, P.; Sharma, R.K.; Kumar, M. Prediction of compressive strength of concrete using artificial neural network and genetic programming. *Advances in Materials Science and Engineering* **2016**, 2016.
23. Amani, J.; Moeini, R. Prediction of shear strength of reinforced concrete beams using adaptive neuro-fuzzy inference system and artificial neural network. *Scientia Iranica* **2012**, *19*, 242–248.
24. Togholi, A.; Mohammadhassani, M.; Suhatri, M.; Shariati, M.; Ibrahim, Z. Prediction of shear capacity of channel shear connectors using the ANFIS model. *Steel Compos Struct* **2014**, *17*, 623–639.
25. by ACI Committee 318, R. Building Code Requirements for Structural Concrete (ACI 318-14), Commentary on Building Code Requirements for Structural Concrete (ACI 318R-14). An ACI Standard and Report. American Concrete Institute, 2014, Vol. ISBN: 978-0-87031-930-3.
26. Mostofinejad, D. *Rainforced concrete Structures, Based on ACI 318-05 and Iranian Concrete Code*; Vol. II, Arkan Scince, ISBN:964-2591-04-9, 2008.
27. Developers, E.S.; Diagram.com, B.M. Free Bending Moment and Shear Force. Link:<https://bendingmomentdiagram.com/free-calculator/reinforced-concrete-beam-calculator/>. SkyeCiv.
28. Tahouni, S. *Design of reinforced concrete buildings*; University of Tehran, ISBN:978-964-03-6363-8, 2010.
29. Kainia, A. *Analysis and design of concrete structures*; Industrial University of Isfahan, ISBN:964-6122-08-6, 2010.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.