Journal of Applied Sciences

ISSN 1812-5654

science alert

ANSInet
an open access publisher
http://ansinet.com

Journal of Applied Sciences 14 (4): 317-324, 2014
ISSN 1812-5654 / DOI: 10.3923/jas.2014.317.324
© 2014 Asian Network for Scientific Information

Time Window, Spike Time and Threshold Boundary for Spiking Neural Network Applications

Abdullah H. Almasri and Shahnorbanun Sahran
Center for Artificial Intelligence, Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, Bangi, Darul Ehsan, 43600, Selangor Malaysia

Abstract: Assigning threshold value plays an important role in the temporal coding Spiking Neural Network (SNN) as it determines when the neuron should fire, the time window parameter plays a significant role in the SNN performance. This study does two things: First it proposes a mathematical method to find out the threshold boundary in the temporal coding SNN models and second it outlines the input time window boundary which leads to specify the spike time boundary. The latter was used at the former. The threshold boundary method was applied to two learning algorithms i.e., Spiking-Learning Vector Quantization (S_LVQ) and Self-Organizing Weight Adaption for SNN (SOWA_SNN), for both classification and clustering pattern recognition applications, respectively. This method finds the threshold boundary mathematically in both learning models above and observes that the minimum and maximum value of the threshold does not depend on the time input window, time coding or delay parameters in SNN. With regard to the input time window, it finds that specification beyond the parameter boundary affects the computational network cost and performance; also it finds that the delay and the time coding parameters play a significant role in assigning the time window boundary.

Key words: Spiking neural network, threshold, time window, spike time, classification, clustering

INTRODUCTION

A special class of Artificial Neural Networks (ANN) is the third generation called Spiking Neural Networks (SNN), where neuron models communicate by sending and receiving action potentials ("spike trains"). During the last couple of years, investigational proof has accumulated. This concept proves that most biological neural systems encode data with the use of spikes (Hopfield, 1995). These experimental results from neurobiology have led to the study of spiking neural networks in more detail, which employ spiking neurons as computational units (Maass, 1997). Due to this property whenever a fast and efficient computation is required a SNN is principally a suitable to do so (e.g., speech recognition) where the timing of the input signals and the firing signal carries important information. In terms of computation, spiking neural networks have more power than both sigmoidal gates and perceptrons (Maass, 1997). The main objective of comprehending the capabilities and restrictions of this new type of spiking neural network provides additional information for theoretical investigation of the third generation of neural network models (Maass, 1997). The spiking neuron's mathematical models do not provide a full explanation of the enormously complex computational function of a biological neuron. The computational units of the previous two generations of neural network models are simplified models and focus on only a few concepts of biological neurons. However, in comparison with the previous two models, they are substantially more realistic. Specifically, the mathematical models express much better the actual output of a biological neuron and so they allow for an investigation on a theoretical level the potential of using time as a resource for a computation and communication (Maass, 1997). The importance of threshold value for learning in SNN: In the simplest (deterministic) model of a spiking neuron one assumes that a neuron, (v) fires whenever its potential (p) reaches a certain threshold (θ). This potential (p) is the sum of the so-called EPSP (excitatory postsynaptic potentials) and IPSP (inhibitory postsynaptic potentials), which result from the firing of other neurons (u) that are connected through a synapse to neuron (v) (Maass, 1997). Furthermore, it has been assumed that fast changes of the value of w(t) are also necessary for computations in

Corresponding Author: Abdullah H. Almasri, Center for Artificial Intelligence,
Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Darul Ehsan,
43600, Selangor Malaysia

biological neural systems (Maass, 1997). The existing literature on spiking neural network computations is related results of neurobiology (Maass, 1997). The theoretical investigation of spiking neural networks is not a new research field. In fact, it has a long tradition in theoretical neurobiology, biophysics and theoretical physics. On the other hand, a mathematically extensive analysis of the computational power of spiking neural networks has not been fully investigated (Maass, 1997). Maass (1997) claims that such an analysis will be useful in understanding the computational power in complex biological neural systems. SNNs have turned out to be very powerful (Maass, 1997) but there is still not much known about possible learning and higher computational mechanisms (Natschlager and Ruf, 1998).

