A NONLINEAR SYNAPTIC NEURAL NETWORK BASED ON EXCITATION AND INHIBITION

CHANG LI

Abstract. Based on the probabilistic features of synaptic excitation and inhibition, we propose and analyze a synaptic neural network. This synaptic neural network consists of synapses and neurons as well as their connections with synaptic graphs and neuronal networks. It simulates some of the computational properties of biological neurons and synapses. We define a non-linear synapse as a function of the excitatory probability of alpha-channel and the inhibitory probability of beta-channel. As a showcase, we create examples of synaptic neural networks. They're single synapse, dual synapse, and many synapse neural networks. In particular, we have concluded Boolean logic as a special case of synaptic neural networks. For practical applications, we provided and analyzed two solutions for synaptic neural networks to Content Addressable Memory (CAM) and Traveling Salesman Problem (TSP). In conclusion, synaptic neural networks can solve large-scale and difficult AI problems.

1. Introduction

The research that simulates the biological neural network has made progress. After the classic model of [McCulloch and Pitts (1946)], many neural network models have been proposed, such as the nonlinear graded-response model and the perception model of the layer network.

These neural network models abstract some computational properties of the biological neural network. At the same time, they illustrate the capability to solve hard problems such as pattern recognition, associative memory, and deep learning.

The synaptic neural network model proposed in this article is based on the properties of excitation and inhibition of the biological neurons and synapses. A synapse consists of an excitatory input, an inhibitory input, and an output while a neuron summons the inputs from synapses and other neurons to decide its activity. A synaptic neural network is a non-linear dynamical system that is connected by synapses and neurons. It may run into a stable state, an oscillating state or a chaos state. Even a simple network can demonstrate the extremely complicated computational behavior.

Date: October 25, 2017.

Key words and phrases. synaptic neural network, neural network, nonlinear, neuron, synapse, deep learning, memory, cam, optimization problem, tsp, probability, Boolean logic, excitation, inhibition.

Non-linearity, excitation, and inhibition are the fundamental behaviors of the synaptic neural network. Other characteristics are the non-determination and randomness of the synaptic neural network which can be understood by probability and topology.

2. Non-linear Synapse and Probability

Suppose a synaptic neural network contains neurons in which they are connected through non-linear synapses. A synapse consists of an input from the excitatory α-channel, an input from the inhibitory β-channel, and an output channel which sends a value to other synapses or neurons. Synapses may form a graph to accept inputs from other neurons and output to a neuron. In advance, many synaptic graphs can construct a neuron graph. Let's start to describe a synapse.

Definition 1. *Suppose S(x,y) is the non-linear function of a synapse from its output channel,*

$$S(x,y) = \alpha x(1 - \beta y) \tag{2.1}$$

where $x \in (0.0, 1.0)$ *is the input variable from excitatory* α*-channel and* $\alpha > 0.0$ *is the parameter of excitatory channel;* $y \in (0.0, 1.0)$ *is the input variable from inhibitory* β*-channel and* $\beta > 0.0$ *is the parameter of the inhibitory channel.*

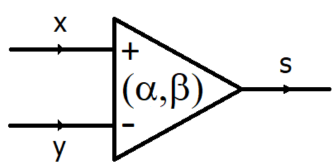

Figure 1. A Non-linear Synapse $S(x,y) = \alpha x(1 - \beta y)$

Now we are going to explain the equation (2.1) from the probability theory.

Considering the membrane of a synapse to which neurons are connected. There are many channels (holes) on the surface of the membrane of a synapse where the various kinds of ions can flow in and out. It is reasonable to assume below:

(i) The channels on the membrane are divided into two types: one is called excitatory channel (α-channel), its opening increases the activity of the neuron; another is called inhibitory channel (β-channel), its opening limits the activity of the neuron. It is called channel selectivity.

(ii) An assumed variable of the neuron (i.e. conductivity rate) is related to the number of opening channels. The more the opening excitatory channels, the bigger the variable value; the more the opening inhibitory channels, the smaller the variable value.

