1. The Input/Output Relationship

In the last thirty years, the research foci of Neural Networks have changed due to properties discovered in Neural Networks. In the early days, the focus was to identify an efficient way to determine the acceptable (or best) parameter values of a network (cf. Rumelhart and McClelland 1986). Then there came the days to investigate the properties of different Neural Network structures, including the number of hidden layers, the number of hidden nodes, the form of activation functions, and the connections between different layers of nodes (cf. Hertz, Krogh, and Palmer 1991). Recently, there has been much research on extracting rules from Neural Networks (cf. Andrews, Diederich, and Tickle 1995; Tsaih, Hsu, and Lai 1998; Maire 1999; Browne and Sun 2001; Setiono, Leow, and Zurada 2002; Saito and Nakano 2002; Hanson and Negishi 2002; and Rabuñal, Dorado, Pazos, Pereira, and Rivero 2004). With all the progresses in Neural Networks, there remains one unsettled question: A systematic, cause-and-effect explanation of the input/output relationship of Neural Networks.

For instance, Huang and Babri (1998) proposed an elegant construction method to set up a real-valued single-hidden layer feed-forward neural network (SLFN) with N hidden nodes to successfully learn N distinct samples with zero error. However, they left two issues worthwhile to follow up: (i) Does HB_SLFN, the SLFN constructed by the method in Huang and Babri (1998), possess any definitive characteristics that differentiate it from SLFNs obtained from other algorithms (or construction methods)? (ii) Is it possible to have an alternative construction that results in a number of adopted hidden nodes significantly less than N? A thorough exploration of the input/output relationship of HB_SLFN may address these two issues. To shed light on the unsettled question about the real-valued SLFN, Tsaih and Wan (2007) explored the *preimage* – the collection of inputs of a given output, and showed that the preimage for a specific output value is either a single manifold or multiple disjoint manifolds, whose form is either linear or nonlinear.

In a *correlated* real-valued SLFN, the weight vectors in the input layer of all its hidden nodes are linearly dependent. We can identify that HB_SLFN is a *correlated* SLFN. Through applying the preimage analysis to HB_SLFN to explore the characteristics of the correlated SLFN, we consolidate two hyperplane principles. Then, with the illustration of applying these two hyperplane principles to the m-bit parity problem, we show that these two hyerplane principles have significant implications in constructing a correlated SLFN that perfectly fits N distinct samples with less than N hidden nodes. The feature of a fewer number of hidden nodes is generally regarded as desirable for preventing over-learning. The number of hidden nodes for a Neural Network has been addressed by Huang and Huang (1991) and Arai (1993). However, both papers considered binary-valued outputs from the hidden nodes and obtained a lower bound of N-1 hidden nodes to have a perfect fit for a sample of size N. To take the training advantages of back propagation of errors, most researchers use continuous-valued hidden node outputs, as we do here.

The remainder of this paper is organized as follows. Section II briefly introduces the construction method proposed in Huang and Babri (1998). Section III summarizes the application of the preimage analysis to HB_SLFN. Section IV shows the powerful usage of the correlated SLFN. Finally, discussion and suggestions for future research are offered in Section V.

2. The Construction Method of Huang and Babri (1998)

Let the parameters of HB_SLFN be as follows:

w^H_{ji} ≡ the weight between the ith input variable and the jth hidden node, where the superscript H throughout the paper refers to quantities related to the hidden layer;

$\mathbf{w}^H_j$ ≡ $(w^H_{j1}, w^H_{j2}, \ldots, w^H_{jI})^{\mathrm{T}}$, where $(\cdot)^{\mathrm{T}}$ denotes the transpose of $(\cdot)$;

w^H_{j0} ≡ the bias value of the jth hidden node;

w^o_j ≡ the weight between the jth hidden node and the output node, where the superscript o throughout the paper refers to quantities related to the output layer;

$\mathbf{w}^o$ ≡ $(w^o_1, w^o_2, \ldots, w^o_N)^{\mathrm{T}}$; and

w^o_0 ≡ the bias value of the output node.

Let $\mathbf{x}^c$ and t^c be the cth input pattern and the corresponding target value, respectively, $c = 1, ...,$ N; $\mathbf{T} \equiv (t^1, t^2, \ldots, t^N)^{\mathrm{T}}$ be the vector of desirable outputs for the N input samples; $x_{01} > x_{02}$ *be* two arbitrary pre-specified constants. The construction method of Huang and Babri (1998) works for *any* activation function g as long as $g(x_{01}) \neq \lim_{x \to +\infty} g(x)$. The procedure is first to arbitrarily choose a vector $\mathbf{w}$ and label $\mathbf{x}^c$ such that

$$\mathbf{w}^{\mathrm{T}}\mathbf{x}^1 < \mathbf{w}^{\mathrm{T}}\mathbf{x}^2 < \ldots < \mathbf{w}^{\mathrm{T}}\mathbf{x}^N. \tag{1}$$

Then calculate $\mathbf{w}_j^H$ and w_{j0}^H from eqt. (2) and eqt. (3), in which the values of $\mathbf{w}_j^H$ and w_{j0}^H are independent of the target outputs $\{t^c\}$:

$$\mathbf{w}_j^H = \begin{cases} \mathbf{0}, & \text{if } j = 1, \\ \dfrac{x_{02} - x_{01}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}\mathbf{w}, & \text{if } 2 \leq j \leq N. \end{cases}; \tag{2}$$

$$w_{j0}^H = \begin{cases} x_{02}, & \text{if } j = 1, \\ \dfrac{x_{01}\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - x_{02}\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}, & \text{if } 2 \leq j \leq N. \end{cases} \tag{3}$$

Let a_j^c be the jth *activation value* for the cth input, i.e., the output of the jth hidden node for input $\mathbf{x}^c$, when $(\mathbf{w}_j^H, w_{j0}^H)$ of HB_SLFN are set as in eqt. (2) and (3). Then

$$a_1^c \equiv g(x_{02}), \text{ and} \tag{4}$$

$$a_j^c \equiv g\Big(\frac{x_{02} - x_{01}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}\mathbf{w}^{\mathrm{T}}\mathbf{x}^c + \frac{x_{01}\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - x_{02}\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^j - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}\Big) \;\forall\; 2 \leq j \leq N. \tag{5}$$

