

ANALYSIS OF THE SIMULATED ANNEALING METHOD IN CLASSIC BOLTZMANN MACHINES

P. Grabusts

Rezekne Higher Education Institution
90, Atbrivoschanas Alley, Rezekne, LV - 4600, Latvia

Phone: (46) 23798

Fax: (46) 23709

E - mail: peter@anna.cs.ru.lv.

ABSTRACT

The paper analyses a model of a neural net proposed by Hinton et al (1985). They have added noise to a Hopfield net and have called it *Boltzmann machine* (BM) drawing an analogy with the behaviour of physical systems with noises. The concept of *simulated annealing* is analysed. The experiment aimed at testing the state of thermal equilibrium for a Boltzmann net with three neurons, specified threshold values and weights at two different temperatures, $T=1$ and $T=0,25$, is described.

KEYWORDS

Recurrent networks, Boltzmann machine, simulated annealing, learning algorithm.

1. INTRODUCTION

A classic Boltzmann machine is treated as a neural net defined in (Ackley, Hinton and Sejnowski, 1985). The present paper illustrates concepts of *simulated annealing* and *thermal equilibrium* by an example

of the partial net. The technique of transition probabilities computation at different temperatures is described that makes use of Markov chains.

2. BOLTZMANN MACHINES AND HOPFIELD NETS

Two papers by Hopfield (1982, 1984) were of crucial importance in pointing out the connections between brain model and physical systems. Ackley, Hinton and Sejnowski (1985) proposed a neural model called *Boltzmann machine* that was a cross between a Hopfield network and the computation technique called *simulated annealing* which was described by Kirkpatrick, Gelatt and Vecchi (1983).

To understand Hinton's idea, one should imagine the interpretation of an energy landscape (see Fig.1). Each state of the system could be associated with a point in the energy landscape where there might exist points with either local minimum (E_1) and / or global minimum (E_2).

For the ball to be able to transit from state E_1 to state E_2 , changes in the ball energy state are necessary. In the intuitive sense, this is an activation function or the concept of noise proposed by Hinton. The problem is how to find the global minimum of the function. Hopfield networks are efficient for finding local minimums, however, they do not find global ones.

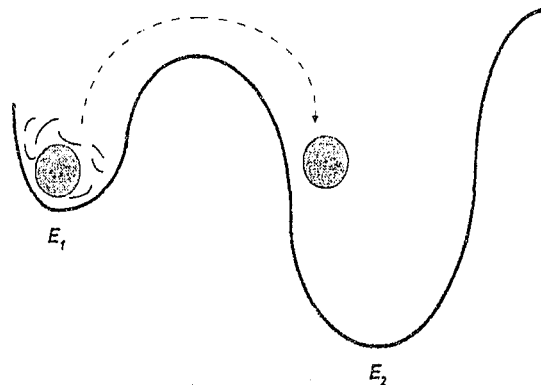


Fig.1. A noisy ball finds the lowest hill.

In real world physical systems, with energy landscape there is associated the concept of temperature. In 19th century, Austrian physicist Ludvig Boltzmann showed that the energy of gas molecules depends on

the temperature. If a system is able to easily transit from one state to another, the most probable configuration of its being in any state could be computed as follows:

$$\frac{P_{E1}}{P_{E2}} = e^{-(E1-E2)/T} \quad (1)$$

Hinton therefore used the name of Boltzmann to convey the idea that the energy of the state of a neural net could be changed depending on the temperature. At $t=0$ the net behaves similarly as the Hopfield model. Hinton has proposed a network operation mode that could find the global minimum, which is called *simulated annealing*, that is the gradual decrease of the temperature in the network.

