Аграрен университет – Пловдив, Научни трудове, т. LVIII, 2014 г.
Agricultural University – Plovdiv, Scientific Works, vol. LVIII, 2014

МОДЕЛ НА ПАМЕТ ЧРЕЗ НЕВРОННА МРЕЖА С ФУНКЦИОНАЛНОСТ НА RS-ТРИГЕР
A MEMORY MODEL THROUGH A NEURAL NETWORK WITH SR-LATCH FUNCTIONALITY

Никола Цайкин
Nikola Tsaikin

E-mail: tsaikin@au-plovdiv.bg

Abstract

The paper presents a developed model of a memory element based on a neuron with a log sigmoid transfer function. The element steadily memorizes one bit in the course of the iterations and works on the principle of the electronic SR-latch. It is proved by means of a graphical presentation that the SR-latch functionality can be realized by a single neuron only. The function *Iterative gradient* is defined, describing the internal iterative dynamics of the model under static external influences. The theoretic estimation of the weight coefficients of the model is confirmed by the experimentally obtained ones.

Key words: Hopfield neural networks, memory, SR-latch, iterative gradient.

INTRODUCTION

One of the most essential functions of Neural Networks (NN) is the recognition of input configurations and their juxtaposing with output ones. When NN is observing a given environment with the sensors, connected to it, it can be trained to recognize and indicate certain characteristics of this environment. Recording signals which indicate events is an important part of the logical processing of information and NN have potential to store information over time (Damara et al., 2013) using the recurrent networks (NARX) according to the models of Hopfield (Li et al., 1989) and Elman (Elman, 1990). In the electronic systems for logical processing the basic memory element is the electronic latch. It can store one bit which corresponds to an existence or a lack of an event. The basic electronic latch is the asynchronous SR-latch (Phister, 1958) with Set and Remove inputs. Most of the other types of latches have additional inputs for gating signals typical for the electronic devices. SR-latches, made by NN are designed to implement shift registers and binary counters and other digital devices in order to reduce energy consumption (Ninomiya et al., 1994).

MODEL OF A NEURON WITH THE FUNCTIONALITY OF A SR-LATCH

The purpose of this task is to create a configuration on Hopfield neural networks (Behl et al., 2013), to realize memory, which stores one bit for an infinitely long time. Similarly to the electronic latch, the configuration must have two inputs for Set and Remove, which establishes one output in a logical 1 or 0. At a state of logical 0 at the two inputs, the output must stay for an infinitely long time according to the last activated input.

The question that arises is what reaction must NN be trained to, when both of the inputs are activated simultaneously. This input combination is forbidden to the electronic latches, but this limit is hard to be imposed to the environment in observation in NN. That is why the following assumption must be done. If both inputs for Set and Remove are activated in logical 1, this is considered to be unclear information, therefore lack of information, so the simulated with NN latch has to keep the output without any change. I.e. the reaction of the input combination {1,1} is equalized to the reaction of the input combination {0,0}. Like this we get the table of true values (table 1), which the network must be trained to, where the third input parameter $M=Y_{(t-1)}$ shows the output in the moment t-1, i.e. the output of the previous iteration. The obtained 8 configurations are the 8 states of the simulated latch.

Table 1

Table of true values. States.

№	Is	Ir	M	Y	Description	x1,5	Exp
1	0	0	0	0	Memorizing 0	0	-2,14
2	0	0	1	1	Memorizing 1	1,5	2,18
3	0	1	0	0	Pressing 0	-1	-6,67
4	0	1	1	0	Transmission to 0	0,5	-2,35
5	1	0	0	1	Transmission to 1	1	2,37
6	1	0	1	1	Pressing 1	2,5	6,69
7	1	1	0	0	Memorizing 0	0	-2,15
8	1	1	1	1	Memorizing 1	1,5	2,17