Let $\mathbf{a}^c \equiv (g(x_{02}), a_2^c, \ldots, a_N^c)^{\mathrm{T}}$ and $\mathbf{M} \equiv (\mathbf{a}^1, \mathbf{a}^2, \ldots, \mathbf{a}^N)^{\mathrm{T}}$. Huang and Babri (1998) showed that the N samples in $\{\mathbf{x}\}$ space are mapped to N distinctive points in the activation space such that the matrix $\mathbf{M}$ is invertible. With the bias w_0^o set to zero,

$$\mathbf{w}^o = \mathbf{M}^{-1}\mathbf{T} \tag{6}$$

can always be found to match $w_0^o + \sum_{j=1}^{N} w_j^o a_j^c$ *to* t^c *without any error.*

3. The Preimage Analysis of HB_SLFN

To explore the input/output relationship of a general HB_SLFN, we apply the concept of preimage analysis proposed in Tsaih and Wan (2007). We present the result below.

By the structure of HB_SLFN, the net input to the *j*th hidden node $\nu_j = w_{j0}^H + \sum_{i=1}^{I} w_{ji}^H x_i$, $j = 1, \ldots, N$; the *j*th activation value $a_j = \tanh(\nu_j)$, $j = 1, ..., N$; the output $y = w_0^o + \sum_{j=1}^{N} w_j^o a_j$; and the function $f \equiv w_0^o + \sum_{j=1}^{N} w_j^o \tanh\left(w_{j0}^H + \sum_{i=1}^{I} w_{ji}^H x_i \right)$. Denote a particular collection of w_{ji}^H, w_{j0}^H, w_j^o, and w_0^o by θ. Given θ, f is the composite of the following mappings: the *input mapping* Φ_I:$\Re^I \rightarrow \Re^N$ that maps an input $\mathbf{x}$ to a net input $\mathbf{\nu}$ (i.e., $\mathbf{\nu} = \Phi_I(\mathbf{x})$); the *activation mapping* Φ_A:$\Re^N \rightarrow (-1, 1)^N$ that maps a net input $\mathbf{\nu}$ to an activation value $\mathbf{a}$ (i.e., $\mathbf{a} = \Phi_A(\mathbf{\nu})$); and the *output mapping* Φ_O: $(-1, 1)^N \rightarrow (w_0^o - \sum_{j=1}^{N} |w_j^o|, w_0^o + \sum_{j=1}^{N} |w_j^o|)$ that maps an activation value $\mathbf{a}$ to an output y (i.e., $y = \Phi_O(\mathbf{a})$). Note that the range of Φ_A and the domain of Φ_O are set as $(-1, 1)^N$ because $a_j = \tanh(\nu_j)$ and $-1 < |\tanh(\nu_j)| < 1$ for $-\infty < \nu_j < \infty$ for any j. For the same reason, the range in the output space $\Im \equiv (w_0^o - \sum_{j=1}^{N} |w_j^o|, w_0^o + \sum_{j=1}^{N} |w_j^o|)$ contains all achievable output values. For ease of reference in later sections, we also call R^I the *input space*, R^N the *net input space*, and $(-1, 1)^N$ the *activation space*.

For some θ, f may not be a subjective function. In that case, $f(\{\Re^I\}) \subset \Im$, i.e., there exists $y \in \Im$ such that $y \notin f(\{\Re^I\})$. Such an output value y is referred to as *void* with respect to the given θ; otherwise, y is *non-void*. For instance, assume $\sum_{j=1}^{N} w_j^o a_j$ is equal to zero with respect to the given θ. Then for all $y \in \Im$ such that $y \neq w_0^o$, $y \notin f(\{\Re^I\})$, i.e., $y = w_0^o$ is the only non-void output value in this case. Formally, the followings are defined for a given $\boldsymbol{\theta}$:

(a) The *image* of an input $\mathbf{x} \in \Re^I$ is $y \equiv f(\mathbf{x})$ for $y \in \Re$.

(b) An output value $y \in \Re$ is *void* if $y \notin f(\{\Re^I\})$, i.e., for all $\mathbf{x} \in \Re^I$, $f(\mathbf{x}) \neq y$; otherwise y is *non-void*.

(c) The *preimage* of a non-void output value y is $f^{-1}(y) \equiv \{\mathbf{x} \in \Re^I | \ f(\mathbf{x}) = y\}$. The preimage of a void output value y is the empty set.

(d) The *internal-preimage* of a non-void output y is the collection $\{\mathbf{a} \in (-1, 1)^N | \ \Phi_O(\mathbf{a}) = y\}$ on the activation space.

(e) A subset $\mathbf{A}$ of the activation space is *non-void* if for each $\mathbf{a} \in \mathbf{A}$, there exists $\mathbf{x}$ in the input space such that $\Phi_A \circ \Phi_I(\mathbf{x}) = \mathbf{a}$.

For any non-void y, $f^{-1}(y) \equiv \Phi_I^{-1} \circ \Phi_A^{-1} \circ \Phi_O^{-1}(y)$, with

$$\Phi_O^{-1} : \Im \rightarrow (-1, 1)^N \text{ and } \Phi_O^{-1}(y) \equiv \{\mathbf{a} \in (-1, 1)^N | \ \sum_{j=1}^{N} w_j^o a_j = y - w_0^o\}, \tag{7}$$

$$\Phi_A^{-1} : (-1, 1)^N \rightarrow \Re^N \text{ and } \Phi_A^{-1}(\mathbf{a}) \equiv \{\mathbf{v} \in \Re^N | \ v_j = \tanh^{-1}(a_j), j = 1, ..., N\}, \tag{8}$$

$$\Phi_I^{-1} : \Re^N \rightarrow \Re^I \text{ and } \Phi_I^{-1}(\mathbf{v}) \equiv \bigcap_{j=1}^{N} \{\mathbf{x} \in \Re^I \ | \mathbf{x}^T \mathbf{w}_j^H = v_j - w_{j0}^H\}, \tag{9}$$

where $\tanh^{-1}(x) \equiv 0.5 \ln\left(\frac{1+x}{1-x}\right)$.

