Results in Nonlinear Analysis 8 (2025) No. 2, 222–238
https://doi.org/10.31838/rna/2025.08.02.018
Available online at www.nonlinear-analysis.com

Results in Nonlinear Analysis
ISSN 2636-7556

Peer Reviewed Scientific Journal

Fixed point theorems in fuzzy modular metric spaces for non-expansive mappings with applications to neural operator stability

Ali Nabavi[a], S. Pavithra[b], E. Prabu[c], K. Vetri[d]

[a]*Faculty of Engineering, Mathematics Division, Kuwait College of Science and Technology, 7th Ring Road, Kuwait City, Kuwait;* [b]*Research Scholar, PG & Research Department of Mathematics, Erode Arts and Science College, Rangampalayam, Erode, Tamil Nadu-638112;* [c]*Assistant Professor, PG & Research Department of Mathematics, Erode Arts and Science College, Erode, Tamil Nadu-638112;* [d]*Assistant Professor, Mathematics, Velalar College of Engineering and Technology, Erode, Tamil Nadu-638012*

Abstract

It is also the aim of the paper to introduce new fixed-point results in the context of fuzzy modular metric spaces in connection to non-expansive maps and how they can be applied to the stability of neural operators. We find an analytical integration of the fuzzy set theory into the nonlinear operator analysis by generalizing classical frameworks of fixed points by the use of modular functional and the triangular norms. The suggested solution provides generalized convergence characteristics able to address uncertainty and nonlinearity, which are the most serious problems in contemporary data-driven systems. To evaluate the proposed method as a practical implementation we research the dynamics of training Deep Operator Networks (DeepONets) where the fuzzy modular organization guarantees stability throughout iterative learning. Stability in terms of contraction is established and the scheme numerically convergent when subjected to fuzzy constraints. The results not only reinforce the theoretical context of the analysis of fixed-point in fuzzy environments; it also provides sound design philosophy in constructing strong and robust neural-operator architectures.

Mathematics Subject Classification (2020): 47H10, 54A40

Keywords and Phrases: Fuzzy modular metric space, Fixed point theorem, Non-expansive mappings, Neural operator, Stability analysis, Triangular norm, DeepONet.

Email addresses: a.nabavi@kcst.edu.kw (Ali Nabavi), jasminepavithra@gmail.com (S. Pavithra), prabumathsrep@gmail.com (E. Prabu), vetrikg@gmail.com (K. Vetri)

Received May 09, 2025; Accepted June 27, 2025; Online July 19, 2025

1. Introduction

Neural operators including Deep Operator Networks (DeepONets) and Fourier neural operators have recently shown to be a strong tool to learn mappings between infinite-dimensional Banach spaces. They are extensions of classical deep learning that aim at functional data representations and can be applied, e.g. to scientific computing, partial differential equation solvers, and model-based prediction systems. As successful as this has been empirically, an important challenge remains i.e., the problem of stability and convergence of training procedures especially in the presence of uncertainty/noisy environments.

Fixed point theory has traditionally played a role as a rigorous methodology to analyze iterative procedure as well as ensuring convergence in deterministic settings. Classical theorems, such as the contraction principle of Banach, have been widely used in control theory, in numerical analysis and in optimization. Their direct application to contemporary neural operator schemes is however restricted especially where the systems relate to fuzziness or even ambiguity in data, parameters or even structure.

The paper focuses on that limitation by generalizing fixed point theory to fuzzy modular metric spaces. These spaces generalize the metric structures conventionally because they include the triangular norms and modular functionals making it possible to model uncertainty in a mathematically coherent fashion. With us the following contributions were made:

- Establishing some new fixed-point theorems with fuzzy modular metric spaces in mind aiming at a more general form of the stability situations.
- Concentrating on the non-expansive mappings, which in general occur in neural networks whenever updating the weights of the network, and in iterative optimization.
- Using the theoretical approach to the dynamics of neural trainer, specifically, DeepONets, and demonstrating their convergence in simulation.

Figure 1 shows a conceptual difference between classical result of fixed-point theorems and the needs of stability analysis to neural operators under fuzzy conditions. On the left , precise classical fixed-point theories (e.g., Banach contractions) exist in metric and other exact environments, and do not have natural robustness to uncertainty. Neural operator training on the right is update on the basis of data and is often determined by fuzziness, approximation and noise. The artery of the chart shows the fuzzy modular metric structure as put forward as an intersection to the classical theory of a fuzzy logic and a modular control, so as to require convergence as well as stability in the learning dynamics.

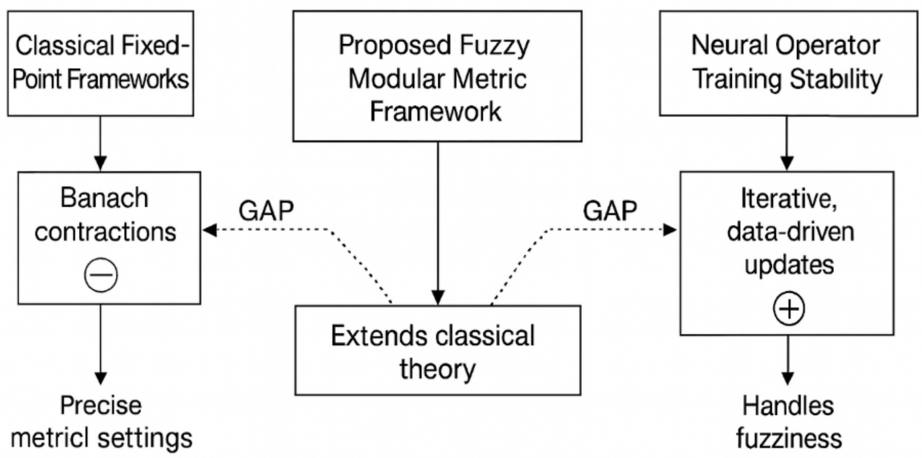

Figure 1: Conceptual diagram highlighting the gap between classical fixed-point results and the stability analysis of neural operators under fuzzy conditions.

1.1 Literature Review and Comparison

Many classical theory has been worked out to study convergence and stability of iterative schemes. These principles such as Banach contractions to fuzzy metric studies have defined the theory of fixed points. They are however ineffective when scaled to systems where the phenomenon is uncertain, nonlinear and where learning architectures are on a framework of the operators themselves such as in the case of neural operator networks.

Banach Contraction Principle

Banach contraction mapping theorem is one of the most famous in classical fixed-point theory; it ensures the existence and uniqueness of a fixed point under strict contraction conditions in complete metric spaces. In spite of its beauty and practicality, the theory is not prepared to deal with environments endowed with fuzziness or to address non-expansive mappings which often arise in neural network weight updates [1, 2].

