Design of GBSB Neural Associative Memories Using Semidefinite Programming

Jooyoung Park, Hyuk Cho, and Daihee Park

Abstract—**This paper concerns reliable search for the optimally performing GBSB (generalized brain-state-in-a-box) neural associative memory given a set of prototype patterns to be stored as stable equilibrium points. First, we observe some new qualitative properties of the GBSB model. Next, we formulate the synthesis of GBSB neural associative memories as a constrained optimization problem. Finally, we convert the optimization problem into a semidefinite program (SDP), which can be solved efficiently by recently developed interior point methods. The validity of this approach is illustrated by a design example.**

Index Terms—**Associative memory, convex optimization, generalized brain-state-in-a-box, linear matrix inequality, semidefinite program.**

I. Introduction

RECENT interest in neural associative memories has been spurred by the seminal work of Hopfield [1], who has shown that fully interconnected feedback neural networks trained by the Hebbian learning rule can function as a new concept of associative memories. Since then, numerous neural-network models have been proposed with synthesis methods for realizing associative memories [2]. Also, many studies on how well they perform as associative memories appeared. In general, the desirable characteristics emphasized in the performance evaluation include the following [2]–[4]: asymptotic stability of each prototype pattern; large domain of attraction for each prototype pattern; small number of stable equilibrium points that do not correspond to prototype patterns (i.e., spurious states); global stability.

Among the various kinds of promising neural models showing good performance is the so-called BSB (brain-state-in-a-box). This model was first proposed by Anderson *et al.* in 1977 [5], and has been regarded as particularly suitable for implementing associative memories. In this paper, we focus on developing a synthesis procedure for associative memories based on an advanced form of the BSB model, which is often referred to as generalized BSB (GBSB). The GBSB model was proposed and studied by Hui and Zak [6], and is now considered to be more appropriate for realizing associative memories than the BSB model in several respects [3], [7]. Both the BSB and GBSB have been studied extensively as effective tools for realizing associative memories. Perfetti [8] analyzed qualitative properties of the BSB model and formulated the design of BSB-based associative memories as a constrained optimization in the form of a linear programming with an additional nonlinear constraint. Also, he proposed an iterative algorithm to solve the constrained optimization problem, and illustrated the algorithm with some design examples. Lillo *et al.* [3] analyzed the dynamics of the GBSB model, and presented a novel synthesis procedure for GBSB-based associative memories. Their procedure utilizes a decomposition of interconnection matrix, which results in asymmetric interconnection structure, asymptotic stability of the desired memory patterns, and small number of spurious states. In [9], they incorporated the learning and forgetting capabilities into the synthesis method of [3]. Also, in [7], they proposed "designer" neural network for the synthesis of GBSB-based associative memories.

In this paper, we derive some guidelines for the synthesis of GBSB-based associative memories, and formulate the synthesis as a constrained optimization problem. Next, we convert the optimization problem into an SDP[1] (semidefinite program), which consists of a linear objective and constraints in the form of LMI's (linear matrix inequalities). Since efficient interior point algorithms are now available to solve SDP's (i.e., find the global optimum of a given SDP efficiently within a given tolerance or find a certificate of infeasibility) [10]–[12], recasting the synthesis problem to an SDP is equivalent to finding a solution to the original problem. In this paper, MATLAB LMI Control Toolbox [12] is used as an optimum searcher to solve the synthesis problem formulated as an SDP.

Throughout this paper, we use the following definitions and notation, in which R^n denotes the normed linear space of real n-vectors with the Euclidean norm $\|\cdot\|$. A symmetric matrix $\mathbf{A} \in R^{n\times n}$ is positive definite if $\mathbf{x}^T\mathbf{A}\mathbf{x} > 0$ for any $\mathbf{x} \neq \mathbf{0}$, and $\mathbf{A} > \mathbf{0}$ denotes this. Also, $\mathbf{A} > \mathbf{B}$ denotes that $\mathbf{A}$–$\mathbf{B}$ is positive definite. For a symmetric matrix $\mathbf{W} \in R^{n\times n}$, $\lambda_{\min}(\mathbf{W})$ and $\|\mathbf{W}\|$ denote the minimum eigenvalue and the induced norm defined by $\max_{\mathbf{x}\neq\mathbf{0}} \|\mathbf{W}\mathbf{x}\|/\|\mathbf{x}\|$, respectively. H_n denotes the hypercube $[-1, +1]^n$. By a binary vector, we mean a bipolar binary vector whose element is either $+1$ or -1, and the set of all these binary vectors in H_n is denoted by B_n. The usual Hamming distance between two vectors $\mathbf{x}^*$ and $\mathbf{x}$ in B_n is denoted by $H(\mathbf{x}^*, \mathbf{x})$.

