Quality Solution of Logic Programming in Hopfield Neural Network

M S M Kasihmuddin\textsuperscript{1}, M A Mansor\textsuperscript{2}, S Alzaeemi\textsuperscript{1}, M F M Basir\textsuperscript{1} and S Sathasivam\textsuperscript{1}

\textsuperscript{1}School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Pulau Pinang, Malaysia.
\textsuperscript{2}School of Distance Education, Universiti Sains Malaysia, 11800 USM, Pulau Pinang, Malaysia.

Email: shareduwan@usm.my; asyrafman@usm.my; shehab_alzaeemi@yahoo.com; mfaishalmbasir@usm.my; sathasivam@usm.my

Abstract. The dynamical behaviours of Artificial neural network (ANN) system are strongly dependent by its network structure. In that sense, the output of ANN has long suffered from a lack of interpretability and variation. This has severely limited the practical usability of ANN in doing logic programming. The work presents an integrated representation of 2 Satisfiability (2SAT) in different Hopfield Neural Network (HNN) models. Neuron states of HNN always converge to minimum energy but the solution produced always confined in limited number of solution space. The main purpose of this study is to explore the quality of the solution obtained from HNN. It has been shown that HNN only retrieves limited neuron states with the lowest minimum energy. This finding will lead to a better understand of logic programming in HNN.

1. Introduction
In recent years, the output of Hopfield Neural Network (HNN) has been greatly utilized in the field of pattern recognition [1], operational research [2], constraint optimization problem [3], data mining [4] and many more [5]. HNN consist of neurons that were connected by synaptic weight. One of the beneficial property of HNN is the minimization of energy (towards local or global minimum) when the neuron changes state. The search of optimal HNN for a given problem is an increasingly complex task. Data will be “learned” by ANN and from this data, HNN will provide researcher with useful output to tackle various real life problems. In this case, HNN will act as a black box model that handles computation internally before it can produce a meaningful output for user. The pursuit of understanding the HNN model attained a new level when symbol and rules has been included in the HNN computation. This perspective inspires more researcher to merged the idea of logic programming with HNN.

One of the earliest model of this merger were proposed by Wan Abdullah [6] and Sathasivam [7]. In these studies, Horn clauses has been implemented in HNN by minimizing the inconsistencies of logical clauses. The proposed model obtained synaptic weight by comparing cost function with Lyapunov energy equation. This significant hybridization motivates more exploration towards the capability of logical rule and neural network. Hamadneh [8] proposed a new paradigm by proposing logic
programming in Radial Basis function neural network. In terms of HNN variant, several hybrid network has been proposed such as Boltzmann-HNN [9], Mean Field Theory-HNN [10] and Kernel-HNN [11]. All the proposed network mentioned above reduce the number of local minimum solution (>90% global minima ratio) during retrieval phase of HNN. Although the quality of the minimum energy obtained from the network has improved, the behavior of the retrieved neuron state that corresponds to the embedded logical rule is poorly understood.

The most common problem emerged in HNN is the repetitive pattern of neuron state. Although the pattern obtained from the network is considered global minimum energy, it was believed that there were other neuron states which leads to global minimum energy (in different location in solution space). In some case, repetitive final neuron state might reduce the accuracy of the network or overfitting. The correct paradigm in examining the quality of neuron state in HNN model lies in retrieval phase. Bipolar feature vector is one of the most common representation of similarity measures. Similarity measures plays an important role in classification and pattern recognition. Ever since Jaccard [12], proposed a similarity measure to classify species in 1901, numerous binary similarity approaches have been proposed in various field. In the field of ANN, similarity analysis can be utilized to examine the quality of the final neuron state [13-14]. Interestingly, this is the first paper that explore the behavior of final state for logic programming in HNN. Most studies put much emphasis and effort to create hybrid HNN model that achieve the most global minimum energy. The rational behind the implementation of similarity measure is to reveal the repetitive nature of neuron state in HNN. This paper will critically explore the behavior of the neuron states in several established HNN models in literature.