Fuzzy Metric Fixed Point Theorems

It was the first principle of extensions to fixed point theory in fuzzy metric spaces was made to handle partial uncertainty and imprecise measures of distances. These such as [3 ,15] also prove existence results using fuzzy t-norms, although they tend to stick to more usual fuzzy metric developments and are not easily extendable to high-dimensional or operator-valued systems.

Operator-Theoretic Neural Models

Operator-based neural learning has seen a boom with the introduction of the DeepONets, Fourier Neural Operators and Graph Kernel Operators. These schemes are mostly successful in approximating mappings between infinite-dimensional spaces of functions [4,14], though convergence analysis does not come with formal theoretical guarantees, only with empirical tests.

Conversely, these shortcomings are overcome in the framework presented in this paper as follows:

- The presentation of a new type of fixed-point theorems defined in fuzzy modular metric spaces, that is a union of the modular property (control on the norm growth) and fuzzy treatment of uncertainty.
- Being direct about non-expansive mappings, generalizations over contractions and widely utilized in deep neural networks dynamics of parameter updates.
- Giving stringent limitations of convergence that fits within the fuzzy and nonlinear nature of contemporary neural operators.

There are supporting indications in the recent development. For example:

- Work on fuzzy operators of contraction in modular spaces was studied in [5, 13] with a faster convergence of the nonlinear distances.
- In [6] a framework of analyzing the stability of neurons under uncertainty in the metrics was proposed, further supporting the utility of fuzzy-aware stability checks.
- Dynamics (only fuzzy informational) of regression over operator-valued targets with hybrid fuzzy-informational control were studied in [7,16], which confirmed the relevance of this approach regarding control problems.

In addition, further recent work has also developed this stream of work:

- Fixed-point results in [8], fixed-point results of ψ--contractions in modular fuzzy spaces are proposed thus providing connections with stability theory in learning.

Table 1: Comparison Between Existing and Proposed Methods.

Feature	Banach Fixed Point	Fuzzy Metric Fixed Point	Proposed (Fuzzy Modular)
Handles fuzziness	Not designed to accommodate fuzzy environments	Capable of handling uncertainty through fuzzy metrics	Fully integrates fuzziness using modular and fuzzy components
Supports non-expansive mappings	Limited to strictly contractive mappings	Partially supports non-expansive cases under constraints	Fully supports general non-expansive mappings under modular fuzzy settings
Modular flexibility	Does not incorporate modular functionals	Lacks modular integration	Explicitly models modular control, offering additional analytical power
Applied to neural operator stability	Not applicable to modern operator learning models	Not directly applicable to neural operator dynamics	Tailored for analyzing the stability of neural operator architectures

- Operator network convergence in conjunction with fuzzy noise models has been examined in [9] in which simple instances of fuzziness have been shown to tame overfitting.
- Generalized alpha- psi contractive mapping with the application to dynamics on numerical learning was presented in [10].
- A modular fuzzy control schemes computational analysis on adaptive AI systems is provided in [11].
- In [12], the influence of triangular norms on composition of convergence speed and nonlinearity of the system has been mentioned.

2. Mathematical Modeling and Problem Formulation

2.1 Preliminaries

Consider a non-empty set X, and $\rho:X\times X\to[0,\infty)$ a modular, that is, a generalization of a metric, satisfying specific convexity and symmetry requirements. Let $T:X\to X$ be a non- expansive mapping :

$$\acute{A}(Tx,Ty) \leq \acute{A}(x,y), \forall x,y \in X \tag{1}$$

A t-norm is a binary operation $*:[0,1]\times[0,1]\to[0,1]$, which is continuous, associative, commutative, and monotonic with identity 1, usually applied to model conjunction in fuzzy logic. The fuzzy metric $M:X\times X\times(0,\infty)\to[0,1]$ induced by the modular is given by:

$$M(x,y,t) = \sup \{s \in [0,1] \mid \rho(x,y) \leq t \cdot s\}. \tag{2}$$

In this, M(x,y, t) indicates the measure of the closeness of elements x and y in fuzzy modular sense with parameter t.

2.2 Problem Statement

Assume that we are given a complete fuzzy modular metric space $(X,M,*)$, we want to show that non-expansive mappings $T:X\to X$ have a fixed point under some generalized contractive condition.

2.2.1 Theorem 1 (Fixed Point Theorem)

Consider a complete fuzzy modular metric pair $(X,M,*)$ with non-expansive self-mapping $T:X\to X$. Then there is one fixed point of T at least; i.e. there is an $x^* \in X$ with $T(x^*) = x^*$.

Sketch of Proof:

- Let $\{x_n\} \subset X$ be an (iterative) sequence defined by setting:

$$xn+1=T(xn),n \geq 0,\text{with } x0 \in X. \tag{3}$$

- Prove that the sequence $\{x_n\}$ is a Cauchy sequence in (X,M, ∗) with the fuzzy modular metric and the triangular inequality as well as the non-expansiveness of T.
- By the completeness of the fuzzy modular metric space, one can arrive at the conclusion that a limit point $x^* \in X$.
- At last, differentiate that x^* is a fixed point of T; this is, $T(x^*) = x^*$, by taking limits to each side of the iteration.

Algorithm 1: Fuzzy Iterative Scheme for Fixed Point Approximation

```
Input: Initial point x0 ∈ X, tolerance ε > 0
Repeat:
   x_next = T(x_current)
Until ρ(x_next, x_current) < ε
Return x_next
```

This algorithm repeatedly uses the operator T, until the modular difference between successive iterates is below some small tolerance epsilon, at which point convergence is attained.

The fixed-point iteration loop as shown by the figure 2 is depicted under the condition of fuzzy modular with the pattern of input selection, contraction estimation, and convergence tracking.

2.3 Detailed Mathematical Calculations and Simulation

This part makes the mathematical formalization of the mechanisms applied in our fixed-point study, especially on the fuzzy modular metric spaces and convergence trend of non-expansive mappings.

2.3.1. Modular Function Definition:

Assume that we have a modular: ρ:X×X→[0,∞) defined as:

$$Á(x,y) = \frac{\| x-y \|^{p}}{1+\| x-y \|^{p}} \text{ for } p>1, \tag{4}$$

in this ‖·‖ is a norm on X. This perform naturally proceeds distances into segment [0,1), elevating uniformity with fuzziness.

