2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

# A New Strategy to Modify Hopfield by Using XOR Operation

***Rusul Hussein Hasan***

College of Law, University of Baghdad, Iraq, Baghdad

Received 12/06/2023, Revised 27/08/2023, Accepted 29/08/2023, Published Online First 20/02/2024, Published 01/09/2024



## Abstract

The Hopfield network is one of the easiest types, and its architecture is such that each neuron in the network connects to the other, thus called a fully connected neural network. In addition, this type is considered auto-associative memory, because the network returns the pattern immediately upon recognition, this network has many limitations, including memory capacity, discrepancy, orthogonally between patterns, weight symmetry, and local minimum. This paper proposes a new strategy for designing Hopfield based on XOR operation; A new strategy is proposed to solve these limitations by suggesting a new algorithm in the Hopfield network design, this strategy will increase the performance of Hopfield by modifying the architecture of the network, the training and the convergence phases, the proposed strategy based on size of pattern but will avoid learning similar pattern many time, whereas the new strategy XOR shows tolerance in the presence of noise-distorted patterns, infinite storage capacity and pattern inverse value. Experiments showed that the suggested method produced promising results by avoiding the majority of the Hopfield network's limitations. In additional it learns to recognize an infinite number of patterns with varying sizes while preserving a suitable noise ratio.

**Keywords:** Auto-Associative Memory, Hopfield Network, Neural Network, Pattern Recognition and XOR Operation**.**

## Introduction

Traditional programs can find solutions to problems for which a clear algorithm can be developed to reach the solution, or whose solution is fixed and never changes. These programs can save hundreds of millions of patterns, but can these programs recognize the contents of the images they keep? The neural network came to solve these problems, as it means trying to imitate the human way of thinking, and therefore when starting to program the neural network, one must train it on a wide range of patterns, that later, if a similar or a close pattern comes along, the neural network will be able to recognize it [1, 2].

Basic neural networks consist of three layers of interconnected artificial neurons (see Fig .1) [2].

- Input layer: the entrance to this layer is information from the outside world; the input nodes analyze, process, or classify the information and pass this information to the next layer.
- Hidden layer: there are a large number of hidden layers in artificial neural networks; they take their input from other hidden layers or from the input layer. Each hidden layer parses the output from the previous layer and processes it further, passing it on to the output layer.

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

- Output layer: this layer gives the final results for the processing operations performed by the neural networks.

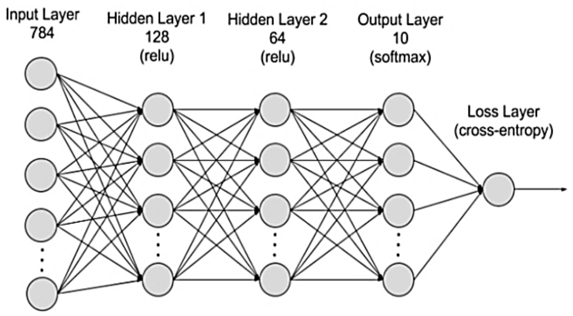



**Figure 1. Neural Networks Layers.**

In general, most of the tasks of neural networks are of two types: classification or association. In the first type, the classification, an image is entered and classified into one of the existing categories, for example, an animal image is classified into mammals, reptiles, or any other type [3,4].

The other type is associative memory, defined as a form of artificial neural network, which stores input patterns with their matching output patterns, and it was designed to call a pattern from a noise-distorted form, an associative memory block diagram is shown in Fig. 2 that performs associative mapping among vector (x) as input and vector (v) as output [5,6]. There are two types of associations: the first is hetero-associative memory, and the other is auto-associative memory. In the first type, an output is completely different from the input, for example, the network input is an audio file, while the output is a text representing the things that the network understood from the sound, as shown in Fig .3. In the other type, the output is the same as the input, it can be used when the network is trained on several patterns, if a slightly distorted pattern is inserted, the network will recognize the pattern and return the original pattern as shown in Fig. 4, Hopfield network is a type of auto-associative memory [6-9].

