Analysis on Bidirectional Associative Memories with Multiplicative Weight Noise

Chi Sing Leung[1], Pui Fai Sum[2], and Tien-Tsin Wong[3]

[1] Department of Electronic Engineering, City University of Hong Kong
eeleungc@cityu.edu.hk
[2] Institute of Electronic Commerce, National Chung Hsing University Taiwan
[3] Department of Computer Science and Engineering,
The Chinese University of Hong Kong

Abstract. In neural networks, network faults can be exhibited in different forms, such as node fault and weight fault. One kind of weight faults is due to the hardware or software precision. This kind of weight faults can be modelled as multiplicative weight noise. This paper analyzes the capacity of a bidirectional associative memory (BAM) affected by multiplicative weight noise. Assuming that weights are corrupted by multiplicative noise, we study how many number of pattern pairs can be stored as fixed points. Since capacity is not meaningful without considering the error correction capability, we also present the capacity of a BAM with multiplicative noise when there are some errors in the input pattern. Simulation results have been carried out to confirm our derivations.

1 Introduction

Associative memories have a wide range of applications including content addressable memory and pattern recognition [1,2]. An important feature of associative memories is the ability to recall the stored patterns based on partial or noisy inputs. One form of associative memories is the bivalent additive bidirectional associative memory (BAM) [3] model. There are two layers, F_X and F_Y, of neurons in a BAM. Layer F_X has n neurons and layer F_Y has p neurons. A BAM is used to store pairs of bipolar patterns, $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$'s, where $h = 1, 2, \cdots, m$; $\boldsymbol{X}_h \in \{+1, -1\}^n$; $\boldsymbol{Y}_h \in \{+1, -1\}^p$; and m is the number of patterns stored. We shall refer to these patterns pairs as *library pairs*. The recall process is an iterative one starting with a stimulus pair $(\boldsymbol{X}^{(0)}, \boldsymbol{Y}^{(0)})$ in F_X. After a number of iterations, the patterns in F_X and F_Y should converge to a fixed point which is desired to be one of the library pairs.

BAM has three important features [3]. Firstly, BAM can perform both heteroassociative and autoassociative data recalls: the final state in layer F_X represents the autoassociative recall, while the final state in layer F_Y represents the heteroassociative recall. Secondly, the initial input can be presented in one of the two layers. Lastly, BAM is stable during recall. In other words, for any connection matrix, a BAM always converges to a stable state. Several methods have been proposed to improve its capacity [4,5,6,7,8].

M. Ishikawa et al. (Eds.): ICONIP 2007, Part I, LNCS 4984, pp. 289–298, 2008.

Although the capacity of BAM has been intensively studied with a perfect laboratory environment consideration[9,10,11,12,13], practical realization of BAM may encounter the problem of inaccuracy in the stored weights. All the previous studies assume that the stored weights matrix is noiseless. However, this is not always the case when training a BAM for some real applications. One kind of weight faults is due to the hardware or software precision [14,15]. For example, in the digital implementation, when we use a low precision floating point format, such as 16-bit half-float[16], to represent trained weights, truncation errors will be introduced. The magnitude of truncation errors is proportional to that of the trained weights. Hence, truncation errors can be modelled as multiplicative weight noise[17,18].

This paper focuses on the quantitative impact of multiplicative weight noise to the BAM capacity. We will study how many number of pattern pairs can be stored as fixed points when multiplicative weight noise presents. Since capacity is not meaningful without considering the error correction capability, we also present the capacity of BAM with multiplicative noise when there are some errors in the input pattern. The rest of this paper is organized as follows. Section 2 introduces the BAM model and multiplicative weight noise. In section 3, the capacity analysis on BAM with multiplicative weight noise is used. Simulation examples are given in Section 4. Then, we conclude our work in Section 5.

2 BAM with Multiplicative Weight Noise

2.1 BAM

The BAM, as proposed by Kosko [3], is a two-layer nonlinear feedback heteroassociative memory in which m library pairs $(\boldsymbol{X}_1, \boldsymbol{Y}_1), \cdots, (\boldsymbol{X}_m, \boldsymbol{Y}_m)$ are stored, where $\boldsymbol{X}_h \in \{-1, 1\}^n$ and $\boldsymbol{Y}_h \in \{-1, 1\}^p$. There are two layers of neurons in BAM; layer F_X has n neurons and layer F_Y has p neurons. The connection matrix between the two layers is denoted as W.