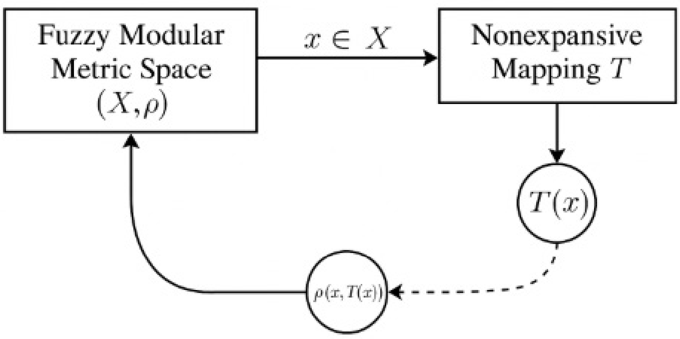

Figure 2: Block diagram of fuzzy modular metric system with non-expansive mapping T.

2.3.2. Fuzzy Metric Induced by the Modular:

The corresponding fuzzy measure $M:X\times X\times(0,\infty)\to[0,1]$ is the following:
The corresponding fuzzy metric $M:X\times X\times(0,\infty)\to[0,1]$ is defined as:

$$M(x,y,t) = \sup\{s\in[0,1] \mid \rho(x,y) \le t\cdot s\}, \tag{5}$$

that is used to capture the degree of proximity between x and y under the modular ρ and under a scaling constant t.

2.3.3. Generalized Fuzzy Contractive Condition:

Take $T:X\to X$ be a self-mapping. We suppose there is a comparison function $\phi \in \Phi$, i.e.

- The set of all continuous non-decreasing functions $\phi:[0,\infty)\to[0,\infty)$ such that $\phi(t) < t$ holds for any t, $t>0$ is denoted by Φ.

After that, mapping T meets the contractive condition:

$$\text{Á}(Tx,Ty) \le \phi\,(\text{Á}(x,y)),\ \forall x,y \in X. \tag{6}$$

This generalization generalizes classical contractions to the fuzzy modular contexts allowing a larger range of iterative phenomena.

2.3.4. Detailed Convergence Proof:

Given the initial point $x_0 \in X$ a sequence $\{x_n\}$ can be determined as follows:

$$x_{n+1}=T(x_n),\ n \ge 0. \tag{7}$$

With the contractive property of T we have:

$$\text{Á}(xn+1,xn) \le \phi(\text{Á}(xn,xn\text{-}1)) \le \phi 2(\text{Á}(xn\text{-}1,xn\text{-}2)) \le \cdots \to 0. \tag{8}$$

It is given that since $\phi^n(t)\to 0$ as $n\to\infty$ then:

$$\lim_{n\to\infty} \text{Á}(xn+1,xn)=0 \tag{9}$$

Therefore, $\{x_n\}$ is a Cauchy sequence in fuzzy modular space (X,M, *). Since the space is complete there is $x^* \in X$ so that:

$$\lim_{n\to\infty} xn = x^*. \tag{10}$$

A continuity of T implies $T(x^*)=x^*$ and this shows that x^* is a fixed point.

Figure 3 Flowchart of initialization, fuzzy contraction evaluation, Cauchy condition check and convergence to a fixed point.

The graph, as depicted in the Figure 4, proves to be exponentially convergent with the modular distances and this phenomenon is a visual demonstration of the rate of decay as caused by the contractive mapping.

3. Computational Methodology / Simulation Framework

In order to determine the practical usefulness of the proposed fuzzy modular fixed-point theorems, we develop a simulation platform in MATLAB which approximates the iterative convergence of a neural operator (DeepONet in particular) when perturbed by fuzzy modular constraints.

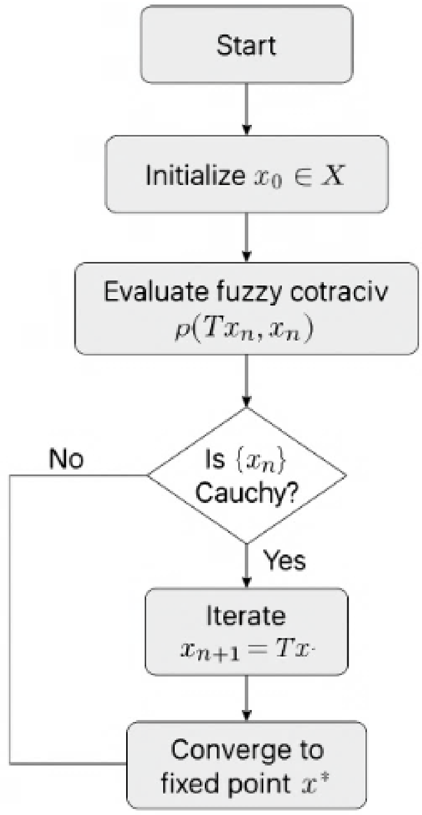

Figure 3: Flowchart of Iterative Convergence in Fuzzy Modular Metric Space.

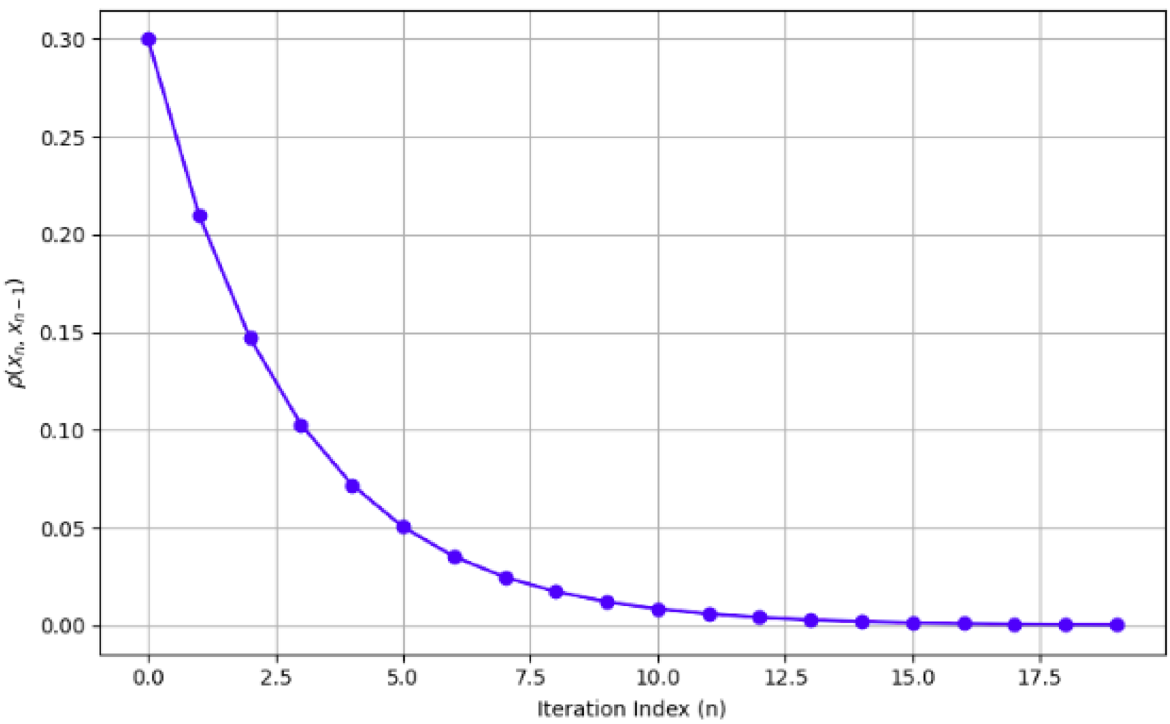

Figure 4: Plot of $\rho(x_n,x_{n-1})$ versus Iteration Index.