2. 2 Satisfiability Representation

Various optimization problem can be translated into 2 Satisfiability (2SAT) representation. 2SAT is a logical rule that comprises of strictly 2 literals per clause [15]. The three important component of 2SAT representation is as follows

1. A set of \( n \) variables, \( x_1, x_2, ..., x_n \).
2. A set of literals. A literal is a variable \( T \) or a negation of variable \( \lnot T \).
3. A set of \( m \) distinct clauses; \( C_1, C_2, ..., C_m \) connected with logical AND \( (\wedge) \). Each \( C_i \) consist of exactly 2 literals and connected with logical OR \( (\vee) \).

Hence the general formula for 2SAT can be defined as

\[
P_{2SAT} = \bigwedge_{i=1}^{m} C_i \text{ where } C_i = \bigvee_{j=1}^{2} (x_{ij}, y_{ij})
\]

Each of the variable can only take bipolar value \( x_{ij}, y_{ij} \in \{-1, 1\} \) which represents FALSE and TRUE respectively. Several variant of 2SAT has been studies such as MAX-2SAT [16] and weighted 2SAT [17]. The goal of 2SAT is to find the optimal assignment that makes the following formula becomes true.

3. Logic Programming in Hopfield Neural Network

HNN is a classical recurrent network with symmetrical connected synaptic weight corresponds to the interconnected neurons. This network comprises of good characteristics such as parallel computation, fast convergence and effective content addressable memory (CAM) capability [18]. Consider the bipolar neuron state in HNN that is denoted as \( S_i(t) \in \{-1, 1\} \) where \( i = 1, 2, 3, ..., N \). The general asynchronous updating rule of HNN is given by:
\[ S_i = \begin{cases} 1 & \text{if } \sum_j W_{ij} S_j > \xi \\ -1 & \text{Otherwise} \end{cases} \]  

where \( W_{ij} \) is the weight for unit \( j \) to \( i \) and \( \xi \) refers to the threshold of the HNN. The implementation of 2SAT in HNN is denoted as HNN-2SAT. In this case, HNN-2SAT will consider the 2 neurons per clause. During retrieval phase, the local field of HNN-2SAT is:

\[ h_i = \sum_{j=1, i \neq j}^{N} W_{ij}^{(2)} S_j + W_{i}^{(1)} \]

where \( i \) and \( j \) are corresponded to neurons \( N \). Synaptic weight of Equation (3) is limited to be symmetric \( W_{ij}^{(2)} = W_{ji}^{(2)} \) and zero diagonal \( W_{ii}^{(2)} = W_{jj}^{(2)} = 0 \). The local field is prominent to properly squash the retrieved output before generating the final state. According to several studies [19], the local field of HNN is able to locate the global solution correctly. Hence, the final neuron state is as follows:

\[ S_i(t+1) = \text{sgn}[h_i(t)] \]  

In order to examine the “correctness” of the 2SAT rule embedded in HNN, final neuron state is further examined by using the Lyapunov energy function

\[ H_{2SAT} = -\frac{1}{2} \sum_{i=1, i \neq j}^{N} \sum_{j=1, i \neq j}^{N} W_{ij}^{(2)} S_i S_j - \sum_{i=1}^{N} W_{i}^{(1)} S_i, \quad k = 2 \]

Interestingly, any initial neuron state of 2SAT that has been updated by using Equation (4) will arrive to global minimum energy [21]. This structure guarantees the convergence of the neuron state to minimum solution. To further investigate the quality of the retrieval phase, other variants of HNN will be explored.