This paper is organized as follows: In Section II, we briefly review the fundamentals on GBSB, and present some new qualitative properties of the GBSB model. Also, based on these, we formulate the synthesis of GBSB-based associative memories as a constrained optimization problem. In Section III, we show the process for the recast of the opti-

Manuscript received July 7, 1998; revised January 20, 1999.

J. Park is with the Department of Control and Instrumentation Engineering, Korea University, Chochiwon, Chungnam, 339-800, Korea.

H. Cho and D. Park are with the Department of Computer Science, Korea University, Chochiwon, Chungnam, 339-800, Korea.

Publisher Item Identifier S 1045-9227(99)05974-3.

[1] In general, SDP means "semidefinite programming" or a "semidefinite program" (i.e., a semidefinite programming problem).

mization problem into an SDP, and consider a design example to illustrate the validity of the SDP-based approach. With the concrete simulation results from the design experiment, we compare the performance of the GBSB designed by the proposed method with the associative memories designed by other methods. Finally, in Section IV, concluding remarks are given.

II. Background Results

The dynamics of the GBSB model is described by the following state equation:

$$\mathbf{x}(k+1) = g(\mathbf{x}(k) + \alpha(\mathbf{W}\mathbf{x}(k) + \mathbf{b})) \tag{1}$$

where $\mathbf{x}(k) \in R^n$ is the state vector at time k, $\alpha > 0$ is the step size, $\mathbf{W} \in R^{n\times n}$ is the weight matrix, $\mathbf{b} \in R^{n\times 1}$ is the bias vector, and g: $R^n \to R^n$ is a linear saturating function whose ith component is a function defined as follows:

$$g_i([x_1 \cdots x_i \cdots x_n]^T) = \begin{cases} 1 & \text{if } x_i \geq 1 \\ x_i & \text{if } -1 < x_i < 1 \\ -1 & \text{if } x_i \leq -1. \end{cases}$$

The GBSB model is a generalized version of the BSB network proposed by Anderson *et al.* [5] and it differs from the original network for the presence of the bias vector $\mathbf{b}$.

In the discussion on the stability of the GBSB model, we use the following definitions.

- A point $\mathbf{x_e} \in R^n$ is an equilibrium point of system (1) if $\mathbf{x}(0) = \mathbf{x_e}$ implies $\mathbf{x}(k) = \mathbf{x_e}$, $\forall k > 0$.
- An equilibrium point $\mathbf{x_e}$ of system (1) is stable if for any $\epsilon > 0$, there exists $\delta > 0$ such that

$$\|\mathbf{x}(0) - \mathbf{x_e}\| < \delta \text{ implies } \|\mathbf{x}(k) - \mathbf{x_e}\| < \epsilon, \qquad \forall k > 0.$$

- An equilibrium point $\mathbf{x_e}$ of system (1) is asymptotically stable if it is stable and there exists $\delta > 0$ such that

$$\mathbf{x}(k) \to \mathbf{x_e} \text{ as } k \to \infty \text{ if } \|\mathbf{x}(0) - \mathbf{x_e}\| < \delta.$$

- System (1) is globally stable if every trajectory of the system converges to some equilibrium point.

The criteria on the stability of the GBSB model are now well established [3], [13], [14].

- A vertex $\mathbf{x}$ of the hypercube H_n is an equilibrium point of system (1) if and only if

$$x_i\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) \geq 0, \qquad \forall i \in \{1, \cdots, n\}. \tag{2}$$

- A vertex $\mathbf{x}$ of the hypercube H_n is an asymptotically stable equilibrium point of system (1) if

$$x_i\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) > 0, \qquad \forall i \in \{1, \cdots, n\}. \tag{3}$$

- System (1) is globally stable if the weight matrix $\mathbf{W}$ is symmetric and

$$\lambda_{\min}(\mathbf{I} + \alpha\mathbf{W}) > -1. \tag{4}$$

In general, designing neural associative memories based on the stability criteria only does not result in satisfactory results. Additional guidelines should be provided to address other performance indexes such as the size of the domain of attraction for each prototype pattern. In [8], Perfetti proposed some guidelines for the BSB model based on the conjecture that the absence of equilibrium points near stored patterns would increase their domains of attraction, and the experimental results there showed that such strategy is very effective in reducing the number of spurious states as well as in increasing the attraction basins for prototype patterns. In this paper, we utilize the same strategy, and the GBSB counterparts of Perfetti's results [8] are given in the following.