If the opening probability of the excitatory channels at time t is x and the opening probability of inhibitory channels at time t is y, then the opening probability of the connective synapse is $x(1-y)$. Suppose the linear relation of opening channel's probability and neuron variable (conductivity), the active probabilistic variable value of a synapse can be represented as $\alpha x(1-\beta y)$. The α is the excitatory parameter (α-channel) while the β is the inhibitory parameter (β-channel).

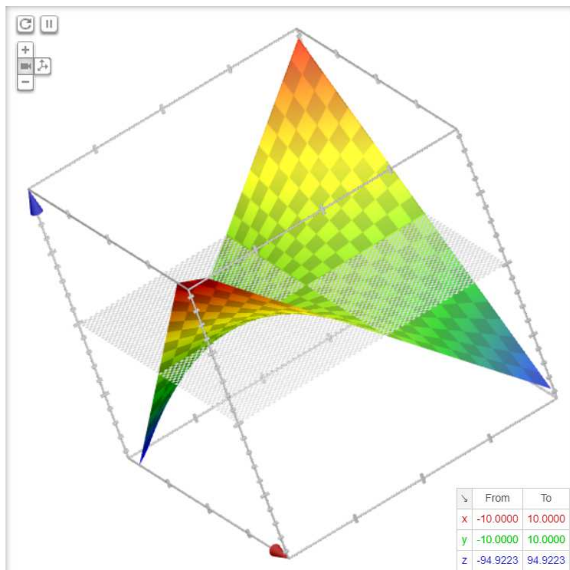

FIGURE 2. Synaptic Function $S(x,y)=x(1-y)$

The biological research has demonstrated that the structure and the change of the membrane protein is the key to the neuron activity. The change of the channel protein on neuron and synaptic membrane (i.e. potential gate control channel and chemical gate control channel shows the selectivity and threshold) can explain the interaction among neurons and synapses. [David H. Hubel, Stevens C.F., Eric R. Kandel, et,al 1985][3].

The Na+ channel illustrates the effect of an excitatory α-channel. The Na+ channel allows the Na+ ions flow in the membrane and make the conductivity increase, then produce excitatory post-synapse potential. The K+ channel illustrates the effect of an inhibitory β-channel. The K+ channel that lets the K+ ions flow out of the membrane shows the inhibition. This makes the control channel of potential gate closing and generates inhibitory post-potential of the synapse.

Other kinds of channels (i.e. Ca channel) have more complicated effects. Biological experiments showed that there were only two types of channels in a synapse while a neuron may be related to more types of channels on the membrane. Experiments illustrated that while a neuron is firing, it will generate a series of spiking where the spiking rate (frequency) reflects the strength of stimulation.

In statistical physics, the relation between probability distribution P_r and the energy E_r is $P_r = e^{-bE_r}$ where the e^{-bE_r} is called Boltzmann factor. Replacing x and y in Eq.[2.1] by the probability distribution, there is the formula $\alpha e^{-au}(1-\beta e^{-bv})$. We can assign various probability distribution as the inputs of a synapse. Therefore, a synaptic neural network can be considered as a dynamic probability network.

The stochastic attribute of channel's opening and closing shows that the opening time of special channels is random. The procedure that chemical material affects the control channel of a chemical gate is completed by a random procedure of the mixing of the tokens of the small bulbs and membrane. A large number of channels opening and closing shows the stochastic change. The probabilistic explanation of neuron and synapse activity does make sense for biological neurons and synapses.

3. Synaptic Neural Network

The features of a synaptic neural network contain highly interconnected excitatory and inhibitory synapses that form a non-linear dynamical system. It is flexible to construct a synaptic neural network with various topologies.

Definition 2. *Suppose α_{ki} is the parameter of the non-linear excitatory synapse between neuron x_k and neuron x_i while β_{kj} is the parameter of the nonlinear inhibitory synapse between neuron x_k and neuron x_j. The input value of the excitatory channel of the synapse is x_k and the input value of the inhibitory channel of the synapse is x_j, the output to neuron x_k is described below,*

$$x_k = \alpha_{ki} x_i (1 - \beta_{kj} x_j) \tag{3.1}$$

The activation function of the neuron x_k is

$$x_k = \begin{cases} \delta & x_k < \epsilon \ (notfiring) \\ 1-\delta & x_k > 1-\epsilon \ (firing) \end{cases} \tag{3.2}$$

where $\alpha_{ki}, \ \beta_{kj} \ > \ 0; \ \delta, \epsilon \in (0.0, 1.0)$

To input and output variables, the equation (3.2) is a threshold function. A vector $X = (x_1, x_2, ..., x_n)$ can represent 2^n limited points by the firing and not firing of output variables x_k. If the vector is $X_{n_i} = (x_{n_1}, x_{n_2}, ...)$ after the state transformation, then the vector X is mapped to the vector X_{n_i}. All the vectors of X that maps to the X_{n_i} constructs an element in the classification set.