3. THE ARCHITECTURE OF BM

In what follows, we will use denotations and a net with three nodes (see Fig. 2). (Alexander and Morton, 1991):

For the above net, the following holds:

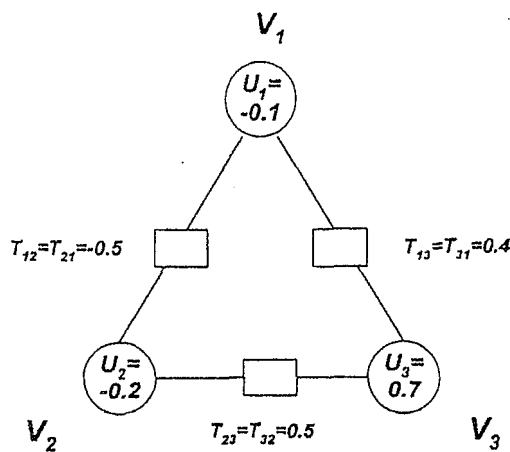


Fig.2. The net nodes.

$$V_i = 1, \text{ if } \sum_{i \neq j} T_{ij} V_j > U_i \text{ and}$$

$$V_i = 0, \text{ if } \sum_{i \neq j} T_{ij} V_j < U_i \quad (2)$$

where V_i stands for the firing state of the neuron ($V_i=0$ when not firing and $V_i=1$ when firing); U_i is the threshold of the neuron and $\sum_{i \neq j} T_{ij} V_j - U$ (3)

is the activation of the neuron, T_{ij} being the weight linking neuron i to neuron j .

The effect of temperature might be shown as alternation of probabilities (see Fig.3):

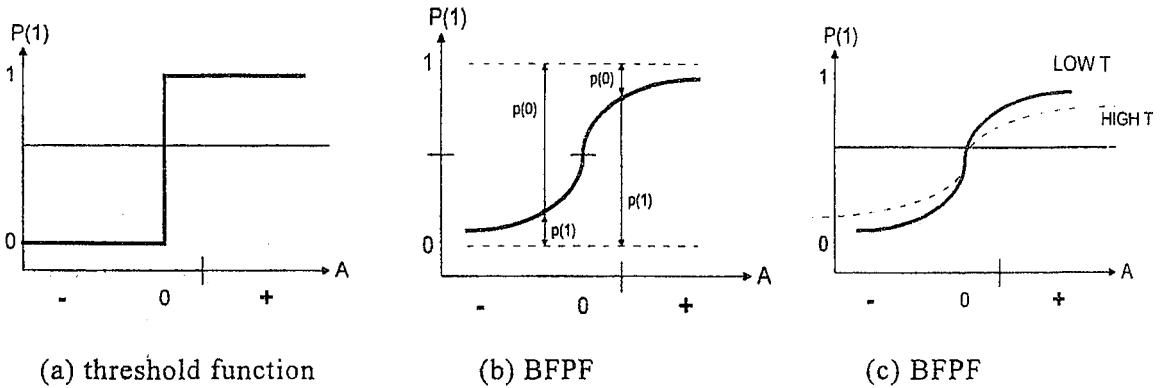


Fig.3. The effect of temperature on firing probability.

Hinton has shown that Boltzmann function correctly characterizes this effect:

$$p(1) = \frac{1}{1 + e^{-\frac{A}{T}}}$$

We shall refer to this as the Boltzmann firing probability function (BFPF). This function has been plotted in Fig.4 for temperatures of 0,5 and 0,25 (in arbitrary units):

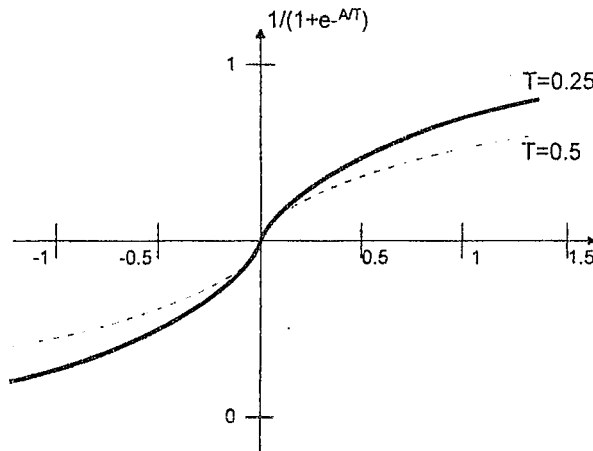


Fig.4. The S-shaped BFPF.