The encoding equation, as proposed by Kosko, is given by

$$W = \sum_{h=1}^{m} \boldsymbol{Y}_h \, \boldsymbol{X}_h{}^T \tag{1}$$

which can be rewritten as

$$w_{ji} = \sum_{h=1}^{m} x_{ih} y_{jh} \,, \tag{2}$$

where $X_h = (x_{1h}, x_{2h}, \cdots, x_{nh})^T$ and $Y_h = (y_{1h}, y_{2h}, \cdots, y_{ph})^T$.

The recall process employs interlayer feedback. An initial pattern $\boldsymbol{X}^{(0)}$ presented to F_X is passed through W and is thresholded, and a new state $Y^{(1)}$ in F_Y is obtained which is then passed back through W^T and is thresholded again, leading to a new state $X^{(1)}$ in F_X. The process repeats until the state of BAM converges. Mathematically, the recall process is:

$$\boldsymbol{Y}^{(t+1)} = \operatorname{sgn}\left(W\,\boldsymbol{X}^{(t)}\right), \text{ and } \boldsymbol{X}^{(t+1)} = \operatorname{sgn}\left(W^T \boldsymbol{Y}^{(t+1)}\right), \tag{3}$$

where sgn(·) is the sign operator:

$$\mathrm{sgn}(x) = \begin{cases} +1 & x > 0 \\ -1 & x < 0 \\ \text{state unchanged} & x = 0 \end{cases} .$$

Using an element-by-element notation, the recall process can be written as:

$$y_i^{(t+1)} = \mathrm{sgn}\left(\sum_{i=1}^{n} w_{ji} x_i^{(t)}\right), \text{ and } x_j^{(t+1)} = \mathrm{sgn}\left(\sum_{j=1}^{p} w_{ji} y_i^{(t+1)}\right), \tag{4}$$

where $x_i^{(t)}$ is the state of the ith F_X neuron and $y_j^{(t)}$ is the state of the jth F_Y neuron. The above bidirectional process produces a sequence of pattern pairs $(X^{(t)}, Y^{(t)})$: $(X^{(1)}, Y^{(1)})$, $(X^{(2)}, Y^{(2)})$, $\cdots$. This sequence converges to one of the fixed points (X^f, Y^f) and this fixed point ideally should be one of the library pairs or nearly so. A fixed point (X^f, Y^f) has the following properties:

$$\boldsymbol{Y}^f = \mathrm{sgn}(W\,\boldsymbol{X}^f) \text{ and } \boldsymbol{X}^f = \mathrm{sgn}(W^T\,\boldsymbol{Y}^f). \tag{5}$$

Hence, a library pair can be retrieved only if it is a fixed point. One of the advantages of Kosko's encoding method is the ability of incremental learning, i.e., the ability of encoding new library pairs to the model based on the current connection matrix only. With the Kosko's encoding method, BAM can only correctly store up to $\frac{\min(n,p)}{2\log\min(n,p)}$ library pairs. When the number of library pairs exceeds that value, a library pair may not be stored as a fixed point.

2.2 Multiplicative Weight Noise

In some electrical circuits, inaccuracy occurs in the implementation of trained weights. Errors in the weights may be caused by quantization error due to limited bits used to store the trained weights, or percentage error due to voltage perturbation. In the digital implementation, such as DSP and FPGA, the trained weights are usually stored as floating-point format. Floating-point representation of real numbers is more desirable than integer representation because floating-point provides large dynamic range. When we use a low precision floating point format, such as 16-bit half-float[16], to represent trained weights, truncation errors will be introduced. The magnitude of the truncation errors is proportional to that of the trained weights[19]. Hence, the truncation errors can be modelled as multiplicative weight noise.

An implementation of a weight w_{ji} is denoted by $\tilde{w}_{ji}$. In multiplicative weight noise, each implemented weight deviates from its nominal value by a random percent, i.e.,

$$\tilde{w}_{ji} = w_{ji} + b_{ji}\, w_{ji} \tag{6}$$

where b_{ji}'s are identical independent mean zero random variables with variance σ_b^2. The density function of b_{ji}'s is symmetrical.

3 Analysis on BAM with Multiplicative Weight Noise

3.1 Capacity

We will investigate the BAM's memory capacity when the multiplicative weight noise presented. The following assumptions and notations are used.