It is important to know that a nonlinear relation can act as an amplifier that rapidly amplifies initial little variance to bigger in the final states. The equations of synapses are the expression of the excitation and inhibition with nonlinear effects in a synaptic neural network.

It is reasonable to assume that there are two classes of communication systems for a synaptic neural network. A fired neuron outputs a spiking plus to synapses of other neurons through axon while these synapses convert the spiking into energy and complete a continuous non-linear computing then send to neurons which decide their firing state. A synaptic neural network may have a loop connection that constructs a continuous computing until some conditions satisfied.

In our current model of the synaptic neural network, the role of a neuron acts as an oscillator to send out spiking pluses has been ignored. So a neuron simply inputs and outputs a value from an activation function.

3.1. **Single-Synapse.** The Logistic Equation [3.3] has been deeply studied and widely used in physics, ecology, and chaos. [Peityen H.O., Richter P.H. (1986)][12]

Definition 3. *The recurrence relationship of a single-synapse is defined as*

$$x^{(n+1)} = \alpha x^{(n)}(1 - \beta x^{(n)}) \tag{3.3}$$

If $0 \leq \alpha \leq 3$ the Eq.[3.3] has a single attractor, and it iterates to a stable state. If $3 < \alpha < 3.4444$ there is a limit ring, the dynamical system has 2, 4, 8, ... finite attractors. If $\alpha > 3.57$, the chaos beginning, the series x^n jumps from here to there indeterminately, and the set of attractors have the attribute of fractals. One

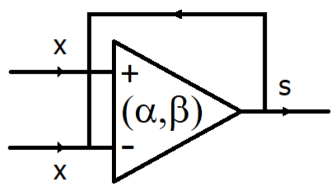

FIGURE 3. Single-Synapse in a Loop

chaos theory regards that the stretching and folding is the basic operation of chaos. [Crutchfield J.P., Farmer J.D., Packard N.H., Shaw R.S. (1986)][2]. In Eq.[3.3] the item $\alpha x^{(n)}$ represents the stretching (excitation) operation, and the item $1 - \beta x^{(n)}$ represents the folding (inhibition) operation. This implies that chaos can exist in the synaptic neural network.

3.2. **Dual Synapses.** Many types of connection for dual synapses exist. The symmetric loop connection below constructs a memory unit that can store the state of this simple synaptic network.

Definition 4. *Suppose* $\alpha_1, \alpha_2 > 1, \beta_1, \beta_2 > 0$ *and* $x^{(0)}, y^{(0)} \in (0.0, 1.0)$*,*

$$\begin{cases} x^{(n+1)} = \alpha_1 x^{(n)}(1 - \beta_1 y^{(n)}) \\ y^{(n+1)} = \alpha_2 y^{(n)}(1 - \beta_2 x^{(n)}) \end{cases} \tag{3.4}$$

since $x^{(n+1)} - y^{(n+1)} = \alpha(x^{(n)} - y^{(n)})$ if $(\alpha_1 = \alpha_2, \beta_1 = \beta_2)$ and $x^{(0)} \leq y^{(0)}$ then $x^{(n)} \leq y^{(n)}$. If $x^{(0)} = y^{(0)}$, then the Eq.[3.4] is the same as the Eq.[3.3] If $\alpha > 1, \beta = 1$ then

$$\begin{cases} x^{(0)} > y^{(0)} \to (1, 0) \\ x^{(0)} = y^{(0)} \to (0, 0) \\ x^{(0)} < y^{(0)} \to (0, 1) \end{cases} \tag{3.5}$$

In the final state, the 1 represents the $1 - \epsilon$ and the 0 represents ϵ, a very small real number. The dynamical system of Eq.[3.4] constructs an oscillator with two stable states and its initial x, y value corresponds to different stable state. If $x^{(0)}$ and $y^{(0)}$ is not equal exactly, the synaptic neural network converges to (1, 0) or (0, 1) rapidly. It completes a comparative operation and amplifies the small difference.