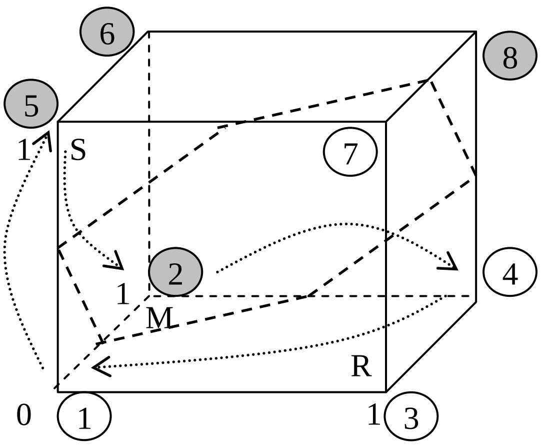

***Fig. 1**. Presenting the 8 states in a three-dimensional space {R,S,M}*

The eight states from the table can be viewed as apexes of a cube in a three-dimensional space with a system of coordinates {S, R, M}∈{0,1} where S and R are the inputs and M is the output from the previous iteration $Y_{(t-1)}$ (fig. 1). The state is marked on the apexes of the cube – in the white counters Y is a logical 0, in the grey ones – logical 1. The transitions between the 4 working states are marked with arrows. It can be seen that the states for the logical 1 and 0 can be split into two non-intersecting groups with plane S-R+M-0,5=0. On figure 1 it is shown with dotted line in section with the cube.

Another way to present the 8 states is a number line (fig. 2a), where S, R and M form total value L=S-R+1,5M and M is multiplied by a suitable coefficient 1,5. These values are presented in the column "x1,5" from table 1. On figure 2 the numbers of the states are presented above the number line. It can be seen that the states of Y for 1 and 0 are divided into two non-intersecting groups. This means that the logical function of the trigger according to table 1 can be realized using only one neuron which through the three weight transmission coefficients of S, R and M and the internal deviation of the bias B can realize the demarcation plane from figure 1.

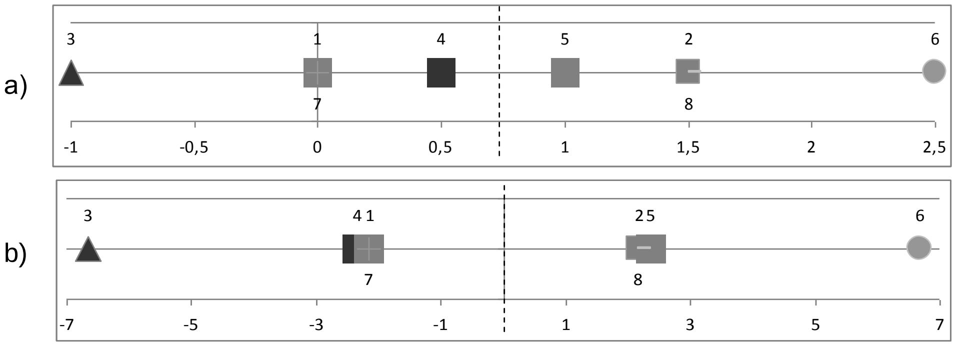

Fig. 2. *Presenting the eight states with number lines*

What is offered here is a neuron network with a functionality of a SR-latch for recording of 1 bit, realized with 1 neuron, 2 inputs for Set and Remove, feedback and bias (fig. 4c). Before investigating the dynamics of this model, the dynamic of the neuron is going to be studied, only with feedback m, without external inputs and bias (fig. 4a). The aim is to prove that a neuron with a log-sigmoid transfer function and feedback has the property to support equilibrium output level.

In the electronic latch the functionality of the recording is realized through the mutual stop/release of the other transistor when the collector of one of the transistors gives a signal to the other transistor. In this way stable state on the exit is maintained, until a signal from the driving inputs is received which violates the stability and dynamics of the mutual stop/release and leads to turning the latch over. This is the reason to study the dynamics of a neuron with a log-sigmoid transmission function and feedback regarding its ability to maintain stable output levels.

FUNCTION "ITERATIVE GRADIENT"

Let us introduce the function $G(Y_{(t-1)})$, which will be defined as "iterative gradient" on the output of the neuron.

$$G(Y_{(t-1)}) = Y_{(t)} - Y_{(t-1)} \tag{1}$$

$$Y(i+m) = 1/(1+e^{-(i+m)}) \tag{2}$$

where i is the total value of the set of external inputs and biases and $m = Y_{(t-1)}$ is the feedback.