From (7), with the given $\mathbf{w}^o$ and w_0^o, $\Phi_O^{-1}(y)$ is the linear equation $\sum_{j=1}^{N} w_j^o a_j = y - w_0^o$. Thus, geometrically for each non-void output value y, $\Phi_O^{-1}(y)$ is a hyperplane in the activation space. As y changes, $\Phi_O^{-1}(y)$ forms parallel hyperplanes in the activation space; for any change of the same magnitude in y, the corresponding hyperplanes are spaced by the same distance. The activation space is entirely covered by these parallel $\Phi_O^{-1}(y)$ hyperplanes, orderly in terms of the values of y. These parallel hyperplanes form a (linear) scalar field (Tsaih, 1998): For each point $\mathbf{a}$ of the activation space, there is only one output value y whose

$\Phi_O^{-1}(y)$ hyperplane passes point **a**; all points on the same (internal preimage) hyperplane are associated with the same y value.

From (8), Φ_A^{-1} is a separable function such that each of its components lies along a dimension of the activation space; moreover, Φ_{Aj}^{-1} is a monotone bijection that defines a one-to-one mapping between the activation value a_j and the net input value ν_j of the jth hidden node. Furthermore, from (9), for the jth hidden node, because $\mathbf{w}_j^H$ and w_{j0}^H are given constants, Φ_I^{-1} defines a hyperplane $\{\mathbf{x} \in \Re^I | \mathbf{x}^{\mathrm{T}} \mathbf{w}_j^H = \nu_j - w_{j0}^H\}$ in the input space. Thus, for a given activation value a_j, $\Phi_I^{-1} \circ \Phi_{Aj}^{-1}(a_j)$ defines the hyperplane $\{\mathbf{x} \in \Re^I | \mathbf{x}^{\mathrm{T}} \mathbf{w}_j^H = \tanh^{-1}(a_j) - w_{j0}^H\}$ in the input space.

The hyperplanes associated with $\Phi_I^{-1} \circ \Phi_{Aj}^{-1}$ in the input space have properties analogous to hyperplanes associated with Φ_O^{-1} in the activation space: The hyperplanes associated with $\Phi_I^{-1} \circ \Phi_{Aj}^{-1}$ are parallel and form a (linear) scalar activation field in the input space; for each point **x** of the input space, there is only one activation value a_j whose $\Phi_I^{-1} \circ \Phi_{Aj}^{-1}(a_j)$ hyperplane passes point **x**; all points on the $\Phi_I^{-1} \circ \Phi_{Aj}^{-1}(a_j)$ hyperplane are associated with the activation value a_j. Each hidden node gives rise to an activation field, and N hidden nodes set up N independent activation fields in the input space. Thus, for a given $\boldsymbol{\theta}$, the preimage of an activation value **a** by $\Phi_I^{-1} \circ \Phi_A^{-1}$ is the intersection $\bigcap_{j=1}^{N} \{\mathbf{x} \in \Re^I \,|\, \mathbf{x}^{\mathrm{T}} \mathbf{w}_j^H = \nu_j - w_{j0}^H\}$.

For ease of reference, denote the intersection $\bigcap_{j=1}^{N} \{\mathbf{x} \in \Re^I \,|\, \mathbf{x}^{\mathrm{T}} \mathbf{w}_j^H = \nu_j - w_{j0}^H\}$ by $\{\mathbf{x} | \mathbf{W}^H \mathbf{x} = \boldsymbol{\omega}(\mathbf{a})\}$, where $\mathbf{W}^H \equiv (\mathbf{w}_1^H, \mathbf{w}_2^H, \ldots, \mathbf{w}_N^H)^{\mathrm{T}}$ is the matrix of weights between the hidden nodes and the input layer, $\omega_j(a_j) \equiv \tanh^{-1}(a_j) - w_{j0}^H$ for all $1 \le j \le N$ and $\boldsymbol{\omega}(\mathbf{a}) \equiv (\omega_1(a_1), \omega_2(a_2), \ldots, \omega_N(a_N))^{\mathrm{T}}$. Given an activation value **a** (and $\boldsymbol{\theta}$), $\boldsymbol{\omega}(\mathbf{a})$ is simply a vector of known component values; the conditions that relates with the activation value **a** and the input value **x**,

$$\mathbf{W}^{H}\mathbf{x} = \omega(\mathbf{a}), \tag{10}$$

is a system of N simultaneous linear equations with I unknowns. From eqt. (2), every row vector of $\mathbf{W}^{H}$ is linearly dependent with the vector **w**, i.e., $rank(\mathbf{W}^{H}) = 1$, in which and hereafter $rank(\mathbf{D})$ is the rank of matrix **D**.

Let $(\mathbf{D_1} \vdots \mathbf{D_2})$ be the augmented matrix of two matrices $\mathbf{D_1}$ and $\mathbf{D_2}$ (with the same number of rows). $\mathbf{W}^{H}\mathbf{x} = \omega(\mathbf{a})$ is a set of inconsistent simultaneous equations if $rank(\mathbf{W}^{H} \vdots \omega(\mathbf{a})) = rank(\mathbf{W}^{H}) + 1$ (Murty 1983, p. 108). In this case, the corresponding point **a** is void. Otherwise, **a** is non-void. For a non-void **a**, the set of equations in eqt. (10) defines an affine space of dimension I - 1 in the input space. The discussion establishes Lemma 1 below.

Lemma 1: (a) For HB_SLFN, an activation vector **a** in the activation space is void if its corresponding $rank(\mathbf{W}^{H} \vdots \omega(\mathbf{a}))$ equals 2. (b) For HB_SLFN, the set of input values **x** mapped onto a non-void **a** forms a hyperplane in the input space.