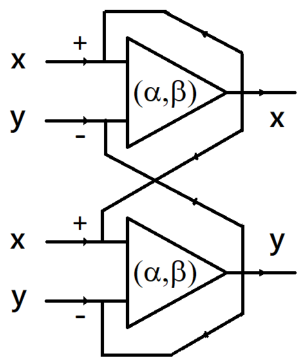

FIGURE 4. Dual Synapses as a Memory Unit

3.3. **N Synapses.** Let's check the function $S(x, y) = \alpha x(1 - \beta y)$. If α is large, the $S(x, y)$ is large. But if β is large, the $S(x, y)$ is small. In many constrain conditions the inhibitory condition is easy to express. Generally, α can be set as the same value, the different inhibition can be expressed by β. One insight into the neural network is the minimization principle which can be represented by a different value β.

Definition 5. *Suppose there are m neurons and $(m-1)m$ synapses, they can be connected as a matrix of synapses below*

$$x_i^{(n+1)} = \alpha x_i^{(n)}(1 - x_1^{(n)})(1 - x_2^{(n)})...(1 - x_m^{(n)}) \tag{3.6}$$

$i = 1..m, \alpha \geq 1.0, 0.0 < x_i < 1.0,$ *$x_1, x_2, ..., x_m$ inputs,* $\alpha_{ij} = \alpha^{1/(m-1)}, \beta_{ij} = 1.0.$

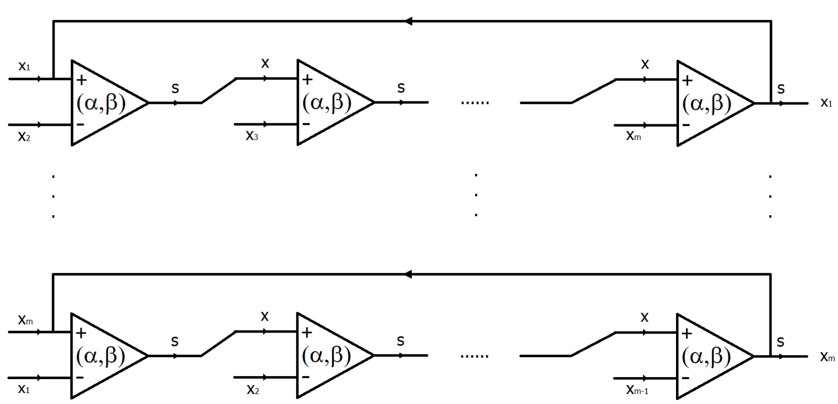

FIGURE 5. m Neurons and $(m-1)m$ Synapses.

After iteration, the neural network may have arrived in the last stable state. The x_i has a value 1 if the x_i is the maximum value. otherwise, the x_i is 0. Eq.[3.6] is a stable oscillator with m states, it generates the maximum value among m variables like a softmax function. This kind of synaptic neural network can be applied to build CAM (Content Addressable Memory) and solve Optimization Problem.

4. Boolean Logic

Selecting suitable α and β in the synaptic neural network, we can express the discrete DIFF (DIFFERENCE) in Boolean Logic. Actually when we set $\alpha = \beta = 1$, The function of the synapse becomes $x(1-y)$ which is the DIFF (DIFFERENCE) logic $x\hat{y}$. Since Boolean Logic is 1-bit binary logic with value 0 or 1, we can build a 1-bit synaptic neural network.

Definition 6. *The* Boolean Logic $DIFF, NOT, AND, OR, XOR$ *are represented by synaptic equation below,*

$$\begin{cases} DIFF: & x\hat{y} = x(1-y) \\ NOT: & \hat{x} = 1-x \\ AND: & x\&y = xy \\ OR: & x|y = 1-(1-x)(1-y) \\ XOR: & x \oplus y = 1-(1-x+xy)(1-y+xy) \end{cases} \tag{4.1}$$

In binary state, x, y have value 0 or 1.

