Proceedings of the 1994 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI'94) • Las Vegas, NV • Oct. 2-5, 1994

New Hyper-Distributed Hyper-Parallel AI Approach Based on Chaotic Bifurcations and Synchronizations

Dianxun Shuai * Yoichiro Watanabe
Department of Computer Science and Technology
Xian University of Electronic Science and Technology
Box 710071
Xian, Shaanxi, China
Department of Knowledge Engineering and Computer Science, Faculty of Engineering
Doshisha University
Box 610-03
Tanabe, Kyoto, Japan

Abstract– For problem solving in the artificial intelligence, this paper presents a new hyper-distributed hyper-parallel approach based on the bifurcations and synchronizations of the hierarchical distributed chaotic dynamic systems. By using Chua's circuits arrays. the realization of the hyper-distributed hyper-parallel heuristic algorithms for real-time search of any implicit AND/OR graph is discussed. The approach not only combines the advantages of both the traditional sequential symbolic logic and the conventional neural network approaches, but also overcomes their drawbacks in many respects.

I. INTRODUCTION

We can decompose a lot of complicated problems, in many different ways, into the equivalent combinations of such subproblems that can also be decomposed further unless entirely simplified into the primitive unsolvable or the primitive solvable problems. By this approach, problem solving eventually results in the search of an equivalent implicit AND/OR graph. The efficient search is one of the most fundamental and crucial to the artificial intelligence (AI) and the knowledge engineering (KE).

Many search algorithms based on the traditional sequential symbolic logic have been proposed so far for robot motion planning, job scheduling, disease or defect diagnosing, knowledge data-base retrieving, programming language parsing, game playing, theorem proving, natural language comprehension, , pattern recognition, and so forth. Although the algorithms have the advantages in the respects, such as the determinate acquisition of existential optimal solutions and the easy use of some heuristic knowledge to guide the search, they almost can but be carried out sequentially with complex control strategies, lower processing efficiency and great difficulties in the systolic implementation.

On the other hand, however, the distributed parallel approaches to the search and the optimization, based on the ordinary neural networks like the Hopfield recursive network or the feedforward multilayer networks, might neither guarantee the network, for any initial state and any input, always to converge to the stable states with the minimum energy corresponding to the optimal solutions, nor real-timely obtain the solutions. Sometimes only approximate solutions as the compromises between the optimal and the feasible solutions might be obtained. Many parameters of the neural networks must be determined by some time-consuming simulations or by experiences. Furthermore, it is very hard to introduce, as the case may be during the solving process, some heuristic knowledge into this kind of neural networks.

As for the real-time search, most of the existent AI approaches, based either on the symbolic logic or on the neural networks, have a common essential drawback that at a time of solving process only one state of the state space can be provided as inputs, whereby the disposal of the state space is still sequential virtually.

For the real-time search of any implicit AND/OR graph, authors have presented in [1] and [2] the search concepts of propagations and competitions of concurrent waves, the heuristic distributed parallel algorithms, $\mathcal{P}^*$, $\mathcal{P}_h^*$ and $\mathcal{P}_{AO}^*$, and the implementation scheme based on dynamical clustering competitive neural networks. Nevertheless, the conveyances of cost function values and heuristic knowledge values between cells are implemented implicitly via universal clocks, whereby there are still many connections

*Supported by the CNNSF under Grant No.69073316, the CNHTDP under No.5120402 and the research foundation of Doshisha University in Japan.

0-7803-2072-7/94 $4.00 © 1994 IEEE

between the clocks-coding unit and every neural cell. Furthermore, for state transitions and controls, every cluster of neural cells has comparatively complicated structure.