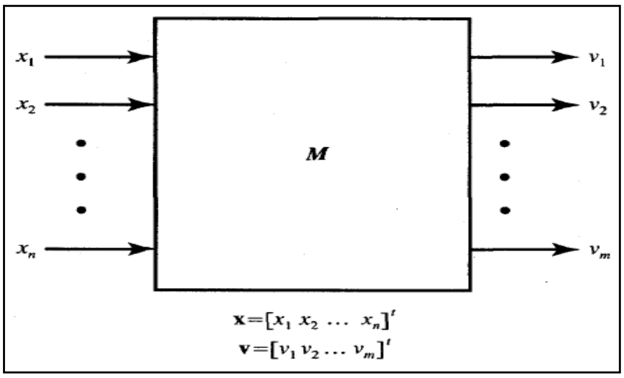



**Figure 2. Block Diagram of Associative Memory.**

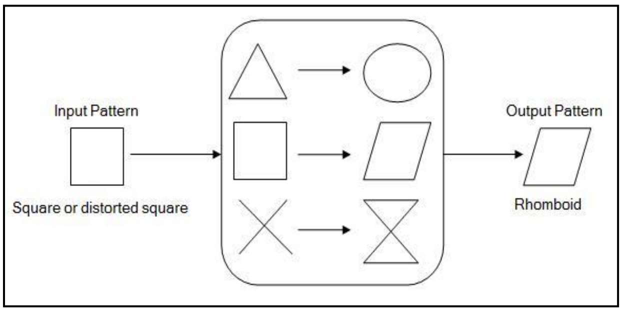



**Figure 3. Hetero- Associative Memory.**

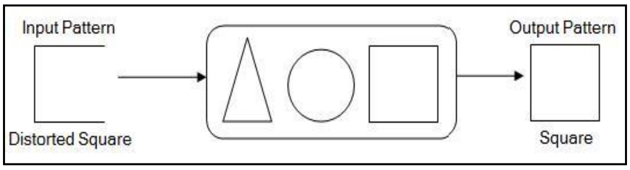



**Figure 4. Auto- Associative Memory.**

### Hopfield Neural Network

The network of the Hopfield is one type of neural network that is classified as single-layer, which contains one layer of weights, and the neurons in it are often split into input neurons that receive signals from the outside world, and output neurons that give the results[7]. In traditional single-layer networks, the input neurons are completely connected to the output neurons, but are not connected to other input neurons, and the output neurons are not linked to other output neurons. One of the most important applications of the Hopfield network is shape recognition and optimizing problem-solving (finding the shortest path in the famous traveling salesman problem)[9-11].

There are some important considerations regarding the Hopfield network: [9, 12]

- The neurons in this model have two outputs, one of which is inverting and the other non-inverting.
- Rather than being its input, each neuron's output should be the input of other neurons.
- *Wij* is used to represent the weight or connection strength.
- Links have both excitatory and inhibitory effects. If the neuron's output matches its input, it would be excitatory; otherwise, it would be inhibitory.
- Weights of this network should be symmetric, *i.e.* $wji = wij$.

**Hopfield Network Architecture**

The Hopfield network consists of several connected neurons, whose effective values update simultaneously and independently of the rest of the neurons, and all neurons in it are input and output neurons at the same time, where the output of all the nodes is linked to the input of all other nodes. Fig. 5 illustrates the Hopfield network operation, where the output from $Y_1$ - $Yi$ *and* $Yn$ have the weights $w_{12}$ - $w_{1i}$ *and* $w_{1n}$ respectively[9, 12].

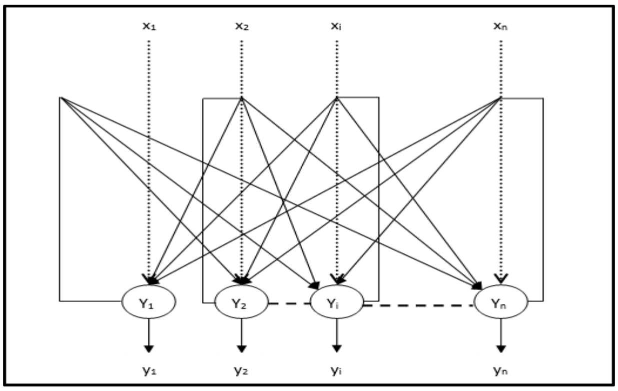



**Figure 5. Hopfield Network Operation.**

There are two phases for the work of the Hopfield network, the training and the convergence phases, as shown in Fig. 6, to training pattern in this network needs to create symmetrical weights ($t_{ij} = t_{ji}$), and these weights will not change if there is more than one training pattern ($S_1$, $S_2$, … $S_n$) additional operations will be performed on all symmetric weights for all training patterns ($t_1$, $t_2$, … $t_n$) to generate associative weights for each training pattern [9, 10].

