Differential Association and Operational Equivalence of Discrete Hopfield Networks

Aristidis Likas and Andreas Stafylopatis

National Technical University of Athens
Department of Electrical and Computer Engineering
Division of Computer Science
157 73 Zographou, Athens, Greece

Email: arly@theseas.ntua.gr, andreas@theseas.ntua.gr

Abstract

An original transform is presented which, given a binary Hopfield neural network and a state vector of this network, creates a new binary Hopfield network of the same size and establishes a correspondence between the states of the two networks during operation. The derived network operates in a differential manner with respect to the initial one, in that its state vector represents the deviation of the corresponding state of the original network from the base state used for the transform. As a consequence, the energy of the second network accepts an analogous interpretation. The transform exhibits several interesting properties which are proved and discussed. Moreover, the notion of operational equivalence is introduced and it is shown that equivalence classes of binary Hopfield networks can be defined on the basis of the transform.

1 Introduction

The Hopfield network [4] constitutes a well known model of neural computation. It is a recurrent neural network with computational units which can have either discrete states [4] (taking values in $\{0, 1\}$ or {-1, 1}) or continuous states [5] (taking values in [0,1]). Each computational unit performs the simple computation of the weighted sum of its inputs and uses a transfer function to determine its new state. Several applications of the Hopfield network have been reported mainly in the fields of associative memory [7, 3] and combinatorial optimization [1, 6]. Also, this model has been the basis for the development of other neural network models, such as the Boltzmann Machine [2, 1].

This paper deals with the binary Hopfield neural network, with the state of each unit in {0,1}. We use the notation (W, θ) to denote a Hopfield network having connection weights w_{ij} $(i = 1, \dots, n,\ j = 1, \dots, n)$ with $w_{ii} = 0$ and threshold values θ_i $(i = 1, \dots, n)$. The energy of the network at a binary vector state $y = (y_1, \dots, y_n)$ is denoted by $E^{(W,\theta)}(y)$ and is equal to:

$$E^{(W,\theta)}(y) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j w_{ij} - \sum_{i=1}^{n} \theta_i y_i \tag{1}$$

Moreover, we are interested in networks that operate *asynchronously*. At each time instant, one unit i is selected randomly and the quantity $\sum_{j=1}^{n} y_j w_{ji} + \theta_i$ is calculated. If it is positive, we set $y_i = 1$, otherwise we set $y_i = 0$. This operation is equivalent to computing the change in the network's energy that will result if the selected node i changes state:

$$\delta E_i^{(W,\theta)}(y) = (2y_i - 1)(\sum_{j=1}^{n} w_{ji} y_j + \theta_i) \tag{2}$$

If $\delta E_i^{(W,\theta)}(y) < 0$ then the change is accepted, otherwise it is rejected. In the case of symmetrical weights $(w_{ij} = w_{ji})$, it is ensured that starting from an arbitrary initial state the network will settle into a state corresponding to an energy minimum [4]. In that state, $\delta E_i^{(W,\theta)}(y) > 0$ for all $i = 1, \dots, n$, i.e., no further change is possible. In general, there exist many states corresponding to local minima of the energy function.

It will be shown here that, once a Hopfield network (W, θ) and a state vector $x = (x_1, \dots, x_n)$ are given, it is possible to construct another Hopfield network (F, ζ) that is related to (W, θ) in the following manner. Consider that the two networks operate in parallel starting from states y^0 and z^0, respectively, such that $y^0 = z^0 \oplus x$, where $\oplus$ denotes the exclusive or operator. At each step during operation, an index i is selected and the respective node with index i is considered for update in each one of the networks. Then, at any time instant, if node i of (F,ζ) is in state 1, node i of the original network has a state value that is complementary to the value of x_i. Accordingly, if node i of (F,ζ) is in state 0, node i of the original network has a state value equal to x_i. In other words, the state vectors y and z of the two networks satisfy the relation $y = z \oplus x$. Thus, the state vector of the derived network represents the deviation of the corresponding state of the original network from the state vector x through which the transform takes place. In this sense, the derived network can be viewed as operating in a differential manner with respect to the original network. This differential association between states of the two networks implies a meaningful relationship regarding the respective energy functions.