Under the above mentioned circumstances, this paper presents a new hyper-distributed hyper-parallel heuristic approach based on the bifurcations and the synchronizations of hierarchical distributed chaotic dynamic systems. The approach can be used for complicated problems solving in the AI and the KE areas. By using Chua's circuits arrays, the waves' concurrent propagations and competitions required by the hyper-distributed hyper-parallel heuristic algorithms for real-time search of any implicit graph are realized. Compared with both the traditional sequential symbolic logic and the conventional neural network approaches, the approach of this paper combines their advantages and overcomes their drawbacks in many aspects, so that it is characterized by the hyper-distributed hyper-parallel real-time processing performance, successful acquisition of the existential optimal solutions, easy utilization of the heuristic knowledge during solving process, simultaneous offering of massive states of the state space as inputs at a time in the solving process, local connections between processing cells, rather simple structures of networks and neurons, the feasibility of the VLSI implementation, and so on.

II. Discrete Nonlinear Dynamic Model For $\mathcal{M}^*$ And Search

The problem $\mathcal{M}^*$ is defined by

$$g(x, a(x)) = \min_{a} \{c(a(x)) + g(y, a(y))\}, \qquad (1)$$

where $x, y \in (\Im \cup \aleph)$; $a(x), a(y) \in \wp$; $\Im$ and $\aleph$ are the finite sets of the terminal states and nonterminal states, respectively; $\wp$ is a set of operators. In order to obtain the minimum $g(x_0, a(x_0))$ for a given initial state x_0, the best operator $a(x)$ in response to state x should be taken, which generates the immediate cost $c(a(x))$ and the next state y.

The model can be used for formalizing many problems in the areas, such as dynamic programming, machine learning, optimal control, information coding, software engineering, automation theory, and artificial intelligence.

As for AI problem solving, searching an implicit AND/OR graph $\mathcal{G}$ results in

$$g(n, a(n)) \stackrel{\mathcal{G}}{=} \min_{a} \{ \sum_{p_1(a(n))}^{p_k(a(n))} c(n, n_i) + \sum_{p_1(a(n))}^{p_k(a(n))} g(n_i, a(n_i)) \}, \qquad (2)$$

where edges (n, n_i) with the cost value $c(n, n_i)$, for $i = p_1(a(n)), \cdots, p_k(a(n))$, belong to the same hyper-edge of node n, and k is the possible maximum number of the edges of a hyper-edge. The set composed of all the primitive solvable and primitive unsolvable problem nodes corresponds to $\Im$ of model $\mathcal{M}^*$, with $a(n) = \phi$, and $g(n, a(n)) = 0$ or $g(n, a(n)) = \infty$ for primitive solvable or primitive unsolvable node n, respectively. The empty operator ϕ implies doing nothing. The best operator $a(n)$ means the optimal hyper-edge selection from the set of hyper-edges of n.

In terms of power set of state nodes, the above AI search model can be regarded as a discrete dynamical system defined by the state equation

$$\mathbf{X}_{t+1} = \mathcal{A}(\mathbf{X}_t), t = 0, 1, 2, \ldots, \qquad (3)$$

where $\mathbf{X} \in 2^{R^m}$ is the state set within the power set of states $n \in R^m$, and $\mathcal{A}$ maps the current state set $\mathbf{X}_t$ to the next state set $\mathbf{X}_{t+1}$ under the optimal constraint shown in (2). Starting with an initial state n_0, if repeated applications of $\mathcal{A}$ generate such obits that can't extend any longer and has finite value of total costs, thus search process finishes successfully. Nevertheless, because the map function $\mathcal{A}$ could be expressed only formally in an implicit form, it is very difficult and almost impossible for the map $\mathcal{A}$ to be found both previously and explicitly, and then directly to be used in AI problem solving.

Consequently, we propose a hyper-distributed hyper-parallel representation, a state of $\mathbf{X}_t$ expressed by a processing cell and the mapping $\mathcal{A}$ embodied by the changes of the states of processing cells. The processing cells are organized as the hierarchic arrays, each processing cell is a chaotic nonlinear dynamic system composed of the Chua's circuits.