Pham *et al.* (2008) claimed that in order to keep relevant neurons active, a low threshold value was assigned initially and increased after each training epoch in small equal steps to a preset value. Initial threshold values were set to 60*0.5*0.7 and increased up to 60*0.5*0.83. Here 60 is the number of input neurons and 0.5 is the average connection weight. Pham *et al.* (2007) set the threshold value to 60*0.5*0.5, here 60 is the number of input neurons and 0.5 is the average connection weight for (SOWA_SNN). Pham and Sahran (2006) the threshold θ is a constant and is equal for all neurons in the network (Shahnorbanun *et al.*, 2010). The threshold θ is a constant and is equal for all neurons in the network.

Many SNN learning algorithms have been proposed for supervised learning (Bohte *et al.*, 2002a; Xin and Embrechts, 2001; Pham and Sahran, 2006; Ruf and Schmitt, 1997; Sporea and Gruning, 2012) and clustering (Bohte *et al.*, 2002b; Natschlager and Ruf, 1998; Pham *et al.*, 2007), however as the author concern, none of them has mentioned clearly the guidelines for selecting threshold value. They used to select it empirically to give the best result. Threshold plays an important role in SNN learning as it determines when the neuron should fire and the input time window parameter plays a significant role in the SNN performance. Here the question raised is; can one set the right and suitable threshold parameter mathematically depending on other parameters in SNN. This study addresses this question and findings showed that indeed one can find mathematically the threshold boundary where the threshold cannot go beyond the boundary. The suitable threshold must be found within this range and based on this study the average value is recommended as it's almost the same as proposed by Pham *et al.* (2007) and Pham and Sahran (2006).

This study presents two main concepts. The first concept is to find out the threshold boundary and select a suitable threshold value within this range, using a mathematical model. The threshold depends on the relation between the spike time and its boundary. The second concept is to outline the input time window boundary. The range of the input time window is required to specify the spike time boundary. The spike time boundary is, in turn, required to find the threshold boundary. The different resulted issues are discussed when selecting the input time window range, with regard to the computational cost. The latter was used at the former. Two SNN learning algorithm S_LVQ (for classification application) and SOWA_SNN (for clustering application) have been selected to apply the method to find out the threshold boundary.

How may the correct threshold value for the SNN be assigned? What may the threshold value actually be? From the threshold boundary the suitable threshold value may be assigned from that range. Further studies on selection of the correct threshold needs to be done in the future.

This work studies two models for learning in temporal coding SNNs to find out the threshold boundary. The first one has been applied for classification application and the second one for clustering application. The main purpose is to find the threshold boundary as in Eq. 1:

$$\theta \in [\theta_{min}, \theta_{max}] \qquad (1)$$

MATERIALS AND METHODS

Time window parameter boundary: The time window (tw) plays a significant role in the SNN performance (Pham *et al.*, 2007). Usually the time window boundary is assigned experimentally $tw_{exp} \in [tw_{min.exp}, tw_{max.exp}]$ to get the best result with no guidelines on how to select it and what the effects by selecting it on the computation cost of SNN learning.

The issue here is to assign the right value for the (tw) parameter depending on the other parameters at the pre-processing stage. It is suggested to observe the value for (tw) from the Spike Time parameter (st) which passed to the spike response function $\varepsilon(st)$. The Spike Response Function (SRF) is basically a generalized leaky-integrate-and-fire model. This describes the biophysical mechanisms of the neuron mainly by means of its membrane potential. In addition, this model gives much importance to the time lap taken from the last firing event. The model describes the state of a neuron j at time t by the sate variable $u_j(t)$ (Maass and Bishop, 2001). So the spike time is assigned as appear in Eq. 2:

$$\begin{gathered} st = tw-(input+delay);\ (st \geq 0) \Rightarrow \\ tw-(input+delay) \geq 0 \Rightarrow tw \geq (input+delay) \end{gathered} \qquad (2)$$

The following Eq. 3 and 4 depend on Eq. 2 to specify the boundary for tw∈[tw_{min}, tw_{max}] where, input∈[tc_{min}, tc_{max}] and Delay∈[d_{min}, d_{max}]. where, tw_{min} and tw_{max} refereed to the minimum and maximum value of time window respectively which specified here, d_{min} and d_{max} refereed to the minimum and maximum value of delay, tc_{min} and tc_{max} refereed to the minimum and maximum value of temporal coding, respectively:

$$tw_{min} = tc_{min}+d_{min} \tag{3}$$

$$tw_{max} = tc_{max}+d_{max} \tag{4}$$

Assigning the value for tw plays a significant role in the SNN performance, experimental assigning will cause the SNN to face different problems:

- If $tw_{min.exp}$<tw_{min} then any neuron at the hidden or output layer will not allow to be activated at the range ($tw_{min.exp}$, tw_{min}) as (st<0) for all neurons within this range and that will affect the SNN performance. This increases the network computational cost without any benefit
- If $tw_{min.exp}$>tw_{min} then some of the neurons at the hidden or output layers will not get the chance to be activated within the range (tw_{min}, $tw_{min.exp}$) as (st) could be greater than zero within this range for some neurons and that will affect the SNN performance
- If $tw_{max.exp}$>tw_{max} then all the neurons at the hidden or output layers will always be activated within the range (tw_{max}, $tw_{max.exp}$) as (st>0) for all the neurons within this range and that will increase the computational cost of SNN learning
- If $tw_{max.exp}$<tw_{max} then some of the neurons at the hidden or output layers will not get the chance to be activated within the range ($tw_{max.exp}$, tw_{max}) as (st) could be greater than zero within this range and that will affect the SNN performance

By outlining the input time window boundary, the number of free parameters which used to be assigned experimentally decreased by one, as the time window would be assigned depends on the delay and the time coding parameters which assigned by the experimenter at the preprocessing stage.

Spike time boundary: Outlining the time window shows that the (tw) parameter should be in the range (tw_{min}, tw_{max}) as outlined in Eq. 3 and 4. The (tw) boundary depends on the temporal coding and delay parameters and from those functions the spike time st∈[st_{min}, st_{max}] is defined on a closed interval and could be specified depending on Eq. 2 as follows:

$$st_{min} = tw_{min}-tw_{min} = 0 \tag{5}$$

$$st_{max} = tw_{max}-tw_{min} \tag{6}$$

The Eq. 5 and 6 define the st_{min} and st_{max}, respectively; this will help to find out the threshold boundary as (st) is defined on closed interval; as will come in the rest of this study.

Threshold boundary for classification application (S_LVQ): The network architecture for spiking learning vector quantization (S_LVQ) (Pham and Sahran, 2006) consists of three layers (Input, Hidden and Output), is a feed-forward network, is fully connected between input layer and hidden layer, has multiple delayed synaptic terminals and is partially connected between hidden and output layer as shown in Fig. 1. The connection is characterized by weight and delay value. The details of this network, which have been used by (Pham and Sahran, 2006) for control chart datasets, are as follows: N_{input} = 60, N_{hidden} = 24, N_{output} = 6 and N_{sub} = 16 where N_{input} refers to the number of input neurons, N_{hidden} refers to the number of hidden neurons, N_{output} refers to the number of the output neurons and N_{sub} refers to the number of the sub connection.

At the hidden layer each neuron receives (N_{input}) input from the input layer and consists of (N_{sub}) as each synapse between input layer and hidden layer consists of (N_{sub}) connections. This means each neuron at the hidden layer receives (N_{input}*N_{sub}) input at a time. The synapse potential for each sub-connection could be calculated by Eq. 7:

$$\rho_{sub} = \varepsilon(st_i) * wght_i; \quad \begin{cases} st = tw - (\text{Input} + \text{Delay}); (st \geq 0) \\ \rho_{sub}: \text{Synapse Potential For Each SubConnection} \end{cases} \tag{7}$$

At the hidden layer the minimum or maximum value any neuron could have is the minimum or maximum threshold value that could be assigned to the S_LVQ, respectively; the Eq. 8 and 9 define the θ_{min} and θ_{max}, respectively:

$$\theta_{min} = N_{input} * \underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Min}}} \left(\sum_{i=1}^{N_{sub}} \varepsilon(st_i) \right) * wght_{min} \tag{8}$$

$$\theta_{max} = N_{input} * \underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Max}}} \left(\sum_{i=1}^{N_{sub}} \varepsilon(st_i) \right) * wght_{max} \tag{9}$$