Lemma 1: Let $\mathbf{x} \in B_n$ be an asymptotically stable equilibrium point of GBSB (1), and let $\mathbf{x}^* \in B_n$ be a vertex which differs from $\mathbf{x}$ in the component x_i for $i \in J \subset \{1, \cdots, n\}$. If

$$x_i\left(\sum_{j=1}^{n} w_{ij}x_j^* + b_i\right) > 0 \tag{5}$$

for at least one integer $i \in J$, then $\mathbf{x}^*$ is not an equilibrium point.

Proof: Consider an integer $i \in J$. By condition (5), we have

$$x_i^*\left(\sum_{j=1}^{n} w_{ij}x_j^* + b_i\right) = -x_i\left(\sum_{j=1}^{n} w_{ij}x_j^* + b_i\right) < 0.$$

Thus, by criterion (2), the vertex $\mathbf{x}^*$ is not an equilibrium point.

Corollary 1: Suppose that $\mathbf{x} \in B_n$ is an asymptotically stable equilibrium point of GBSB (1). If $w_{ii} = 0$ for $i = 1, \cdots, n$, then none of the vertices $\mathbf{x}^*$ such that $H(\mathbf{x}^*, \mathbf{x}) = 1$ is an equilibrium point.

Proof: Assume that $x_i^* = -x_i$ and $x_j^* = x_j$ for $\forall j \neq i$. Since $\mathbf{x}$ is an asymptotically stable equilibrium point, we have

$$\begin{aligned} x_i\left(\sum_{j=1}^{n} w_{ij}x_j^* + b_i\right) &= x_i\left(\sum_{j\neq i} w_{ij}x_j^* + b_i\right) \\ &= x_i\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) > 0. \end{aligned}$$

By Lemma 1, this implies that $\mathbf{x}^*$ is not an equilibrium point.

Corollary 2: Suppose that $\mathbf{x} \in B_n$ is an asymptotically stable equilibrium point of GBSB (1) and that k is an integer in $\{1, \cdots, n\}$. If

$$x_i\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) > 2k \max_j |w_{ij}|, \qquad \forall i \in \{1, \cdots, n\} \tag{6}$$

then none of the vertices $\mathbf{x}^*$ satisfying $0 < H(\mathbf{x}^*, \mathbf{x}) \leq k$ is an equilibrium point.

Proof: Let $\mathbf{x}^* \in B_n$ be a vertex satisfying $0 < H(\mathbf{x}^*, \mathbf{x}) \leq k$, and let i be an integer such that $x_i^* \neq x_i$. Note that $\delta \triangleq \mathbf{x}^* - \mathbf{x}$ satisfies

$$\left|\sum_{j=1}^{n} w_{ij}\delta_j\right| \leq 2k \max_j |w_{ij}|. \tag{7}$$

By (6) and (7), we have

$$\begin{aligned} x_i\left(\sum_{j=1}^{n} w_{ij}x_j^* + b_i\right) &= x_i\left\{\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) + \sum_{j=1}^{n} w_{ij}\delta_j\right\} \\ &\geq x_i\left(\sum_{j=1}^{n} w_{ij}x_j + b_i\right) - 2k \max_j |w_{ij}| \\ &> 0 \end{aligned}$$

which establishes the corollary by Lemma 1.

Remark 1: The zero-diagonal condition of Corollary 1 also guarantees that only binary steady states are to be observed [8].

Remark 2: Corollary 2 implies that the maximization of the left-hand side of (6) generally leads to better performance with respect to the size of the domain of attraction for each prototype pattern and the number of spurious states. However, as noted in [8], this maximization should be sought under the condition of the boundedness of the weight matrix $\mathbf{W}$.