The convergence phase begins once the Hopfield network is initialized with unknown patterns, and this procedure is repeated until there is no change in the output of the network throughout successive rounds. Following that, the procedure is terminated, the recovered pattern is compared to stored patterns, and a class is assigned to it [9, 13].

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

Input: n Patterns.
Output: Convergence Pattern.

Step one: Set connection weights:

$$t_{ij} = \begin{cases} \sum_{s=0}^{M-1} x_i^s x_j^s & i \neq j \\ 0 & i = j, 0 \leq i, j \leq n-1 \end{cases}$$

Where
N: is elements number in input vector.
M: is patterns number.
$t_{ij}$ : is representing the connection weight from i to j nodes.
s: is the training pattern which it saved.
s: is the training pattern which it saved.
$x_i$ : is an element in pattern (+1 / -1).
Step two: prepares the unknown patterns:

$$\mu(0) = x_i, \quad 0 \leq i \leq n-1$$

where:
$\mu(t)$ : is the node output i at time t.
Step three: loops until convergence:

$$\mu(t+1) = f_k\left[\sum_{i=0}^{n-1} t_{ij}\mu(t)\right] \quad 0 \leq i \leq n-1$$

where:
$f_k$ : is the hard limiting function.
Step four: End

**Figure 6. Hopfield Neural Network**

## Related Works

Díaz de León JL, Gamino Carranza A.,[14] In this article, the extended operations (XOR/ XNOR), which are the new auto inverse operations produced from the original operations (XOR and XNOR, respectively), are used to build a binary associative memory model, This model produces two different types of associative memory: the first one is built using the extended XOR operation's maximum called max type (XOR-AM max), and other one is built using the extended XNOR operation's minimum called min type (XOR-AM min). The max type (XOR-AM max) shows tolerance against the presence of distortion by dilative noise patterns, while the min type (XOR-AM min) shows tolerance against the presence of erosive anamorphic patterns; these types of memory converge in one step, using the same extended operator of XOR/XNOR for the training phase and the convergence phase, working in auto-associative memory, and demonstrating the unlimited storage capacity that comes with auto-associative memory. Finally, the results of computer simulations of novel memories based on extended XOR/XNOR (XOR-AM) are presented, they have equal or better rendering compared with other associative memories.

Folli V, et al[15], in this article, the storage performance of the generalized Hopfield network under the model at finite N, where the diagonal elements of the conductance matrix are considered to be different from zero, is examined in this article. It also shows that as the number of stored patterns is raised past a certain point, the retrieval errors start to decrease until they are less than unity for P N. It gives a mathematical expression for the number of retrieval errors. It shows that the amount of patterns (P) stored in the network determines the strength of the trade-off between effectiveness and efficiency by appropriately changing the connection weights. The number of stored memories required to reach the ideal storage capacity is significantly more than previously reported for PN and the diagonal adjacency matrix members are not required to be zero.

Kasihmuddin et al[16], as a Bezier properties validation technique, this research proposed rebuilding the Bezier curve model by solving the satisfaction problem in Hopfield neural network and representing the properties of the Bezier model in two satisfiability (2SAT). Then, In order to detect the presence of any non-Bezier curve, the generated Bezier model will be combined with a Hopfield neural network. The results of this paper are evaluated in terms of computation time and the global Bezier model. The majority of the models generated by HNN-2SAT are Bezier curve models.

Mohd Asyraf et al [17], the HNN-3SAT with Hyperbolic Tangent Activation Function and the traditional McCulloch-Pitts function was proposed in this research. The purpose of this research is to look at the accuracy of the patterns generated by HNN-3SAT; the results of HNN-3SAT discussion depend on running time and global pattern-SAT.