To characterize a BM, the following is required:

- (1) Compute activation function by formula (3) for each state $V_1 V_2 V_3$:

For example: $V_1 V_2 V_3 = 0 1 0$.

$$A_1 = T_{12}V_2 + T_{13}V_3 - U_1 = -0,5 + 0,1 = -0,4$$

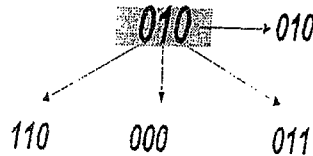
$$A_2 = T_{12}V_1 + T_{23}V_3 - U_2 = 0 + 0 + 0,2 = 0,2$$

$$A_3 = T_{13}V_1 + T_{23}V_2 - U_3 = 0,5 - 0,7 = -0,2.$$

(2) Compute probability for each neuron by formula (4) at various temperatures [$p(0) = 1 - p(1)$].

Neuron number	T=0.25		T=1.0	
	p(1)	p(0)	p(1)	p(0)
1.	0.17	0.83	0.4	0.6
2.	0.69	0.31	0.55	0.45
3.	0.3	0.7	0.45	0.55

(3) By using values from point (2), transition probabilities should be computed. In general case, if a network consists of N elements, a neuron might remain in the same state or to transit to N other states. From state 010 the following transitions could be obtained:



Transition probabilities are computed as follows:

$$\left[V_j(p(1)_j + (1 - V_j)p(0)_j) \right] / 3 \tag{5}$$

Determine probabilities for T=1:

$$010 \rightarrow 110 - p(1)_1 = 0,13$$

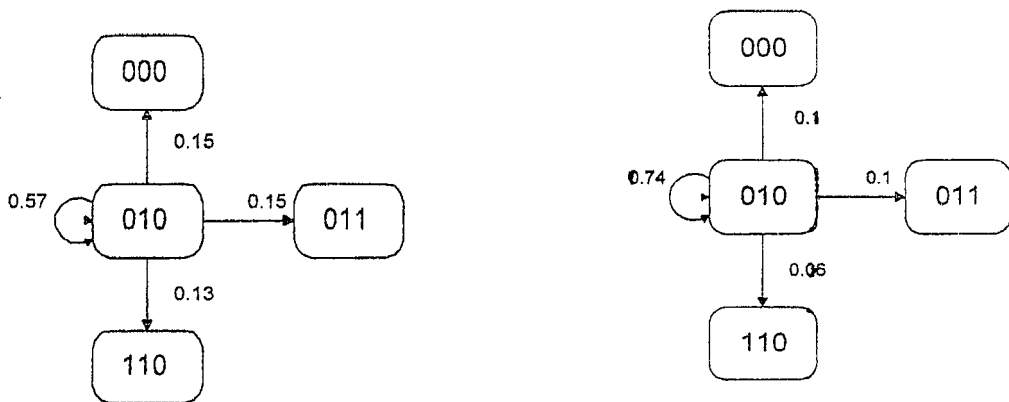
$$010 \rightarrow 000 - p(0)_2 = 0,15$$

$$010 \rightarrow 011 - p(1)_3 = 0,15$$

$$010 \rightarrow 010 - p = 0,57.$$

Application of this formula for state 010 for each neuron at different temperatures is illustrated in Fig. 5.