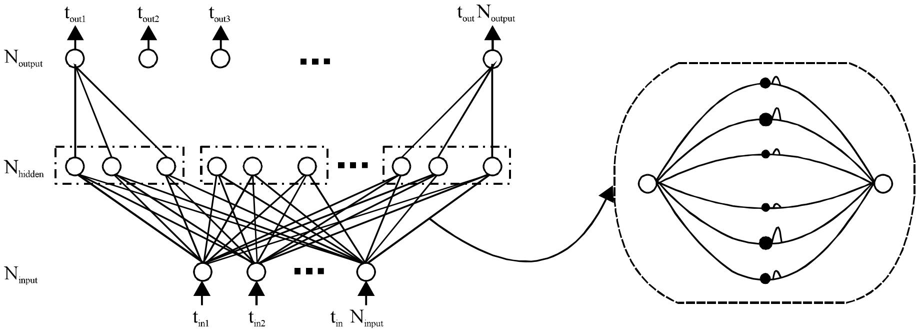

Fig. 1: Spiking learning vector quantization (S_LVQ) architecture. Redrawn from Pham and Sahran, (2006)

```
[Set $N_{sub}$; Spike Response Function $\varepsilon(st)$
[SpikeTimeMax=stm; DelayInterval=di
for i=1 to i<$N_{sub}$
  for j=1to($N_{sub}$-i)
    If((stm-j.di)>0)
      calculate sum of $\varepsilon(st_i)$
    end if
  end for
  for k=1to(i-1)
    if((stm+k.di)<twindow)
      calculate sum of $\varepsilon(st_i)$
    end if
  end for
end for
Find Min & Max $\varepsilon(st_i)$
```

Fig. 2: Pseudo code for calculating the maximum and minimum ε(st) for a hidden neuron received from one input in spiking learning vector quantization (S_LVQ)

The value of:

$$\underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Min}}}\left(\sum_{i=1}^{N_{sub}}\varepsilon(st_i)\right)$$

and:

$$\underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Max}}}\left(\sum_{i=1}^{N_{sub}}\varepsilon(st_i)\right)$$

is calculated by applying the following pseudo code as shown in Fig. 2.

The value of N_{input}, N_{sub} and the weight boundary [$wght_{min}$, $wght_{max}$] are known. The boundary of ε(st) needs to be found. The absolute minimum and maximum value for the ε(st) function could always be found mathematically; as the Spike Time (st) parameter is defined on a closed interval as showed in Eq. 5 and 6. The (st) is defined by Eq. 10:

$$st = tw - (Input + Delay);\left[\begin{array}{c} Input\in(0,100) \text{ as taken in S\_LVQ} \\ Delay\in(10,160) \text{ as taken in S\_LVQ} \\ tw\in(10,260) \text{ as outlined (Eq.3\&4)} \\ st\in(0,250) \text{ as specified (Eq.5\&6)} \end{array}\right. \tag{10}$$

The spike response function ε(st) function which has been proposed by Pham and Sahran (2006) is described by Eq. 11 where a is time constant membrane and b is time constant for synapse and is graphically described as in Fig. 3a.

$$\varepsilon(st)=\left(\frac{1}{1-\frac{a}{b}}\right)\left(e^{-\frac{1+st}{b}}-e^{-\frac{1+st}{a}}\right);\left(\left(c=\frac{1}{1-\frac{a}{b}}\right),a\neq b\right) \tag{11}$$

In this model st∊(0, 250), i.e., the function is defined on a closed interval and the absolute minimum and maximum value could be derived within this interval as in the following steps:

Step 1: Compute ε(st)':

$$\varepsilon(st)'=c.\left[-\frac{1}{b}e^{-\frac{1+st}{b}}+\frac{1}{a}e^{-\frac{1+st}{a}}\right] \tag{12}$$

Step 2: Find the critical points of ε(st) in (0, 250) as follows:

$$\varepsilon(st)'=0\rightarrow st=(a.b).\left(\frac{\ln(a)-\ln(b)}{a-b}\right)-1 \tag{13}$$

The value of (st) will be the same whether a>b or a<b.The absolute minimum and maximum of ε(st) does not depend on the value of time input window(tw), time coding (Input) or delay; thus θ_{min} and θ_{max} doesn't depend on them too. Its minimum and maximum value depends only on the value of a time constant for membrane (a) and time constant for synapse (b).