Kareem EIA et al[18], the Hopfield neural network will be enhanced by this research's proposal for multi-connect architecture associative memory (MCA) by changing the network's architecture, learning, and convergence processes. By eliminating most of the Hopfield network restrictions; this update aims to improve associative memory neural network performance. Generally, MCA is a single-layer network that works in the learning and convergence phases. It uses auto-association tasks. MCA was created using two guiding ideas. First, rather than relying on the size of the pattern, the smallest network size will be chosen. Second, only the limited portions of the pattern will be learned in order to avoid having to learn the same portions repeatedly. The results of the trials are encouraging when MCA exhibits highly effective associative memory by bypassing the majority of the Hopfield network constraints. In contrast to the conventional Hopfield neural network, the results showed that the MCA network can learn and recognize an infinite number of patterns of varied sizes with an acceptable percentage of noise rates.

## Proposed Method for New Strategy XOR Operation

A new strategy based on the XOR operation is proposed to modify Hopfield auto-associative memory. These modifications include the algorithm for the training and convergence phases; in general, the proposed XOR associative memory processes (training phase and convergence phase) are less complex than other neural networks, because the size of the grid will be the same as the size of the pattern, regardless of the number of entries to the network, because it will store the entered patterns in a dimensional array, and this will be very effective for real-time pattern recognition, In addition, it does not need many complex mathematical operations, as the X-OR process is one of the simplest mathematical operations.

The proposed new strategy has the ability to overcome the majority of associative memory learning-phase limitations and is able to deal with automatic associative memory problems in the convergence phase, such as pattern inverse value, local minima, and ratio limits. Permissible percentage of noise rate and show infinite storage capacity. In addition, there is no need to convert the pattern to bipolar representation because XOR operations deal with 0 and 1. As shown in next sections.

### Training Phase

Algorithm 1 shows the training phase in the proposed method, the training process is considered one of the most important processes that affect the efficiency of the network, thus, a comprehensive modification of the training phase was proposed.

**Algorithm 1: Training Phase**

| |
|---|
| Input: Training Patterns. |
| Output: N- Dimensional Matrix (*MD*) |
| Step_1: Initialize N-dimensional (*MD*)<br>**Matrix Size= Width*Height* Dimensional*<br>Where Dimensional = Number of Training patterns.<br>Step_2: Convert Pattern to Binary Each Pixel (1, 0);<br>Step_3: Store Training Patterns in N- Dimensional Matrix (*MD*)<br>**End** |

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

The training algorithm illustrates that the input for the XOR strategy is a stream of patterns, it is used as a training pattern, and then these patterns are stored in N-dimensional matrixes as shown in Fig. 7

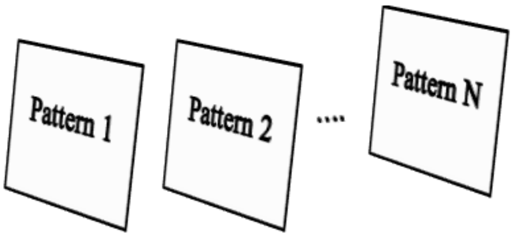



**Figure 7. N- Dimensional Matrixes to Store Patterns**

### Convergence Phase

The convergence phase was modified in order to ensure high efficiency for the XOR strategy, this phase is used to detect unknown patterns after converting them to binary form and applying the XOR operation with the dimensional matrix that was stored in the training phase, Algorithm 2 shows the Convergence Phase

**Algorithm 2: Convergence Phase**

| |
|---|
| Input: Pattern Unknown (*NP*) |
| Output: Convergence the Pattern (*CP*) |
| Step_1: Recall Matrix (*MD*) From Training Phase<br>Step_2: Convert Unknown Pattern to Binary Each Pixel (1, 0)<br>Step_3: Repeat Step_3-1 until Matrix (*MD*) is Ended<br>Step_3-1: Apply XOR Operation Between Unknown Pattern (*NP*) and Matrix (MD) (Each Dimension Separately) and Stored in Variable SV<br>$SV_i = NP \oplus MD_i$<br>Step_4: Summation all Stored Vector ($SV_i$)<br>$SumSV = \sum_{i=1}^{n} SV$<br>Step_5: Select minimum *SumSW* from stored pattern number to Convergence the pattern:<br>$minvalue = \min(SumSVi)$<br>Where: min function use to select minimum value to locate the pattern in matrix (*MD*).<br>Step_6: Return the Pattern with location of minimum value from Matrix (*MD*) that recalled in the Step_1.<br>End |

### Test Examples

This section provides an example of the proposed XOR strategy; all steps of the training phase and convergence phase are illustrated in this example. This example will train in two patterns as shown in Fig. 8.