If the values of x and y are continuous, the non-linear synapse can be computed in continuous variables. The Synaptic Logic Circuits can be implemented with these basic Boolean synapses.

5. CAM (Content Addressable Memory)

CAM is an application of neural network. A CAM synaptic neural network forms a parallel mapping from inputs to outputs. The information is stored as the weights of the connection among synapses. Memorized data is recalled by their contents. Adjust the local parameters and initial values might change the memorized content by collective computation of network. Look at the connection of CAM, each neuron connects all the other neurons by direct inhibition and it also forms a loop to itself.

Definition 7. *The rule of transformation is*

$$x_i = r d_{ii} x_i (1 - d_{ij} x_j) \tag{5.1}$$

x_i is the output of a neuron or a synapse, the matrix d_{ij} is the link strength between neuron i and j. d_{ii} is the excitatory parameter while d_{ij} is the inhibitory parameter and r is the excitatory constant.

Definition 8. *Selecting a matrix d_{ij} below, we can construct a CAM.*

$$d_{ij} = \begin{cases} \sum(1 - V_i * V_j) & i \neq j \\ \sum V_i * V_j & i = j \end{cases} V_i = \begin{cases} \epsilon & x_i = 0 \\ 1-\epsilon & x_i = 1 \end{cases} \epsilon > 0 \tag{5.2}$$

Suppose U is the set of memorized word, the length of vector $X = (x_1, ..., x_N)$ is N, the number of memorized words in U is n. x_i is equal to 0 or 1, the binary words of X (i.e. 1000101 ...) represents the memorized information in the synaptic neural network. ϵ is a very small positive real number which represents 0 of a continuous variable and $1 - \epsilon$ represents 1. If the bit i of the memorized word is 1 then its $V_i = 1 - \epsilon$, otherwise $V_i = \epsilon$.

Since $d_{ij} = d_{ji}$, the matrix is a symmetric Hermit matrix which means that x_i and x_j have the same inhibitory strength. This selection of memorized matrix is similar to other CAM models [Hopfield J.J. 1982][7]. Hermit matrix shows the stability of the neural network as a dynamical system.

Initial input of x_i is a continuous variable, and the word of 0 and 1 is the last stable state. If the corresponding input state is equal to the output state, then the neural network memorizes the word, otherwise, it corresponds to another stable state. This procedure is called recall of memory.

The vector length N=3, 50 and the memorized word length n=1,2,3 have been selected for the numerical experiments. The result is summarized as below:

(i) *Constant State.* 000 ... is a special state of the synaptic neural network and is always memorized in the synaptic neural network.

(ii) *Complementary Memory.* The complementary state of a memorized state is always a correct answer. That is if 100 is memorized then 011 is memorized as well.

(iii) *Repeat Memory.* Memory a lot of data information repeatedly does not change the initial memory result.

(iv) *Overload.* To N=3, n=1,2, the r can be chosen that make all memory information response correctly. When $r \geq 3.0$, no r can make all the response correctly. For example, memory 100, 011 and 010, the 010 cannot response correctly. It shows the number of content correlation is going to affect the correctness of memory.

(v) *Excitatory Parameter.* The numerical experiments showed that the value of r plays an important role on result. If r is large, the network responses the memorized state with maximum strength, not all states can response correctly, that is the sensibility is low and it cannot find the little difference. If r is too small, the state without memorized vector can also be responded to correctly, that is the sensibility is too high.

When N=3, n=1,2, there exists r that all the memorized data responses correctly. If $n \geq 3$, there are some errors. Because there is 2^N information can be memorized, the maximum correct rate of all is 25%. When N is large the maximum correct rate cannot be more than 25

In particular, N=50, n=6, and floating point real value $ZERO = 0.00001$, real value $ONE = 0.999$, and the special memorized sequence is 0...00001, 0...00110, 0...01010, 0...10001, 0...10011, 0...11010. The experiment shows that the recalling strength which corresponds to the memorized value is $(5, 3, 4, 6, 1, 2)$. The smaller the value, the bigger the strength to different r.