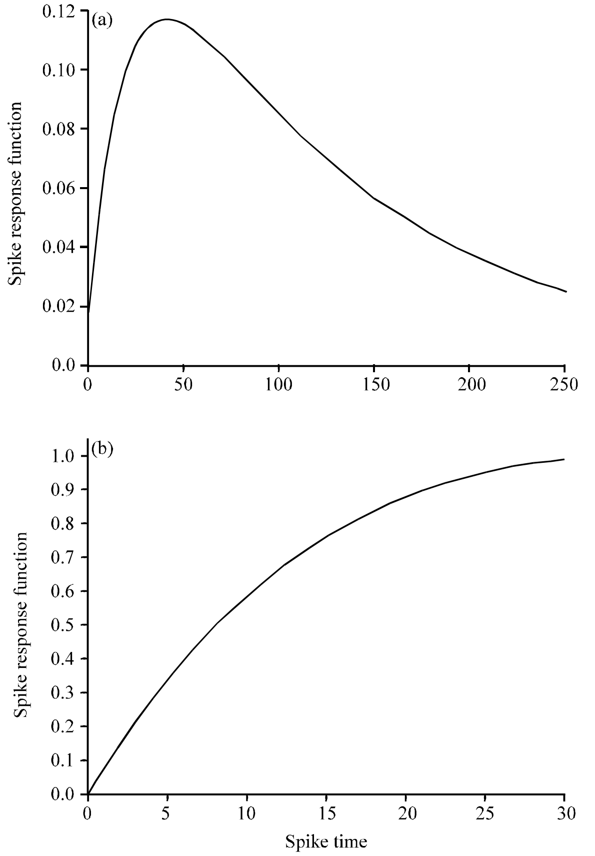

Fig. 3(a-b): Spike response function (a) For spiking learning vector quantization (S_LVQ). As proposed in Pham and Sahran (2006) and (b) For self-organizing weight adaptation spiking neural network (SOWA_SNN). As taken in Pham *et al.* (2007)

Step 3: Calculate the value of ε(st) at the critical point; where (a = 120, b = 20) as it has been taken by Pham and Sahran (2006)

At st = 42→ε(42) = 0.116.

Step 4: The absolute minimum and maximum value is (0.008) and (0.116), respectively on (0, 250) and the boundary for ε(st) is as in Eq. 14:

$$\varepsilon\epsilon[\varepsilon_{min}, \varepsilon_{max}] = [0.008, 0.116] \quad (14)$$

Step 5: By applying the Pseudo Code in Fig. 2 using MATLAB to calculate the minimum and maximum ε(st) for a hidden neuron received from one input in S_LVQ and then find out the θ_{min} and θ_{max} as in Eq. 15 and 16, respectively:

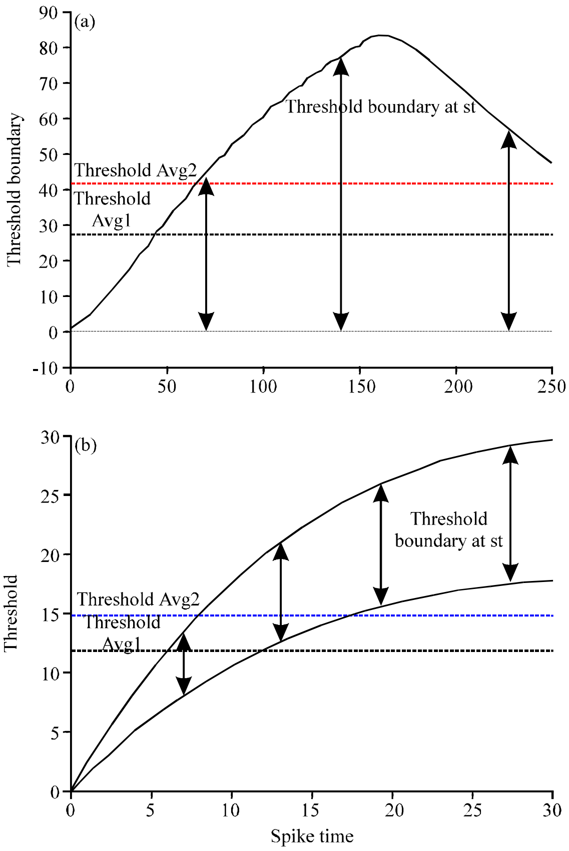

Fig. 4(a-b): The relation between threshold boundary and the spike time (a) In spike learning vector quantization (S_LVQ) and (b) In self-organizing weight adaptation spiking neural network (SOWA_SNN)

$$\mathrm{Max}_{j=1,k=1}^{N_{output},N_{input}}\left(\sum_{i=1}^{N_{sub}}\varepsilon(st_i)\right)=1.3876\Rightarrow\theta_{max}=60*1.3876*1=83.2583 \quad (15)$$

$$\mathrm{Min}_{j=1,k=1}^{N_{output},N_{input}}\left(\sum_{i=1}^{N_{sub}}\varepsilon(st_i)\right)=0.4297\Rightarrow\theta_{min}=60*0.4297*0=0 \quad (16)$$

Here the threshold boundary in S_LVQ is θϵ[0, 83.2583]. Figure 4a shows the relation between the threshold boundary and the spike time.

