Alpha-Beta Bidirectional Associative Memories Based Translator

María Elena Acevedo-Mosqueda, Cornelio Yáñez-Márquez, and Itzamá López-Yáñez

Centro de Investigación en Computación, IPN, Mexico City

Summary
Bidirectional Associative Memories (BAM) based on Kosko's model are implemented through iterative algorithms and present stability problems. Also, these models along with other models based on different methods, have not been able to perfectly recall all trained patterns. In this paper we present an English-Spanish / Spanish-English translator based on a new BAM model denominated Alpha-Beta BAM, whose process is non iterative and does not require to find stable states. The translator recalls the whole set of learned patterns, even when the presented word is incomplete.

Key words:
Bidirectional Associative Memories, Alpha-Beta Associative Memories, perfect recall, transaltor.

Introduction

The first bidirectional associative memory (BAM), introduced by Kosko [1], was the base of many models presented later. Some of this models substituted the learning rule for an exponential rule [2-4]; others used the method of multiple training and dummy addition in order to reach a greater number of stable states [5], trying to eliminate spurious states. With the same purpose, linear programming techniques [6], the descending gradient method [7-8] have been used, besides genetic algorithms [9] and BAM with delays [10-11]. Other models of non iterative bidirectional associative memories exist, such as morphological BAM [12] and Feedforward BAM [13]. All these models have arisen to solve the problem of low pattern recall capacity shown by the BAM of Kosko; however, none has been able to recall all the trained patterns. Also, these models demand the fulfillment of some specific conditions, such as a certain Hamming distance between patterns, solvability by linear programming, orthogonality between patterns, among other.

The model of bidirectional associative memory presented in this paper is based on the Alpha-Beta associative memories [14], is not an iterative process, and does not present stability problems. Pattern recall capacity of the Alpha-Beta BAM is maximal, being $2^{\min(n,m)}$, where n and m are the input and output pattern dimensions, respectively. Also, it always shows perfect pattern recall without imposing any condition.

Manuscript received May 5, 2006.
Manuscript revised May 25, 2006.

In section 2 we present the Alpha-Beta autoassociative memories, base of our new model of BAM, and the theoretical sustentation of Alpha-Beta BAM. In section 3 the model is applied to words translation. Conclusions follow in section 4.

2. Alpha-Beta Bidireccional Associative Memories

In this section the proposed model of bidirectional associative memory is presented. However, since it is based on the Alpha-Beta autoassociative memories, a summary of this model will be given before presenting the Alpha-Beta BAM model.

2.1 Alpha-Beta Associative Memories

Basic concepts about associative memories were established three decades ago in [15-17], nonetheless here we use the concepts, results and notation introduced in the Yáñez-Márquez's PhD Thesis [14]. An associative memory **M** is a system that relates input patterns, and outputs patterns, as follows: $\mathbf{x}\rightarrow\mathbf{M}\rightarrow\mathbf{y}$ with **x** and **y** the input and output pattern vectors, respectively. Each input vector forms an association with a corresponding output vector. For k integer and positive, the corresponding association will be denoted as $(\mathbf{x}^k, \mathbf{y}^k)$. Associative memory **M** is represented by a matrix whose ij-th component is m_{ij}. Memory **M** is generated from an *a priori* finite set of known associations, known as the fundamental set of associations.

If μ is an index, the fundamental set is represented as: $\left\{\left(x^\mu, y^\mu\right) \mid \mu = 1,2,\ldots,p\right\}$ with p the cardinality of the set. The patterns that form the fundamental set are called fundamental patterns. **M** is *autoassociative* if it holds that $\mathbf{x}^\mu = \mathbf{y}^\mu \ \forall \mu \in \{1,\ldots,p\}$, otherwise it is *heteroassociative*; in this latter case it is possible to establish that $\exists \mu \in \{1,\ldots,p\}$ for which $x^\mu \neq y^\mu$. A distorted version of a pattern $\mathbf{x}^k$ to be recuperated will be denoted as $\tilde{x}^k$. If when feeding a distorted version of x^ϖ with $\varpi = \{1,2,\ldots,p\}$ to an associative memory **M**, it happens that the output corresponds exactly to the associated pattern y^{ϖ}, we say that recall is perfect.

