A Visualization of Weight Solutions of a Neural Network Model of Associative Memory

Akira Imada

Dokuz Eylül University
Computer Engineering Department
Bornova, Izmir, Turkey
akira@cs.deu.edu.tr

Abstract

We apply evolutionary computations to the Hopfield model of associative memory. In the model, a number of patterns can be stored in the network as attractors if synaptic weights are determined appropriately. Thus far, we have explored the weight space and found many different such solutions of weight configurations. One of our goals of this study is to learn the number and distribution of these weight solutions, which is still an open problem. To address this issue, we test a method of visualizing these solutions in high-dimensional weight space.

I. INTRODUCTION

Although the fully connected neural network model of associative memory, such as the Hopfield's model, is nowadays said to be impractical, many interesting issues from physics point of view are still unknown. The number and distribution of attractors in synaptic weight space are one of these examples. This paper is a step toward learning the distribution of these weight solutions as a function of number of patterns to be stored.

Using evolutionary computations, we have found many different solutions that give a fully connected neural network a function of associative memory. Especially, the Breeder Genetic Algorithm (BGA) [1] gave us a wide variety of such solutions that store almost an upper bound of storage capacity of a network [2], which are in general a very difficult to be searched for.

Each solution is a point in weight space of high dimensionality, and they are visualized by being mapped into points in the two-dimensional (2D) space with the distance relation among the original points remaining as much as possible. Shine et al. [3] and Collins [4] argued such a technique, and Collins calls this technique *"Sammon Mapping"* after Sammon [5] who proposed this technique originally. (Shine et al. [3] call this *"Distance Map"*.) Since the technique is an optimization problem, we can also employ a genetic algorithm (GA) to solve this problem.

In the following two subsections, associative memory model and Sammon mapping are described more in detail.

Associative Memory

Associative memory is a dynamical system which has a number of stable states with a domain of attraction around them [6]. If the system starts at any state in the domain, it will converge to the stable state. In 1982, Hopfield [7] proposed a fully connected neural network model of associative memory in which information is stored by being distributed among neurons, and retrieved from dynamically relaxed neurons' states. In the model, some of the appropriate configurations of synaptic weights give the network a function of associative memory.

The Hopfield model consists of N neurons and N^2 synapses. Each neuron state is either active ($+1$) or quiescent (-1). When an arbitrary N-bit *bipolar pattern*, a sequence of $+1$ and -1, is given to the network as an initial state, the dynamical behavior of neurons' states afterwards are characterized by the strengths of the N^2 synapses. The synaptic strengths are called *weights*, and the weight from neuron j to neuron i is denoted as w_{ij} in this paper. Provided the synaptic weights are determined appropriately, network can store some number of patterns as attractors. Hopfield employed the so-called Hebb's [8] rule to prescribe the weights [1] That is, to store p bipolar patterns ξ^ν:

$$\xi^\nu = (\xi_1^\nu, \cdots, \xi_N^\nu), \qquad \nu = 1, \cdots, p$$

the weight values are determined as:

$$w_{ij} = \frac{1}{N} \sum_{\nu=1}^{p} \xi_i^\nu \xi_j^\nu \quad (i \neq j), \quad w_{ii} = 0. \tag{1}$$

An instantaneous state of a neuron is updated asynchronously (one neuron at a time) as:

$$s_i(t+1) = sgn\left(\sum_{j \neq i}^{N} w_{ij} s_j(t) \right),$$

where $s_i(t)$ is a state of the i-th neuron at time t. If an initial state converges to one of the stored patterns ξ^ν as an equilibrium state, then the pattern is said to be *recalled*. Furthermore, if an initial state chosen from the stored patterns remains unchanged from the beginning, then the pattern is said to be stored as a *fixed point*.

In analyzing the Hopfield model, there have been basically two different approaches: one is to explore *pattern space* searching for attractors under a specific weight configuration, and the other is to explore *weight space* searching for an appropriate weight configuration that stores a given set of patterns. To be more specific, the former is an analysis of the

[1] The rule was at first advocated by Hebb [8] in 1949 and later formalized by Cooper [9] in 1973.