4. Neuron State Analysis of HNN-2SAT

4.1. Benchmark State

The key component of analyzing the final state of neuron is by comparing the retrieved state with an ideal neuron state. In this section, analysis of final state of neuron in HNN-2SAT model will be studied. Benchmark state is defined as the ideal neuron state retrieved from the HNN model. The benchmark neuron state is given as follows

\[ S_i^{\text{max}} = \begin{cases} 1 & ,x \\ -1 & ,\neg x \end{cases} \]  

where \( x \) and \( \neg x \) are positive and negative literal in 2SAT formula respectively. Based on equation (41), if the logical rule reads \( P_{2SAT} = (A \lor B) \land (C \lor \neg D) \land (\neg E \lor \neg F) \), the benchmark state of the neuron is given as \( S_A^{\text{max}} = 1, S_B^{\text{max}} = 1, S_C^{\text{max}} = 1, S_D^{\text{max}} = -1, S_E^{\text{max}} = -1, S_F^{\text{max}} = -1 \) or \( S_i^{\text{max}} = (1,1,1,-1,-1,-1) \). Worth mentioning that the final energy of \( S_i^{\text{max}} \) is always global minimum solution or \( H_{S_i^{\text{max}}} - H_{P_{2SAT}}^{\text{min}} \leq \partial [21] \). \( \partial \) is a tolerance value for energy difference in HNN. Since most of the neuron state retrieved in HNN achieve global minimum energy [22], \( S_i^{\text{max}} \) is considered as a perfect benchmark state in comparing the final state of different HNN model.
4.2. Similarity Metrics

Analyzing the behavior of the final neuron state in HNN-2SAT is a challenging task. In this section, the final neuron state retrieved that corresponds to 2SAT logical rule will be analyzed by using similarity metrics. Several similarity metrics were identified to explore the lack of variation of HNN models. In this case, instead of comparing logic with logic, the comparison will be made based on the individual neuron state. Hence, the general comparison between benchmark state and the final neuron state is given by:

\[ C_{S_{\text{max}}S_i} = \left\{ \left(S_{i_{\text{max}}}, S_i \right) \mid i = 1, 2, \ldots, n \right\} \]

(7)

The further specification of the equation (42) is defined as follows

- \( l \) is the number of \( \left(S_{i_{\text{max}}}, S_i \right) \) where both elements have the value 1 in \( C_{S_{\text{max}}S_i} \).
- \( m \) is the number of \( \left(S_{i_{\text{max}}}, S_i \right) \) where \( S_{i_{\text{max}}} \) is 1 and \( S_i \) is -1 in \( C_{S_{\text{max}}S_i} \).
- \( n \) is the number of \( \left(S_{i_{\text{max}}}, S_i \right) \) where \( S_{i_{\text{max}}} \) is -1 and \( S_i \) is 1 in \( C_{S_{\text{max}}S_i} \).
- \( o \) is the number of \( \left(S_{i_{\text{max}}}, S_i \right) \) where both elements have the value -1 in \( C_{S_{\text{max}}S_i} \).

The size of the neuron string is given as \( n = l + m + n + o \). By using the above information, similarity coefficient for all HNN model is given as follows

Jaccard’s Index [23]

\[ J\left(S_{i_{\text{max}}}, S_i \right) = \frac{l}{l + m + n} \]

(8)

Sokal-Sneath 2 [24]

\[ SS\left(S_{i_{\text{max}}}, S_i \right) = \frac{l}{l + 2(m + n)} \]

(9)

For example, if the benchmark state reads \( S_{i_{\text{max}}} = (1,1,1,1,1,-1,-1) \) and the final state of the neuron is given by \( S_i = (1,1,1,-1,1,-1) \). The value of the similarity index for a particular HNN-2SAT model based on equation (8) and (9) are given as follows

\[ J\left(S_{i_{\text{max}}}, S_i \right) = 0.4 \]

\[ SS\left(S_{i_{\text{max}}}, S_i \right) = 0.25 \]

(10)

Worth mentioning that, high similarity index signifies low variation of final neuron state compared to benchmark neuron state. The aim of this paper is to find HNN model that has the highest variation of final neuron state.