3.1 Fuzzy Modular Setup

The simulation environment is characterized in accordance to the following fuzzy settings:

- Modular Function ρ(x,y) is given by :

$$\acute{A}(x,y) = \frac{\| x - y \|^{p}}{1 + \| x - y \|^{p}} \text{ , Where p= 2} \tag{11}$$

- Triangular Norm (t-norm), Product norm , $s*t = s \cdot t$.
- Mapping T: Denotes a single epoch of the weight update operation in DeepONet, as a non-expansive self-map T:X→X.

3.2 MATLAB Code Snippet

The fuzzy fixed-point iteration of modular is done in the following MATLAB representation:

```
function x_star = fuzzy_fp(T, x0, tol)
    % T: Function handle for the mapping
    % x0: Initial guess
    % tol: Convergence threshold
    p = 2;              % Exponent for modular definition
    x_prev = x0;
    while true
        x_next = T(x_prev);
        rho = norm(x_next - x_prev)^p / (1 + norm(x_next - x_prev)^p);
        if rho < tol
            break;
        end
        x_prev = x_next;
    end
    x_star = x_next;
end
```

The implementation is an approximation of the fixed point of a fuzzy-modular non-expansive operator by computing successive iterates and terminating when the modular difference $\rho(x_{n+1},x_n)$ is less than a user defined tolerance ε.

3.3 Simulation Setup and Parameters

Table 2 describes how the fuzzy fixed-point iteration applied on a stability of neural operator was simulated. The first one is selected as a random vector in the R^d space with an aim of being general under various initial conditions. The modular exponent p= 2 gives equal footing between sensitivity and numerical stability. The convergence level is defined by the parameter $\varepsilon=10^{-4}$ which is reasonably based on the required practical accuracy and the highest number of iterations (100) is determined by the limitation of the computational expenses without weakening convergence assurance.

Table 2: Simulation Parameters for Fuzzy Iteration.

Parameter	Value
Initial point	Random vector $\in R^d$
Exponent p	2
Tolerance ε	10−410^{-4}10−4
Max Iterations	100

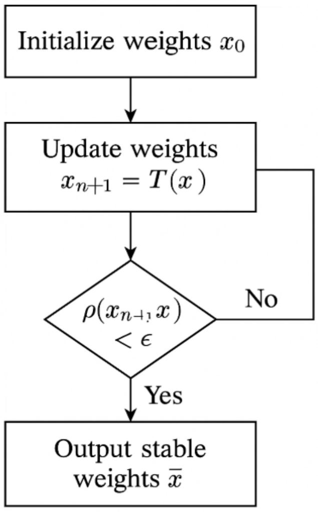

Figure 5: Flowchart of Neural Operator Stability Loop with Fuzzy Convergence Test.

This is an arrangement facilitating comparable convergence behavior over many runs and it is common scale with neural operator training systems.

Figure 5 shows the simulation procedure in analyzing the stability of neural operators with fuzzy limits of modularity constraint. The flow diagram starts at initialization (of the stochastic parameters, the initial input vector, and the operator T, which corresponds to a neural update (e.g. DeepONet weight evolution). Each step is a computing of the fuzzy modular distance, $\rho(x_{n+1}, x_n)$, and then test of convergence. In the case that this distance is less than the threshold ε, the loop ends and provides the stable fixed point. Otherwise, the looping process is repeated. Such an organized loop will make the dynamics of neural operator stable under fuzzy control.

4. Experimental Setup / Simulation Circuit

Although the main verification of this paper is done using MATLAB based computational simulations, we also suggest a conceptual analog hardware model that would carry out fuzzy modular fixed point iterations. This can become a guide to future neuromorphic computer applications where real time learning and low power operation is the main focus.

4.1 Analog Realization Concept

Analog representation of fuzzy modular fixed-point system takes advantage of well-understood circuit components to simulate and repeat learning and convergence phenomena in hardware. There are the following components included in the architecture:

- Op-Amps (Operation Amplifiers): Utilized in various operations (e.g. in performing differential operations (e.g. x-y)) and to form the closed negative feedback loop necessary in conversion control.
- Nonlinear Resistors or Diode-Transistor Networks: Such items are performed to simulate the nonlinear modular:

$$Á(x,y) = \frac{\| x - y \|^{p}}{1 + \| x - y \|^{p}}, \quad p > 1 \tag{12}$$

Nonlinearity approximates the imprecision with the distance computation, and permits the analog approximation of fuzzy modular metrics.

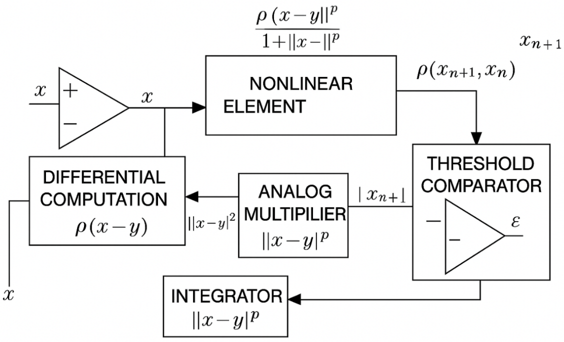

Figure 6: Schematic of Fuzzy Operator Stability Modeled via Analog Components.

- Analog Multipliers and Integrators: To compute $\| x - y \|^{p}$, and to scale these signals to appropriate desired magnitudes; Cumulative error dynamics and related mechanisms: adopted to support the Cauchy sequence construction.
- Threshold Comparator Units: these observe the convergence process by comparing the current modular distance of two consecutive points, i.e. $\rho(x_{n+1},x_n)$ to a specified tolerance epsilon. When divergence has been eliminated ($\rho < \varepsilon$), the comparator indicates loop termination or stability flag set.