(4) On obtaining thus transition diagrams for all the states, one could then draw the BM state diagram for each temperature. In practice, however, this is a labour-intensive process, that is why another technique is used. A chain of events $S_0, S_1, S_2, \dots, S_{m-1}$ is given and a system is known for which the events follow each other with known probabilities $p(i, j)$. The system might be represented as a matrix $m \times m$. This technique is referred to as Markov chain. It enables one to determine the probability of system's being in any state and at any time. The results are given in Table 1 where the number of state is set by a binary number, e.g. $S_2=010$. The shaded column shows the state used for illustration.



(a) Boltzmann behaviour (T=1.0)

(b) T=0.25

Fig.5. Transition probabilities at different temperatures.

Table 1

Next State	CURRENT STATE							
	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
S_0	0.55	0.31	0	0	0.13	0	0	0
S_1	0.02	0.08	0	0.02	0	0.04	0	0
S_2	0.23	0	0.24	0.23	0	0	0.27	0
S_3	0	0.31	0	0.58	0	0	0	0.17
S_4	0.2	0	0	0	0.71	0.26	0.26	0
S_5	0	0.3	0	0	0.08	0.47	0	0.1
S_6	0	0	0.06	0	0.08	0	0.24	0.1
S_7	0	0	0	0.17	0	0.23	0.23	0.63

Markov chain for T=0,25

Next State	CURRENT STATE							
	S ₀	S ₁	S ₂	S ₃	S ₄	S ₅	S ₆	S ₇
S ₀	0.53	0.22	0.15	0	0.16	0	0	0
S ₁	0.11	0.35	0	0.11	0	0.13	0	0
S ₂	0.18	0	0.57	0.18	0	0	0.2	0
S ₃	0	0.22	0.15	0.54	0	0	0	0.17
S ₄	0.18	0	0	0	0.56	0.19	0.2	0
S ₅	0	0.21	0	0	0.14	0.5	0	0.15
S ₆	0	0	0.13	0	0.14	0	0.42	0.15
S ₇	0	0	0	0.17	0	0.18	0.18	0.53

Markov chain for T=1,0

It is known that the probability of being in state S_i at time t is P_i(t), so the probability of being in some state j at time t+1, P_j(t+1) may be worked out by adding up all the probabilities of entering that state, taking into account the probability of being in the previous state:

$$P_j(t+1) = \sum P_i(t) p(i, j) \tag{6}$$

For example, say the probability of being in any state at t=0 is 0,125 (that is 1/8). The probability of the system being in state S₃ will be as follows:

$$\begin{aligned}
 P_3(1) &= P_0(0)p(0,3) + P_1(0)p(1,3) + P_2(0)p(2,3) + P_3(0)p(3,3) + P_4(0)p(4,3) + \\
 &+ P_5(0)p(5,3) + P_6(0)p(6,3) + P_7(0)p(7,3) = \\
 &= 0,125*0 + 0,125*0,22 + 0,125*0,15 + 0,125*0,54 + 0,125*0 + 0,125*0 + \\
 &+ 0,125*0 + 0,125*0,17 = 0,125(0,22 + 0,15 + 0,54 + 0,17) = 0,135.
 \end{aligned}$$

4. STIMULATED ANNEALING

By using the possibility to compute network nodes in any state and at any time, one could observe what happens if the temperature of the network is changed. Let us turn to formula (6) and compute probabilities of being in any network state at time t using a Markov chain for T=1 and T=0,25. The results are given in Table 2.

Starting with equal probabilities (i.e. 1/8), the system then reaches t=7 at the temperature equal to 1. By testing experimentally the next steps, one can conclude that the probabilities are not actually changed at this temperature. According to Hinton, this effect is called *thermal equilibrium*. The probabilities will start changing when the temperature is changed. From Table 2 it is seen that the temperature is decreased

up to 0,25. Thermal equilibrium is reached again at $t=15$. If we transit to $T=0$ at this point then a final state of the network could be obtained:

T. Time	P(0)	P(1)	P(2)	P(3)	P(4)	P(5)	P(6)	P(7)
0 28	0	0	0,494	0	0,313	0	0	0,193

The network is in three states what gives the local minimum of the system. At the end of the annealing process the system is seen to finish in the stable states of the net with a probability related to their energy. If the initial conditions of point (2) are satisfied, then it can easily be seen that state S_2 with the least energy, 0,1 , has the highest final probability, 0,494, state S_4 with the energy 0,1 at the beginning has the final probability 0,313 and, at last, state S_7 has the lowest final probability. For all the other states, the final probabilities are equal to zero.