5. Similarity Analysis of HNN Models

During retrieval phase, neuron state will be updated by capitalizing the synaptic weight obtained during learning phase. During learning phase, all randomized 2SAT clauses will be used to derive the optimal cost function by minimizing the logical inconsistencies. Consequently, the main task of the all the proposed HNN is to create a “model” that behave according to 2SAT logical rule. In this section, the basic algorithm of HNN-2SAT that complies with the work of [9 -11, 22] will be implemented. The following algorithm shows the implementation of basic HNN-2SAT:

1. Transform 2SAT clauses to Boolean algebra (if applicable).
2. Initialize the neuron state by assigning neuron with the variable in 2SAT formula.

3. Derive the cost function of 2SAT by assigning \( A = \frac{1}{2} (1 + S_d) \) or \( \neg A = \frac{1}{2} (1 - S_d) \). In this case, \( \wedge \) and \( \vee \) represents multiplication and addition. Note that the neuron state reads true if \( S_d = 1 \) and reads false if \( S_d = -1 \).

4. Obtain satisfied interpretation corresponds to 2SAT formula.

5. Obtain the synaptic weight matrix of the HNN-2SAT model corresponds to each 2SAT clause.

6. Compute the lowest minimum energy of HNN-2SAT by using equation (5).

7. Obtain the benchmark state of 2SAT formula by using equation (6).

8. Compute the final neuron state via equation (3) and (4).

9. Compute the final energy of the HNN-2SAT by using equation (5). If the final energy is within the tolerance value, the final energy is considered as global minimum energy.

10. For every neuron state that reach global minimum solution, compute similarity index by using equation (8) until (11).

The HNN model that have been explored in this paper are HNN-2SAT [22], HNN with Boltzmann Machine (BHNN-2SAT) [9], HNN integrated with Kernel machine (KHNN-2SAT) [11] and HNN integrated with Mean Field Theory MFTHNN-2SAT [10]. All HNN models utilized simulated datasets by generating random 2SAT clauses. In order to obtain a fair comparison among all HNN-2SAT models, all source code will be implemented via Microsoft Visual Basic C++ 2013 for Windows 10. Similar device will be used in every simulation to avoid possible bad sector. Table 1 until Table 4 stated all the parameters involved in each HNN-2SAT models.

| Table 1. List of parameters in HNN-2SAT [22] |
| Parameter | Parameter value |
| Neuron Combination | 100 |
| Tolerance Value (\( \bar{\alpha} \)) | 0.001 |
| Number of learning (\( \theta \)) | 100 |
| No_ Neuron String | 100 |
| Selection_Rate | 0.1 |

| Table 2. List of parameters in KHNN-2SAT [11] |
| Parameter | Parameter value |
| Neuron Combination | 100 |
| Tolerance Value (\( \bar{\alpha} \)) | 0.001 |
| Number of learning, (\( \theta \)) | 100 |
| No_ Neuron String | 100 |
| Selection_Rate | 0.1 |
| Type of Kernel | Linear Kernel |

| Table 3. List of parameters in BHNN-2SAT [9] |
| Parameter | Parameter value |
| Neuron Combination | 100 |
| Tolerance Value (\( \bar{\alpha} \)) | 0.001 |
| Number of learning, (\( \theta \)) | 100 |
| No_ Neuron String | 100 |
| Selection_Rate | 0.1 |
| Temperature (\( T \)) | 70 |

| Table 4. List of parameters in MFTHNN-2SAT [10] |
| Parameter                  | Parameter value |
|----------------------------|-----------------|
| Neuron Combination         | 100             |
| Tolerance Value (\(\varepsilon\)) | 0.001           |
| Number of learning, (\(\theta\)) | 100             |
| No_Neuron String           | 100             |
| Selection Rate             | 0.1             |
| Temperature (\(T\))       | 70              |
| Activation Function        | Hyperbolic Tangent (HTAF) |

6. Result and Discussion

Figure 1. Jaccard Index of HNN Models.