- The dimensions, n and p, are large. Also, $p = rn$, where r is a positive constant.
- Each component of library pairs $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$'s is a ± 1 equiprobable independent random variable.
- $EU_{j,h}$ is the event that $\sum_i^n \tilde{w}_{ji} x_{ih}$ is equal to y_{jh} (the j-th component of the library pattern $\boldsymbol{Y}_h$). Also, $\overline{EU}_{j,h}$ is the complement event of $EU_{j,h}$.
- $EV_{i,h}$ is the event that $\sum_j^p \tilde{w}_{ji} y_{jh}$ is equal to x_{ih} (the i-th component of the library pattern $\boldsymbol{X}_h$). Also, $\overline{EV}_{i,h}$ is the complement event of $EV_{i,h}$.

With the above assumptions and the multiplicative weight noise, we will introduce Lemma 1 and Lemma 2. They will assist us to derive the capacity of BAM with multiplicative weight noise.

Lemma 1: The probability Prob($\overline{EU}_{j,h}$) is approximately equal to

$$Q\left(\sqrt{\frac{n}{(1+\sigma_b^2)m}}\right)$$

for $j = 1, \cdots, n$ and $h = 1, \cdots, m$, where $Q(z) = \frac{1}{\sqrt{2\pi}} \int_z^\infty \exp(\frac{-z^2}{2}) dz$.

Proof: Event $\overline{EU}_{j,h}$ means that $\text{sgn}\left(\sum_{i=1}^n \tilde{w}_{ji} x_{ih}\right) \neq y_{jh}$. From (2) and (6), we have

$$\begin{aligned}\sum_{i=1}^n \tilde{w}_{ji} x_{ih} &= \sum_{i=1}^n w_{ji}(1+b_j i) x_{ih} = \sum_{i=1}^n (\sum_{h'}^m y_{jh'} x_{ih'})(1+b_j i) x_{ih} \\ &= n y_{jh} + \sum_{i=1}^n (\sum_{h' \neq h}^m y_{jh'} x_{ih'}) x_{ih} + \sum_{i=1}^n (\sum_{h'=1}^m y_{jh'} x_{ih'}) b_{ji} x_{ih} \qquad (7)\end{aligned}$$

Without loss of generality, we consider the library pair $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$ having all components positive: $\boldsymbol{X}_h = (1, \ldots, 1)^T$ and $\boldsymbol{Y}_h = (1, \ldots, 1)^T$. This consideration is usually used [13] and does not affect our results. We can easily verify this by use of conditional probability. Now,(7) becomes

$$\sum_{i=1}^n \tilde{w}_{ji} x_{ih} = n + \sum_{i=1}^n (\sum_{h' \neq h}^m y_{jh'} x_{ih'}) + \sum_{i=1}^n (\sum_{h'=1}^m y_{jh'} x_{ih'}) b_{ji} \qquad (8)$$

$$= n + \sum_{i=1}^n \alpha_{ji} + \sum_{i=1}^n \beta_{ji}\,, \qquad (9)$$

where α_{ji}'s are independent identical zero mean random variables (i.e., $E[\alpha_{ji}] = 0$, and $E[\alpha_{ji}\alpha_{ji'}] = 0$ for $i \neq i'$) and the variance of α_{ji}'s, denoted as $\mathrm{Var}[\alpha_{ji}]$, is equal to $(m-1)$. Since b_{ji}'s are independent identical zero mean random variables and they are independent of $(\sum_{h'=1}^{m} y_{jh'}x_{ih'})$'s. Hence, β_{ji}'s are independent identical zero mean random variables, where $E[\beta_{ji}] = 0$, $E[\beta_{ji}\beta_{ji'}] = 0$ for $i \neq i'$), $\mathrm{Var}[\beta_{ji}] = \sigma_b^2 m$.