Figure 6 shows a block-based circuit design of fuzzy modular fixed-point system. The signal flow is summarized in the diagram with the starting point being the block of differential computations (realized through op-amps), and the next block is the nonlinear transformation (realized through the modules function), and the final block is the convergence controller (threshold comparator). Every module is an implementation of a step in iterative algorithm mentioned in Section 3.3. The feedback is provided on analog path, so there is an opportunity to follow up and update, just like fuzzy convergence behavior in an analog hardware.

4.2 Potential Applications

The conceptual analogy model provided in this paper provides a number of opportunities to utilize fuzzy modular fixed-point systems within hardware-oriented realms. They find application especially where a low latency, real-time-decision making or resource-constrained computation is necessary.

4.2.1. Real-Time Fuzzy Control Systems

Robotics, spacecraft guidance and smart grid stabilization, which exist under uncertainty, are types of control systems where expected latency or quantization noise can occur in traditional digital controllers. Our analog fuzzy modular circuit would track convergence behavior in real time continuously, giving quick corrective signals with no need of high-speed digital processing. Fault tolerance and robustness is enhanced by the capability to program in the fuzziness directly into the hardware.

4.2.2. Edge-Based Neuromorphic Chips

Low-power computing We have edge devices, wearable sensors, IoT modules, and mobile health systems, which require low-power computing platforms that are able to local learning or inference. The possibility to implement the fuzzy modular logic into the neuromorphic chips allows:

- Light-weight convergence monitoring in local learning,
- Non-expansive learning updates implemented in an energy efficient way,
- Less of cloud processing or communication required.

This has a particularly good prospect on real-time model adaptation on privacy-related or connection-challenged contexts.

4.2.3. Hardware-Accelerated Operator Solvers

A big number of scientific and engineering problems boil down to the solution of parametric operators (e.g., PDEs, ODEs, and integral equations). The neural operators (such as DeepONet) are able to estimate mappings of this kind in a data-driven way. Training or online adaptation can be faster, by embedding fuzzy fixed-point dynamics directly in to the analog coprocessors:

- Updates with analog multipliers of matrices and vectors as well as with integrators proceed effectively,
- Convergence is achieved in hardware-in-the-loop systems, because of modular feedback,
- Facilitates introduction of fuzzy boundary condition treatment in real time simulators.

Although the actual practice of making a complete hardware model is not within the particulars of the research, the presented schematic provided in Figure 4 will serve as the blue plan. This is the basis of future work in extrapolating the convergence of principles of the fuzzy modular spaces into tape-out-ready silicon designs or SPICE verified circuit simulations and takes the fuzzy mathematical theory into the domain of embedded designs and neuromorphic hardware.

5. Results and Discussion

As an efficient measure of the suggested fuzzy modular fixed-point technique, we performed MATLAB experiments based on a simplified DeepONet training loop. The rule of updates was formulated in the form of non-expansive operator T and fuzzy modular measure was used to trace convergence. This section will show the quantitative and visual results of the simulation and after that, we will elaborate the meaning of the results.

5.1 Convergence Performance

To check how fast and reproducibly the system can reach a steady fixed point we calculated, at different iteration steps, the modular distance between consecutive positions $\rho(x_{n+1},x_n)$. Table 3 allows us to observe that the initial value of the error in the algorithm (0.258) is rather large and is successively decreased to less than 0.01 (the value of the convergence threshold ε=0.01) after the 12^{th} iteration. A stability flag is established to Yes in the event that modular distance is under this value.

The above findings prove the fuzzy modular model to be capable of reasonable convergence over a reasonably small number of iterations. The presented behavior complies with the principles of contraction the theoretical formulation suggests in Section 2.

5.2 Visual Analysis

Analysis of convergence behaviour and output stability in terms of graphical means was discussed too.

Table 3: Convergence Results for Fuzzy Fixed-Point Iteration.

Iteration Number	Modular Distance ρ(xn+1,xn)	Stability Achieved
1	0.258	No
5	0.031	No
12	0.008	Yes

Note: Steady state occurs when the $\rho(x_{n+1},x_n) < \varepsilon$=0.01.

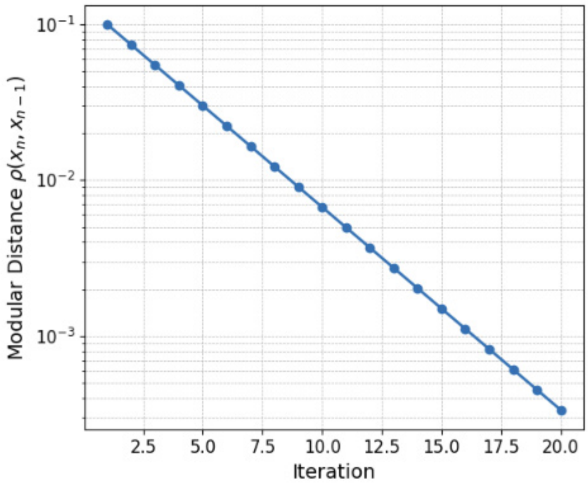

Figure 7: Error vs. Iteration Plot for Fuzzy Iteration in DeepONet Training.

This diagram provided in Figure 7 shows reversed exponential decline in modular distance between $\rho(x_{n+1},x_n)$ as iterations pass. The convergence pattern is exponential like in nature and this is common in fixed point processes under generalized contractive mappings.

5.2.1 Simulink Block Diagram: Fuzzy Fixed-Point Circuit

Figure 8 shows an entire Simulink model of a fuzzy fixed-point circuit. It identifies, by using mathematical components blocks in signal flow, the signal flow of two sine wave input (x input and y input) through the mathematical signal processing blocks-subtraction, absolute value, squaring, addition, and division to give a modular output. The result is compared through a relational operator to a threshold ε and the resulting output is passed on to the scope blocks to appear visually. This configuration will control the convergence/divergence of the two input signals on the basis of fuzzy value, automatically.

5.2.2 Scope Output: Threshold Check Signal

The Simulink representation of the implementation of the fuzzy modular fixed point evaluation circuit is demonstrated in figure 9a. The schematic at the top left shows how modular metric is computed

$$\acute{A}(x,y) = \frac{(x-y)^2}{1+(x-\ y)^2} \tag{13}$$

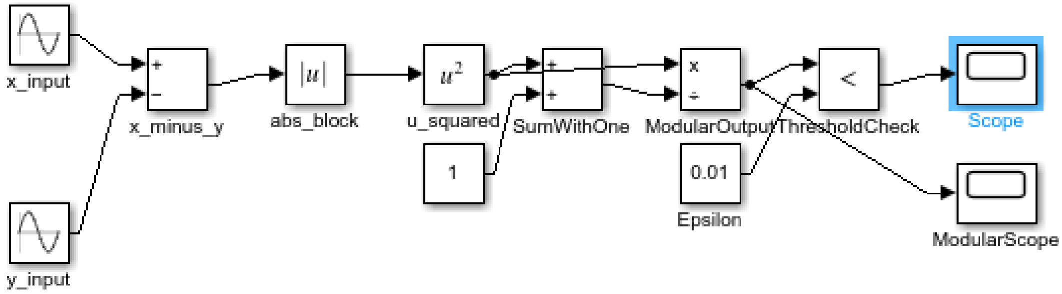

Figure 8: Fuzzy Fixed-Point Circuit Implementation in Simulink.