In Section 2 the mathematical formulation of the transform is presented and the correspondence in the operation of the original and the resulting network is established. In Section 3 some interesting properties are derived and a number of issues concerning the operation of the binary Hopfield network are discussed. The notion of operational equivalence relation on a set of binary Hopfield networks is introduced and discussed in Section 4, while conclusions and directions for future research are presented in Section 5.

2 Differentially Associated Networks

Consider a Hopfield neural network (W,θ) with n processing units and a state vector $x = (x_1,\ldots,x_n)$, with $x_i \in \{0,1\}$ for $i = 1,\ldots,n$.

Theorem 1. *For each binary Hopfield network (W,θ) and state vector x of this network, there exists a binary Hopfield network (F,ζ) of equal size such that for each binary state z of (F,ζ) the following holds:*

$$E^{(W,\theta)}(x \oplus z) = E^{(W,\theta)}(x) + E^{(F,\zeta)}(z) \tag{3}$$

where the parameters f_{ij} $(i = 1,\ldots,n, j = 1,\ldots,n)$ and ζ_i $(i = 1,\ldots,n)$ of the network (F,ζ) are specified as follows:

$$f_{ij} = (2x_i - 1)(2x_j - 1)w_{ij} \tag{4}$$

$$\zeta_i = -dE_i^{(W,\theta)}(x) = -(2x_i - 1)(\sum_{j=1}^{n} w_{ij}x_j + \theta_i) \tag{5}$$

Proof. Let $y = x \oplus z$. Each component $y_i = x_i \oplus z_i$ $(i = 1,\ldots,n)$ of y can be written as:

$$y_i = x_i - \kappa_i z_i \tag{6}$$

where $\kappa_i = 2x_i - 1$ $(i = 1, \ldots, n)$. The energy of the network (W, θ) in state y is

$$
\begin{aligned}
E^{(W,\theta)}(y) &= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} y_i y_j w_{ij} - \sum_{i=1}^{n}\theta_i y_i \\
&= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i - \kappa_i z_i)(x_j - \kappa_j z_j) w_{ij} - \sum_{i=1}^{n}\theta_i (x_i - \kappa_i z_i)
\end{aligned}
\tag{7}
$$

Taking into account that $w_{ij} = w_{ji}$, after performing some algebra we find:

$$
\begin{aligned}
E^{(W,\theta)}(y) &= -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} x_i x_j w_{ij} - \sum_{i=1}^{n}\theta_i x_i \\
&\quad -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\kappa_i\kappa_j w_{ij} z_i z_j + \sum_{i=1}^{n}\kappa_i(\sum_{j=1}^{n} x_j w_{ji} + \theta_i) z_i
\end{aligned}
\tag{8}
$$

and from the energy definition it follows that

$$E^{(W,\theta)}(y) = E^{(W,\theta)}(x) + E^{(F,\zeta)}(z) \tag{9}$$

where

$$E^{(F,\zeta)}(z) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} z_i z_j f_{ij} - \sum_{i=1}^{n}\zeta_i z_i \tag{10}$$

with f_{ij} $(i = 1, \ldots, n, j = 1, \ldots, n)$ and ζ_i $(i = 1, \ldots, n)$ as given by equations (4) and (5) respectively. $\square$

Let us denote by $\mathcal{D}$ the transform that leads to the construction of the network (F, ζ). Thus, the relation $(F, \zeta) = \mathcal{D}_x(W, \theta)$ means that the network (F, ζ) results from the network (W, θ) and the state vector x, through application of equations (4) and (5). We shall refer to x as the *base state* of the transform. It is obvious that since $w_{ij} = w_{ji}$ and $w_{ii} = 0$, the resulting network is also a symmetric network with $f_{ij} = f_{ji}$ and $f_{ii} = 0$ $(i = 1, \ldots, n, j = 1, \ldots, n)$.

In addition to providing the mathematics for the construction of the network (F, ζ), the above theorem states that the energies of the two networks in states y and z (with $y = x \oplus z$) differ by a constant quantity $E^{(W,\theta)}(x)$ corresponding to the energy of the first network in the base state x. This correspondence in energy implies an analogy of the two networks during operation as indicated by the following proposition.