For large n, the summations, $\sum_{i=1}^{n} \alpha_{ji}$ and $\sum_{i=1}^{n} \beta_{ji}$, tend to normal with variances equal to $(m-1)n$ and $\sigma_b^2 mn$,respectively. Besides, the sum of the two normal random variables is still a normal random variable. Hence, (9) becomes

$$\sum_{i=1}^{n} \tilde{w}_{ji}x_{ih} = n + \sum_{i=1}^{n} \alpha_{ji} + \sum_{i=1}^{n} \beta_{ji} = n + \alpha + \beta\,. \tag{10}$$

Note that β_{ji}'s are independent of $(\sum_{h'=h}^{m} y_{jh'}x_{ih'})$'s. We have $E[\alpha\beta] = 0$, $E[\alpha+\beta] = 0$ and $\mathrm{Var}[\alpha+\beta] = (m-1)n + \sigma_b^2 mn$. Event $\overline{EU_{j,h}}$ means that $\alpha+\beta < -n$. Hence, $\mathrm{Prob}(\overline{EU_{j,h}}) \approx Q\left(\sqrt{\frac{n}{(m-1)+\sigma_b^2 m}}\right)$. For large m, $Q\left(\sqrt{\frac{n}{(m-1)+\sigma_b^2 m}}\right) \approx Q\left(\sqrt{\frac{n}{(1+\sigma_b^2)m}}\right)$. (Proof completed).

Using the similar way, we can have Lemma 2.

Lemma 2: The probability Prob($\overline{EV_{i,h}}$) is approximately equal to

$$Q\left(\sqrt{\frac{p}{(1+\sigma_b^2)m}}\right)$$

for $i = 1, \cdots, p$ and $h = 1, \cdots, m$.

Now, we start to estimate the capacity. Let the probability that a library pair $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$ is fixed point be P_*:

$$\begin{aligned} P_* &= \mathrm{Prob}\left(EU_{1h} \cap \cdots \cap EU_{nh} \cap EV_{1h} \cap \cdots \cap EU_{ph}\right) \\ &= 1 - \mathrm{Prob}\left(\overline{EU}_{1h} \cup \cdots \cup \overline{EU}_{nh} \cup \overline{EV}_{1h} \cup \cdots \cup \overline{EV}_{ph}\right) \\ &\geq 1 - p\mathrm{Prob}(\overline{EU}_{jh}) - n\mathrm{Prob}(\overline{EV}_{ih})\,. \end{aligned} \tag{11}$$

From Lemmas 1 and 2, (11) becomes

$$P_* \geq 1 - pQ\left(\sqrt{\frac{n}{(1+\sigma_b^2)m}}\right) - nQ\left(\sqrt{\frac{p}{(1+\sigma_b^2)m}}\right). \tag{12}$$

Letting $P_B = pQ\left(\sqrt{\frac{n}{(1+\sigma_b^2)m}}\right)$ and $P_A = nQ\left(\sqrt{\frac{p}{(1+\sigma_b^2)m}}\right)$, we get

$$P_* \geq 1 - P_B - P_A\,. \tag{13}$$

If z is large,

$$Q(z) \approx \exp\left\{ -\frac{z^2}{2} - \log z - \frac{1}{2}\log 2\pi \right\}, \tag{14}$$

which is quite accurate for $z > 3$. Using the approximation (14),

$$\begin{aligned} P_A &= \exp\left\{ \log p - \frac{n}{2(1+\sigma_b^2)m} - \frac{1}{2}\log\frac{n}{(1+\sigma_b^2)m} - \frac{1}{2}\log 2\pi \right\} \\ &= \exp\left\{ \log r + \log n - \frac{n}{2(1+\sigma_b^2)m} - \frac{1}{2}\log\frac{n}{(1+\sigma_b^2)m} - \frac{1}{2}\log 2\pi \right\} \\ &= \exp\left\{ \log n - \frac{n}{2(1+\sigma_b^2)m} - \frac{1}{2}\log\frac{n}{(1+\sigma_b^2)m} + \text{constant} \right\} \end{aligned} \tag{15}$$

Clearly, if $m < \frac{n}{2(1+\sigma_b^2)\log n}$, P_A tends zero as n tends infinity. Similarly, we can get that as $p \to \infty$ and $m < \frac{p}{2(1+\sigma_b^2)\log p}$, $P_B \to 0$. To sum up, for large n and p, If

$$m < \frac{\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)} \tag{16}$$

then $P_* \to 1$. That means if the number m of library pairs is less than $\frac{\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)}$, a library pair is with a very high chance to be a fixed point. So the capacity of BAM with multiplicative weight noise is equal to $\frac{\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)}$.