Given the set of binary prototype patterns $\mathbf{x}^{(1)}, \cdots, \mathbf{x}^{(m)}$, the background results outlined above allow us to find the GBSB performing optimally by solving

$$\begin{aligned} \max \quad & \delta \\ \text{s.t.} \quad & x_i^{(k)}\left(\sum_{j=1}^{n} w_{ij}x_j^{(k)} + b_i\right) - \delta > 0, \qquad i = 1, \cdots, n \\ & \quad k = 1, \cdots, m \\ & w_{ii} = 0, i = 1, \cdots, n \\ & \mathbf{W} = \mathbf{W}^T \\ & \|\mathbf{W}\| < c \\ & \lambda_{\min}(\mathbf{I} + \alpha\mathbf{W}) > -1 \end{aligned} \tag{8}$$

where α and c are the constants describing the step size and the norm-bound of $\mathbf{W}$, respectively. The best part of this formulation is that it can be transformed into a semidefinite program, thus its global optimum can be found by interior point methods.

III. SDP-Based Synthesis and a Design Example

In this section, we establish an SDP-based synthesis procedure for the GBSB neural associative memories by transforming the nonlinear constraints of (8) into LMI's, and present an example to demonstrate the applicability of the procedure.

An LMI is any constraint of the form

$$\mathbf{A}(\mathbf{z}) \triangleq \mathbf{A_0} + z_1\mathbf{A_1} + \cdots + z_N\mathbf{A_N} > \mathbf{0} \tag{9}$$

where $\mathbf{z} \triangleq [z_1 \cdots z_N]^T$ is the variable, and $\mathbf{A_0}, \cdots, \mathbf{A_N}$ are given symmetric matrices. Since $\mathbf{A}(\mathbf{x}) > \mathbf{0}$ and $\mathbf{A}(\mathbf{y}) > \mathbf{0}$ imply that $\mathbf{A}((\mathbf{x}+\mathbf{y})/2) > \mathbf{0}$, LMI (9) is a convex constraint on the variable $\mathbf{z}$. Note that multiple LMI's $\mathbf{A}^{(1)}(\mathbf{z}) > \mathbf{0}, \cdots, \mathbf{A}^{(p)}(\mathbf{z}) > \mathbf{0}$ can be expressed as the single LMI $\mathbf{diag}(\mathbf{A}^{(1)}(\mathbf{z}), \cdots, \mathbf{A}^{(p)}(\mathbf{z})) > \mathbf{0}$. Thus, there is no distinction between a set of LMI's and a single LMI. It is well-known that an optimization problem with a linear objective and LMI constraints, which is called a semidefinite program, can be efficiently solved by interior point methods [10]–[12], and a toolbox of MATLAB for convex problems involving LMI's is now available [12].

Optimization problem (8) has not only linear constraints but also nonlinear constraints, which prevent us from applying a linear programming technique such as the simplex method. However, the nonlinear constraints can be easily converted to LMI's. First, since the matrix norm condition $\|\mathbf{W}\|^2 < c^2$ is equivalent to

$$\mathbf{x}^T\mathbf{W}^T\mathbf{W}\mathbf{x} < \mathbf{x}^T(c^2\mathbf{I})\mathbf{x}, \qquad \forall \mathbf{x} \neq \mathbf{0}$$

this constraint can be reduced to

$$c\mathbf{I} - \mathbf{W}^T(c\mathbf{I})^{-1}\mathbf{W} > \mathbf{0}. \tag{10}$$

Utilizing the Schur complement [11], we can see that (10) is equivalent to the following LMI:

$$\begin{bmatrix} c\mathbf{I} & \mathbf{W} \\ \mathbf{W}^T & c\mathbf{I} \end{bmatrix} > \mathbf{0}.$$

Next, consider the eigenvalue condition $\lambda_{\min}(\mathbf{I}+\alpha\mathbf{W}) > -1$. Since $\mathbf{I} + \alpha\mathbf{W}$ is real symmetric, its eigenvalues are real, and corresponding eigenvectors can be chosen to be real orthonormal [15]. Thus, its spectral decomposition [15] can be written as

$$\mathbf{I} + \alpha\mathbf{W} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$$

where the eigenvalues of $\mathbf{I}+\alpha\mathbf{W}$ appear on the diagonal of $\mathbf{\Lambda}$, and $\mathbf{U}$, whose columns are the real orthonormal eigenvectors of $\mathbf{I} + \alpha\mathbf{W}$, satisfies $\mathbf{U}\mathbf{U}^T = \mathbf{U}^T\mathbf{U} = \mathbf{I}$. Note that $\lambda_{\min}(\mathbf{I} + \alpha\mathbf{W}) > -1$ is equivalent to