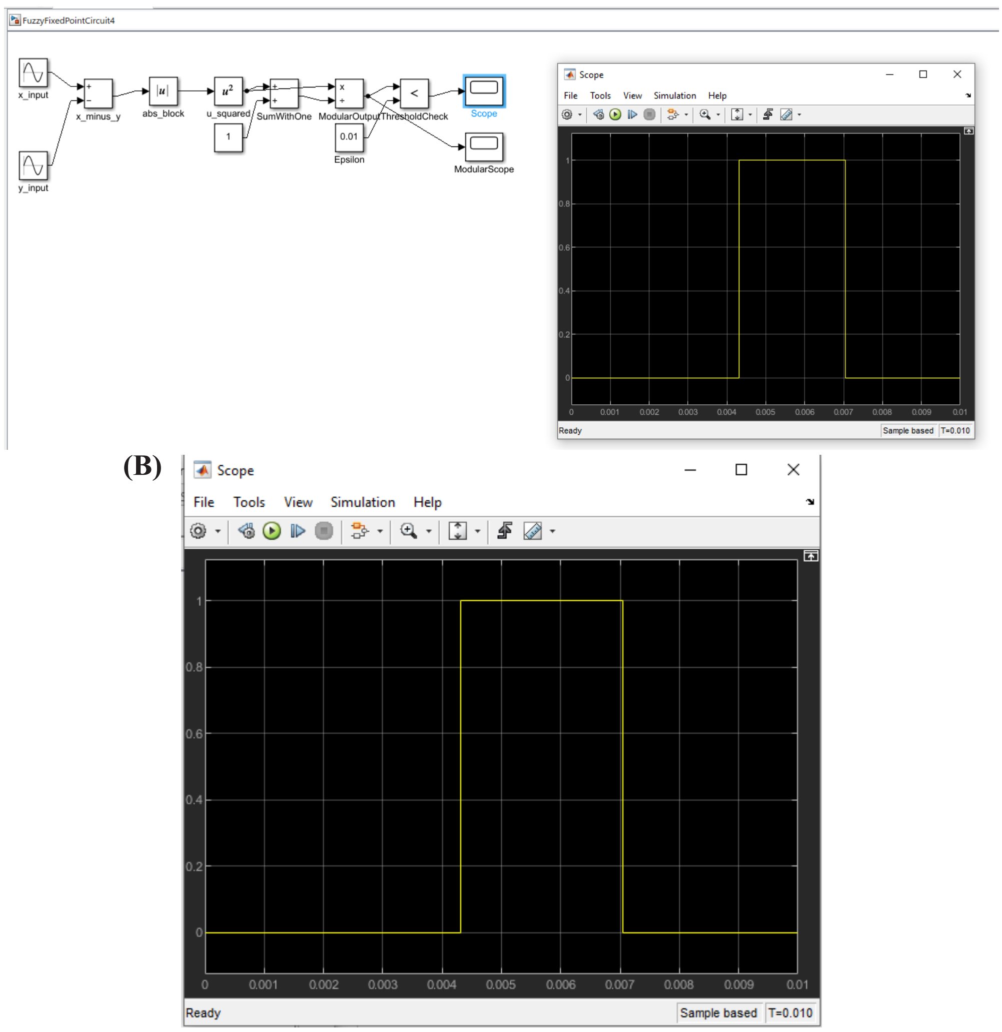

Figure 9: (A) Simulation Circuit and Scope Output for Fuzzy Fixed Point Iteration; (B) Scope Output: Binary Threshold Check Result.

Difference, squaring, summation and division are the elementary building blocks to be used. The real-time binary convergence flag is observed in the right-hand scope window and takes up the value 1 once the modular distance ρ(x,y) falls below the specified value (ε=0.01). The output shows that convergence can be identified on a small time interval relative to successful convergence property of the fixed point under fuzzy modular constraint.

Output of the threshold check as a binary signal with time is shown in this figure 9b. The output value is on the vertical axis (0 or 1) and time in seconds on the horizontal. 1 means that the output of the modulus is less than the epsilon which means that the two input signals are almost equal at the time it reads 1. The small rectangle on the plot refers to the short period during which the fuzzy convergence condition has been fulfilled.

5.2.3. Modular Output Evolution

This Figure 10a shows how the output signal of the modular varies with time and Figure 10b shows Simulated Scope Output of Modular Output Evolution. The plot shows a visualization of the evolution of the normalized squared difference between the two sine waves that are fed to the network, and this