By definition, *the non-void set*, the set of all non-void **a**'s in the activation space, is

$$\{\mathbf{a} \in (-1, 1)^{N} | \; a_j = tanh(\mathbf{x}^{\mathrm{T}}\mathbf{w}_j^{H} + w_j^{o}) \; \forall \, j \geq 1, \mathbf{x} \in \Re^{I}\}.$$

From eqt. (2) and eqt. (3), the non-void set of HB_SLFN equals

$$\{\mathbf{a} \in (-1, 1)^{N} | \; a_1 = tanh(x_{02}), \, a_j = tanh(\frac{x_{02} - x_{01}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j} - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}\mathbf{w}^{\mathrm{T}}\mathbf{x} + \frac{x_{01}\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j} - x_{02}\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}{\mathbf{w}^{\mathrm{T}}\mathbf{x}^{j} - \mathbf{w}^{\mathrm{T}}\mathbf{x}^{j-1}}) \; \forall \, j \geq 2, \mathbf{x} \in \Re^{I}\}.$$

Note that for $j \geq 2$, a_j is in fact the *tanh* transform of inputs in the form $\beta_{j1}\mathbf{w}^{\mathrm{T}}\mathbf{x} + \beta_{j2}$, where β_{j1} and β_{j2} are constants for the given **w** and sample inputs $\{\mathbf{x}^{c}\}$. Now, from eqt. (1) and eqt. (2), $\mathbf{w}_2^{H}$ is a non-zero vector. Thus, all a_j's for $j \geq 3$ can be represented in terms of a_2 and the non-void set in the activation space is equivalent to

$$\{\mathbf{a} \in (-1, 1)^N |\ a_1 = tanh(x_{02}),\ a_2 \in (-1, 1),\ a_j = tanh(\frac{\mathbf{w}^T\mathbf{x}^2 - \mathbf{w}^T\mathbf{x}^1}{\mathbf{w}^T\mathbf{x}^j - \mathbf{w}^T\mathbf{x}^{j-1}} tanh^{-1}(a_2) + \frac{x_{01}(\mathbf{w}^T\mathbf{x}^j - \mathbf{w}^T\mathbf{x}^2) - x_{02}(\mathbf{w}^T\mathbf{x}^{j-1} - \mathbf{w}^T\mathbf{x}^1)}{\mathbf{w}^T\mathbf{x}^j - \mathbf{w}^T\mathbf{x}^{j-1}})\ \forall\ j \geq 3\}. \quad (11)$$

To deduce the properties of the non-void set, check that eqt. (11) is in the form

$$\{\mathbf{a} \in (-1, 1)^N |\ a_1 = tanh(x_{02}),\ a_2 \in (-1, 1),\ a_j = tanh(\delta_{j1} tanh^{-1}(a_2) + \delta_{j2})\ \forall\ j \geq 3\}, \quad (12)$$

where the actual values of the non-zero constants δ_{j1} and δ_{j2} are unimportant for subsequent discussion. The non-void set is characterized by a single basis $a_2 \in (-1, 1)$ and the function $tanh(\delta_{j1} tanh^{-1}(a_2)+\delta_{j2})$ translates an open set $a_2 \in (-1, 1)$ into an open set for a_j. In other words, as stated in Lemma 2, geometrically the non-void set of HB_SLFN is a 1-*manifold*. A p-manifold is a Hausdorff space **X** with a countable basis such that each point x of **X** has a neighborhood that is homomorphic with an open subset of $\Re^p$. A 1-manifold is often called a curve (Munkres 1975). For our case, we are working with Euclidean spaces, the commonest among Hausdorff spaces.

Lemma 2. The non-void set of HB_SLFN is a 1-manifold.

Remarks. (i) Suppose that $|a_j| = |a_2|$ for all $j \geq 3$. Then the non-void set of HB_SLFN is in fact (part of) the line (segment) $\{\mathbf{a} \in (-1, 1)^N |\ a_1 = tanh(x_{02}),\ a_2 = a \in (-1, 1),\ a_j = a \text{ (or } -a)\ \forall\ j \geq 3\}$. (ii) For each a_j such that $|a_j| \neq |a_2|$, the form $a_j = tanh(\delta_{j1} tanh^{-1}(a_2)+\delta_{j2})$ indicates a degree of non-linearity of the manifold.

Let $\mathbf{A}(y)$ be the intersection of $\Phi_O^{-1}(y)$ and the non-void set in the activation space. $\mathbf{A}(y)$ is called the *internal-preimage* of y in the activation space, since any point in it is linked to the preimage in the input space. Mathematically, for HB_SLFN and each non-void y,

$$\mathbf{A}(y) \equiv \{\mathbf{a} \in (-1, 1)^N |\ w_2^o a_2 + \sum_{j=3}^{N} w_j^o a_j = y - w_1^o tanh(x_{02}),\ \ \mathbf{w}^o = \mathbf{M}^{-1}\mathbf{T},$$

$$a_1 = tanh(x_{02}),\ a_2 \in (-1, 1),\ a_j = tanh(\delta_{j1} tanh^{-1}(a_2)+\delta_{j2})\ \forall\ j \geq 3\}. \tag{13}$$

Effectively $\mathbf{A}(y)$ of HB_SLFN is the intersection of a hyperplane and a 1-manifold. Geometrically, when the non-void set is a line, each (non-empty) $\mathbf{A}(y)$ is the intersection of a hyperplane with a line, which gives a point in the activation space. When the non-void set is a nonlinear 1-manifold, each (non-empty) $\mathbf{A}(y)$ is the intersection of a hyperplane with a curve, which gives rise to either a point or a collection of several disjoint points in the activation space.

As a subset of $\Phi_O^{-1}(y)$, we have the following Lemma 3 and $\mathbf{A}(y)$'s are aligned orderly according to $\Phi_O^{-1}(y)$'s. Furthermore, $\mathbf{A}(y)$'s associated with all non-void y's form an *internal-preimage field* in the activation space, i.e., there is one and only one $\mathbf{A}(y)$ located upon each non-void $\mathbf{a}$; and for any $\mathbf{a}$ on $\mathbf{A}(y)$, its output value is equal to y.

Lemma 3. For a non-void output value y, all points in the set $\mathbf{A}(y)$ are at the same hyperplane.