Proposition 1. *For the state vectors y, x and z, with $y = x \oplus z$ (and, therefore, $z = x \oplus y$), the following relationship holds for $i = 1, \ldots, n$:*

$$dE_i^{(W,\theta)}(y) = dE_i^{(F,\zeta)}(z) \tag{11}$$

where $(F,\zeta) = \mathcal{D}_x(W,\theta)$.

Proof. From equation (2) we have:

$$dE_i^{(W,\theta)}(y) = (2y_i - 1)(\sum_{j=1}^{n} w_{ij}y_j + \theta_i) \tag{12}$$

$$dE_i^{(F,\zeta)}(z) = (2z_i - 1)(\sum_{j=1}^{n} f_{ij}z_j + \zeta_i) \tag{13}$$

Since $y_i = x_i \oplus z_i$ $(i = 1,\ldots,n)$, which is equivalent to $y_i = x_i - \kappa_i z_i$ with $\kappa_i = 2x_i - 1$, we have that:

$$2y_i - 1 = 2(x_i - (2x_i - 1)z_i) - 1 = -(2x_i - 1)(2z_i - 1) = -\kappa_i(2z_i - 1) \tag{14}$$

Substitution in equation (12) yields:

$$\begin{aligned} dE_i^{(W,\theta)}(y) &= -\kappa_i(2z_i - 1)(\sum_{j=1}^{n} w_{ij}y_j + \theta_i) \\ &= -\kappa_i(2z_i - 1)(\sum_{j=1}^{n} w_{ij}(x_j - \kappa_j z_j) + \theta_i) \end{aligned}$$

and, finally,

$$dE_i^{(W,\theta)}(y) = -\kappa_i(2z_i - 1)(\sum_{j=1}^{n} w_{ij}x_j + \theta_i) + \kappa_i(2z_i - 1)\sum_{j=1}^{n} \kappa_j z_j w_{ij} \tag{15}$$

Now, using equations (4) and (5), equation (13) takes the form:

$$dE_i^{(F,\zeta)}(z) = (2z_i - 1)(\sum_{j=1}^{n} \kappa_i\kappa_j w_{ij}z_j - \kappa_i(\sum_{j=1}^{n} x_j w_{ij} + \theta_i)) \tag{16}$$

which is the same as equation (15). $\square$

The above proposition provides insight into the operation of the network (F,ζ), which operates in a *differential* manner with respect to the original network.

For a binary Hopfield network, let us define an *update sequence* as a sequence of indices i_k $(1 \leq i_k \leq n)$, such that the corresponding nodes of the network are successively considered for update during operation. For a given update sequence, we define the *execution profile* of the network as a sequence of binary values δ_k, where the value 1 (0) means that the state of the corresponding node i_k has been changed (has not been changed). Finally, for a given update sequence, the *energy profile* of the network is the sequence of energy changes dE_{i_k} computed when the corresponding node is considered for update. It is obvious that, as far as changes of state are performed depending on energy changes in a deterministic way, an energy profile uniquely determines the execution profile.

Proposition 1 implies that if the same update sequence is applied to both networks (W, θ) and (F, ζ) starting from states y^0 and z^0, respectively, such that $y^0 = z^0 \oplus x$, then the two networks exhibit the same energy profile, and, consequently, the same execution profile.

A node transition (energy decrease) in the network (F, ζ) that sets a node i to the 'on' state has the meaning that there is a decrease in the energy of the network (W, θ) if the state of node i changes with respect to the value x_i, i.e., node i is set to $1-x_i$. Equivalently, a state transition in the network (F, ζ) that sets a node i to the 'off' state means that there is an energy decrease in (W, θ) if node i of this network is set to the value x_i.