3.2 Error Correction

In this section, we will investigate the capacity of BAM with weight noise when the initial input is a noise version X_h^{noise} of a library pattern $\boldsymbol{X}_h$. Let X_h^{noise} contains ρn bit errors, where ρ is the input noise level. If

$$\boldsymbol{Y}_h = \text{sgn}(\tilde{W}\,\boldsymbol{X}_h^{noise}) \tag{17}$$

$$\boldsymbol{X}_h = \text{sgn}(\tilde{W}^T\,\boldsymbol{Y}_h) \tag{18}$$

$$\boldsymbol{Y}_h = \text{sgn}(\tilde{W}\,\boldsymbol{X}_h)\,, \tag{19}$$

then a noise version $\boldsymbol{X}_h^{noise}$ can successfully recall the correct the desire library pair $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$. Similarly, we hope that a noise version $\boldsymbol{Y}_h^{noise}$ of $\boldsymbol{Y}_h$ can successfully recall the correct the desire library pair $(\boldsymbol{X}_h, \boldsymbol{Y}_h)$. We will study under what condition of m, the probability of successful recall tends to one.

Define $EU_{j,h}^{noise}$ be the event that $\sum_i^n \tilde{w}_{ji} x_{ih}^{noise}$ is equal to y_{jh} (the j-th component of the library pattern $\boldsymbol{Y}_h$). Also, $\overline{EU}_{j,h}^{noise}$ is the complement event of $EU_{j,h}^{noise}$. Also, define $EV_{i,h}^{noise}$ be the event that $\sum_j^p \tilde{w}_{ji} y_{jh}^{noise}$ is equal to x_{ih} (the i-th component of the library pattern $\boldsymbol{X}_h$). Also, $\overline{EV}_{i,h}^{noise}$ is the complement event of $EV_{i,h}^{noise}$. With the above definition, we can following the proof of Lemma 1 to get the following two lemmas.

Lemma 3: The probability $Prob(\overline{EU}_{i,h}^{noise})$ *is approximately equal to*

$$Q\left(\sqrt{\frac{(1-2\rho)n}{(1+\sigma_b^2)m}}\right)$$

for $i = 1, \cdots, p$ *and* $h = 1, \cdots, m$.

Lemma 4: The probability $Prob(\overline{EV}_{i,h}^{noise})$ *is approximately equal to*

$$Q\left(\sqrt{\frac{(1-2\rho)p}{(1+\sigma_b^2)m}}\right)$$

for $i = 1, \cdots, p$ *and* $h = 1, \cdots, m$.

Define P_{**} be the probability that a noise version with ρ fraction of errors can recall the desired library pair. It is not difficult to show that

$$P_{**} \geq 1 - p\Big(\text{Prob}(\overline{EU}_{jh}) + \text{Prob}(\overline{EU}_{jh}^{noise})\Big) - n\Big(\text{Prob}(\overline{EV}_{ih} + \text{Prob}(\overline{EV}_{ih}^{noise}))\Big). \tag{20}$$

From Lemmas 1-4, To sum up, for large n and p, if

$$m < \frac{(1-2\rho)\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)} \tag{21}$$

then $P_{**} \to 1$. That means, when there are ρn (or ρp) bit errors in the initial input, the capacity of BAM with multiplicative weight noise is equal to $\frac{(1-2\rho)\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)}$.

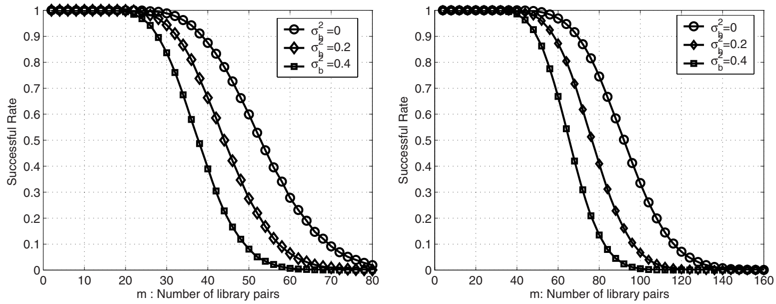
4 Simulation

4.1 Capacity

We consider two dimensions, 512 and 1024. For each m, we randomly generate 1000 sets of library pairs. The Kosko's rule is then used to encode the matrices. Afterwards, we add the multiplicative weight noise to the matrices. The variances σ_b^2 of weight noise are set to $0, 0.2, 0.4$.