III. Hierarchic Distributed Chaotic Model For AI Problem Solving

A. Architecture of the System

The architecture of the hyper-distributed hyper-parallel implementation of AI problem solving, shown in Fig. 1, basically consists of three sections : the cellular neural network section (CNNS), the evaluation section (ES) and the generator section (GS). The CNNS makes use of the bifurcation phenomena and synchronization mechanisms of the distributed chaotic dynamical systems composed of the two-layer two-dimension arrays of Chua's circuits, in order to implement the concurrent propagations of states wave, the concurrent transmissions of cost function values and heuristic values, and the competitive activation mechanisms. The ES simultaneously provides, in terms of the bias current sources, every cellular cell of CNNS with the current practical costs and the estimated heuristic costs. The GS is responsible for establishing new cells of the CNNS , and making up the connection patterns and bifurcation parameters of cells. In principle, the GS can be implemented by using some kinds of association memory or other ordinary approaches without any essential difficulties.

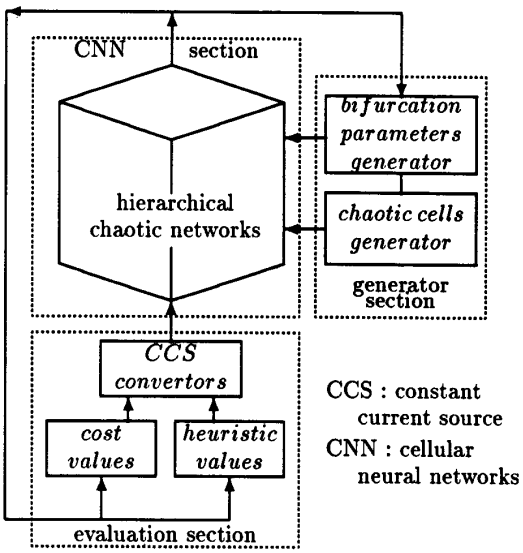
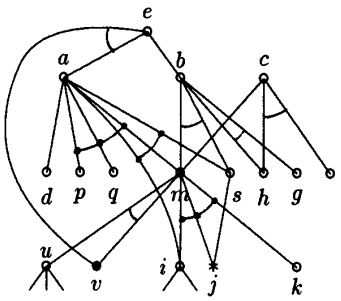


Figure 1: The architecture of chaotic network implementation for AI problem solving



node j is primitive unsolvable.
node v is primitive solvable.

Figure 2: The example of a partial implicit AND/OR graph resulted from some AI problem.

From below we are mainly concerned with the CNNS. Both the first layer and the second layer of the CNNS are the two-dimension array, each cell of which is a chaotic dynamic system. Using the Fig. 2 example of an implicit AND/OR graph derived from some AI problems the partial structures of the first and the second layer are illustrated in Fig. 3 and Fig. 4, respectively. A cell of the Chau's circuit in the first layer represents a discovered node or a discovered edge of an implicit AND/OR graph, as that cell $\mathbf{m_1}$ and cell $(\mathbf{m_1}, \mathbf{i_1})$ of Fig. 3 correspond to node m and edge (m, i) of Fig. 2, respectively. A cell of the second layer represents a discovered node or a hyper-edge of an implicit AND/OR graph, as that the cell $\mathbf{m_2}$ and cell $[(\mathbf{i_2}, \mathbf{j_2}, \mathbf{k_2}), \mathbf{m_2}]$ of Fig. 4 correspond to node m and hyper-edge $[(i, j, k), m]$ of Fig. 2, respectively. The linear resistor of every Chua's cell is taken as the bifurcation parameter under the control of relevant states of the adjacent cells, that are indicated over the linear resistor of Fig. 3, 4, 5. There is a branch of constant current source (CCS) in every cell, which conducts the necessary cost function values and some heuristic knowledge values via the cell into the networks. The CCS also performs as another bifurcation parameter. The rectifier unit I or unit II between cells in the first or the second layer performs the such rectifier-like function as makes the minimum current of all the existing inputs flow through it. The connection between the first and second layer is shown in Fig. 5, and there are intercouplings only between such cells as represent the same node of the implicit graph.