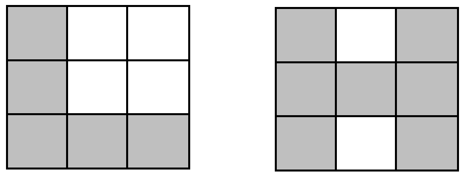


**Figure 8. Two Patterns (P1 and P2)**

Initialize N-dimensional (MD)

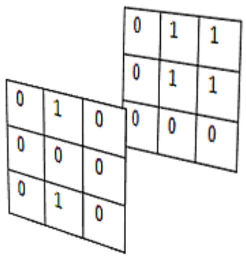
*Matrix Size= 3*3*2

Each pattern must be converted to binary pixels (0 and 1)

L= [0, 1, 1; 0, 1, 1; 0, 0, 0]

H= [0, 1, 0; 0, 0, 0;0, 1, 0]

Store pattern in 2- dimensional matrix because there are only two patterns




Then applying the convergence phase to one unknown pattern, the unknown pattern Un-Pat," as shown in Fig. 9.

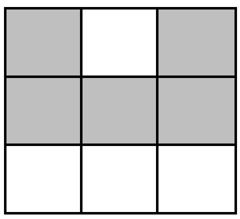
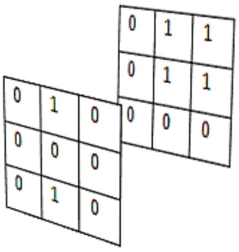


**Figure 9. Unknown Pattern**

Recall Matrix (MD)) from training phase

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal



Convert unknown pattern to binary

Unknown pattern= [0, 1, 0; 0, 0, 0; 1, 1, 1]

For each pattern in Matrix (MD) apply XOR operation with unknown pattern.

| Pattern 1 | Pattern 1 |
|---|---|
| MXORD1 = 0⊕0=0 | MXORD2 = 0⊕0=0 |
| MXORD1 = 1⊕1=1 | MXORD2 = 1⊕1=0 |
| MXORD1 = 1⊕ 0=1 | MXORD2 = 0⊕0=0 |
| MXORD1 = 0⊕0=0 | MXORD2 = 0⊕0=0 |
| MXORD1 = 1⊕0=1 | MXORD2 = 0⊕0=0 |
| MXORD1 = 1⊕0=1 | MXORD2 = 0⊕0=0 |
| MXORD1 = 0⊕1=1 | MXORD2 = 0⊕1=1 |
| MXORD1 = 0⊕1=1 | MXORD2 = 1⊕1=0 |
| MXORD1 = 0⊕1=1 | MXORD2 = 0⊕1=1 |

## Results and Discussion

To evaluate the performance of the proposed New Strategy for Designing the Hopfield network using XOR Operation, several experiments will be presented in this section, and the results of these experiments will be analyzed and discussed. These experiments are applied to the conventional Hopfield network and to the proposed XOR strategy to make a comparison between the two. These experiments highlight the efficiency of the proposed method by dealing with Hopfield network constraints. The experiments were conducted on a set of patterns, and these patterns are the alphabetical letters with d different sizes, as these patterns (10*10, 16*16, 32*32, 64*64 and 128*128) were used to evaluate these experiments. The experiments are carried out in two phases (the learning and the convergence phases). In the learning phase, a set of patterns are learned for one time only, as the proposed strategy is based on the size of the pattern but will avoid learning similar patterns many times, while the known pattern is detected using the convergence phase.