Hence, the initial and suitable threshold value will be within this range. Eq. 17 and 19 are the equations suggested to calculate the suitable threshold:

$$\theta_{avg1}=N_{input}*\frac{\mathrm{Max}\sum_{i=1}^{N_{sub}}\varepsilon(st_i)+\mathrm{Min}\sum_{i=1}^{N_{sub}}\varepsilon(st_i)}{2}*wght_{avg}=27.2595 \quad (17)$$

Where:

$$\text{wght}_{\text{avg}} = \frac{\text{wght}_{\text{max}} + \text{wght}_{\text{min}}}{2} \tag{18}$$

Or:

$$\theta_{\text{avg2}} = \frac{\theta_{\text{min}} + \theta_{\text{max}}}{2} = 41.62915 \tag{19}$$

Compared the θ value which has been used by Pham and Sahran (2006) firstly with the θ_{avg1}, both are very closed and that supports that equation is the suitable formula to get the suitable threshold in this case and secondly with the θ_{avg2} which its value is far away from the one the (Pham and Sahran, 2006) used.

Threshold boundary for clustering application (SOWA_SNN): The network architecture for self-organizing weight adaptation spiking neural network (SOWA_SNN) (Pham *et al*., 2007) consists of two layers (Input and Output) and is fully connected between input layer and output layer as shown in Fig. 5. The details of this network, which have been used by Pham *et al*. (2007), are as follows: $N_{\text{input}} = 60$ and $N_{\text{output}} = 64$, where N_{input} refers to the number of input neurons and N_{output} refers to the number of the output neurons.

At the output layer each neuron receives (N_{input}) from the input layer. This means each neuron at the output layer receives (N_{input}) input at a time. The synapse potential for each connection could be calculated by Eq. 20:

$$\rho_i = \varepsilon(\text{st}_i) * \text{wght}_i; \begin{cases} \text{st} = \text{tw} - (\text{Input} + \text{Delay}); (\text{st} \geq 0) \\ \rho_i\text{: Synapse Potential For Each Connection} \end{cases} \tag{20}$$

At the output layer the minimum and maximum value any neuron could have is the minimum and maximum threshold the SNN could be assigned respectively. The Eq. 21 and 22 define the θ_{min} and θ_{max} consequently:

$$\theta_{\text{min}} = N_{\text{input}} * \underset{j=1,k=1}{\overset{N_{\text{output}},N_{\text{input}}}{\text{Min}}} (\varepsilon(\text{st}_i)) * \text{wght}_{\text{min}} \tag{21}$$

$$\theta_{\text{max}} = N_{\text{input}} * \underset{j=1,k=1}{\overset{N_{\text{output}},N_{\text{input}}}{\text{Max}}} (\varepsilon(\text{st}_i)) * \text{wght}_{\text{max}} \tag{22}$$

The value of N_{input} and the weight boundary [wght_{min}, wght_{max}] are known. The boundary of ε(st) needs to be found, the absolute minimum and maximum value for the ε(st) function could always be found mathematically; as the Spike Time (st) parameter is defined on a closed interval as shown in Eq. 5 and 6. The (st) is equal to the time constant for excitatory spike response function as defined in Eq. 23:

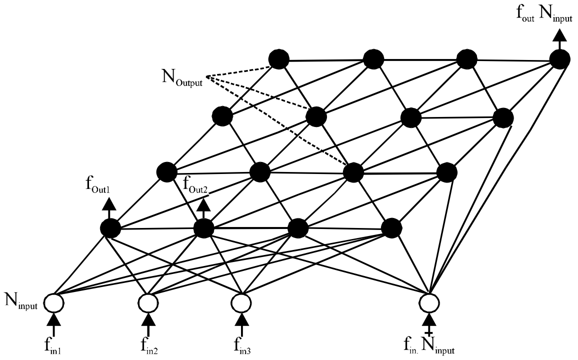

Fig. 5: Self-organizing weight adaptation spiking neural network (SOWA_SNN) architecture. Redrawn from Pham *et al.* (2007)

$$\text{st} = \text{tw} - (\text{Input} + \text{Delay}); \begin{bmatrix} \text{Input} \in (0,30) \text{ as taken in SOWA\_SNN} \\ \text{Delay} = 0 \text{ as taken in SOWA\_SNN} \\ \text{tw} \in (0,30) \text{ as outlined (Eq.3 \& 4)} \\ \text{st} \in (0,30) \text{ as specified (Eq.5 \& 6)} \end{bmatrix} \tag{23}$$