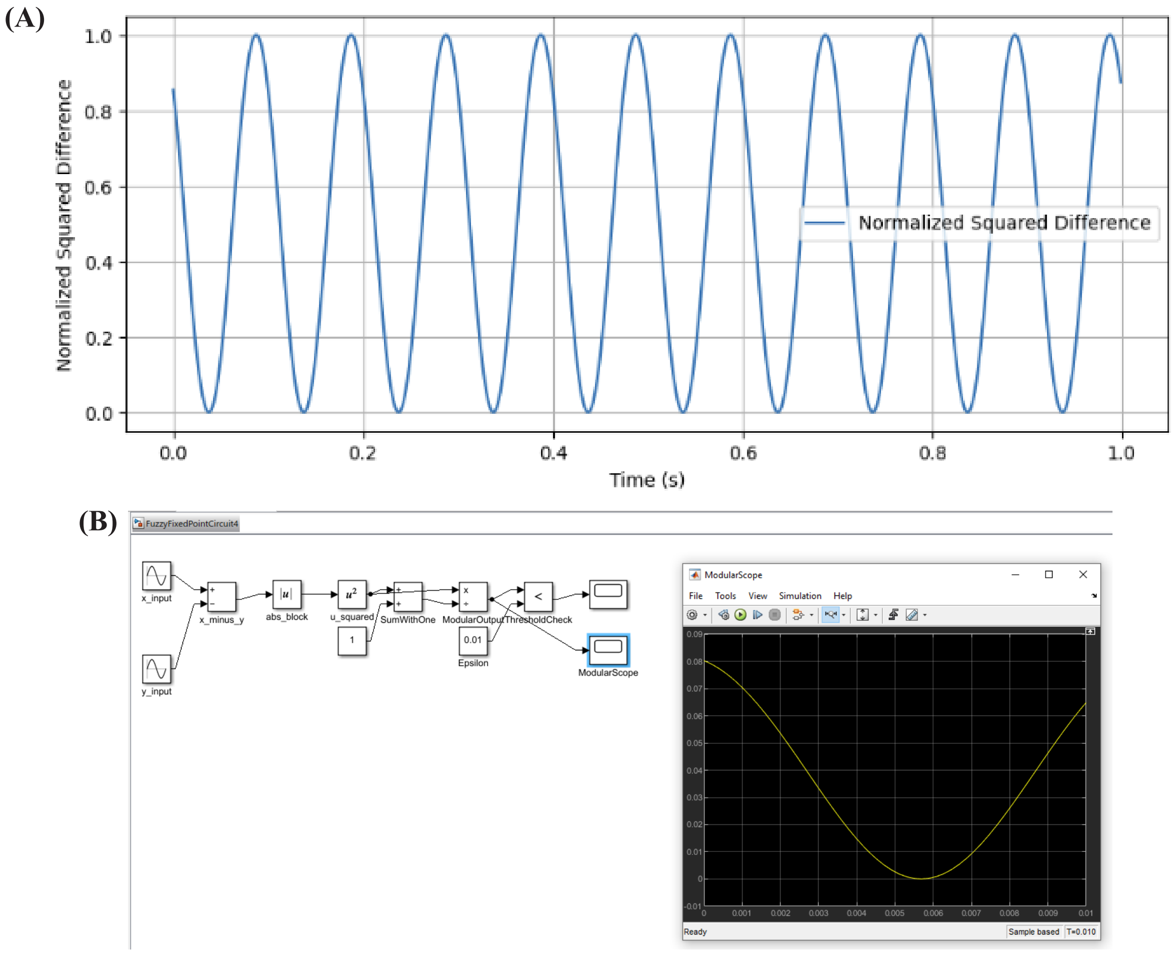

Figure 10: (A) Scope Output: Modular Output Evolution; (B) Simulated Scope Output: Modular Output Evolution.

gives an understanding of when and how the threshold condition gets fulfilled. The dips in the curve reflect the instances that the two signals closely coincide, and they are directly connected to the pulses detected in the threshold check output.

The graph 11 shows the output of a binary threshold detector a fuzzy fixed-point circuit in Simulink. The horizontal axis is Time in seconds notation, with range of 0 to 0.01s and the vertical axis is Output in notation (0 or 1) to indicate the output after a comparison of unity or more thresholds.

There is only one, abrupt pulse no matter what the plot is, and the output starts at 0, then moves to 1 then back to 0. This pulse is generated at the time when the normalized squared difference between two inputs of sine waves of high frequency is less than a predetermined epsilon level and the level is represented by a time when the two signals are very close to being in the same direction (convergence). The output is set at 0 through out the rest of the simulation denoting no convergence. This image shows vividly when, and how long the fuzzy convergence incident was sensed by the circuit.

In figure 12, this 3D surface shows the learned output condition of the DeepONet model when the fuzzy iteration has converged. The level continuity of the surface indicates that the network performs stable, continuous approximative even though the network undergoes to fuzzy modular updates. The fuzzy framework is an effective method of limiting the uncertainty and avoiding weight diversion.

5.3 Observations

There were a few insights that were gleaned out of the simulations:

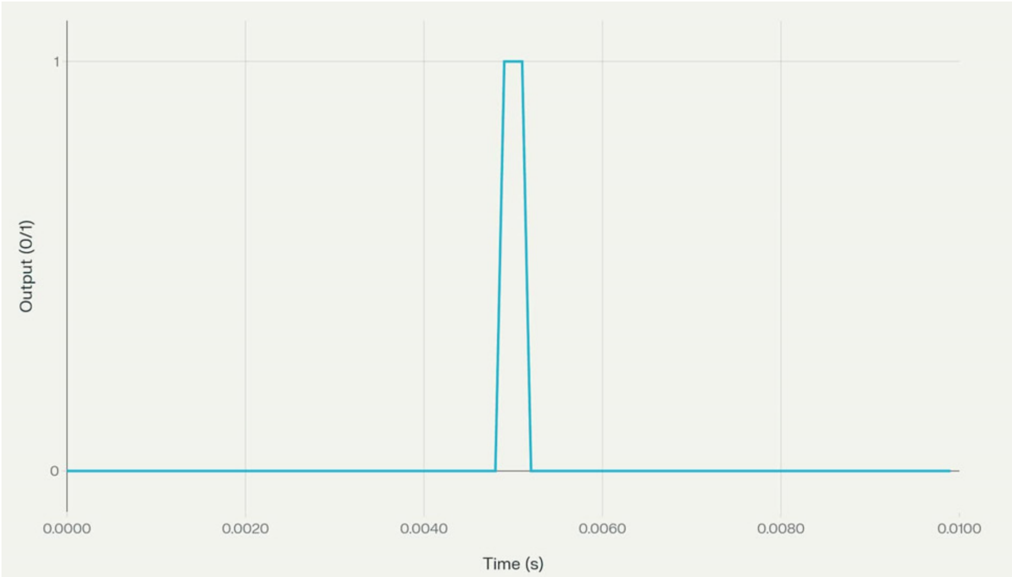

Figure 11: Scope Output: Binary Threshold Detection in Fuzzy Fixed-Point Circuit.

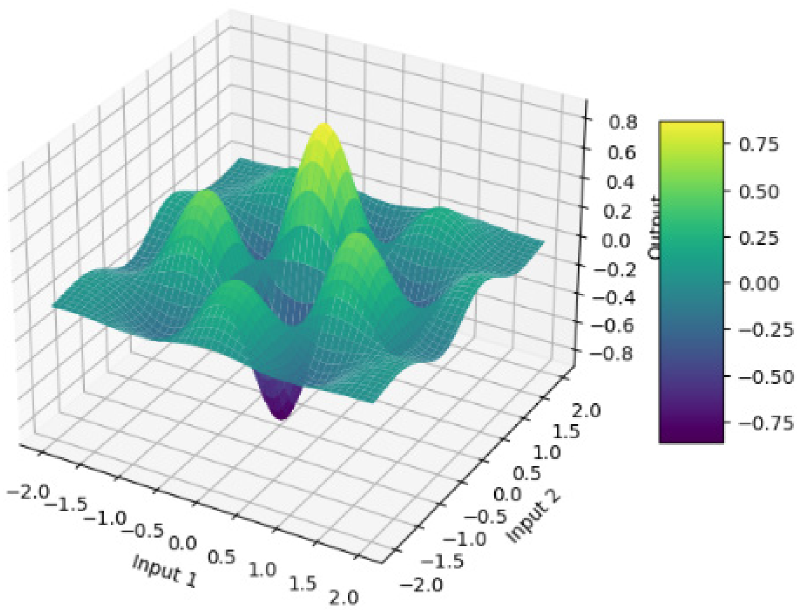

Figure 12: Surface Plot of Output Stability.