As a result, there exists an exact correspondence in operation between the networks (W, θ) and (F, ζ). In fact, observation of the operation of one of the networks and knowledge of the correspondence between network states ($y = x \oplus z$), provides knowledge concerning the evolution of the other network. Therefore, one can switch during operation from one network to the other. At each time instant, if network (F, ζ) reaches a state z, then (in the case of deterministic updates) we can be certain that the operation of network (W, θ) would lead to the state $y = x \oplus z$, and vice versa. We can say that the two networks are *operationally equivalent*, a notion that will be discussed in a later section.

From another perspective, at each time instant, the state vector z of (F, ζ) provides the difference between the corresponding state y that the network (W, θ) would reach and the base state x. Moreover, Theorem 1 reinforces the view of (F, ζ) as a differential network, since it ensures that if state z of (F, ζ) corresponds to the difference between the state vectors y and x of the original network, then the energy of (F, ζ) in state z provides the difference between the energy values of (W, θ) that correspond to the states y and x.

3 Other Properties

1. *$\mathcal{D}_0(W, \theta) = (W, \theta)$ where we denote by 0 the zero vector. This means that the zero vector can be considered as the identity element of the transform $\mathcal{D}$.*

   Proof. Since $2x_i - 1 = -1$ for $i = 1, \ldots, n$, equations (4) and (5) yield $f_{ij} = w_{ij}$ and $\zeta_i = \theta_i$, i.e., the resulting network is identical to the original. □

2. *If $z = x \oplus y$ and y is an equilibrium state (local minimum) of (W, θ), then z is an equilibrium state of $(F, \zeta) = \mathcal{D}_x(W, \theta)$. Also, if z is an equilibrium point of*

$(F,\zeta) = \mathcal{D}_x(W,\theta)$, then $y = x \oplus z$ is an equilibrium point of (W,θ).

Proof. As stated in the introduction, once a Hopfield network is in a local minimum state v, then $dE_i(v) > 0$ for $i = 1, \ldots, n$. Considering the states y and $z = x \oplus y$ of the networks (W,θ) and (F,ζ), Proposition 1 states that the corresponding energy differences $dE_i^{(W,\theta)}(y)$ and $dE_i^{(F,\zeta)}(z)$ are equal for each i. Consequently, if one of the states y and z corresponds to a local minimum of one network then the other state will also correspond to a local minimum of the other network. $\square$

An interesting result that stems from this property is the following. Suppose that we use the Hopfield network as an associative memory and we are given a set of m vectors $\{y^1, \ldots, y^m\}$ that should be stored in the network. This essentially means that we have to apply a learning rule (e.g. Hebbian rule) and construct a Hopfield network (W,θ) with the property that the m stored vectors $\{y^1, \ldots, y^m\}$ correspond to equilibrium points (local minima) of the resulting network. If this network is transformed with respect to a vector x, then the network $(F,\zeta) = \mathcal{D}_x(W,\theta)$ will have the vectors $\{d^1, \ldots, d^m\}$ as equilibrium points, where $d^k = x \oplus y^k$ $(k = 1, \ldots, m)$. In this way, we have stored the sequence of vectors $\{d^1, \ldots, d^m\}$ in the network (F,ζ). This, of course, could have been done using the learning algorithm to construct a network (W',θ') that explicitly stores the sequence d^k. It would be interesting to study the relationship between the two networks (F,ζ) and (W',θ') and compare their reconstruction capabilites in conjunction with the learning algorithm used.

3. *If $(F,\zeta) = \mathcal{D}_x(W,\theta)$ then $\mathcal{D}_x(F,\zeta) = (W,\theta)$. More generally, if $(F,\zeta) = \mathcal{D}_x(W,\theta)$, then $\mathcal{D}_y(F,\zeta) = \mathcal{D}_{x\oplus y}(W,\theta)$.*