The spike response function ε(st) which has been used by Pham *et al.* (2007) is given by Eq. 24 for clustering application and is graphically described as in Fig. 3b:

$$\varepsilon(\text{st}) = \frac{\text{st}}{a}.e^{1-\frac{\text{st}}{a}} \tag{24}$$

In this model st∈(0, 30), i.e., the function is defined on a closed interval and the absolute minimum and maximum value could be derived within this interval as the following steps:

Step 1: Derivation of ε(st)'

$$\varepsilon(\text{st})' = \frac{1}{a}.\left(1 - \frac{\text{st}}{a}\right).e^{1-\frac{\text{st}}{a}} \tag{25}$$

Step 2: Find out the critical points of ε(st) in (0, 30) as follows:

$$\varepsilon(\text{st})' = 0 \rightarrow \text{st} = a \tag{26}$$

The absolute minimum and maximum of ε(st) does not depend on the value of time (tw), time coding (Input) or delay (Delay). It depends only on the value of the time constant for excitatory spike response function (a). This finding shows that the θ_{min} and θ_{max} value is not affected by them too.

Step 3: Calculate the value of ε(st) at the critical point where (a = 35) as it has been taken by Pham *et al.* (2007)

At st = 35→ε(35) = 1.

Step 4: The absolute minimum and maximum value of ε(st) is (0) and (0.98877), respectively on (0, 30), then the range for ε(st) is as in Eq. 27:

$$\varepsilon \epsilon [\varepsilon_{min}, \varepsilon_{max}] = [0, 0.98877] \quad (27)$$

Step 5: By applying Eq. 21 and 22 to calculate the sum of ε(st) for one output neuron:

$$\theta_{min} = 60*0*0.3 = 0 \quad (28)$$

$$\theta_{max} = 60*0.9887*0.5 = 29.6631 \quad (29)$$

Then, the threshold boundary in SOWA_SNN is θϵ[0, 29.6631]. Figure 4b shows the relation between the threshold boundary and a spike time.

Hence, the initial and suitable threshold value will be within this range. Equation 30 and 32 are the equations suggested to calculate the suitable threshold:

$$\theta_{avg1} = N_{input} * \frac{\underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Min}}} (\varepsilon(st_i)) + \underset{j=1,k=1}{\overset{N_{output},N_{input}}{\text{Max}}} (\varepsilon(st_i))}{2} * wght_{avg} = 11.86524 \quad (30)$$

Where:

$$wght_{avg} = \frac{wght_{max} + wght_{min}}{2} \quad (31)$$

$$\theta_{avg2} = \frac{\theta_{max} + \theta_{min}}{2} = 14.83155 \quad (32)$$

Compared the θ value which has been used by Pham *et al.* (2007) firstly with the θ_{avg2}, both are very closed and that supports that Eq. 32 is the suitable equation to get the suitable threshold in this case and secondly with the θ_{avg1} which its value is far away from the one (Pham *et al.*, 2007) used.

RESULTS AND DESCUSSION

This study outlines the input time window boundary which leads to specify the spike time boundary. With regard to the input time window, it is found that specification beyond the parameter boundary affects on the computational network cost and performance and found that the delay and the time coding parameters play a significant role in assigning the time window boundary. By outlining the input time window boundary, the number of free parameters which used to be assigned experimentally decreased by one, as the time window would be assigned depends only on the delay and the time coding parameters which assigned by the experimenter. Many SNN learning algorithms have been proposed for supervised and unsupervised learning (Bohte *et al.*, 2002a, b; Xin and Embrechts, 2001; Pham and Sahran, 2006; Ruf and Schmitt, 1997; Sporea and Gruning, 2012; Natschlager and Ruf, 1998; Pham *et al.*, 2007). However, to the best of the author knowledge, this is the first time where a clear discussion on outlining input time window parameter has given.