Suppose the correct rate = (correct response of memorized data number)/(memorized data number). If $r \in [1.1, 1.5]$, all stored data can be recalled correctly, however, some not stored data can also be recalled correctly, the correct rate is 58%. If $r \in [1.8, 2.0]$, the stored data cannot be recalled completely, but the correct rate is 80%. If $r = 1.6749222$ then the maximum correct rate is archived at 87.5%, no r can be selected over that rate.

Remark 1. *The rule of CAM model is similar to the Hebbian rule. The capability of associative memory is in the correlation of the nonlinear synapses. Memory and learning are related to the conductivity of neurons and synapses. A pair of neurons in the different state will decrease the connective strength while a pair of neurons in the same state will increase the connective strength.*

In the CAM model, associative strength matrix is decided by inhibitory parameters. Because the parameters are decided by the number of 0 and 1 in the sequence, different place of the sequence is equal to the value of the excitatory parameter by 0 or 1.

Remark 2. *Another model of CAM is constructed by layer network that is decided by the content of the memory. Like RNN, to a memorized sequence, the different value of bits is connected by inhibition channels, all inhibitory parameters are 1 while excitatory parameters are r. Second data is memorized by the output of the first layer as the input of the second layer, and so on. Finally, the last layer outputs value to the first layer and forms the memorized network by feedback.*

Although the experiment shows the capability of associative memory, it is absent flexibility and regularity in somewhat. The memory implemented by the topological connection can be applied to inference after the training of synaptic neural network.

Remark 3. *In the previous CAM model, the parameter matrix is set in fixed. However, we can change the strength according to the inputs of neurons by Hebbian learning rule such that* $\partial d_{ij} = -\partial V_i \partial V_j$. *Decreasing inhibitory connection strength* d_{ij} *of different states, and increasing* d_{ij} *of the same state, that will increase the reliability of memorized state. Since* $d_{ij} = d_{ji}$, *two connected neurons have the same inhibitory parameter. Besides, the initial states of* x_i *can be set in random values.*

6. Optimization Problem

An optimization problem is a problem that searches for the maximum or minimum under some constraint conditions. These problems are often in combinatorial complexity. In the real world, many optimization problems face the fact that to find the best solution the exponential time has to be taken. Therefore the optimization problems can be converted to find 'good' solutions in polynomial time.

'Traveling Salesman Problem' (TSP) is an NP-complete problem that its features and algorithms have been widely analyzed. [Garey M.R., Johnson D.S. (1979)[5], Karp R.M. (1986))[10]]. Meanwhile, it is one of the hardest problems in the NP-complete problems. Hopfield and Tank has presented a method that produces approximate solution with an analog neural network, and showed the capability to solve this problem rapidly. ([Hopfield J.J., Tank, D.W. (1985)][9]. Below I am going to discuss how to solve the TSP by a synaptic neural network.

Definition 9. *Suppose there are n cities A, B, .., AB, AC, ..., the distance from A to B is $d_{A,B}$, the Traveling Salesman Problem is to search a route that the salesman visits each city once and only once, returns to the beginning city, and requires the sum of the distance is minimum.*

$$x_{ij} = r^{4n-2} x_{ij} \prod_{k \neq i}^{n} (1 - x_{ik}) \prod_{k \neq j}^{n} (1 - x_{kj}) \prod_{k=1}^{n} (1 - d_{k,i} x_{kj\ominus 1}) \prod_{k=1}^{n} (1 - d_{k,i} x_{kj\oplus 1}) \tag{6.1}$$

$$1 \leq i, j, k \leq n, 0 < r, 0 < d_{ij} \leq 1, d_{ij} = d_{ji} \tag{6.2}$$

$$j \ominus 1 = \begin{cases} j-1 & j > 1 \\ n & j \leq 1 \end{cases} j \oplus 1 = \begin{cases} j+1 & j < n \\ 1 & j \geq 1 \end{cases} \tag{6.3}$$

To exactly represent the problem, the matrix of n^2 is used. The row represents the city name, the column represents the order in the cities are visited. The value 1 of the item at (x, i) represents that the city x is visited at time i, otherwise, value 0 means not visited at time i. The condition is translated to that there are only 1 in each rows and columns. The total visited distance is $d_{A,X_1} + d_{X_1,X_2} + ... + d_{X_n,A}$ where X_i is the city that its ith column is 1 and A is the beginning city.