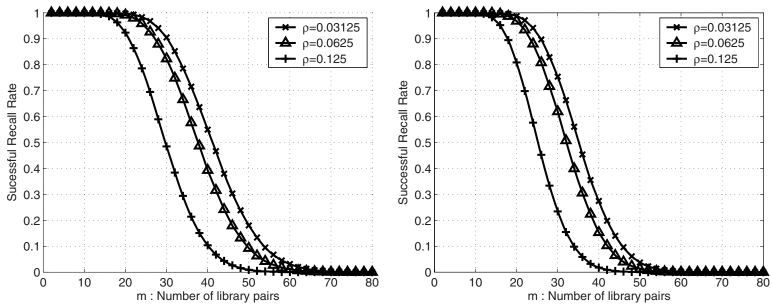
Figure 1 shows the percentage of a library pair being successfully stored. From the figure, as the variance of weight noise increases, the successful rate decreases. This phenomena agrees with our expectation.

From our analysis, i.e., (16), for $n = p = 512$, a BAM can store up to 41, 34, and 29 pairs for σ_b^2 equal to 0, 0.2 and 0.4, respectively. From Figure 1(a), all the corresponding successful rates are very large. Also, there are a sharply decreasing changes in successful for $\{m > 41, \sigma_b^2 = 0\}$, $\{m > 34, \sigma_b^2 = 0.2\}$, and $\{m > 29, \sigma_b^2 = 0.4\}$.



(a) $n = p = 512$ (b) $n = p = 1024$

Fig. 1. Successful rate of a library pair being a fixed point. (a) The dimension is 512. (b) The dimension is 1024. For each value of m, we generate 1000 sets of library pairs.



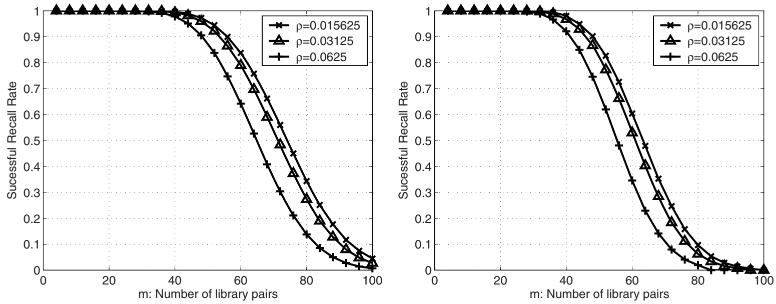
(a) weight noise level $\sigma_b^2 = 0.2$ (b) weight noise level $\sigma_b^2 = 0.4$

Fig. 2. Successful recall rate from a noise input. The dimension is 512. For each value of m, we generate 1000 sets of library pairs. For each library pattern, we generate 10 noise versions.

Similarly, from (16), for $n = p = 1024$, a BAM can store up to 73, 61, and 52 pairs for σ_b^2 equal to 0, 0.2 and 0.4, respectively. From Figure 1(b), all the corresponding successful rates are also large. Also, there are a sharply decreasing changes in successful for $\{m > 73, \sigma_b^2 = 0\}$, $\{m > 61, \sigma_b^2 = 0.2\}$, and $\{m > 52, \sigma_b^2 = 0.4\}$. To sum up, the simulation result is consistent with our analysis (16).

4.2 Error Correction

The dimension is 512. We consider two weight noise levels, $\sigma_b^2 = 0.2, 0.4$,and three input error levels, $\rho = 0.003125, 0.0625, 0.125$. For each m, we randomly generate 1000 sets of library pairs. The Kosko's rule is then used to encode the matrices. Afterwards, we add the multiplicative weight noise to the matrices. For each library pair, we generate ten noise versions. We then feed the noise versions as initial input and check whether the desire library can be recalled



(a) weight noise level $\sigma_b^2 = 0.2$ (b) weight noise level $\sigma_b^2 = 0.4$

Fig. 3. Successful recall rate from a noise input. The dimension is 1024. For each value of m, we generate 1000 sets of library pairs. For each library pattern, we generate 10 noise versions.

or not. Figures 2 and 3 shows the successful recall rate. From the figures, as the input error level ρ increases, the successful rate decreases. This phenomena agrees with our expectation.

From our analysis, i.e., (21), for the dimension $n = p = 512$ and weight noise level $\sigma_b^2 = 0.2$, a BAM can store up to 32, 30, and 26 pairs for the input error level ρ equal to 0.03125, 0.0625 and 0.125, respectively. From Figure 2(a), all the corresponding successful rates are large. Also, there are a sharply decreasing changes in successful recall rates for $\{m > 32, \rho = 0.03125\}$, $\{m > 30, \rho = 0.0625\}$, and $\{m > 26, \rho = 0.125\}$. For other weight noise levels and dimension, we obtained similar phenomena.