XOR operation result stored in variable $SV_i$

SV1=[0,0,1,0,1,1,1,1,1]

SV2=[0,0,0,0,0,0,1,0,1]

Summation all Stored Vector (SV)

$$SumSV1 = \sum_{i=1}^{n} SV$$

SumSv1=6

SumSv2=2

Select the minimum value by using min function to determine the stored pattern number *(minvalue)* to locate the pattern.

$$minvalue = \min(SumSV)$$

Minvalue=2

The proposed XOR strategy recognized the unknown pattern correctly, where return the pattern 2

**Experiment 1: Network Capacity**

In this experiment, training both the Hopfield network and the proposed method (XOR Operation) on the maximum number of patterns (these patterns were alphabetical letters with size (16 * 16)) without noise, and the training stops when one of the two networks fails to recognize the most pattern storage.

Figs. 10 and Fig. 11 show the result of this experiment, the convergence Hopfield network started failing with four stored patterns, and with six and seven stored patterns failed to recognize most of the patterns (that means: When the stored patterns number increases, the percentage of convergence of the Hopfield network decreases). While the convergence rate of the proposed XOR strategy was 100% even if the stored patterns number increases the same convergence rate will be maintained.

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986



| No. Of Stored Pattern | Hopfield Neural Network | | | | | | | XOR Operation | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | A | B | C | D | E | F | | A | B | C | D | E | F | |
| 1 | A | | | | | | | A | | | | | | |
| 2 | A | B | | | | | | A | B | | | | | |
| 3 | A | B | C | | | | | A | B | C | | | | |
| 4 | A | B | C | D | | | | A | B | C | D | | | |
| 5 | A | B | C | D | E | | | A | B | C | D | E | | |
| 6 | A | E | C | D | E | E | | A | B | C | D | E | F | |
| 7 | A | B | C | D | E | E | C | A | B | C | D | E | F | G |

**Figure 10. Network Capacity Experiment**

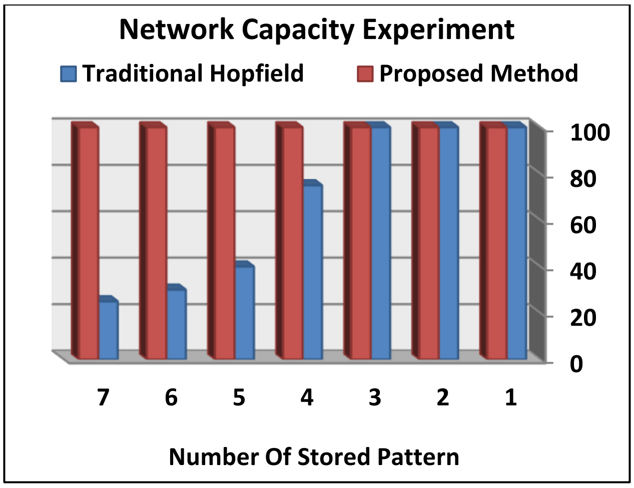



**Figure 11. The Diagram Illustrates for Network Capacity Experiment to Comparison between Hopfield Network and the XOR strategy.**

**Experiment 2: Noise Percentage Rate**

In this experiment, using different noise percentage rates (random noise from 10% to 90%) with different pattern sizes to calculate the convergence ratio. This experiment proves that the proposed method deals with large patterns more efficiently than with small patterns.

Fig. 12 shows the allowable percentage of noise between different noise ratios with different size patterns, (100) Patterns were used in this experiment, the results showed the proposed method convergence response was more efficient when the patterns size is increased. The efficiency of the proposed method is very high whenever the noise ratio is low, while the network efficiency decreases when the noise ratio increases, as patterns are not recognized.