Now for HB_SLFN and any non-void y, its preimage is

$$f^{-1}(y) \equiv \{\mathbf{x} \in R^I |\ \mathbf{w}^T \mathbf{x} = \frac{(\mathbf{w}^T\mathbf{x}^2 - \mathbf{w}^T\mathbf{x}^1)}{x_{02} - x_{01}} tanh^{-1}(a_2) + \frac{x_{01}\mathbf{w}^T\mathbf{x}^2 - x_{02}\mathbf{w}^T\mathbf{x}^1}{x_{02} - x_{01}},$$

$$w_2^3 a_2 + \sum_{j=3}^{N} w_j^3 tanh(\beta_{j1} tanh^{-1}(a_2) + \beta_{j2}) = y - w_1^3 tanh(x_{02}),\ \ \mathbf{w}^3 = \mathbf{M}^{-1}\mathbf{T},\ a_2 \in (-1, 1)\}. \tag{14}$$

Given $\boldsymbol{\theta}$, the input space is entirely covered by a grouping of (preimage) hyperplanes that forms a *preimage field*. That is, there is one and only one (preimage) hyperplane passing through each $\mathbf{x}$; and the corresponding output value of the network to this $\mathbf{x}$ is the y value associated with this (preimage) hyperplane.

From Lemma 1(b), when the associated A(y) is a single point, the preimage $f^{-1}(y)$ of a non-void output value y is a hyperplane. Similarly, when the associated A(y) consists of disjoint points, $f^{-1}(y)$ is a collection of several disjoint hyperplanes. Table 1 shows that the preimage $f^{-1}(y)$ of HB_SLFN is dictated by the property of the associated internal-preimage A(y). Note that the preimage hyperplanes are aligned orderly because for $f^{-1}(y) \equiv \{\mathbf{x} \in R^I|$ $\mathbf{x}=\Phi_I^{-1} \circ \Phi_A^{-1}(\mathbf{a})$, $\mathbf{a} \in \mathbf{A}(y)\}$, $\mathbf{A}(y)$'s are aligned orderly according to $\Phi_O^{-1}(y)$'s and the mapping of $\Phi_I^{-1} \circ \Phi_A^{-1}$ is a monotone bijection that defines a one-to-one mapping between an activation vector and a hyperplane.

Table 1. The relationship between the internal-preimage A(y) and the preimage $f^{-1}(y)$ of a non-void output value y of HB_SLFN

	A(y) is a single point	A(y) consists of disjoint points
The nature of $f^{-1}(y)$	a hyperplane	several disjoint hyperplanes

Note that, given $\mathbf{w}_j^H$, w_{j0}^H, and $\mathbf{w}^o$ in eqt. (2), (3) and (6), the output value y in regard to an arbitrary input $\mathbf{x}$ can be represented as $w_1^o\, tanh(x_{02})+\sum_{j=2}^{N} w_j^o\, tanh(w_{j0}^H+\sum_{i=1}^{I} w_{ji}^H x_i)$, since a_1 always equals $tanh(x_{02})$. Thus $w_1^o\, tanh(x_{02})$ can serve as the bias of the output node such that there are only N-1 *effective* hidden nodes. Furthermore, from eqt. (3), vectors $\mathbf{w}_j^H$ for $2 \le j \le N$ are linearly dependent. In other words, the HB_SLFN as constructed is a *correlated* SLFN, since the weight vectors (from the input layer) of all its (effective) hidden nodes are linearly dependent on each other. By the proof of Huang & Babri (1998), any N distinct samples can be fitted with zero error through a correlated SLFN. Furthermore, the above preimage analysis has established the following Lemma 4.

Lemma 4: For a correlated SLFN, the preimage field is formed from a collection of (preimage) hyperplanes.

Without any loss of generality, let $\boldsymbol{\beta}^T\mathbf{x} + \theta$ be a preimage hyperplane of the correlated SLFN, where $\boldsymbol{\beta} \equiv (\beta_1, \beta_2, \ldots, \beta_I)^T$ being the normal vector and θ the interception of the hyperplane. From eqt. (14), $\boldsymbol{\beta}$ is in parallel with $\mathbf{w}$ stated in eqt. (1). Furthermore, from eqt.

(2), $\mathbf{w}_j^H \equiv \alpha_j \mathbf{w}$ for all j, where $\alpha_j \neq 0$ for all j and $\alpha_{j_1} \neq \alpha_{j_2}$ for all $j_1 \neq j_2$; and from eqt. (9), $\mathbf{w}_j^H$ determines the orientation of the activation hyperplane in the input space corresponding to the jth hidden node. Thus, we have Lemma 5.

Lemma 5: For a correlated SLFN, the activation hyperplanes in the input space corresponding to all hidden nodes are parallel, and the preimage hyperplane is parallel with the activation hyperplane.

4. The Reduction of Hidden Nodes – Alternative Constructive Methods from Preimage Analysis

In this section, we demonstrate the powerful usage of the correlated SLFN with the application to the m-*bit parity problem. For the* m-*bit parity problem,* I *equals* m *and* N *equals* 2^m*, where the* 2^m *input samples can be regarded as the vertices of an* m-*dimensional hypercube with any two adjacent vertices having different target values. The fact of any two adjacent vertices having different target values renders the* m-*bit parity problem a nonlinearly separable problem and a standard benchmark for the performance of new algorithms for Neural Networks (cf. Hohil, Liu, and Smith 1999; Lavretsky 2000; Arslanov, Ashigaliev, and Ismail 2002; Liu, Hohil, and Smith 2002; Iyoda, Nobuhara, and Hirota 2003; and Urcid, Ritter, and Iancu 2004 for various issues of the* m-*bit parity problem). Without the loss of generality, take* $x_i^c \in \{-1, 1\}$ *for all* c *and* i*, and set the required output to* t *for odd number of +1's in input, and to* -t *otherwise.*

For the m-*bit parity problem, the constructive method of Huang & Babri (1998) has a requirement stated in eqt. (1) that leads to a correlated SLFN with* 2^m*-1 effective hidden nodes, hereafter denoted as SLFN1. From Lemma 5, the normal vector of the preimage hyperplane of SLFN1 is parallel with the* ***w*** *vector picked according to eqt. (1). There is a distinctive preimage hyperplane passing through each of the input samples and there are a total of* 2^m *distinctive (and linearly independent) points in the activation space of* 2^m *dimensions. Thus, the matrix M in eqt. (6) is of full rank, the corresponding inverse matrix* M^{-1} *exists, and eqt. (6) sets up a unique weight vector* $\boldsymbol{w}^o$.