The Alpha-Beta associative memories are able to operate in two different modes: *max* and *min*. The operator α is useful at the learning phase, and the operator β is the basis for the pattern recall phase.

The heart of the mathematical tools used in the Alpha-Beta model, are two binary operators designed specifically for these memories. These operators are defined as follows: First, we define the sets A={0,1} and B={0,1,2}, then the operators α and β are defined as is shown in table 1.

Table 1. α and β binary operators

α: AxA→B

x	y	$\alpha(x,y)$
0	0	01
0	1	00
1	0	10
1	1	01

β: BxA→A

x	y	$\beta(x,y)$
00	0	0
00	1	0
01	0	0
01	1	1
10	0	1
10	1	1

The sets A and B, the α and β operators, along with the usual $\wedge$ (minimum) and $\vee$ (maximum) operators, form the algebraic system (A,B,α,β,$\wedge$,$\vee$) which is the mathematical basis for the Alpha-Beta associative memories.

Below are shown some characteristics of Alpha-Beta autoassociative memories:

1. The fundamental set takes the form $\{(\mathbf{x}^{\mu},\mathbf{x}^{\mu}) \mid \mu=1,2,...,p\}$.
2. Both input and output fundamental patterns are of the same dimension, denoted by n.
3. The memory is a square matrix, for both modes, $\mathbf{V}$ and $\mathbf{\Lambda}$. If $\mathbf{x}^{\mu} \in A^n$ then

$$v_{ij} = \bigvee_{\mu=1}^{p} \alpha\left(x_i^{\mu}, x_j^{\mu}\right) \quad (1)$$

and $$\lambda_{ij} = \bigwedge_{\mu=1}^{p} \alpha\left(x_i^{\mu}, x_j^{\mu}\right) \quad (2)$$

and according to α: A x $A \rightarrow B$, we have that v_{ij} and $\lambda_{ij} \in B$, $\forall i \in \{1, 2, ..., n\}$ and $\forall j \in \{1, 2, ..., n\}$.

In the recalling phase, when a pattern $\mathbf{x}^{\mu}$ is presented to memories $\mathbf{V}$ and $\mathbf{\Lambda}$, the i-th components of recalled patterns are:

$$\left(\mathbf{V}\Delta_{\beta}\mathbf{x}^{\omega}\right)_i = \bigwedge_{j=1}^{n} \beta(v_{ij}, x_j^{\omega}) \quad (3)$$

and $$\left(\mathbf{\Lambda}\nabla_{\beta}\mathbf{x}^{\omega}\right)_i = \bigvee_{j=1}^{n} \beta(\lambda_{ij}, x_j^{\omega}) \quad (4)$$

2.2 Alpha-Beta Bidirectional Associative Memories

The model proposed in this paper has been named Alpha-Beta BAM since Alpha-Beta associative memories, both *max* and *min*, play a central role in the model design. However, before going into detail over the processing of an Alpha-Beta BAM, we will define the following.

In this work we will assume that Alpha-Beta associative memories have a fundamental set denoted by $\{(\mathbf{x}^{\mu}, \mathbf{y}^{\mu}) \mid \mu = 1, 2, \ldots, p\}$ $\mathbf{x}^{\mu} \in A^n$ and $\mathbf{y}^{\mu} \in A^m$, with $A = \{0, 1\}$, $n \in \mathbf{Z}^+$, $p \in \mathbf{Z}^+$, $m \in \mathbf{Z}^+$ and $1 < p \leq \min(2^n, 2^m)$.

Definition 1 (One-Hot) *Let the set A be A* = {0, 1} *and* $p \in \mathbf{Z}^+$, $p > 1$, $k \in \mathbf{Z}^+$, *such that* $1 \leq k \leq p$. *The k-th one-hot vector of p bits is defined as vector* $h^k \in A^p$ *for which it holds that the k-th component is* $h_k^k = 1$ *and the ret of the components are* $h_j^k = 0$, $\forall j \neq k$, $1 \leq j \leq p$.