   Proof. Let $z = x \oplus y$, $(G^1,\delta^1) = \mathcal{D}_y(F,\zeta)$ and $(G^2,\delta^2) = \mathcal{D}_{x\oplus y}(W,\theta)$. It is sufficient to show that $g^1_{ij} = g^2_{ij}$ and $\delta^1_i = \delta^2_i$ for $i = 1, \ldots, n$ and $j = 1, \ldots, n$. Since $x_i \oplus y_i = x_i - (2x_i - 1)y_i$, we have:

$$2z_i - 1 = 2(x_i - (2x_i - 1)y_i) - 1 = -(2x_i - 1)(2y_i - 1) \tag{17}$$

   Thus, using the transform equation (4), we obtain for $i = 1, \ldots, n$ and $j = 1, \ldots, n$:

$$\begin{aligned} g^1_{ij} &= (2y_i - 1)(2y_j - 1)f_{ij} = (2y_i - 1)(2y_j - 1)(2x_i - 1)(2x_j - 1)w_{ij} \\ &= (2z_i - 1)(2z_j - 1)w_{ij} = g^2_{ij} \end{aligned} \tag{18}$$

   From equation (5) we have that for $i = 1, \ldots, n$:

$$\delta^1_i = -(2y_i - 1)(\sum_{j=1}^{n} f_{ij}y_j + \zeta_i) \tag{19}$$

$$\delta_i^2 = -(2z_i - 1)(\sum_{j=1}^{n} w_{ij}z_j + \theta_i) \tag{20}$$

Taking into account equations (4), (5) and (17), equation (19) can be written:

$$\delta_i^1 = -(2y_i - 1)\left[\sum_{j=1}^{n}(2x_i - 1)(2x_j - 1)w_{ij}y_j - (2x_i - 1)(\sum_{j=1}^{n} x_j w_{ij} + \theta_i))\right] \tag{21}$$

Also using equation (17), equation (20) takes the form:

$$\delta_i^2 = (2x_i - 1)(2y_i - 1)\left[\sum_{j=1}^{n} w_{ij}(x_j - (2x_j - 1)y_j) + \theta_i)\right] \tag{22}$$

It can be easily observed that the right-hand parts of equations (21) and (22) are identical. □

4. *If y is the global minimum state of (W, θ), then $z = x \oplus y$ is the global minimum state of $(F, \zeta) = \mathcal{D}_x(W, \theta)$ and inversely, if z is the global minimum state of (F, ζ), then $y = x \oplus z$ is the global minimum state of (W, θ).*

   Proof. This is a direct consequence of the fact that when the networks (W, θ) and (F, ζ) are in the states y and z respectively (with $z = x \oplus y$), their respective energy values differ by a constant amount. Thus, minimum energy values will also differ by the same constant and will correspond to states y and z with $z \oplus y = x$, where x is the base state of the transform. □

Based on the last property, it is apparent that a state x is the global minimum state of a binary Hopfield network (W, θ), if and only if the network $(F, \zeta) = \mathcal{D}_x(W, \theta)$ has the zero state as the global minimum state. From equation (1) giving the energy function of a Hopfield network, it is easy to observe that a sufficient condition for the zero state to be a global minimum state is that the weights and thresholds of the network should have nonpositive values. This is translated in our case as $f_{ij} \leq 0$ and $\zeta_i \leq 0$ for $i = 1, \ldots, n$ and $j = 1, \ldots, n$. Using equations (4) and (5), which describe the transform, we derive the following sufficient condition.

Proposition 2. *If a state $x = (x_1, \ldots, x_n)$ of a binary Hopfield network (W, θ) satisfies the conditions:*

$$(1 - 2x_i)(1 - 2x_j)w_{ij} \leq 0, \quad i = 1, \ldots, n, j = 1, \ldots, n \tag{23}$$

$$(dE)_i(x) \geq 0, \quad i = 1, \ldots, n \tag{24}$$

then x corresponds to the global minimum state of (W, θ).

It must be noted that the above inequalities constitute a rather strict condition for the global minimum state, but can prove helpful when using the binary Hopfield network (or the closely related Boltzmann Machine) to solve optimization problems. In such problems, one is interested in finding the state that corresponds to the global energy minimum, since this state gives the optimal solution to the problem. The above proposition constitutes an attempt to relate the global minimum state with the network parameters in an explicit way. It can also be used to construct a binary Hopfield network with a given state as the global minimum state.