Hamiltonian energy as a function of all the possible configurations of bipolar pattern to be given to the network, where synaptic weights are pre-specified using a learning algorithm, usually the Hebb's rule, so that the network stores a set of p given patterns. In this context, the model for $p = 1$ corresponds to the Mattis [10] model of spin-glass, in which the Hamiltonian energy has two minima, while the model for infinitely large p corresponds to the Sherrington-Kirkpatrick [11] model, in which the synaptic weights become Gaussian random variables. Analyses of the former type have been made in between these two extreme cases (see Amit [12]). The latter analysis, on the other hand, was addressed by Gardner [13]. She discussed the optimal weight configurations for a *fixed* number of given patterns in terms of the volume of the solutions in weight space, suggesting that the volume shrinks to vanish when p approaches to $2N$. In short, the former approach searches for the optimal pattern configurations which minimize the Hamiltonian energy *in pattern space* with the weights being fixed, while the latter searches for the weight configurations *in weight space* that optimally store a set of given fixed patterns.

In this paper, our concern is on weight space where some points give a network a capability to store a fixed set of patterns, and we call these points solutions.

Sammon Mapping

As Collins [4] wrote, the dimension reduction has been an important technique for visualization of the space of high dimensionality. The Sammon Mapping [5] is one of these techniques. This enables us to map a set of $\mathcal{N}$ points in n-dimensional space to 2D location data so that the distance information is preserved as much as possible, or as Shine et al. [3] wrote *"so that the n-dimensional distances are approximated by 2 dimensional distances with a minimal error."* This problem is an optimization problem.

Shine et al. [3] and Collins [4] proposed a method to solve this problem by a GA, as follows. First, the distance matrix whose entries are Euclidean distances between all possible pairs of $\mathcal{N}$ points in the n-dimensional space is calculated. Then tentative $\mathcal{N}$ points in the 2-dimensional space are determined representing the original $\mathcal{N}$ points in the n-dimensional space. The distance matrix of these $\mathcal{N}$ points in the 2D space is also calculated, which then will be subtracted from the original n-dimensional distance matrix, yielding an error matrix. A GA is used to minimize this error matrix.

II. EXPERIMENT

Since our experiments of the fully-connected neural network model of associative memory have been carried out on networks with 49 neurons, the weight space is $49^2 = 2401$ dimensional. Hence, we restricted the dimensionality to 2401 that is to be reduced to 2 in this paper. Given $\mathcal{N}$ points in the 2401 dimensional space

$$X^1, \cdots, X^{\mathcal{N}}$$

where each point X^k is expressed by 2401 coordinates as

$$X^k = (z_1^k, \cdots, z_{2401}^k).$$

Then the square distance between the m-th point and the n-th point is calculated as

$$R^{mn} = \sum_{i=1}^{2401} (z_i^m - z_i^n)^2.$$

The values for all possible combination of m and n construct a distance matrix. Since the matrix is symmetric with zero diagonal elements, we use the lower triangle elements ($m < n$).

Then we generate $\wp$ sets of the 2D $\mathcal{N}$ points at random, i.e.,

$$\begin{array}{l} \Psi^1(1), \cdots, \Psi^{\mathcal{N}}(1) \\ \Psi^1(2), \cdots, \Psi^{\mathcal{N}}(2) \\ \cdots \\ \Psi^1(\wp), \cdots, \Psi^{\mathcal{N}}(\wp) \end{array}$$

where the k-th point of the i-th set is represented as:

$$\Psi^k(i) = (\zeta_1^k, \zeta_2^k).$$

Thus the i-th set of these $\mathcal{N}$ points has its distance matrix whose elements are

$$r^{mn}(i) = \sum_{i=1}^{2} (\zeta_i^m - \zeta_i^n)^2.$$

The objective function of the i-th sets can be defined as

$$f(i) = \sum_{m<n} (r^{mn}(i) - R^{mn}).$$

Starting with a random configuration of $\mathcal{N}$ points in the 2D space, the GA corrects these points generation by generation applying crossover and mutation[2] to 2D coordinates. The correction is repeated until the error converges to an acceptable minimum.

III. RESULTS AND DISCUSSION

In exploring weight space to search for the solutions, w_{ij} can take an arbitrary real value, there are infinite number of equivalent configurations which differ only by scaling factor. In other words, for any scaling factor κ, κw_{ij} works exactly in the same way as w_{ij} in updating neuron states (see the Equation (1)). These solutions construct hyper-lines emanating from the origin. Therefore, as Gardner [13] did, all the solutions are normalized on the surface of hypersphere of radius one whose center is at the origin, by regarding the solutions on a hyper-line as the cross point of the line and the sphere surface.