However, when a correlated SLFN is adopted, Lemma 4 and Lemma 5 lead to the following hyperplane principle *(i): input patterns with the same required output value are allowed to be on the same preimage hyperplane. Furthermore, Lemma 3 leads to the following* hyperplane principle (ii)*: activation points with the same required output value are allowed to be on the same hyperplane* $\Phi_O^{-1}(\mathrm{y})$.

The vector $\mathbf{w}$ *could be picked according to the hyperplane principle (i) to have a smaller number of distinctive activation points corresponding to all training samples. Meanwhile, according to the hyperplane principle (ii), the matrix* $\boldsymbol{M}$ *in eqt. (6) is not necessarily of full rank, though of course the system of simultaneous equations in eqt. (15) is still consistent.*

$$\boldsymbol{M}\,\mathbf{w}^{o} = \boldsymbol{T} \tag{15}$$

In the following, we provide two alternative constructive methods, each of which results in a correlated SLFN with a number of effective hidden nodes far less than $2^m - 1$. *The first one is derived merely based on the hyperplane principle (i) and the second one is derived based upon the integration of the hyperplane principles (i) and (ii). For the* m*-bit parity problem, the correlated SLFNs released by these two alternative constructive methods are hereafter denoted as SLFN2 and SLFN3, respectively.*

For the m-bit parity problem, the first alternative constructive method is summarized in the following eqts. (16)-(20):

$$\widetilde{w}_j = 1,\ i = 1, \ldots, m; \tag{16}$$

$$\mathbf{w}_j^{H} = 0.5(x_{02} - x_{01})\,\widetilde{\mathbf{w}},\ j = 1, \ldots, m; \tag{17}$$

$$w_{j0}^{H} = 0.5((2j\text{-}m)x_{01} - (2j\text{-}2\text{-}m)x_{02}),\ j = 1, \ldots, m; \tag{18}$$

$$\begin{pmatrix} \alpha \\ \mathbf{w}^{o} \end{pmatrix} = \widetilde{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^{\,0} \\ \hat{t}^{\,1} \\ \vdots \\ \hat{t}^{\,m} \end{pmatrix}; \tag{19}$$

$$w_0^{o} = \alpha \tanh(x_{02}); \tag{20}$$

where $x_{02} > x_{01}$, $\tanh(x_{01}) \neq 1$ and $\hat{t}^{\,k} \equiv (-1)^{k+1} t$ for all k.

Following the hyperplane principle (i), we can pick up a vector $\widetilde{\mathbf{w}}$ stated in eqt. (16) and thus the normal vector of the preimage hyperplane of SLFN2, which is parallel with the vector $\widetilde{\mathbf{w}}$. With vector $\widetilde{\mathbf{w}}$ stated in eqt. (16), the 2^m input patterns are grouped into m+1 groups according to the number of +1's in an input pattern, i.e., the kth group consists of all input patterns with k +1's and (m-k) -1's, k = 0, …, m. Note that the group membership of inputs determines the magnitude of $\widetilde{\mathbf{w}}^{\mathrm{T}}\mathbf{x}^{c}$, whose value equals $2k$-m if $\mathbf{x}^{c}$ belongs to the kth group, and that there is a distinctive preimage hyperplane passing through all input samples of each group. It is satisfactory to map all input samples in the kth group onto one single point in the activation space, since they have the same required output value.

With the assignments of eqt. (17) and eqt. (18), the total 2^m input samples are mapped onto m+1 activation points in $(-1, 1)^m$, $\{\hat{\mathbf{a}}^0, \ldots, \hat{\mathbf{a}}^m\}$, where $\hat{a}_j^k = tanh((k-j+1)x_{02} - (k-j)x_{01})$, $k = 0, \ldots, m$, $j = 1, \ldots, m$. Now let $\tilde{M}_{i1} \equiv tanh(x_{02})$, $i = 1, \ldots, m+1$, $\tilde{M}_{ij} \equiv \hat{a}_{j-1}^{i-1}$, $i = 1, \ldots, m+1$, $j = 2, \ldots, m+1$. Extended from the proof in (Huang & Babri, 1998), there exist x_{02} and x_{01} such that the $(m+1)\times(m+1)$ square matrix $\tilde{\mathbf{M}}$ is invertible and the assignments of eqt. (19) and eqt. (20) can accomplish the learning of m-bit parity problem with zero error.

With eqts. (16)-(20), the non-void activation set, the internal preimage, the preimage and the various inverse functions of SLFN2 are as follows:

- $\Phi_O^{-1}(y) = \{\boldsymbol{a} \mid \sum_{j=1}^{m} w_j^o a_j = y - \alpha \tanh(x_{02}),\ \begin{pmatrix} \alpha \\ \mathbf{w}^o \end{pmatrix} = \mathbf{M}^{-1}\begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^m \end{pmatrix},\ \boldsymbol{a} \in (-1, 1)^m\}$;

- *the non-void set* = $\{\boldsymbol{a} \mid a_j = \tanh(\tanh^{-1}(a_1) + (j-1)(x_{01} - x_{02})),\ 2 \le j \le m,\ a_1 \in (-1, 1)\}$, *which is a (non-linear) curve;*

- $\boldsymbol{A}(y) = \{\boldsymbol{a} \mid \sum_{j=1}^{m} w_j^o a_j = y - \alpha \tanh(x_{02}),\ a_j = \tanh(\tanh^{-1}(a_1) + (j-1)(x_{01} - x_{02})),\ 2 \le j \le m,$

$$\begin{pmatrix} \alpha \\ \mathbf{w}^o \end{pmatrix} = \mathbf{M}^{-1}\begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^m \end{pmatrix},\ a_1 \in (-1, 1)\}; \textit{ and}$$

- $f^{-1}(y) = \{\boldsymbol{x} \mid \boldsymbol{w}^T\boldsymbol{x} = \dfrac{2tanh^{-1}(a_1)}{x_{02} - x_{01}} + \dfrac{(2-m)x_{01} + mx_{02}}{x_{02} - x_{01}},\ w_1^o a_1 + \sum_{j=2}^{m} w_j^o \tanh(\tanh^{-1}(a_1) + (j-1)(x_{01} - x_{02}))$

$$= y - \alpha \tanh(x_{02}),\ \begin{pmatrix} \alpha \\ \mathbf{w}^o \end{pmatrix} = \mathbf{M}^{-1}\begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^m \end{pmatrix},\ a_1 \in (-1, 1)\}.$$

Interestingly, with $x_{02} = 0.5-m$ and $x_{01} = 0.5-m-2$, the synaptic weights obtained from eqt. (17) and (18) are the same as the synaptic weights of the network proposed in Rumelhart et al (1986, pp. 334-335) for learning the m-bit parity problem.