The aim of Boltzmann approximation was to prove that the final probabilities in partial state greatly depend on the energy of the state.

Table 2

T. Time	P(0)	P(1)	P(2)	P(3)	P(4)	P(5)	P(6)	P(7)
1.00 1	0.13250	0.08750	0.14125	0.13500	0.14125	0.12500	0.10500	0.13250
1.00 2	0.13326	0.07630	0.14966	0.13586	0.14770	0.12052	0.10211	0.13457
1.00 3	0.13350	0.07198	0.15417	0.13548	0.15002	0.11715	0.10321	0.13450
1.00 4	0.13372	0.07001	0.15694	0.13498	0.15094	0.11487	0.10457	0.13398
1.00 5	0.13396	0.06899	0.15873	0.13461	0.15133	0.11336	0.10555	0.13345
1.00 6	0.13420	0.06843	0.15993	0.13436	0.15151	0.11238	0.10617	0.13302
1.00 7	0.13441	0.06810	0.16074	0.13421	0.15159	0.11172	0.10655	0.13268
0.25 8	0.13054	0.01528	0.20866	0.13741	0.19140	0.09850	0.06050	0.15671
0.25 9	0.12228	0.01052	0.23237	0.13194	0.20334	0.08186	0.05802	0.15866
0.25 10	0.12019	0.00920	0.24609	0.13000	0.20520	0.07376	0.06000	0.15456
0.25 11	0.12024	0.00869	0.25585	0.12913	0.20451	0.06930	0.06104	0.15024
0.25 12	0.12100	0.00845	0.26317	0.12872	0.20314	0.06656	0.06138	0.14658
0.25 13	0.12189	0.00833	0.26875	0.12851	0.20169	0.06473	0.06143	0.14366
0.25 14	0.12272	0.00826	0.27306	0.12842	0.20038	0.06342	0.06137	0.14137
0.25 15	0.12341	0.00822	0.27639	0.12838	0.19926	0.06246	0.06128	0.13959

5. CONCLUSIONS

All the examples used above to analyse the simulated annealing method might be implemented with the help of either programming language. The computing technique for transition probabilities and thermal equilibrium is necessary for further application in Boltzmann machine learning (Ackley, Hinton and Sejnowsky, 1985; Alexander and Morton, 1991; Kappen, 1995).

6. REFERENCES

- Ackley D. H., Hinton G. E and Sejnowski T. J. A learning algorithm for Boltzmann machines. *Cognitive Science*, 9, 1985. - 147. - 169. p.
- Alexander I., Morton H. *An Introduction to Neural Computing*. - London: Chapman & Hall, 1991.
- Fausett L. *Fundamentals of Neural Networks: Architectures, Algorithms and Applications*. Prentice Hall International Inc., 1994.
- Hopfield J. J. Neural networks and physical systems with emergent collective computational abilities. - USA: *Proc. Natl. Acad. Sci.*, 79, 1982. - 2554. - 2558. p.
- Hopfield J. J. Neurons with graded response have collective computational properties like those of the state neurons. - USA *Proc. Natl. Acad. Sci.*, 81, 1984. - 3088 - 3092. p.
- Kappen H. J. Deterministic learning rules for Boltzmann machines. *Neural Networks*, Vol.8, No.4, 1995. - 537 - 548. p.
- Kirkpatrick S., Gelatt C. D. and Vecchi M. P. Optimization by simulated annealing. *Science*, 220, 1983. - 671- 680.p.