Figure 2. Sokal-Sneath 2 of HNN Models.

Figure 3. Variation of HNN Models.

Figure 4. Zm of HNN Models.
It has been reported that the HNN model with decaying synaptic weight will result in ineffective neuron updates [27]. In this case, all HNN-2SAT model will utilized Wan Abdullah method [6] to find the synaptic weight of the network. After learning phase, all neurons will undergo relaxation [20]. The mentioned setup helps the network to avoid potential local minimum solution. Apparently, local minimum solution will disrupt the similarity measure of the final neuron state. The result in Figure 1 until Figure 4 allow the following observations:

1. MFTHNN-2SAT has the highest value of $Z_m$ compared to other HNN model. This is due to effectiveness of the updating rule of the Boltzmann Machine during high and low temperature. The resulting state will be properly relaxed by using HTAF. HNN-2SAT has the lowest value of $Z_m$ since the traditional updating rule prone to neuron oscillation that results in local minimum energy.

2. HNN-2SAT has the lowest index value for Jaccard and Sokal-Sneath 2. In this case, the neuron retrieved from HNN-2SAT has a lowest similarity with the benchmark state. MFTHNN-2SAT has the highest similarity with the benchmark state. In this case, MFTHNN-2SAT is observed to be bias towards benchmark state. According to variation value, almost 60% of the final state in KHNN-2SAT were similar to benchmark state.

3. This experiment unearth a surprising relationship between similarity index and $Z_m$ value. Higher $Z_m$ value does not signifies lower similarity index. For example, MFTHNN-2SAT model has the highest value of $Z_m$ but the final neuron state is bias to benchmark state. In this case, the model fails to explore more state that lead to global minimum energy.

4. HNN-2SAT and BHNN-2SAT demonstrate the higher value of variation compared to other HNN model. The variation value shows that HNN-2SAT is able to locate other state that leads to global minimum energy. The updating rule of HNN-2SAT may increase the accuracy of HNN models.

5. Unfortunately, the value of traditional error metric such as RMSE does not provide significant insight on the accuracy of the HNN-2SAT. This is due to most of the final state of neuron has reached global minimum solution. Error metric will be beneficial to several study such as [25 - 26].

6. Another perspective that driven the high similarity index is the nature of the dynamical system in HNN models. For instance, the linearized local field in KHNN-2SAT is observed to be bias towards benchmark state. In order to reduce the similarity, several non-linear kernel formulations [11] could increase the variation of the neuron state. BHNN-2SAT and MFTHNN-2SAT has a higher similarity index because the heating of neuron with predefined temperature provide minor oscillation to the neuron state [9 -10].

Another aspect that is not taken into account is computation time. Since all HNN uses the same learning model to find the correct assignment, computation time is not a significant subject to this paper. Overall, the similarity index analysis provides a good measurement to explore the overfitting nature of logic programming in HNN.

7. Conclusion

In this paper, the behavior of the retrieval phase for 2SAT programming in various HNN model have been proposed. The result obtained demonstrates the overfitting nature of HNN-2SAT model. This paper has successfully uncovered the relationship between the global minimum energy and the number of unique final neuron state. In this case, MFTHNN-2SAT has been observed to has the highest similarity measure with regards to the benchmark state. To sum up, this profound behavior HNN-2SAT is only a tip of the iceberg in improving the accuracy and stability of the network. For future work, the metaheuristics-based algorithm can be employed to increase the variation of the solution obtained from HNN-2SAT.