$$\mathbf{\Lambda} > -\mathbf{I}. \tag{11}$$

By pre- and postmultiplying matrix inequality (11) by $\mathbf{U}$ and $\mathbf{U}^T$, respectively, we can obtain an equivalent condition $\mathbf{I}+\alpha\mathbf{W} > -\mathbf{I}$, which is again equivalent to the following LMI:

$$2\mathbf{I} + \alpha\mathbf{W} > \mathbf{0}.$$

Therefore, optimization (8) can be transformed into the following semidefinite program:

$$\begin{aligned} \min \quad & -\delta \\ \text{s.t.} \quad & x_i^{(k)}\left(\sum_{j=1}^{n} w_{ij}x_j^{(k)} + b_i\right) - \delta > 0, \qquad i = 1, \cdots, n \\ & \quad k = 1, \cdots, m \\ & w_{ii} = 0, \qquad i = 1, \cdots, n \\ & \mathbf{W} = \mathbf{W}^T \\ & \begin{bmatrix} c\mathbf{I} & \mathbf{W} \\ \mathbf{W}^T & c\mathbf{I} \end{bmatrix} > \mathbf{0} \\ & 2\mathbf{I} + \alpha\mathbf{W} > \mathbf{0}. \end{aligned} \tag{12}$$

Note that the constraints of (12) can be rewritten in the cannonical form of LMI (9) by defining $z_1, \cdots, z_N$ as the

TABLE I
THE WEIGHT MATRIX **W** OBTAINED BY THE PROPOSED METHOD

0.000	-0.072	-0.242	-0.286	-0.349	0.027	0.045	-0.045	-0.072	0.170
-0.072	0.000	0.035	0.133	-0.072	-0.252	-0.039	0.039	0.203	0.272
-0.242	0.035	0.000	0.134	-0.242	-0.171	0.067	-0.067	0.035	-0.249
-0.286	0.133	0.134	0.000	-0.286	0.103	-0.195	0.195	0.133	-0.199
-0.349	-0.072	-0.242	-0.286	0.000	0.027	0.045	-0.045	-0.072	0.170
0.027	-0.252	-0.171	0.103	0.027	0.000	-0.274	0.274	-0.252	-0.082
0.045	-0.039	0.067	-0.195	0.045	-0.274	0.000	-0.202	-0.039	-0.184
-0.045	0.039	-0.067	0.195	-0.045	0.274	-0.202	0.000	0.039	0.184
-0.072	0.203	0.035	0.133	-0.072	-0.252	-0.039	0.039	0.000	0.272
0.170	0.272	-0.249	-0.199	0.170	-0.082	-0.184	0.184	0.272	0.000

independent scalar entries of δ and $\mathbf{W}$ satisfying $w_{ii} = 0, i = 1, \cdots, n$, and $\mathbf{W} = \mathbf{W}^T$. However, leaving LMI's in a condensed form as in (12) not only saves notation, but also leads to more efficient computation. Several softwares are readily available for solving this problem. An example is the "mincx" function of MATLAB LMI Control Toolbox, which is used as an optimum searcher in this study.

To evaluate the performance of the proposed method, we consider a design example. The dimension of the GBSB model to be considered is $n = 10$, and we wish to store the following five prototype patterns:

$$\mathbf{x}^{(1)} = [-1 \ \ +1 \ \ -1 \ \ +1 \ \ +1 \ \ +1 \ \ -1 \ \ +1 \ \ +1 \ \ +1]^T$$
$$\mathbf{x}^{(2)} = [+1 \ \ +1 \ \ -1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ +1]^T$$
$$\mathbf{x}^{(3)} = [-1 \ \ +1 \ \ +1 \ \ +1 \ \ -1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ -1]^T$$
$$\mathbf{x}^{(4)} = [+1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ -1 \ \ +1 \ \ +1 \ \ +1]^T$$
$$\mathbf{x}^{(5)} = [+1 \ \ -1 \ \ -1 \ \ -1 \ \ +1 \ \ +1 \ \ +1 \ \ -1 \ \ -1 \ \ -1]^T.$$

This is the same set of prototype patterns that was considered in [7] and [8]. Solving the corresponding SDP with $c = 0.7$ and $\alpha = 0.3$, we obtained $\delta = 0.3489$ together with the weight matrix $\mathbf{W}$ shown in Table I and the bias vector

$$\mathbf{b} = [+0.081 \quad +0.606 \quad -0.701 \quad +0.391 \quad +0.081 \quad +0.475 \quad +0.370 \quad -0.370 \quad +0.606 \quad -0.270]^T.$$