Remark 1 In this definition, the value p = 1 is excluded since a one-hot vector of dimension 1, given its essence, has no reason to be.

Definition 2 (Zero-Hot) *Let the set A be A* = {0, 1} *and* $p \in \mathbf{Z}^+$, $p > 1$, $k \in \mathbf{Z}^+$, *such that* $1 \leq k \leq p$. *The k-th zero-hot vector of p bits is defined as vector* $\overline{\mathbf{h}}^k \in A^p$ *for which it holds that the k-th component is* $h_k^k = 0$ *and the ret of the components are* $h_j^k = 1$, $\forall j \neq k$, $1 \leq j \leq p$.

Remark 2 In this definition, the value p = 1 is excluded since a zero-hot vector of dimension 1, given its essence, has no reason to be.

Definition 3 (Expansion vectorial transform) *Let the set A be A* = {0, 1} *and* $n \in \mathbf{Z}^+$,*y* $m \in \mathbf{Z}^+$. *Given two arbitrary vectors* $\mathbf{x} \in A^n$ *and* $\mathbf{e} \in A^m$, *the expansion vectorial transform of order m,* $\tau^e : A^n \rightarrow A^{n+m}$, *is defined as* $\tau^e(\mathbf{x}, \mathbf{e}) = \mathbf{X} \in A^{n+m}$, *a vector whose components are:* $X_i = x_i$ *for* $1 \leq i \leq n$ *and* $X_i = e_i$ *for* $n + 1 \leq i \leq n + m$.

Definition 4 (Contraction vectorial transform) *Let the set A be A* = {0, 1} *and* $n \in \mathbf{Z}^+$,*y* $m \in \mathbf{Z}^+$ *such that* $1 \leq m < n$. *Given one arbitrary vector* $\mathbf{X} \in A^{n+m}$, *the contraction vectorial transform of order m,* $\tau^c : A^{n+m} \rightarrow A^m$, *is defined as* $\tau^c(\mathbf{X}, m) = \mathbf{c} \in A^m$, *a vector whose components are:* $c_i = X_{i+n}$ *for* $1 \leq i < m$.

In both directions, the model is made up by two stages, as shown in figure 1.

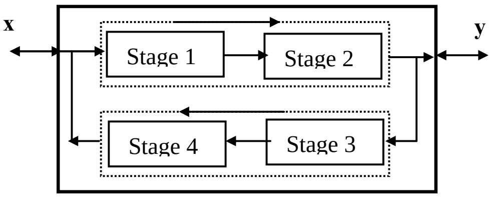

Fig. 1. Graphical schematics of of the Alpha-Beta bidirectional associative memory.

For simplicity, first will be described the process necessary in one direction, in order to later present the complementary direction which will give bidirectionality to the model (see figure 2).

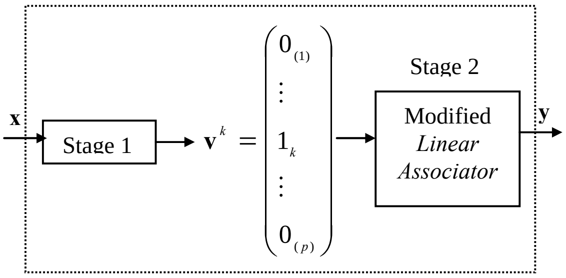

Fig. 2. Schematics of the process done in the direction from **x** to **y**. Here, only Stage 1 and Stage 2 are shown. Notice that $v_k^k = 1$, $v_i^k = 0 \ \ \forall i \neq k$, $1 \le i \le p,\ 1 \le k \le p$

The function of Stage 2 is to offer a $\mathbf{y}^k$ as ouput (k = 1,...,p) given a $\mathbf{x}^k$ as input.

Now we assume that as input to Stage 2 we have one element of a set of p orthonormal vectors. Recall that the *Linear Associator* has perfect recall when it works with orthonormal vectors. In this work we use a variation of the *Linear Associator* in order to obtain $\mathbf{y}^k$, parting from a *one-hot* vector $\mathbf{v}^k$ in its k-th coordinate.