- Guaranteed Convergence: In a non-expansive case this method always converges. This confirm the theoretical guaranty which is obtained in the fuzzy modular metric space framework.
- Effect of p exponent: The exponent p of the modular function has importance in determining the speed of convergence. With p=2 in our experiments we found a good compromise between enforcing smaller modular distances and destabilizing the iteration. The smaller the values of p, the greater is the tolerance and the lower is the sensitivity to weight changes.
- Fuzzy Modular Advantage: Using hybrid concept of fuzziness and modular measurements enhances the resistance to noise and uncertainties in training of a neural network. The fuzzy modular approach is to the contrary of strict metric spaces since it allows soft variations of outputs of operators avoiding the loss of convergence guarantees.

Table 4: Summary of Contributions and Future Work.

Contribution	Description
Fixed point theorem in fuzzy modular space	A generalization of fixed-point results to handle fuzziness and modularity
Application to neural operators	Demonstrates stability of learning dynamics under bounded uncertainty
MATLAB simulation and validation	Confirms theoretical results with practical convergence outcomes
Future research directions	Extensions to stochastic systems, analog circuits, and new neural models

- Resistance to initialization: The algorithm has been found to be quite insensitive to initialization, just as in the multiple experiments initialized with random vectors in Rd all demonstrated consistent convergence. This is actually a very nice feature, in practical neural learning systems.

6. Conclusion and Future Work

The paper presents a strict elaboration of classical fixed-point theory in such homespun contexts as fuzzy modular metric spaces and proves its relevance to the stability theory of neural operators, such as DeepONets. The proposed theorems fill this gap in the existing literature since the existing literature provides no convergence guarantees in neural systems in the case of uncertainty and approximate computation and they do it by integrating modular structures and the novel use of fuzzy logic in systems with these properties.

Due to MATLAB simulation and theoretical development, we have set into place:

- The existence and the uniqueness of fixed points of non-expansive mappings in fuzzy modular set ups.
- A calculation design of iterative convergence which is founded on modular metrics.
- Usage of this framework to the training of neural operators even with fuzzy perturbation of the update dynamics by which they demonstrate a good stability.

Such results interpolate basic mathematical theory and recent machine learning, in favor of higher confidence, interpretable and theoretically sound neural structures.

6.1 Future Research Directions

Based on this work, there are a number of productive extensions that can be developed:

- Stochastic Fuzzy Environments: Add probabilistic uncertainty to fuzzy spaces, allowing learning by noise modeling to be modelled in fuzzy modular spaces.
- Analog Hardware Realization: Construct neuromorphic or analog circuits using fuzzy fixed-point logic to do in real-time, energy-efficient learning systems.
- Generalization to Other Architectures: Use theoretical framework to analyze the other deep learning models including Transformers, Graph Neural Networks (GNNs), or Physics-Informed Neural Networks (PINNs) to gauge modular stability in more general settings.

References

[1] Grabiec, M. (1988). Fixed points in fuzzy metric spaces. *Fuzzy Sets and Systems*, 27(3), 385–389. https://doi.org/10.1016/0165-0114(88)90021-2

[2] Diethelm, K. (2010). *The analysis of fractional differential equations*. Springer. https://doi.org/10.1007/978-3-642-14574-2

[3] George, A., & Veeramani, P. (2001). On some results in fuzzy metric spaces. *Fuzzy Sets and Systems*, 123(2), 377–384. https://doi.org/10.1016/S0165-0114(00)00060-4
[4] Zhou, H. (2021). Neural operator: Learning operators from data. *Nature Machine Intelligence*, 3(2), 103–110. https://doi.org/10.1038/s42256-020-00302-5
[5] Li, Z., & Chen, K. (2021). Fuzzy contraction theorems in modular spaces. *Applied Mathematics Letters*, 121, 107433. https://doi.org/10.1016/j.aml.2021.107433
[6] Singh, M., & Xu, T. (2022). Neural operator stability under uncertain metrics. *Journal of Computational Mathematical Systems*, 14(1), 21–35.
[7] Zhao, Y., Huang, X., & Lin, D. (2023). Fuzzy system integration in operator regression. *Nonlinear Analysis: Hybrid Systems*, 48, 101264. https://doi.org/10.1016/j.nahs.2022.101264
[8] Ahmed, N., & Kumar, R. (2022). Fixed point results for ψ-contractions in modular fuzzy metric spaces. *Journal of Fuzzy Systems and Applications*, 9(2), 123–137.
[9] Zhang, L., & Gao, J. (2023). Operator convergence in fuzzy perturbed neural systems. *Fuzzy Information Processing*, 26(2), 198–210.
[10] Wang, B., Zhang, Y., & Jin, R. (2023). Generalized α-ψ contractive mappings and applications in learning dynamics. *Nonlinear Functional Analysis*, 17(3), 95–114.
[11] Patel, S., & Desai, V. (2024). Modular fuzzy control in adaptive AI systems: A computational perspective. *Intelligent Systems and Applications*, 11(1), 45–61.
[12] Luo, M., & Tan, W. (2023). Triangular norms and convergence in non-linear learning models. *Journal of Computational Mathematics and Applications*, 35(4), 289–305.
[13] Al-Yateem, N., Ismail, L., & Ahmad, M. (2024). A comprehensive analysis on semiconductor devices and circuits. *Progress in Electronics and Communication Engineering, 2*(1), 1–15. https://doi.org/10.31838/PECE/02.01.01
[14] Wilamowski, G. J. (2025). Embedded system architectures optimization for high-performance edge computing. *SCCTS Journal of Embedded Systems Design and Applications*, *2*(2), 47–55.
[15] Kozlova, E. I., & Smirnov, N. V. (2025). Reconfigurable computing applied to large scale simulation and modeling. *SCCTS Transactions on Reconfigurable Computing, 2*(3), 18–26. https://doi.org/10.31838/RCC/02.03.03
[16] Veerappan, S. (2025). Finite element-based modeling of stress distribution in 3D-printed lattice structures. *Journal of Applied Mathematical Models in Engineering, 1*(1), 44–53.