To evaluate the performance of the resulting GBSB, we performed simulations for all possible initial binary states, and summarized the information on the domain of attraction for each prototype pattern in Table II. The entries of the table should be interpreted as follows: "(the entry corresponding to $\mathbf{x}^{(k)}$ and $H = p) = q$" indicates that, out of all possible initial binary states with Hamming distance p, q of them converge to the prototype pattern $\mathbf{x}^{(k)}$. Obviously, having large entries in the table indicates a desirable feature with respect to the domains of attraction for prototype patterns. For the comparison purpose, we performed the same simulations[2] for the GBSB of [7] and for the BSB[3] based on [8], and the results are shown in Tables III and IV, respectively. Note that the entries in the first column of Tables II–IV are all one, which shows that each of the given prototype patterns is stored as a stable equilibrium point in all three cases. Also, note that entries in Table II are comparable to those in Tables III and IV.

To compare the capabilities of these associative memories more concretely, we divided all possible initial binary states into four classes in accordance with the final results of their

TABLE II
DOMAINS OF ATTRACTION FOR THE GBSB DESIGNED BY THE PROPOSED METHOD

	H=0	H=1	H=2	H=3	H=4
$\mathbf{x}^{(1)}$	1	9	34	68	49
$\mathbf{x}^{(2)}$	1	10	45	94	77
$\mathbf{x}^{(3)}$	1	10	44	82	87
$\mathbf{x}^{(4)}$	1	9	34	68	60
$\mathbf{x}^{(5)}$	1	10	39	49	28

TABLE III
DOMAINS OF ATTRACTION FOR THE GBSB OF [7]

	H=0	H=1	H=2	H=3	H=4
$\mathbf{x}^{(1)}$	1	9	30	58	51
$\mathbf{x}^{(2)}$	1	10	38	82	86
$\mathbf{x}^{(3)}$	1	10	43	67	43
$\mathbf{x}^{(4)}$	1	8	32	67	60
$\mathbf{x}^{(5)}$	1	10	41	55	37

TABLE IV
DOMAINS OF ATTRACTION FOR THE BSB BASED ON [8]

	H=0	H=1	H=2	H=3	H=4
$\mathbf{x}^{(1)}$	1	10	34	40	11
$\mathbf{x}^{(2)}$	1	10	37	58	17
$\mathbf{x}^{(3)}$	1	10	35	45	20
$\mathbf{x}^{(4)}$	1	8	30	38	11
$\mathbf{x}^{(5)}$	1	10	36	32	12

[2] In these simulations, the step size was set to $\alpha = 0.3$.

[3] In the synthesis procedure of [8], α is fixed at one. For a meaningful comparison, a BSB network based on [8] has been redesigned with $\alpha = 0.3$.

TABLE V
A COMPARISON OF CONVERGENCE FROM INITIAL BINARY PATTERNS

Number of binary states	GBSB of this paper	GBSB of [7]	BSB based on [8]
# of the best	921	820	502
# of the good	103	140	10
# of the bad	0	0	512
# of the failed	0	64	0

evolutions, and counted the binary states in each class. In Table V, "# of the best" denotes the number of the initial states whose evolutions converge to the closest prototype pattern (in the sense of Hamming distance); "# of the good" denotes the number of the initial states whose evolutions converge to a prototype pattern, but not to the closest one; "# of the bad" denotes the number of the initial states whose evolutions converge to a binary spurious state; and "# of the failed" denotes the number of the initial states whose evolutions fail to reach a vertex. As shown in Table V, the results of the proposed method turned out to be better than those of [7] and [8]. Compared to the GBSB of [7], the GBSB of this paper has the following advantages.

- *Guaranteed Convergence to Prototype Patterns*: The simulation results for the GBSB of this paper have shown that each trajectory starting from an initial binary state converges to a prototype pattern, while 64 out of 1024 initial binary patterns failed to reach a vertex in the simulations for the GBSB of [7].
- *Domains of Attraction in Good Shape*: In the simulations for the GBSB of this paper, almost 90% of initial binary patterns converged to the closest prototype pattern, while this rate dropped to about 80% in the case of the GBSB of [7].

Also, the GBSB of this paper outperforms the BSB based on [8] in the following respects.