This function is useful with several peculiarities. The argument of G is the output of the neuron from the previous iteration $Y_{(t-1)}$. It cannot be calculated with (2), but is assumed to be a parameter with an interval {0,1}. Using (2) only $Y_{(t)}$ has been calculated, where m is replaced by the parameter $Y_{(t-1)}$. The aims of placing $Y_{(t-1)}$ as a parameter is the following: The function G visualizes well only the dynamics of the neuron at static internal inputs, i.e. at memorizing mode. It is not interested in the dynamics of the transmissions, which originates from changes in the set of i in (2). It shows, after ceasing the change of the set i, what gradient the output will run to the equilibrium level. This is important for the graphical representation of figure 3, where the values of $Y_{(t-1)}$ are on the abscissa, not the argument (i+m) of the log-sigmoid function.

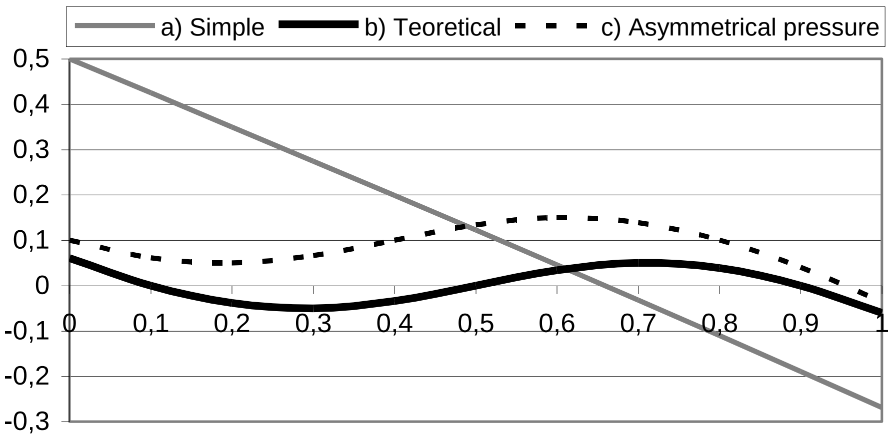

***Fig. 3**. The function "iterative gradient"*

When G=0, the output stays unchanged during the iterations. The way in which G crosses the Y-axis shows the stability of the output level. If the derivative of G in the intersection point is negative, the state is stable and dynamic. If the derivative of G at G=0 is positive, the state is unstable and when $G \approx 0$, the output level will move away from this point. I.e. the direction of intersection of G with the abscissa shows whether the equilibrium levels are self-maintaining or the level drives back from the point G=0. The first to be examined will be the simplest configuration of the neuron with feedback with weight coefficient 1 and without external inputs and bias.

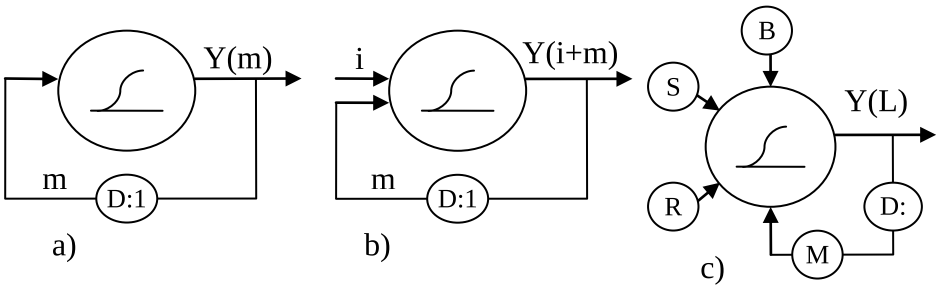

Fig. 4. *Models of neuron with log-sigmoid transfer function for studying of its equilibrium states*

Experimentally, with approximation Δm=0,001, it is found that when m=0,659 Y(m)=m. On figure 3a) is shown G, which is not a straight line, but has a slight S-shaped twist. It is approximated precisely enough to the straight line y=-0,7678x+0,5036, which crosses the abscissa at x=0,6559. It can seen that G crosses the abscissa with negative derivative. This proves the existence of iteration dynamics, which maintains output level Y(m)= 0,659.

THEORETIC ESTIMATION OF THE WEIGHT COEFFICIENTS

On figure 4c) is shown a model of a trigger, realized with one neuron with feedback, bias and inputs for Set and Remove. The weight coefficients of the model can be calculated theoretically with enough preciseness before being obtained experimentally with training of the neuron.