B. Bifurcation Mechanisms

In this paper the Chua's circuit cell is extended by attaching a DC current source I, as shown in Fig. 6, so that each Chua's cell becomes a three-dimensional autonomous affine dynamical system.

The Chua's cell is described by

$$\begin{aligned}
\frac{d\mathbf{x}}{dt} &= \begin{cases} \frac{G}{C_1}\mathbf{y} - \frac{G'_a}{C_1}\mathbf{x} + \mathbf{I} & \text{if } |\mathbf{x}| \le \mathbf{E}, \\ \frac{G}{C_1}\mathbf{y} - \frac{G'_b}{C_1}\mathbf{x} - \frac{\mathbf{I}'}{C_1} + \mathbf{I} & \text{if } |\mathbf{x}| > \mathbf{E}, \end{cases} \\
\frac{d\mathbf{y}}{dt} &= \frac{1}{C_2}\mathbf{Z} - \frac{G}{C_2}(\mathbf{y} - \mathbf{x}), \\
\frac{d\mathbf{Z}}{dt} &= -\frac{1}{L}\mathbf{y},
\end{aligned} \tag{4}$$

where $G'_a = G + G_a, G'_b = G + G_b, \mathbf{I}' = (G_b - G_a)\mathbf{E}$ when $\mathbf{x} < -\mathbf{E}$ (the D_{-1} region) and $\mathbf{I}' = (G_a - G_b)\mathbf{E}$ when $\mathbf{x} > \mathbf{E}$ (the D_{+1} region), $\mathbf{I}$ is the bias current from the DC current source.

The nonlinear resistor N_R has the following piecewise-linear driving-point (DP) characteristic:

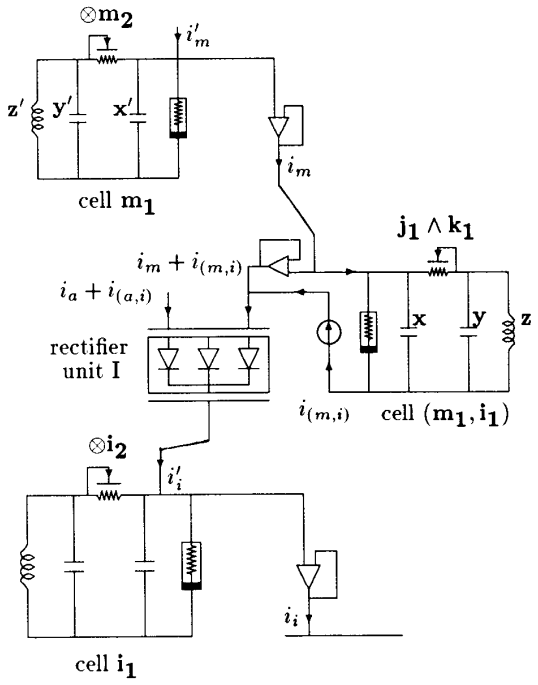


Figure 3: The structure of the first layer chaos array for AI solving.

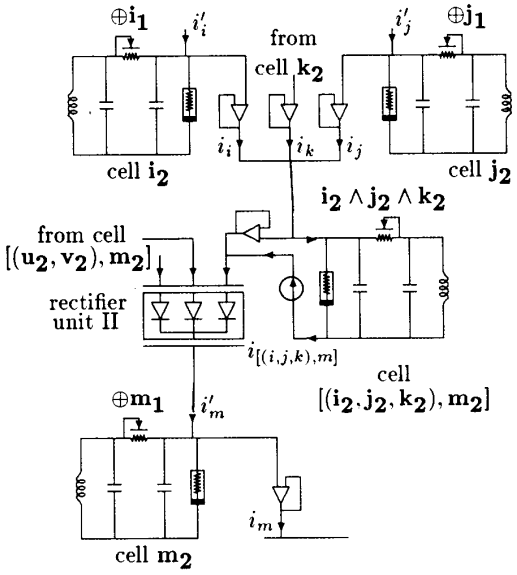


Figure 4: The structure of the second layer chaos array for AI solving.