- *No Spurious State*: The GBSB of this paper has no spurious state at all, while five spurious states, which consist of the negatives of the prototype patterns, were observed in the simulations for the BSB based on [8].
- *Large Domains of Attraction*: In the case of the BSB based on [8], many of initial binary patterns converged to spurious states.[4] On the other hand, the GBSB designed by the proposed method is not hampered by this problem, thus its prototype patterns are endowed with large domains of attraction.

IV. CONCLUDING REMARKS

In this paper, we have addressed problems concerning the reliable synthesis for the optimally performing GBSB neural associative memories given a set of prototype patterns to be stored as stable equilibrium points. Based on known results and newly derived properties of the GBSB model, the design of GBSB-based associative memories was formulated as a constrained optimization problem. Also, by transforming its nonlinear constraints into linear matrix inequalities, the optimization problem was recast to a semidefinite program. This recast is particularly useful in practice, because the interior point methods which can solve semidefinite programs are readily available. The GBSB's designed by the proposed method have many desirable features: Each prototype pattern can be surely stored as an asymptotically stable equilibrium point; global stability is guaranteed; the negatives of given prototype patterns are not automatically stored as stable equilibrium points; a performance index related with the size of domains of attraction for prototype patterns is optimized, thus large attraction basin is expected for each prototype pattern; near the stored prototype patterns, there are no spurious states. A design example was presented to illustrate the proposed method, and the resulting GBSB validated all of the above advantages by outperforming the associative memories designed by other recently developed techniques.

[4] Note that in the BSB model, the negatives of the stored prototype patterns are automatically stored. So the 512 initial states converging to these negatives may not be viewed as errors.

REFERENCES

[1] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," in *Proc. Nat. Academy Sci.*, vol. 79, pp. 2554–2558, 1982.

[2] A. N. Michel and J. A. Farrell, "Associative memories via artificial neural networks," *IEEE Contr. Syst. Mag.*, vol. 10, pp. 6–17, 1990.

[3] W. E. Lillo, D. C. Miller, S. Hui, and S. H. Zak, "Synthesis of brain-state-in-a-box (BSB) based associative memories," *IEEE Trans. Neural Networks*, vol. 5, pp. 730–737, 1994.

[4] G. G. Yen, "Unlearning algorithm in associative memory," in *Proc. 1995 World Congr. Neural Networks*, Washington, D.C., 1995, pp. 1466–1470.

[5] J. A. Anderson, J. W. Silverstein, S. A. Ritz, and R. S. Jones, "Distinctive features, categorical perception, and probability learning: Some applications of a neural model," *Psych. Rev.*, vol. 84, pp. 413–452, 1977.

[6] S. Hui and S. H. Zak, "Dynamical analysis of the brain-state-in-a-box (BSB) neural models," *IEEE Trans. Neural Networks*, vol. 3, pp. 86–100, 1992.

[7] H. Y. Chan and S. H. Zak, "On neural networks that design neural associative memories," *IEEE Trans. Neural Networks*, vol. 8, pp. 360–372, 1997.

[8] R. Perfetti, "A synthesis procedure for brain-state-in-a-box neural networks," *IEEE Trans. Neural Networks*, vol. 6, pp. 1071–1080, 1995.

[9] S. H. Zak, W. E. Lillo, and S. Hui, "Learning and forgetting in generalized brain-state-in-a-box (BSB) neural associative memories," *Neural Networks*, vol. 9, pp. 845–854, 1996.

[10] B. Jansen, *Interior Point Techniques in Optimization: Complementarity, Sensitivity and Algorithms.* Dordrecht, The Netherlands: Kluwer, 1997.

[11] S. Boyd, L. ElGhaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory.* Philadelphia, PA: SIAM, 1994.

[12] P. Gahinet, A. Nemirovski, A. J. Laub, and M. Chilali, *MATLAB LMI Control Toolbox.* Natick, MA: MathWorks Inc., 1995.

[13] A. N. Michel, J. Si, and G. Yen, "Analysis and synthesis of a class of discrete-time neural networks described on hypercubes," *IEEE Trans. Neural Networks*, vol. 2, pp. 32–46, 1991.

[14] R. M. Golden, "Stability and optimization analysis of the generalized brain-state-in-a-box neural network model," *J. Math. Psych.*, vol. 37, pp. 282–298, 1993.

[15] G. Strang, *Linear Algebra and Its Applications*, 3rd ed. Orlando, FL: HBJ, 1988.