For the construction of the modified Linear Associator, its learning phase is skipped and a matrix **M** representing the memory is built. Each column in this matrix corresponds to each output pattern $\mathbf{y}^\mu$. In this way, when matrix **M** is operated with a one-hot vector $\mathbf{v}^k$, the corresponding $\mathbf{y}^k$ will always be recalled.

The task of Stage 1 is: given a $\mathbf{x}^k$ or a noisy version of it ($\tilde{\mathbf{x}}^k$), the *one-hot* vector $\mathbf{v}^k$ must be obtained without ambiguity and with no condition. In its learning phase, stage 1 has the following algorithm:

1. For $1 \le k \le p$ do expansion: $\mathbf{X}^k = \tau^e(\mathbf{x}^k, \mathbf{h}^k)$
2. For $1 \le i \le n$ and $1 \le j \le n$: $v_{ij} = \bigvee_{\mu=1}^{p} \alpha\left(X_i^\mu, X_j^\mu\right)$
3. For $1 \le k \le p$ do expansion: $\overline{\mathbf{X}}^k = \tau^e(\mathbf{x}^k, \overline{\mathbf{h}}^k)$
4. For $1 \le i \le n$ and $1 \le j \le n$:

$$\lambda_{ij} = \bigwedge_{\mu=1}^{p} \alpha\left(\overline{X}_i^\mu, \overline{X}_j^\mu\right)$$

5. Create modified Linear Associator.

$$\mathbf{LAy} = \begin{bmatrix} y_1^1 & y_1^2 & \cdots & y_1^p \\ y_2^1 & y_2^2 & \cdots & y_2^p \\ \vdots & \vdots & \cdots & \vdots \\ y_n^1 & y_n^2 & \cdots & y_n^p \end{bmatrix}$$

Recall phase is described through the following algorithm:

1. Present, at the input to Stage 1, a vector from the fundamental set $\mathbf{x}^\mu \in A^n$, for some index $\mu \in \{1, ..., p\}$.
2. Build vector: $\mathbf{u} = \sum_{i=1}^{p} \mathbf{h}^i$
3. Do expansion: $\mathbf{F} = \tau^e(\mathbf{x}^\mu, \mathbf{u}) \in A^{n+p}$
4. Obtain vector: $\mathbf{R} = \mathbf{V}\Delta_\beta \mathbf{F} \in A^{n+p}$
5. Do contraction: $\mathbf{r} = \tau^c(\mathbf{R}, n) \in A^p$

   If **r** is one-hot vector, it is assured that $k = \mu$, $\mathbf{y}^\mu = \mathbf{LAy} \cdot \mathbf{r}$. STOP. Else:
6. For $1 \le i \le p$: $w_i = u_i$ -1
7. Do expansion: $\mathbf{G} = \tau^e(\mathbf{x}^\mu, \mathbf{w}) \in A^{n+p}$
8. Obtain a vector: $\mathbf{S} = \Lambda \nabla_\beta \mathbf{G} \in A^{n+p}$
9. Do contraction: $\mathbf{s} = \tau^c(\mathbf{S}^\mu, n) \in A^p$
10. If **s** is zero-hot vector then it is assured that $k = \mu$, $\mathbf{y}^\mu = \mathbf{LAy} \cdot \overline{\mathbf{s}}$, where $\overline{\mathbf{s}}$ is the negated vector of **s.** STOP. Else:
12. Do operation $\mathbf{r} \wedge \overline{\mathbf{s}}$, where $\wedge$ is the symbol of the logical AND operator, so $\mathbf{y}^\mu = \mathbf{LAy} \cdot (\mathbf{r} \wedge \overline{\mathbf{s}})$. STOP.

The process in the contrary direction, which is presenting pattern $\mathbf{y}^k$ (k = 1,...,p) as input to the Alpha-Beta BAM and obtaining its corresponding $\mathbf{x}^k$, is very similar to the one described above. The task of Stage 3 is to obtain a one-hot vector $\mathbf{v}^k$ given a $\mathbf{y}^k$. Stage 4 is a modified Linear Associator built in similar fashion to the one in Stage 2.