In the following two subsections, the results of two different experiments are shown. First, in order to see how the Sammon mapping works in our problem, a set of random points are picked up each from three different categories on the hypersphere surface of the 2401 dimensionality, and then observed how they are made visible in the 2D space. Next, the method is applied to our solutions for the associative memory network.

[2] We employ uniform crossover [14] and BGA mutation [1].

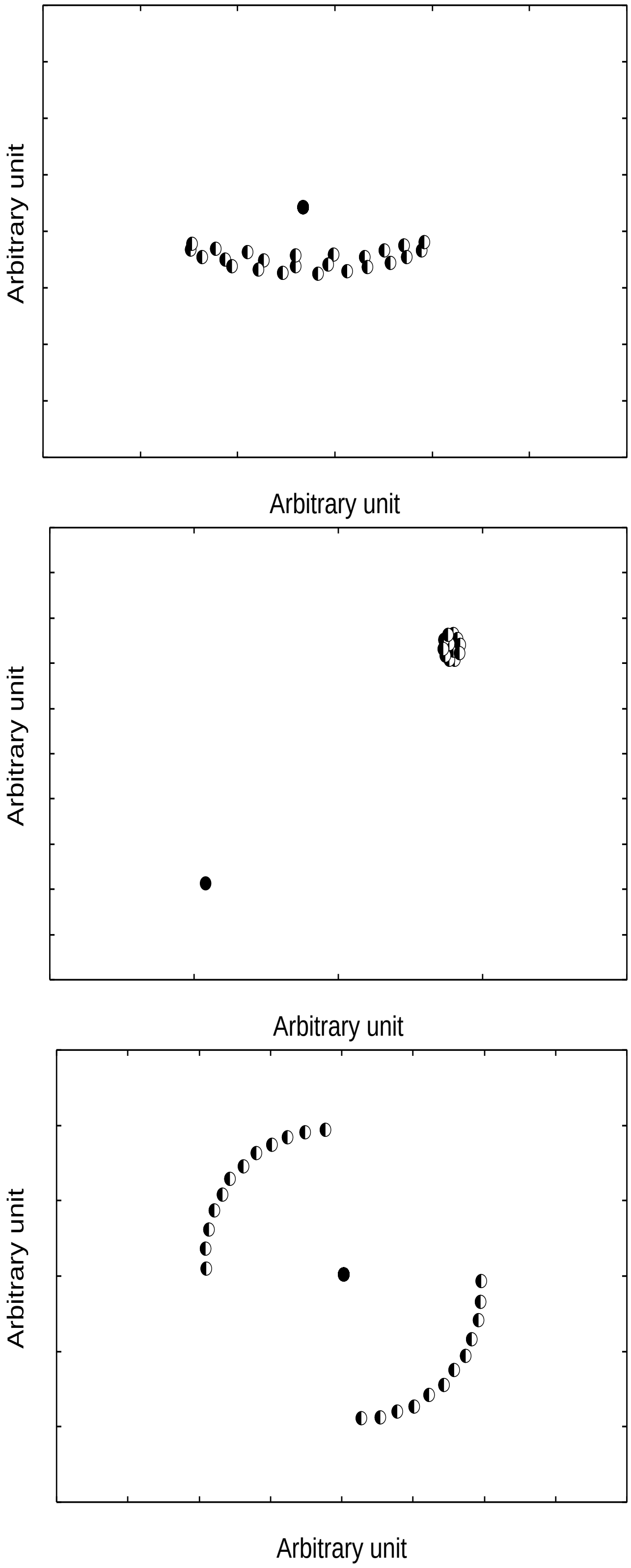

Figure 1: The 2D points mapped from three different regions on hypersphere of the 2401 dimensional space: All positive coordinates region (top); a subset of the region above (middle); and two regions of the symmetric positions (bottom). Filled-in circle • indicates the origin. See text more in detail.