Specifying the spike time range help to determine the threshold boundary. The results show that the threshold parameter boundary could be found for learning in temporal coding SNN for classification (S_LVQ) (Pham and Sahran, 2006) and clustering (SOWA_SNN) (Pham *et al.*, 2007) applications and suggest two formulas for assigning the suitable threshold for each one and compare the threshold value in the original learning algorithm with the suggested one. Results prove that the suitable threshold for S_LVQ is θ_{avg1} as the experimental threshold value assigned by Pham and Sahran (2006) is closed to the θ_{avg1} value suggested in the Eq. 17 and the suitable threshold value for SOWA_SNN is θ_{avg2} as the experimental threshold value assigned by Pham *et al.* (2007) is closed to the θ_{avg2} value suggested in the Eq. 32. In this study, one part of the question have been solved, the threshold boundary can be find mathematically where the threshold cannot go beyond it and then select the suitable threshold within this range. How to select suitable threshold within this range remain an open question and needs to be analyzed more on this issue in the future, to see if the suitable threshold could be determined depends on some other parameters.

CONCLUSION

It was concluded and proved that the θ_{min} and θ_{max} value does not depend on the value of the time input window (tw), time coding (Input) or delay (delay). It rather depends on the value of time constant for membrane (tce), the value of time constant for synapse (tci), N_{input}, N_{hidden}, N_{sub} and N_{output} in S_LVQ learning algorithm which has been proposed by Pham and Sahran (2006) and on the value of the time constant for excitatory spike response function (a), N_{input} and N_{output} in SOWA_SNN learning algorithm which has been proposed by Pham *et al.* (2007).

ACKNOWLEDGMENTS

This research was carried out with support from the Science Fund Grant 01-01-02-SF0694 and Center for Artificial Intelligence, Faculty of Information Science and Technology, UKM Bangi, Malaysia.

REFERENCES

Bohte, S.M., H. la Poutre and J.N. Kok, 2002a. Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer RBF networks. IEEE Trans. Neural Networks, 13: 426-435.

Bohte, S.M., J.N. Kok and H.L. Poutre, 2002b. Error-backpropagation in temporally encoded networks of spiking neurons. Neurocomputing, 48: 17-37.

Hopfield, J.J., 1995. Pattern recognition computation using action potential timing for stimulus representation. Nature, 376: 33-36.

Maass, W. and C.M. Bishop, 2001. Pulsed Neural Networks. 1st Edn., The MIT Press, Cambridge, Massachusetts, ISBN-13: 9780262632218, Pages: 377.

Maass, W., 1997. Networks of spiking neurons: The third generation of neural network models. Neural Networks, 10: 1659-1671.

Natschlager, T. and B. Ruf, 1998. Spatial and temporal pattern analysis via spiking neurons. Network: Comput. Neural Syst., 9: 319-332.

Pham, D.T. and S. Sahran, 2006. Control chart pattern recognition using spiking neural networks. Proceedings of 2nd Virtual International Conference on Intelligent Production Machines and Systems, July 3-14, 2006, Elsevier, Amsterdam, pp: 319-325.

Pham, D.T., M.S. Packianather and E.Y.A. Charles, 2007. Self-organising spiking neural networks trained by weight-and delay-adaptation methods for control chart pattern recognition. Proceedings of the 3rd International Conference (Virtual) on Intelligent Production Machines and Systems, July 3-14, 2006, Cardiff, UK.

Pham, D.T., M.S. Packianather and E.Y.A. Charles, 2008. Control chart pattern clustering using a new self-organizing spiking neural network. Proc. Inst. Mech. Eng. B: J. Eng. Manuf., 222: 1201-1211.

Ruf, B. and M. Schmitt, 1997. Learning temporally encoded patterns in networks of spiking neurons. Neural Process. Lett., 5: 9-18.

Shahnorbanun, S., S.A.S.N. Huda, A. Haslina, O. Nazlia and H. Rosilah, 2010. A computational biological network for wood defect classification. Proceedings of the World Congress on Engineering and Computer Science, Volume 1, October 20-22, 2010, San Francisco, USA., pp: 1-5.

Sporea, I. and A. Gruning, 2012. Supervised learning in multilayer spiking neural networks. Neural Comput., 25: 473-509.

Xin, J. and M.J. Embrechts, 2001. Supervised learning with spiking neural networks. Proceedings of the International Joint Conference on Neural Networks, Volume 3, July 15-19, 2001, Washington, DC., pp: 1772-1777.