4 Operational Equivalence

Definition 1. We say that two binary Hopfield networks (W_1, θ_1) and (W_2, θ_2) of the same size are *operationally equivalent*, if there exists an one-to-one correspondence between the states of the two networks, such that, for each state y of one network and the corresponding state z of the other, the difference in energy caused by a change in the state of each node i $(i = 1, \ldots, n)$ is the same for both networks:

$$dE_i^{(W_1,\theta_1)}(y) = dE_i^{(W_2,\theta_2)}(z) \tag{25}$$

In essence, the notion of operational equivalence implies that, once two networks are operationally equivalent, the knowledge of one of them and of the correspondence between states is sufficient for simulating the operation and evolution of the other network.

Taking into account the above definition it is obvious that the two networks (W, θ) and $(F, \zeta) = \mathcal{D}_x(W, \theta)$ are operationally equivalent, since Proposition 1 ensures that for each state z of the second network there exists a state $y = x \oplus z$ of the first network, such that the above equation is satisfied for every node index i. Moreover, based on Property 3 we can extend this argument.

Proposition 3. *All the binary Hopfield networks that can be constructed from a network* (W, θ) *using the transform* $\mathcal{D}_x(W, \theta)$ *for different values of the base vector x, are operationally equivalent to each other.*

Proof. Consider two binary vectors x^1 and x^2 and two binary Hopfield networks (F^1, ζ^1), (F^2, ζ^2), such that $(F^1, \zeta^1) = \mathcal{D}_{x^1}(W, \theta)$ and $(F^2, \zeta^2) = \mathcal{D}_{x^2}(W, \theta)$. Let also $z = x^1 \oplus x^2$ (and consequently $x^2 = x^1 \oplus z$). Then using Property 3 we obtain:

$$(F^2, \zeta^2) = \mathcal{D}_{x^1 \oplus z}(W, \theta) = \mathcal{D}_z(F^1, \zeta^1) \tag{26}$$

Thus the two networks (F^1, ζ^1) and (F^2, ζ^2) are operationally equivalent since the second network can be derived from the first network through application of the transform $\mathcal{D}$ with respect to the vector $z = x^1 \oplus x^2$. $\Box$

The space of binary Hopfield networks can, thus, be partitioned into equivalence classes; the members of each class can be derived from an arbitrary member of that class through application of the transform using appropriate base states.

5 Conclusions

We have developed an original transform which can be used for the construction of a binary Hopfield network starting from a given binary Hopfield network and a state of that network. The operation of the new network can be characterized from two different viewpoints. On the one hand, the second network provides differential information with respect to the original one and, on the other hand, the two networks satisfy the property of operational equivalence that has been introduced in this paper. Several properties of this transform have been presented along with theoretical results that provide a different insight into the operation of the binary Hopfield network.

In addition to further investigation of theoretical issues, we aim at finding applications where these results could prove useful. We believe that applications of the Hopfield network to cryptography and data security could be developed, since the network resulting from the transform operates equivalently with respect to the original one, but the information about the state of the original network is hidden to one that does not have knowledge of the base state used for the transform.

References

[1] Aarts, E. and Korst, J., *Simulated Annealing and Boltzmann Machines, A Stochastic Approach to Combinatorial Optimization and Neural Computing*, John Wiley & Sons Ltd., 1989.

[2] Ackley, D., Hinton, G. and Sejnowski, T., *A Learning Algorithm for Boltzmann Machines*, Cognitive Science, 9 (1985), 147-169.

[3] Aiyer, S.V.B., Niranjan, M. and Fallside, F., *A Theoretical Investigation into the Performance of the Hopfield Model*, IEEE Trans. on Neural Networks, 1 (1990), 204-215.

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

[5] Hopfield, J., *Neurons with Graded Response have Collective Computational Properties like those of Two-State Neurons*, Proc. of the Nat. Academy of Sciences, USA, 81 (1984), 3088-3092.

[6] Hopfield, J.J and Tank, D.W., *Neural Computation of Decisions in Optimization Problems*, Biological Cybernetics, 52 (1985), 141-152.

[7] McEliece, R.J., Posner, E. C., Rodemich, E.R. and Venkatesh, S.S., *The Capacity of the Hopfield Associative Memory*, IEEE Trans. on Information Theory, 33 (1987), 461-482.