To each neuron, the inhibitory connection of each row and column units keeps only one neuron excitation in each row and column. First two items of product in Eq.[6.1] makes these conditions possible. All the inhibitory parameters of the synapses are set to 1.

The distance of path from city X to next city Y is $d_{X,Y}$. To city X, the distance of the next possible cities is an inhibitory parameter, and it inhibits the connection of X, so the shortest path has the largest probability to be selected. This method is similar to the neighbor heuristic algorithm and will increase the smallest inhibitory items, that is the shortest distance. The city which connected before has the same connected mode. All the neurons are connected so that the first column and the last column is connected in a loop. Last two items of product in Eq.[6.1] implemented the heuristic algorithm.

An example of $n = 10$ is selected to test from [Floulds I.R. (1984)][4]. It is interesting to find that the results changed by various r value so that there are the best solutions, good solutions, and oscillation. Even if a little bit change of r was made, it is possible to generate different distances and visiting sequence.

Although this TSP synaptic neural network does not always generate the best solution, it is similar to a heuristic algorithm to get good by selecting value r. Every synapse has to do $6n-2$ multiplications and $4n-2$ subtractions, and the total of synapses is $2n^2$, therefore each iteration time is $O(n^3)$. Suppose the converge time is $O(n^2)$, the total computing time is in $O(n^5)$.

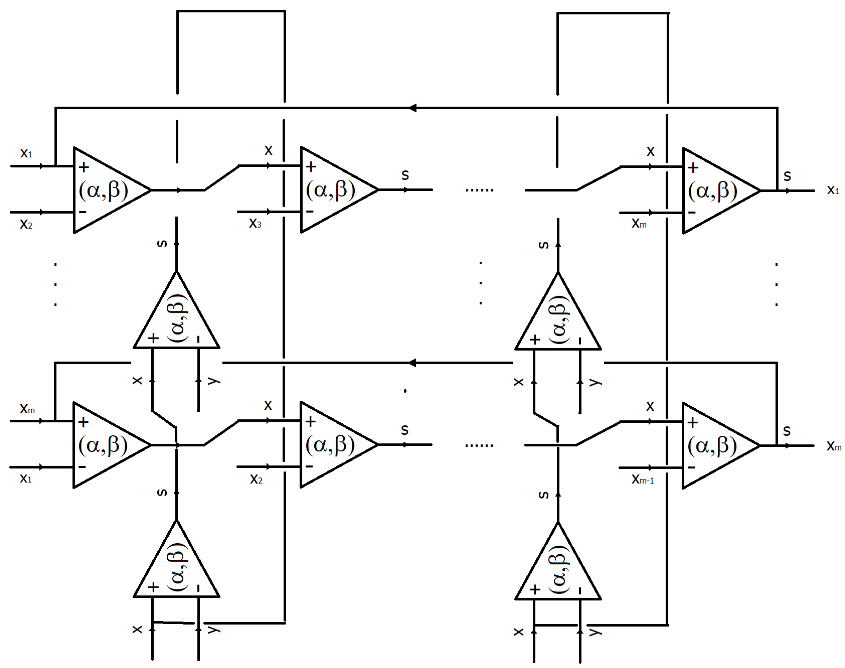

FIGURE 6. Synaptic Neural Network for TSP (Partial Graphs)

There are some interesting features of this synaptic neural network for the optimization problem. Firstly, the algorithm of the synaptic neural network is non-deterministic according to the value r. The assumed r can be selected and achieved to the best solution in polynomial time. If $NP \neq P$ it does not exist an algorithm for all instances so that the rate of the solution with the best solution $RA < \infty$ [Garey M.R., Johnson D.S. (1979)][5]. That is we cannot find a TSP algorithm so that its every solution is very near the best solution. But the research has proved that the non-deterministic algorithms have the capabilities to check an exponential number of possibilities in polynomial time.

Secondly, the synaptic neural network is parallel in computing but it can be in synchronization or non-synchronization. Here we approached the computing mode in synchronization.