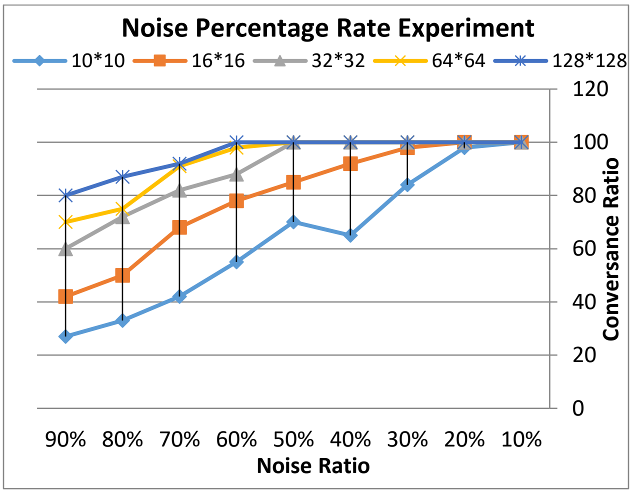



**Figure 12. Noise Percentage Rate Experiment**

## Conclusion

This paper aims to improve the efficiency of the Hopfield neural network's associative memory. This modification was achieved by proposing a new strategy for designing the Hopfield network based on XOR operations, by modifying the network architecture, training phase and convergence phase, thus bypassing most of the limitations suffered by the Hopfield network in particular, and associative memory in general, through the experiments above.

## Authors' Declaration

- Conflicts of Interest: None.

- I hereby confirm that all the Figures and Tables in the manuscript are mine. Furthermore, any Figures and images, that are not mine, have been included with the necessary permission for re-publication, which is attached to the manuscript.
- Ethical Clearance: The project was approved by the local ethical committee at University of Baghdad.

## References

1. Rettig O, Müller S, Strand M, Katic D. Which deep artificial neural network architecture to use for anomaly detection in Mobile Robots kinematic data? Machine Learning for Cyber Physical Systems. Berlin, Heidelberg: Springer Berlin Heidelberg; 2019. p. 58–65. https://doi.org/10.1007/978-3-662-58485-97
2. Zeigler-Hill V, Shackelford T .K, Artificial Neural Networks, Encyclopedia of Personality and Individual Differences. Springer. Cham. 2020. https://doi.org/10.1007/978-3-319-24612-3_300173
3. Ramamurthy G, Swamy TJ. Novel associative memories based on spherical separability. Advances in Intelligent Systems and Computing. Singapore: Springer Nature Singapore; 2022. p. 351–8, https://doi.org/10.1007/978-981-16-7088-6_32
4. Virvou M, Tsihrintzis GA, Jain LC. Introduction to advances in selected artificial intelligence areas. In: Learning and Analytics in Intelligent Systems. Cham: Springer International Publishing; 2022. p. 1–7. https://doi.org/10.1007/978-3-030-93052-3_1
5. Wan G, Wang L, Zou H, Jiang S. A new model of associative memory neural network based on an improved memristor. 39th Chinese Control Conference (CCC). IEEE; 2020, https://doi.org/10.23919/CCC50068.2020.9188654
6. Miikkulainen, R, Hopfield Network. Phung D., Webb, G.I., Sammut, C. (eds) Encyclopedia of Machine Learning and Data Science. Springer, New York, NY, 2023. https://doi.org/10.1007/978-1-4899-7502-7_127-2
7. Sharma N, Kalra K, Sarangi PK, Rani L, Saxena M, Kumar S. Pattern storage & recalling using Hopfield neural network and HOG feature based SVM classifier: An experiment with handwritten Odia numerals. International Conference on Emerging Smart Computing and Informatics (ESCI). IEEE; 2023., https://doi.org/10.1109/ESCI56872.2023.10099928
8. Razzaq AN, Ghazali R, El Abbadi NK, Al Naffakh HAH. Human Face Recognition Based on Local Ternary Pattern and Singular Value Decomposition. Baghdad Sci J. 2022 Oct. 1 [cited 2023 Aug. 19]; 19(5): 1090. https://doi.org/10.21123/bsj.2022.6145
9. Hopfield JJ. Neural networks and physical systems with emergent collective computational abilities. Proc Natl Acad Sci. 1982; 79(8): 2554–8. https://doi.org/10.1073/pnas.79.8.2554
10. Al-Husban A, Karoun RC, Heilat AS, Horani MA, Khennaoui AA, Grassi G, et al. Chaos in a two dimensional fractional discrete Hopfield neural network and its control. Alex Eng J. 2023; 75: 627–38. http://dx.doi.org/10.1016/j.aej.2023.05.078
11. Ashour MAH. Optimized Artificial Neural network models to time series. Baghdad Sci. J. 2022 Aug. 1; 19(4): 0899. https://doi.org/10.21123/bsj.2022.19.4.0899
12. Rusdi N ’afifah, Kasihmuddin MSM, Romli NA, Manoharam G, Mansor MA. Multi-unit Discrete Hopfield Neural Network for higher order supervised learning through logic mining: Optimal performance design and attribute selection. J King Saud Univ - Comput Inf Sci. 2023; 35(5): 101554. http://dx.doi.org/10.1016/j.jksuci.2023.101554
13. Kelleher JD. Deep Learning. The MIT Press; 2019, https://doi.org/10.7551/mitpress/11171.001.0001
14. Díaz de León JL, Gamino Carranza A. New binary associative memory model based on the XOR operation. Appl Algebra Engrg Comm Comput. 2022; 33(3): 283–320. http://dx.doi.org/10.1007/s00200-020-00446-8
15. Folli V, Leonetti M, Ruocco G. On the maximum storage capacity of the Hopfield model. Front Comput Neurosci. 2017; 10. http://dx.doi.org/10.3389/fncom.2016.00144
16. Kasihmuddin MSM, Mansor MA, Sathasivam S. Bezier Curves Satisfiability Model in Enhanced Hopfield Network. International Journal of Intelligent Systems and Applications. 2016 Dec 8;8(12):9–17. http://dx.doi.org/10.5815/ijisa.2016.12.02
17. Mohd Asyraf Mansor, Mohd Shareduwan M. Kasihmuddin, Saratha Sathasivam. Enhanced Hopfield Network for Pattern Satisfiability Optimization. International journal of intelligent systems and applications. 2016 Nov 8;8(11):27–33. http://dx.doi.org/10.5815/ijisa.2016.11.04
18. Kareem EIA, Alsalihy WAHA, Jantan A. Multi-connect architecture (MCA) associative memory: A modified Hopfield neural network. Intell Autom Soft Comput. 2012; 18(3): 279–96. http://dx.doi.org/10.1080/10798587.2008.10643243