Let's code the logical 0 and 1 with input-output levels respectively 0,1 and 0,9. What is marked with L is the summary argument of the log-sigmoid transfer function Y(L), and we have $Y(L_0)$=Y(-2,197)=0,1 and $Y(L_1)$=Y(2,197)=0,9 or L_0=-2,197 and L_1=2,197 for the two logical levels. The four weight coefficients S, R, B and M are searched for. In order to find them, a system of four equations has been built, originating from the four main states from table 1. These are states №1 and №2 for static recording of 0 and 1 and №4 and №5 for transmission to 0 and 1. The system is built according to how the argument L of Y(L) is formed totally, in order to obtain the desired output level.

$$0{,}1M+0{,}1S+0{,}1R+B=L_0=-2{,}197 \quad (3)$$
$$0{,}9M+0{,}1S+0{,}1R+B=L_1=2{,}197$$
$$0{,}9M+0{,}1S+0{,}9R+B=-2{,}197$$
$$0{,}1M+0{,}9S+0{,}1R+B=2{,}197$$

Solution: S= 5,4925; R=-5,4925; B=-2,7463; M=5,4925. (4)

As S and R take part with the same coefficients in the first two equations, a parametric group i= 0,1S+0,1R+B has been created, which does not change in memorizing mode when Is=Ir=0,1. Thus in states №1 and №2 from the table, the internal iteration dynamic is defined by the returned from the output information m and the static deviation i. Let's remind that in (2) the argument f was divided into two components: i – the influence of the external inputs and the bias; m – the feedback. According to the model from figure 4c), G can be built. The S-shape of the function can be seen. Let's remind that the function G of the model in figure 4a) is S-shaped too, but because of the lack of the i component it has insignificant amplitude of deviation and different slope. According to the solution of (4) from the model of figure 4c) G crosses the abscissa in an unstable point at $f_{(t-1)}$=0 and two stable points at the levels for logical 0 and 1. This ensures that the functionality of the model could maintain two stable equilibrium levels and ensures states №1 and №2 for infinitely long time, which coincides with the functionality of the electronic SR-latch. The weight coefficients (4) are theoretically calculated model for memory block with functionality of an electronic SR-latch, according to figure 4c).

TRAINING OF THE NEURON

Two types of input sequences are used for the training (fig. 5). The first type is peak influence with level 0,9 only for one time step in the beginning or the middle of the sequence. The second type is raising the respective input to logical 1. In the first and the second half of the sequence there are two periods of 10 time steps each, in which the inputs are not being changed. In this way, peak and permanent input effects are simulated. The target values for training are also 0,1 and 0,9. If the signals from figure 5 are input combination, after peak influence on the Set input, the target value for the output in the second half will be 0,1. In a similar way of combining are formed 21 sequences with length 23 time steps each, so that all the states from states from the table of trueness (table 1) can be represented. The main task of the training is not to allow undesirable switching after peak or permanent influence and the output level to stay infinitely long time close to the target values of the logical 0 or 1.

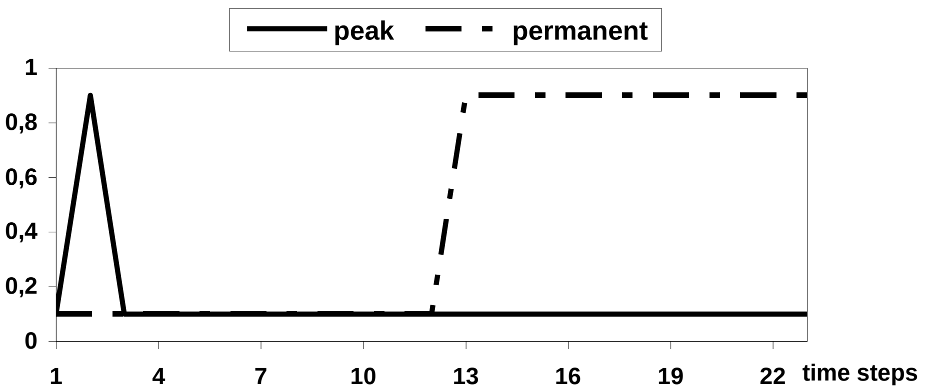

Fig. 5: *Kinds of configurations of the input signals*

The training is done with a training function from the method of Levenberg-Marquardt (Rumelhart et al., 1986). Five trainings at different and random onset initialization of the weights are done. The training stops in all the trials due to minimum gradient of change 1 e-10. In the five trainings the results are the same. The values of the weight coefficients are repeated with an allowance of at least the fourth decimal place. They are:

$$S=5,6442;\ R=-5,6552;\ B=-2,7148;\ M=5,4497 \qquad (5)$$

The values obtained through training coincide with the theoretically calculated ones to a sufficient extent in order to be able to say that they confirm them. As B and M define the internal iteration dynamic, the experimentally obtained values for them coincide to a greater degree with the ones from the theoretical model. That is why the graph of G of the training model almost entirely coincides with the one of the theoretical model and is not shown on figure 3. S and R cancel each other out and the static component i according to the model on figure 4b) is formed entirely from the bias B. S and R actively take part in the changing of the output Y(L) when transmitting.

The aim of the transmission is the central equilibrium point G=0 to be overcome, at $Y_{(t-1)}=0$, with positive derivative G'. Actually with these values S and R from (5) position directly Y(L) right next to the logical 1 and 0.

The weight coefficients (5) are experimentally obtained model of a neuron network with a functionality of an electronic SR-latch according to figure 4c). On figure 2b) and in table 1, column "Exp" we can see the values of the total argument L and their grouping on a number line. The opportunity to split the states for the logical 0 and 1 into two non-intersecting groups is what allows the table of trueness Table 1 to be realized with only one neuron.

THE PROBLEM OF THE PERMANENT ASYMMETRICAL PRESSURE

The model is not suitable for connecting with other parts of neuronal networks because it works correctly only under strictly fixed levels at the inputs for the logical 1 and 0. This kind of restriction is hard to be imposed in the neuronal networks and the system should work correctly with random input levels between 0 and 1 if it is assumed that they are the output levels of other neurons with log-sigmoid transfer function. If on one of the inputs, for example Is, there is a value larger even by 0,1 than Ir during several iterations, from a logical 0 it will become undesirable transmission to levels over 0,9.

This can be seen from the third graphic "pressure" from figure 3c), which is entirely above the abscissa and crosses it only at $Y_{(t-1)}>0,9$. When the iteration gradient is positive, the output will increase in the next iteration by the value of the gradient. Even at this minimally asymmetric pressure on the input, the iteration dynamic will lead to an undesirable transmission.

CONCLUSIONS

The developed model has a value only as a basis for further development in two directions, regarding the overcoming of the disadvantage, connected with the permanent asymmetric pressure. The first one is the development of calibrating layers from neurons, which to equalize the input configurations from random levels to ones with minimal deviations from the logical 0 and 1. The other approach is to define a memory element, driven according to the functionality of an SR-latch by the previous neuron layers, similar to the delayed elements forming tapped delay lines. The obtained configuration from figure 4c) and the weight coefficients (5) can be used as starting point for the defining of a memory element with unified interface to the other neuron layers on the input and the output.

REFERENCES

Behl, I. S., A. Saini, J. Verma, 2013. "Hopfield networks". International Journal of Scientific Research and Education. Volume 1, Issue 7, pp. 154-162, ISSN (e): 2321-7545.

Damara, *A., V. Damara, A. Chanana*, 2013. "Associative memory: a memory". International Journal of Innovative Research in Engineering & Science, issue 2, volume 10, ISSN 2319-5665.

Elman, J. L., 1990. Finding structure in time, Cognitive Science, vol. 14, pp. 179-211.

Li, J., A. N. Michel, and W. Porod, 1989. Analysis and synthesis of a class of neuralnetworks: linear systems operating on a closed hypercube, IEEE Transactionson Circuits and Systems, vol. 36, no. 11, November 1989, pp. 1405-1422.

Ninomiya, H., H. Asai, 1994. "Design and Simulation of Neural Network Digital Sequential Circuits". IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, Vol. E77-A, No. 6, pp. 968-976.

Phister, M., 1958. Logical Design of Digital Computers. Wiley, p.128.

Rumelhart, D., J. McClelland, 1986. Learning internal representations by error propagation, editors. Parallel Data Processing, Vol.1, Chapter 8, the M.I.T.Press, Cambridge, MA pp. 318-362.

Рецензент – проф. д-р Кольо Онков
E-mail: kolonk@au-plovdiv.bg