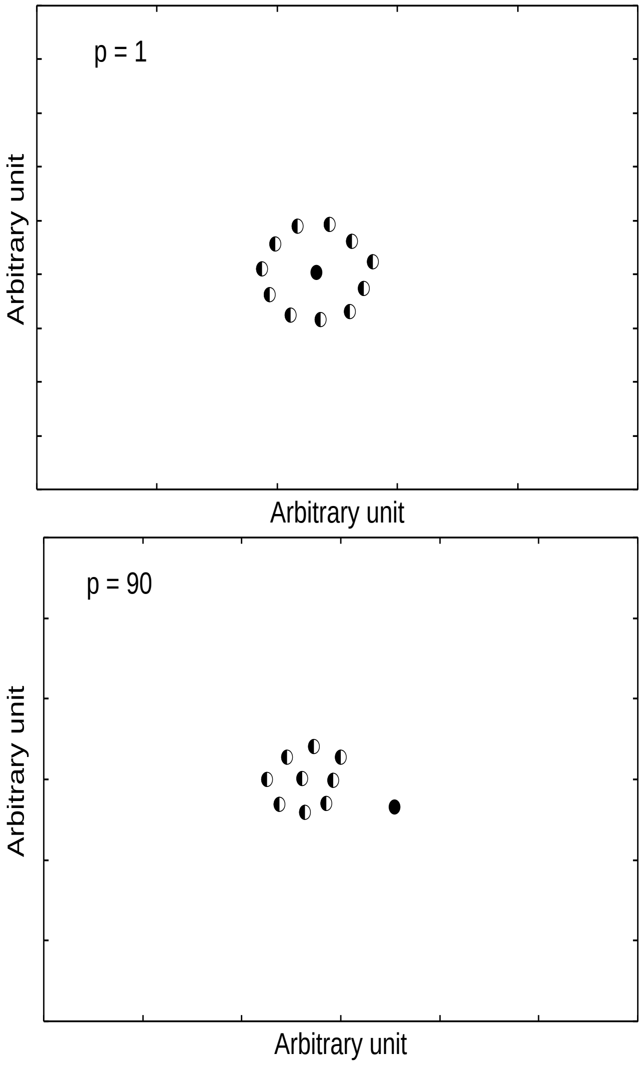

Figure 2: 2D points mapped from solutions in the 2401-dimensional weight space. Filled-in circle $\bullet$ indicates the origin. The number of stored patterns is 1 (top) and 90 (bottom).

Preliminary Test

As a preliminary experiment, random points from the following three categories of region in the 2401 dimensional space: (1) the region whose coordinates are all positive values belonging to $[0, 1]$; (2) the smaller region whose coordinates are picked up from $[0.5, 1]$; (3) two regions whose coordinates are either $[-1, 0]$ or $[0, 1]$. Then they are normalized on the hypersphere surface and are visualized. The results are shown in Figure 1. We can see the results reflect the distribution of the original points in high dimensional space pretty well in all these three cases.

Distribution of Weight Solutions of Associative Memory

As stated earlier, multiple configurations of weights give a network a function of associative memory. The number of these configurations is known to be dependent on p, the number of patterns to be stored. Storing just one pattern gives a maximum number of solutions of weights, while as p approaches twice the number of neurons, all the solutions vanishes [13]. However, the number and distribution as a function of p is still unknown. So far, we have applied some variants of evolutionary algorithms to search for these solutions [2, 15, 16, 17, 18]. Here, we study the solutions found by the BGA among others, since only this algorithm has been able to search for solutions for a wide range of p [16].

Our experiments were carried out on networks with 49 neurons and the BGA found solutions for up to $p = 90$. The solutions that the BGA found are also expected to be different from run to run, as Mühlenbein et al. [1] wrote: *"the BGA mutation scheme is able to optimize many multi-modal functions."* As a preliminary stage of the goal of learning distribution of the solutions as a function of p, we sampled such solutions for $p = 1$ and 90.

We guess that the normalized solutions for $p = 1$ are distributed uniformly on the surface of the hyper-sphere. We show the results of the 2D points mapped from the 2401-dimensional space for the number of solutions $\mathcal{N} = 11$ in Figure 2 (top). We observed that the 2D points corresponding to the solutions are almost uniformly distributed on the circle whose center corresponds to the origin of the 2401-dimensional space.

Next, we apply the technique to solutions also obtained by the BGA runs for $p = 90$. This number of patterns to be stored is almost the upper bound of the storage capacity for a network with 49 neurons, and the solutions are expected to be localized in a small region of weight space [13]. A result for $\mathcal{N} = 9$ is shown in Figure 2 (bottom). Though resolution is not so good, we can imagine the localized solutions.