Acknowledgement

This research is supported by Short Term Grant (203/PMATHS/6315226) by Universiti Sains Malaysia.
References

[1] Melin P and Sanchez D 2018 Multi-objective optimization for modular granular neural networks applied to pattern recognition Information Sciences 460 594

[2] Mohandes S R, Zhang X and Mahdiyar A 2019 A comprehensive review on the application of artificial neural networks in building energy analysis Neurocomputing 340 55

[3] Zhou D and Chi M 2019 Pulse-coupled neural network and its optimization for segmentation of electrical faults with infrared thermography Applied Soft Computing 77 252

[4] Zhang J, Williams S O and Wang H 2018 Intelligent computing system based on pattern recognition and data mining algorithms Sustainable Computing: Informatics and Systems 20 192

[5] Kobayashi M 2018 Storage capacity of rotor Hopfield neural networks Neurocomputing 316 30

[6] Abdullah W A T W 1992 Logic programming on a neural network. International journal of intelligent systems 7(6) 513

[7] Sathasivam S 2006 Logic Mining in Neural Networks (Kuala Lumpur: University of Malaya)

[8] Hamadneh N, Sathasivam S, Tilahun S L. and Choon O H 2012 Learning logic programming in radial basis function network via genetic algorithm Journal of Applied Sciences 12(9) 840

[9] Sathasivam S 2011 Boltzmann machine and new activation function comparison Applied Mathematical Sciences 5(78) 3853

[10] Velavan M, Yahya Z R, Abdul Halif M N and Sathasivam S 2016 Mean field theory in doing logic programming using hopfield network Modern Applied Science 10(1) 154

[11] Alzaeemi S A and Sathasivam S 2018 AIP Conf Proc. 1974 1020107

[12] Rahman M, Hassan M R and Buyya R 2010 Jaccard index based availability prediction in enterprise grids Procedia Computer Science 1(1) 2707

[13] Wang X, Lee F and Chen Q 2019 Similarity-preserving hashing based on deep neural networks for large-scale image retrieval Journal of Visual Communication and Image Representation 61 260

[14] Ahmed A A and Mohammed M F 2018 Sairf: A similarity approach for attack intention recognition using fuzzy min-max neural network Journal of Computational Science 25 467

[15] Bollobás B, Borgs C, Chayes J T, Kim J H and Wilson D B 2001 The scaling window of the 2-SAT transition Random Structures & Algorithms 18(3) 201

[16] Bourjolly J M, Hammer P L, Pulleyblank W R and Simeone B 1992 Boolean-combinatorial bounding of maximum 2-satisfiability Computer Science and Operations Research 23

[17] Porschen S and Speckenmeyer E 2007 Algorithms for variable-weighted 2-SAT and dual problems International Conference on Theory and Applications of Satisfiability Testing 4501 173

[18] Hopfield J J 1982 Proceedings of the National Academy of Sciences 79(8) 2554

[19] Cabrera E and Sossa H 2018 Generating exponentially stable states for a Hopfield Neural Network Neurocomputing 275 358

[20] Sathasivam S 2010 Upgrading logic programming in Hopfield network Sains Malaysiana 39(1) 115

[21] Kasihmuddin M S M, Mansor M A and Sathasivam S 2018 Discrete Hopfield Neural Network in Restricted Maximum k-Satisfiability Logic Programming Sains Malaysiana 47(6) 1327

[22] Kasihmuddin M S M, Mansor M A and Sathasivam S 2017 Hybrid Genetic Algorithm in the Hopfield Network for Logic Satisfiability Problem Pertanika Journal of Science & Technology 25(1) 139

[23] Bag S, Kumar S K and Tiwari M K 2019 An efficient recommendation generation using relevant Jaccard similarity Information Sciences 483 53

[24] Pachayappan M and Panneerselvam R 2018 Materials Today: Proceedings 12285

[25] Kasihmuddin M S M, Mansor M A and Sathasivam S 2018 AIP Conf Proc. 1974 1020020

[26] Mansor M A, Sathasivam S and Kasihmuddin M S M 2018 AIP Conf Proc. 1974 1020072

[27] Mézard M, Nadal J P and Toulouse G 1986 Solvable models of working memories Journal de
physique 47(9) 1457