4. Translator

The programming language used to implement the code of the translator is Visual C++ 6.0. This software has the ability to translate words from English, to Spanish, and visceversa. For the learning phase two text files were used, containing 120 words in English and Spanish, respectively.

With these two files, the Alpha-Beta Bidirectional Associative Memory is built (see figure 3).

Fig. 3. The Alpha-Beta Bidirectional Associative Memory is built by associating 120 words in Spanish with 120 words in English, which were container in two text files.

The learning phase lasts, approximately, 1 minute and 6 seconds, when the program was run on a 2.8 GHz Pentium 4 Sony VAIO Laptop.

Once the Alpha-Beta BAM has been built, a word is written during the recalling phase, either in English or Spanish, and the translation mode is selected. Immediately the word appears in the corresponding language. En example of this can be seen in figure 4, where the word to be translated was "accuracy" and its corresponding translation to Spanish is "exactitud".

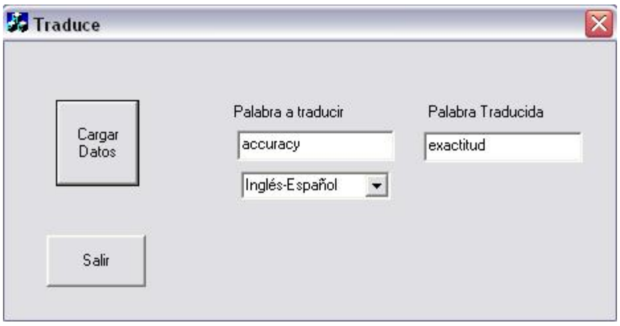

Fig. 4. The word to be translated is written —in this example it was "accuracy"— and the translation mode is selected; immediately, its corresponding Spanish word appears, which in this case is "exactitud".

The translator offers other advantages as well. For instance, suppose only part of the word is entered, say "accur" instead of "accuracy"; the program will give as result "exactitud" (see figure 5).

Fig. 5. The translator recalls perfectly the word associated to "accuracy" even though it is not complete.

Now, assume that instead of writing a "y", an "I" is keyed by mistake. The result is shown in figure 6.

Fig. 6. Even when there is a writing mistake and a "y" is exchanged by an "i", the program recalls ina perfect manner the word "exactitud".

We can see that in this example, a writing mistake, which at pattern level would be interpreted as introducing a noisy pattern, does not impede the performance of the translator.

The advantages presented by the translator reflect the advantages of the Alpha-Beta BAM model. These memories are immune to a certain amount and kinds of noise, properties which still have not been characterized.

Besides these kinds of tests, the full set of 120 words in English was entered, perfectly recalling their corresponding Spanish translations. Also, the 120 words in Spanish were written and, in a perfect manner and without ambiguities, the translator showed the corresponding words in English.

Conclusions

Only a portion of the theoretical basis of Alpha-Beta associative memories, foundation of the Alpha-Beta BAM, was presented. The structure of an Alpha-Beta BAM was

presented, along with the algorithm necessary to implement them.
The cases which the translator can manage, clearly exemplify the advantages shown by the Alpha-Beta BAM. The most important characteristic of an Alpha-Beta BAM is that it can recall, in a perfect manner, all trained patterns: the translator is able to perfectly recall the 240 words contained within it.
The translator was able to recall the word corresponding to an incomplete or misspelled word; this means that Alpha-Beta BAM are capable of recalling clean patterns from noisy patterns.
The kind and amount of noise that an Alpha-Beta BAM can manage has not been characterized yet.

Acknowledgments

The authors would like to thank the Instituto Politécnico Nacional (Secretaría Académica, COFAA, SIP, and CIC), the CONACyT, and SNI for their economical support to develop this work.