2024, 21(9): 3052-3061
https://doi.org/10.21123/bsj.2024.9238
P-ISSN: 2078-8665 - E-ISSN: 2411-7986
Baghdad Science Journal

# إستراتيجية جديدة لتعديل شبكة الهوبفيلد باستخدام عملية XOR

**رسل حسين حسن**

كلية القانون، جامعة بغداد، بغداد، العراق.

## الخلاصة

تعد شبكة هوبفيلد واحدة من أسهل الأنواع الشبكات العصبية، تركيب الشبكة يكون كل خلية عصبية في الشبكة تتصل بالخلية الأخرى، وبالتالي تسمى الشبكة العصبية المتصلة بالكامل .بالإضافة إلى ذلك، يعتبر هذا النوع ذاكرة ارتباطية تلقائية، نظرًا لأن الشبكة تقوم بإرجاع النمط فور التعرف عليه، فإن هذه الشبكة بها العديد من القيود، بما في ذلك سعة الذاكرة، والتباين، والمتعامد بين الأنماط، والاوزان المتماثلة، والحد الأدنى .هذا البحث يقترح استراتيجية جديدة لتصميم شبكة الهوبفيلد باستخدام عملية XOR؛ حيث تم اقتراح إستراتيجية جديدة لحل هذه القيود من خلال اقتراح خوارزمية جديدة، باستخدام هذه الاستراتيجية سيزيد من تحسين أداء شبكة الهوبفيلد من خلال تعديل بنية الشبكة، ومراحل التدريب والتقارب ، وان الاستراتيجية المقترحة تعتمد على حجم النمط .و تتجنب تعلم نمط مشابه عدة مرات، و تظهر قابليتها في التعرف على أنماط مشوهة بالضوضاء، وسعة تخزينها غير محدودة وحل مشكلة النمط المعكوس .أظهرت التجارب أن الطريقة المقترحة لها نتائج جيدة من خلال تجنب غالبية قيود شبكة هوبفيلد .بالإضافة إلى ذلك ، يتعلم التعرف على عدد لا حصر له من الأنماط بأحجام مختلفة مع الحفاظ على نسبة ضوضاء مناسبة.

**الكلمات المفتاحية:** الذاكرة الترابطية التلقائية ,شبكة الهوبفيلد ,الشبكة العصبية ,تمييز الانماط وعمليةXOR .