Finally, the nonlinear relation has the feature of exponential computing. Investigating the change of each synapse and neuron, we find that from beginning although the difference is very small, the difference is amplified, at a stage of processing, the change dramatically arrived to the stable state or in oscillation. This non-linear computing capability of the continuous variable is the key to solve the problem in polynomial time.

7. Conclusion

A non-linear synaptic neural network based on excitation and inhibition has been analyzed and simulated. In particular, Boolean Logic was concluded as a special synaptic neural network. Two applications on both memory (CAM) and optimization problem (TSP) were demonstrated. Like a transistor, a synapse acts as a basic non-linear computing unit to construct the synaptic neural network. We also illustrate that a synaptic graph itself can complete many complex computing tasks. Moreover, the synaptic neural network provides a simple prototype for research of brain with a complex topological connection.

Multi-core CPU, Multi-thread GPU, Multi-cell FPGA are currently suitable computing devices for the implementation of the synaptic neural network. The framework, toolkit, and languages for the synaptic neural network can be effective components and tools to build the ecosystem of the synaptic neural network for AI applications.

In the future, we can apply the Synaptic Neural Network (SNN) on Deep Learning (DL), Recurrent Neural Network (RNN)and Convolutional Neural Network (CNN) construction for new generation AI applications. The implementation of the Synaptic Neural Network (SNN) by semiconductor, optical circuit, and/or chemistry reactor can solve many computing problems in ultra fast speed with big data and low power consumption.

References

[1] Barlow H.B., Levick W.R. (1965)*The mechanism of directionally selective units in rabbit's retina*, J.Physiol., Lond.-UK., 178, 477-504, 1965
[2] Crutchfield J.P., Farmer J.D., Packard N.H., Shaw R.S. (1986)*Chaos*, Scientific American, Vol.255 No.6 Dec. 1986
[3] David H. Hubel, Eric R. Kandel, Walle J.H. Nauta and Michael Feirtag, David H. Hubel and Torsten N. Wiesel, Edward V. Evarts, Norman Geschwind, F.H.C. Crick etc. (1979) *BRAIN*, Scientific American. 241, 1
[4] Floulds I.R. (1984) *Combinatorial Optimization for Undergraduates* Springer-Verlag
[5] Garey M.R., Johnson D.S. (1979) *Computers and Intractability* New York: W.H. Freeman
[6] Hinton G.E., Anderson J.A. edition (1981) *Parallel Models of Associative Memory* (Erlbaum, Hillsadale, NJ, 1981)
[7] Hopfield J.J. (1982) *Neural networks and physical systems with emergent collective computational abilities* Proc.Natl.Acad.Sci. USA 79,2554-2558
[8] Hopfield J.J. (1984) *Neurons with graded response have collective computational properties like those of two-state neurons* Proc.Natl.Acad.Sci. USA 81, 3088-3092
[9] Hopfield J.J., Tank, D.W. (1985) *Neural Computation of Decisions in Optimization Problems* Biol.Cybern.52, 141-152
[10] Karp R.M. (1986) *1985 ACM A.M.TURING AWARD RECIPIENT* Comm. ACM Vol.29, Num 2, 2
[11] Marr D. (1982) *Vision* Freeman, San Francisco, CA.
[12] Peityen H.-O., Richter P.H. (1986) *The Beauty of Fractals* Springe-Verlag, Berlin Heidelberg
[13] Poggio T., Torre V., Koch C. (1985) *Computational vision and regularization theory* Nature Vol.317 26 SEPTEMBER 1985.
[14] Brigoginew I. Stengers I. (1984) *Order out of chaos* Bantam Books, Inc.
[15] Sejnowski T.J., Koch C. and Churchland P.S. (1988) *Computational Neuroscience* Science 241, 1299
[16] Torre V. Poggio T. (1978) *A synaptic mechanism possibly underlying directional selectivity to motion* Proc.R.Soc.Lond.B.202, 409-416

(2900 Warden Ave., Bridlewood Mall 92012, Toronto, Ontario M1W 3Y8, Canada)
Email address, Chang LI: changli@neatware.com
URL: https://www.researchgate.net/profile/Chang_Li59/contributions