For the m-bit parity problem, the second alternative constructive method is summarized as follows: First take the same $\widetilde{\mathbf{w}}$ as in eqt. (16). Then let $w_0^o = 0$ and

$$\mathbf{w}_j^H \equiv \gamma_j\ \mathbf{w},\ j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor, \tag{21}$$

in which γ_j's are arbitrary constants with $\gamma_j \neq 0$ for all j and $\gamma_{j_1} \neq \gamma_{j_2}$ for all $j_1 \neq j_2$. Furthermore,

(1) when m is an odd number: let

$$w_{j0}^H \equiv 0, j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor \text{ and} \tag{22}$$

$$\mathbf{w}^o \equiv \hat{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1} \end{pmatrix}, \tag{23}$$

where $\widetilde{a}_j^k = tanh((m\text{-}2k)\gamma_j)$, $j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor$, $k = 0, \ldots, m$; $\widetilde{\mathbf{a}}^k \equiv (\widetilde{a}_1^k, \ldots, \widetilde{a}^k_{\left\lfloor \frac{m+1}{2} \right\rfloor})^T$, $k = 0, \ldots, m$; $\hat{\mathbf{M}} \equiv (\widetilde{\mathbf{a}}^0, \ldots, \widetilde{\mathbf{a}}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1})^T$; and $\hat{t}^k \equiv (-1)^{k+1} t$ for all k.

(2) when m is an even number: let

$$w_{j0}^H \equiv \gamma_j, j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor \text{ and} \tag{24}$$

$$\mathbf{w}^o \equiv \hat{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1} \end{pmatrix}, \tag{25}$$

where $\widetilde{a}_j^k = tanh((m\text{-}2k\text{+}1)\gamma_j)$, $j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor$, $k = 0, \ldots, m$; $\widetilde{\mathbf{a}}^k \equiv (\widetilde{a}_1^k, \ldots, \widetilde{a}^k_{\left\lfloor \frac{m+1}{2} \right\rfloor})^T$, $k = 0, \ldots, m$; $\hat{\mathbf{M}} \equiv (\widetilde{\mathbf{a}}^0, \ldots, \widetilde{\mathbf{a}}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1})^T$; and $\hat{t}^k \equiv (-1)^{k+1} t$ for all k.

Following the hyperplane principle (i), we learn that the normal vector of the preimage hyperplane of SLFN3 is parallel with the $\widetilde{\mathbf{w}}$ picked in eqt. (16), and that the total 2^m input patterns are mapped onto m+1 activation vectors, $\{\widetilde{\mathbf{a}}^0, \ldots, \widetilde{\mathbf{a}}^m\}$, in the activation space. Now the hyperplane principle (ii) suggests a total of

$\left\lfloor \frac{m+1}{2} \right\rfloor$ adopted hidden nodes and the eqt. (21) to set up an activation space of $\left\lfloor \frac{m+1}{2} \right\rfloor$ dimensions, in which $\tilde{a}_j^k = tanh((m\text{-}2k)\gamma_j + w_{j0}^2)$, $j = 1, \ldots, \left\lfloor \frac{m+1}{2} \right\rfloor$ and $\tilde{\mathbf{a}}^k \equiv (\tilde{a}_1^k, \ldots, \tilde{a}_{\left\lfloor \frac{m+1}{2} \right\rfloor}^k)^{\mathrm{T}}$, $k = 0, \ldots, m$. Meanwhile, this suggestion should be accompanied by a proper arrangement of w_{j0}^H's such that there are a total of $\left\lfloor \frac{m+1}{2} \right\rfloor$ linearly independent activation vectors in $\{\tilde{\mathbf{a}}^0, \ldots, \tilde{\mathbf{a}}^m\}$, and that the system of simultaneous equations in eqt. (15) is consistent. Then the $\left\lfloor \frac{m+1}{2} \right\rfloor \times \left\lfloor \frac{m+1}{2} \right\rfloor$ square matrix $\hat{\mathbf{M}}$ is invertible; the corresponding inverse matrix $\hat{\mathbf{M}}^{-1}$ exists; and the set-up weight vector $\mathbf{w}^o$ can accomplish the learning of m-bit parity problem with zero error. This suggestion and its subsequent calculation lead to the assignment of eqt. (22) and eqt. (24).

By checking all 2^m samples, it is trivial to show that SLFN3 is a solution of the m-bit parity problem. The SLFN3 solution is not surprising, since a similar solution with the sigmoid activation function of hidden nodes was given by Setiono (1997), whose study was based upon one of the results by Sontag (1992).

With eqt. (16) and eqts. (21)-(25), the non-void activation set, the internal preimage, the preimage and the various inverse functions of SLFN3 are as follows:

- $\Phi_O^{-1}(y) = \{\boldsymbol{a} \mid \sum_{j=1}^{\left\lfloor \frac{m+1}{2} \right\rfloor} w_j^o a_j = y,\ \mathbf{w}^o \equiv \hat{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1} \end{pmatrix}, \boldsymbol{a} \in (-1, 1)^{\left\lfloor \frac{m+1}{2} \right\rfloor}\}$;

- *the non-void set* = $\{\boldsymbol{a} \mid a_j = \tanh(\frac{\gamma_j}{\gamma_1} \tanh^{-1}(a_1)),\ 2 \le j \le \left\lfloor \frac{m+1}{2} \right\rfloor,\ a_1 \in (-1, 1)\}$ *which is a (non-linear) 1-manifold;*

- $\boldsymbol{A}(y) = \{\boldsymbol{a} \mid \sum_{j=1}^{\left\lfloor \frac{m+1}{2} \right\rfloor} w_j^o a_j = y,\ a_j = \tanh(\frac{\gamma_j}{\gamma_1} \tanh^{-1}(a_1)),\ 2 \le j \le \left\lfloor \frac{m+1}{2} \right\rfloor,\ \mathbf{w}^o \equiv \hat{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^0 \\ \hat{t}^1 \\ \vdots \\ \hat{t}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1} \end{pmatrix},$

$$\boldsymbol{a} \in (-1, 1)^{\left\lfloor \frac{m+1}{2} \right\rfloor}\}; \textit{ and}$$