5 Conclusion

We have examined the statistical storage behavior of BAM with multiplicative weight noise. The capacity of a BAM is $m < \frac{\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)}$. When the number of library pairs is less that value, the chance of it being a fixed point is very high. Since we expect BAM has certain error correction ability, we have investigated the capacity of BAM with weight noise when the initial input is a noise version of a library pattern. We show that if $m < \frac{(1-2\rho)\min(n,p)}{2(1+\sigma_b^2)\log\min(n,p)}$, a noise version with ρn (or ρp) errors has a high chance to recall the desire library pairs. Computer simulations have been carried out The results presented here can be extended to Hopfield network. By adopting the approach set above, we can easily obtain the result in Hopfield network by replacing $\min(n, p)$ with n in the above equations.

Acknowledgement

The work is supported by the Hong Kong Special Administrative Region RGC Earmarked Grants (Project No. CityU 115606) and (Project No. CUHK 416806).

References

1. Kohonen, T.: Correlation matrix memories. IEEE Transaction Computer 21, 353–359 (1972)
2. Palm, G.: On associative memory. Biolog. Cybern. 36, 19–31 (1980)
3. Kosko, B.: Bidirectional associative memories. IEEE Trans. Syst. Man, and Cybern. 18, 49–60 (1988)
4. Leung, C.S.: Encoding method for bidirectional associative memory using projection on convex sets. IEEE Trans. Neural Networks 4, 879–991 (1993)
5. Leung, C.S.: Optimum learning for bidirectional associative memory in the sense of capacity. IEEE Trans. Syst. Man, and Cybern. 24, 791–796 (1994)
6. Wang, Y.F., Cruz, J.B., Mulligan, J.H.: Two coding strategies for bidirectional associative memory. IEEE Trans. Neural Networks 1, 81–92 (1990)
7. Lenze, B.: Improving leung's bidirectional learning rule for associative memories. IEEE Trans. Neural Networks 12, 1222–1226 (2001)
8. Shen, D., Cruz, J.B.: Encoding strategy for maximum noise tolerance bidirectional associative memory. IEEE Trans. Neural Networks 16, 293–300 (2005)
9. Leung, C.S., Chan, L.W.: The behavior of forgetting learning in bidirectional associative memory. Neural Computation 9, 385–401 (1997)
10. Leung, C.S., Chan, L.W., Lai, E.: Stability and statistical properties of second-order bidirectional associative memory. IEEE Transactions on Neural Networks 8, 267–277 (1997)
11. Wang, B.H., Vachtsevanos, G.: Storage capacity of bidirectional associative memories. In: Proc. IJCNN 1991, Singapore, pp. 1831–1836 (1991)
12. Haines, K., Hecht-Nielsen, R.: A bam with increased information storage capacity. In: Proc. of the 1988 IEEE Int. Conf. on Neural Networks, pp. 181–190 (1988)
13. Amari, S.: Statistical neurodynamics of various versions of correlation associative memory. In: Proc. of the 1988 IEEE Int. Conf. on Neural Networks, pp. 181–190 (1988)
14. Burr, J.: Digital neural network implementations. In: Neural Networks, Concepts, Applications, and Implementations, vol. III, Prentice Hall, Englewood Cliffs, New Jersey (1991)
15. Holt, J., Hwang, J.-N.: Finite precision error analysis of neural network hardware implementations. IEEE Transactions on Computers 42(3), 281–290 (1993)
16. Lam, P.M., Leung, C.S., Wong, T.T.: Noise-resistant fitting for spherical harmonics. IEEE Transactions on Visualization and Computer Graphics 12(2), 254–265 (2006)
17. Bernier, J.L., Ortega, J., Rodriguez, M.M., Rojas, I., Prieto, A.: An accurate measure for multilayer perceptron tolerance to weight deviations. Neural Processing Letters 10(2), 121–130 (1999)
18. Bernier, J.L., Diaz, A.F., Fernandez, F.J., Canas, A., Gonzalez, J., Martin-Smith, P., Ortega, J.: Assessing the noise immunity and generalization of radial basis function networks. Neural Processing Letters 18(1), 35–48 (2003)
19. Sripad, A., Snyder, D.: Quantization errors in floating-point arithmetic. IEEE Transactions on Speech, and Signal Processing 26, 456–463 (1978)