References

[1] Kosko, B., Bidirectional associative memories, *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 18, no. 1, (1988) 49-60.
[2] Jeng, Y.-J.; Yeh, C.-C. & Chiveh, T. D., Exponential bidirectional associative memories**,** *Electronics Letters*, Vol. 26, Issue 11, (1990) 717-718.
[3] Wang, W.-J. & Lee, D.-L., Modified exponential bidirectional associative memories, *Electronics Letters*, Vol. 28, Issue 9, (1992). 888-890.
[4] Chen, S., Gao, H. & Yan, W., Improved exponential bidirectional associative memory, *Electronics Letters*, Vol. 33, Issue 3, (1997) 223-224.
[5] Wang, Y.-F., Cruz J. B., Jr. & Mulligan, Jr., Two coding strategies for bidirectional associative memory, *IEEE Transactions on Neural Networks*, Vol. 1, Issue 1, (1990) 81-92.
[6] Wang, Y.-F., Cruz J. B., Jr. & Mulligan, Jr., Guaranteed recall of all training pairs for bidirectional associative memory, *IEEE Transactions on Neural Networks*, Vol. 1, Issue 6, (1991) 559-56.
[7] Perfetti, R., Optimal gradient descent learning for bidirectional associative memories, *Electronics Letters*, Vol. 29, Issue 17, (1993) 1556-1557.
[8] Zheng, G., Givigi, S. N. & Zheng, W., A New Strategy for Designing Bidirectional Associative Memories, *Lecture Notes in Computer Science*, Springer-Verlag, Vol. 3496, (2005) 398-403.
[9] Shen, D. & Cruz, J. B., Jr., Encoding strategy for maximum noise tolerance bidirectional associative memory, *IEEE Transactions on Neural Networks*, Vol. 16, Issue 2, (2005) 293-300.
[10] Arik, S., Global asymptotic stability analysis of bidirectional associative memory neural networks with time delays, *IEEE Transactions on Neural Networks*, vol. 16, no. 3 (2005) 580-585.
[11] Park, J., Robust stability of bidirectional associative memory neural networks with time delays, *Physics Letters A*, vol. 349, (2006) 494-499.
[12] Ritter, G. X., Diaz-de-Leon, J. L. & Sussner, P., Morphological bidirectional associative memories, *Neural Networks*, Vol. 12, (1999) 851-867.
[13] Wu, Y. & Pados, D. A., A feedforward bidirectional associative memory, *IEEE Transactions on Neural Networks*, Vol. 11, Issue 4, (2000) 859-866.
[14] Yáñez-Márquez, C.: Associative Memories Based on Order Relations and Binary Operators (In Spanish). PhD Thesis. Center for Computing Research, México (2002)
[15] Kohonen, T.: Self-Organization and Associative Memory. Springer-Verlag, Berlin Heidelberg New York (1989).
[16] Hassoun, M. H.: Associative Neural Memories. Oxford University Press, New York (1993).
[17] Kohonen, T.: Correlation Matrix Memories. IEEE Transactions on Computers. 21(4). (1972) 353-359

María Elena Acevedo Mosqueda. Received her BS degree in Engineering with specialization in Computing from the Escuela Superior de Ingeniería Mecánica y Eléctrica (ESIME) at the National Politechnics Institute (IPN) in 1996. She has been teaching at ESIME since 1994. She received her MSc degree, with specialization in Computing, from the Centro de Investigación y de Estudios Avanzados (CINVESTAV) in 2001. Currently, she is undergoing her PhD studies at the Center for Computing Research of the IPN. Her main area of research is bidirectional associative memories.

Cornelio Yáñez Márquez. Received his BS degree in Physics and Mathematics from the Escuela Superior de Física y Matemáticas at IPN in 1989; the MSc degree in Computing Engineering from the CINTEC-IPN in 1995, and the PhD from the Centro de Investigación en Computación of the Instituto Politécnico Nacional (CIC-IPN) in México in 2002, receiving the Lázaro Cárdenas Award 2002. Currently, he is a titular professor at the CIC-IPN. His research interests include associative memories, mathematical morphology and neural networks.

Itzamá López Yáñez. Received his BS degree in Information Systems Engineering at the Monterrey Institute of Technology and Superior Studies (ITESM) Campus Sonora Norte in 2003. Currently, he is undergoing his MSc studies at CIC-IPN.