IV. Conclusion

With our goal being a visualization of weight solutions of neural network model of associative memory, we have described a technique to map a high-dimensional search space to 2D points remaining the distance relation of the source points being as much as possible.

Since the topic has not been the subject of extensive research, as Shine et al. [3] pointed out, we preliminary applied the technique to three different sets of random points picked up from a known shaped region in the 2401 dimensional space, and we have found that the technique can give us an information of the distribution of the original points fairly well.

Then, we applied the technique to our weight space in which we already had the solutions that give a network a function of associative memory. If p, the number of patterns to be stored, is only one, then the solutions are expected to be uniformly distributed in the space. On the other hand, if p is almost the upper bound of the storage capacity, then the solutions are expected to be highly localized. And we obtained the expected results, that is, the solutions are distributed uniformly at random for $p = 1$ and very localized for $p = 90$.

Acknowledgment

We thank Trevor D. Collins for answering us several questions about their GA implementation of the Sammon Mapping via e-mail.

References

[1] Mühlenbein, H., and D. Schlierkamp-Voosen (1996) *Predictive Models for the Breeder Genetic Algorithm I. Continuous Parameter Optimization.* Evolutionary Computation, 1, pp.25–49.

[2] Imada, A., and K. Araki (1997) *Evolution of Hopfield Model of Associative Memory by the Breeder Genetic Algorithm.* Proceedings of the 7th International Conference on Genetic Algorithms, pp.784–791.

[3] Shine, W. B. and C. F. Eick (1997) *Visualizing the Evolution of Genetic Algorithm Search Processes.* Proceedings of the IEEE International Conference on Evolutionary Computation, pp.367–372.

[4] Collins, T. D. (1997) *Using Software Visualization Technology to Help Evolutionary Algorithm Users Validate their Solutions.* Proceedings of the 7th International Conference on Genetic Algorithms, pp.307–314.

[5] Sammon, J. W. (1969) *A Nonlinear Mapping for Data Structure Analysis.* IEEE Transactions on Computers. C-18(5), pp.401–408.

[6] Komlós, J., and R. Paturi (1988) *Convergence Results in an Associative Memory Model.* Neural Networks, 1, pp.239–250.

[7] Hopfield, J. J. (1982) *Neural Networks and Physical Systems with Emergent Collective Computational Abilities.* Proceedings of the National Academy of Sciences, USA, 79, pp.2554-2558.

[8] Hebb, D. O. (1949) *The Organization of Behavior. Wiley.*

[9] Cooper, L. N. (1973) *A Possible Organization of Animal Memory and Learning.* Proceedings of the 24th Nobel Symposium on Collective Properties of Physical Systems, pp.252–264.

[10] Mattis, D. C. (1976) *Solvable Spin Systems with Random Interactions.* Physics Letters, 56A, pp.421–422.

[11] Sherrington, D., and S. Kirkpatrick (1975) *Solvable Model of a Spin Glass.* Physical Review Letters, 35, pp.1792–1796.

[12] Amit, D. J. (1989) *Modeling Brain Function: The World of Attractor Neural Networks.* Cambridge University Press.

[13] Gardner, E. (1988) *The Phase Space of Interactions in Neural Network Models.* Journal of Physics, 21A, pp.257–270.

[14] Syswerda, G. (1989) *Uniform Crossover in Genetic Algorithms.* Proceedings of the 3rd International Conference on Genetic Algorithms, pp.2–9.

[15] Imada, A., and K. Araki (1995) *Genetic Algorithm Enlarges the Capacity of Associative Memory.* Proceedings of the 6th International Conference on Genetic Algorithms, pp.413–420.

[16] Imada, A., and K. Araki (1996) *Lamarckian Evolution of Associative Memory.* Proceedings of IEEE International Conference on Evolutionary Computation, pp.676–680.

[17] Imada, A., and K. Araki (1998) *What does the Landscape of a Hopfield Associative Memory Look Like?* Proceedings of the 7th Annual Conference on Evolutionary Programming, Springer Verlag, Lecture Notes in Computer Science, pp.647–656.

[18] Imada, A. (1999) *Downhill Walk from the Top of a Hill by Evolutionary Programming.* Proceedings of the Congress on Evolutionary Computations, IEEE Press, pp.1414–1418.