- $$f^{-1}(y) = \{\boldsymbol{x} \mid \gamma_1 \boldsymbol{w}^T \boldsymbol{x} = \tanh^{-1}(a_1),\ \ w_1^o a_1 + \sum_{j=2}^{\left\lfloor \frac{m+1}{2} \right\rfloor} w_j^o \tanh\left(\frac{\gamma_j}{\gamma_1} \tanh^{-1}(a_1)\right) = y,\ \ \mathbf{w}^o \equiv \hat{\mathbf{M}}^{-1} \begin{pmatrix} \hat{t}^{0} \\ \hat{t}^{1} \\ \vdots \\ \hat{t}^{\left\lfloor \frac{m+1}{2} \right\rfloor - 1} \end{pmatrix},$$
$$a_1 \in (-1, 1)\}.$$

IV. Discussions and Future Work

The above preimage analysis shows that HB_SLFN, a correlated SLFN resulting from the construction method of Huang & Babri (1998), has preimages in the form of hyperplanes. In contrast, most learning algorithms (or constructive methods) in the literature normally lead to a non-correlated SLFN whose preimage is nonlinear. Meanwhile, the preimage analysis of the correlated SLFN explores hyperplane principles (i) and (ii) that lead to alternative construction methods, fitting samples perfectly but with a fewer number of hidden nodes than HB_SLFN.

In future, it is possible to systematically derive the evolution of internal-preimages and preimages of the correlated SLFN during the learning period. Based upon the knowledge of this evolution of internal-preimages and preimages, it is possible to derive a learning algorithm for the correlated SLFN that fits more accurately with less learning time.

Acknowledgements

This study is supported by the National Science Council of the R.O.C. under Grants No. NSC 92-2416-H-004-004, NSC 93-2416-H-004-015, and NSC 43028F. Thanks are due to Prof. Stuart Dreyfus for frequent discussions on this paper.

References

Andrews, R., Diederich, J., and Tickle, A. (1995). A survey and critique of techniques for extracting rules from trained artificial neural networks. Knowledge Based Systems, *8, 373-389.*

Arai, M. (1993). Bounds on the number of hidden units in binary-valued three-layer neural networks. *Neural Networks*, 6, 855-860.

Arslanov, M. Z., Ashigaliev, D. U., and Ismail, E. E. (2002). N-bit parity ordered neural networks. Neurocomputing, *48, 1053-1056.*

Browne, A., and Sun, R. (2001). Connectionist inference models. Neural Networks, *14, 1331-1355.*

Hanson, S., and Negishi, M. (2002). On the Emergence of Rules in Neural Networks. Neural Computation, *14, 2245-2268.*

Hertz, J., Krogh, A., and Palmer, R. (1991). Introduction to the Theory of Neural Computation. *Redwood City, CA: Addison-Wesley Publishing Company.*

Hohil, M. E., Liu, D. R., and Smith, S. H. (1999). Solving the N-bit parity problem using neural networks. Neural Networks, *12(11), 1321-1323.*

Huang, G., and Babri, H. (1998). Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded nonlinear activation functions. IEEE Transactions on Neural Networks, *9, 224-229.*

Huang, S.-C. and Huang, Y.-F. (1991) Bounds on the number of hidden neurons in multilayer perceptrons. IEEE Transactions on Neural Networks, *2, 47–55.*

Iyoda, E.M., Nobuhara, H., and Hirota, K. (2003) A solution for the N-bit parity problem using a single translated multiplicative neuron. Neural Processing Letters, *18 (3), 213-218.*

Lavretsky, E. (2000) On the exact solution of the Parity-N problem using ordered neural networks. Neural Networks, *13(8), 643-649.*

Liu, D.R., Hohil, M.E., and Smith, S.H. (2002) N-bit parity neural networks: new solutions based on linear programming. Neurocomputing *48, 477-488.*

Maire, F. (1999). Rule-extraction by backpropagation of polyhedra. Neural Networks, *12, 717-725.*

Munkres, J. (1975). *Topology: a first course*. New Jersey: Prentice-Hall, Englewood Cliffs.

Murty, K. (1983) *Linear Programming*. NY: John Wiley & Sons.

Rabuñal, J., Dorado, J., Pazos, A., Pereira, J., and Rivero, D. (2004). A new approach to the extraction of ANN rules and to their generalization capacity through GP. Neural Computation*, 16, 1483-1523.*

Rumelhart, D. E. & McClelland, J. (1986). Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Volume 1 , Cambridge, MA: MIT Press.*

Saito K., and Nakano, R. (2002). Extracting regression rules from neural networks. Neural Networks*, 15, 1279-1288.*

Setiono, R. (1997). On the solution of the parity problem by a single hidden layer feedforward neural network. Neurocomputing*, 16, 225-235.*

Setiono, R., Leow, W. K., and Zurada, J. M. (2002). Extraction of rules from artificial neural networks for nonlinear regression. *IEEE Transactions on Neural Networks*, 13, 564-577.

Sontag E. (1992). Feedforward nets for interpolation and classification. J. Comput. System Sci.*, 45, 20-48.*

Tsaih, R. (1998). An explanation of reasoning neural networks. *Mathematical and Computer Modelling*, 28, 37-44.

Tsaih, R., Hsu, Y., and Lai, C. (1998). Forecasting S&P 500 Stock Index Futures with the Hybrid AI System. Decision Support Systems*, 23, 161-174.*

Tsaih, R., and Wan, Y.-W. (2007). The Rule-Extraction for Neural Networks - Preimage Analysis I. *Working Paper*.

Urcid, G., Ritter, G.X., and Iancu, L. (2004) Single layer morphological Perceptron solution to the N-bit parity problem, *Lecture Notes in Computer Science*, 3287, 171-178.

計畫成果自評：

此研究計畫成果豐碩，已被送到相關研討會發表。其延伸之研究亦在進行中。