$$\mathbf{I_R} = f(\mathbf{V_R}) = \begin{cases} G_b\mathbf{V_R} + (G_b - G_a)\mathbf{E} & \text{if } \mathbf{V_R} < -\mathbf{E}, \\ G_a\mathbf{V_R} & \text{if } -\mathbf{E} \le \mathbf{V_R} \le \mathbf{E}, \\ G_b\mathbf{V_R} + (G_a - G_b)\mathbf{E} & \text{if } \mathbf{V_R} > \mathbf{E}, \end{cases} \quad (5)$$

where $\mathbf{E} > 0, G_a < 0$, and $G_b < 0$.

At first, the following consequences are well known for the case of bias current $\mathbf{I} = 0$. When $G > |G_a|$ or $G < |G_b|$, the circuit has a unique equilibrium point P^0 at the origin, and when $G > |G_a|$ it also has two virtual equilibria P_v^- and P_v^+. Because $G_a < 0$, the equilibrium point P^0 is unstable and the vector field in the region D_0 pushes trajectories away from it until crossing the boundary and entering the outer region D_{+1} or D_{-1}. In the outer regions, although that the virtual eqilibria P_v^- and P_v^+ are stable or unstable depends on $G_b > 0$ or $G_b < 0$, trajectories are pushed by the dissipative or nondamping vector field and all converge toward the corresponding virtual eqilibria P_v^- and P_v^+ until it enters the D_0 degion. As result, a periodic steady-state trajectory called a limit-cycle attractor P^c is produced.

Except the above cases, the circuit has three equilibrium points at P^-, P^0, and P^+. Assume that the Jacobian matrix of the above state equations has eigenvalues: $\gamma_k, \sigma_k \pm j\omega_k, k = 0, 1$, and has the corresponding real eigrnvector $E^r(P)$ and the complex eigenplane $E^c(P)$, where P is the label of a equilibrium point. Taking R as the bifurcation parameter, the parameters of the circuit can be selected, so that when R is large the outer equilibrium points P^- and P^+ are stable (namely, $\gamma_1 < 0, \sigma_1 < 0, \omega_1 \neq 0$) and the inner equilibrium point P^0 is unstable (namely, $\gamma_0 > 0, \sigma_0 < 0, \omega_0 \neq 0$)[5]. Depending on the initial state of the circuit, the system remains at one outer equilibrium point or the other, say at point P^+. As the R is decreased, the σ_1 becomes positive, whereby the outer equilibrium points become unstable. Because the γ_1 remains negative, the trajectories in the D_1 region converge toward the complex eigenplane $E^c(P^+)$ and spiral away from the point P^+ along $E^c(P^+)$ until they enter the D_0 region. For $\gamma_0 > 0$ and $\sigma_0 < 0$, the equilibrium point at the origin in the D_0 region is characterized by that the trajectories follow a helix of exponentially decreasing radius whose axis lies in the direction of $E^r(P^0)$. Therefore, trajectories that enter D_0 from D_{+1} either cross over to D_{-1} or are turned back toward D_{+1}. The stable *period-k* limit cycle, $k = 1, 2, \cdots$, and furthermore the chaotic Spiral-Chua strange attractor will occur.

Next, we consider the case of the existence of a shunt DC current I. The equilibrium poins are defined by

$$
\begin{aligned}
0 &= \tfrac{G}{C_1}(\mathbf{y}-\mathbf{x}) - \tfrac{1}{C_1}\mathbf{I}_R + \mathbf{I} \\
0 &= \tfrac{1}{C_2}\mathbf{Z} - \tfrac{G}{C_2}(\mathbf{y}-\mathbf{x}) \\
0 &= -\tfrac{1}{L}\mathbf{y}.
\end{aligned} \tag{6}
$$

The equilibrium points can be determined graphically by intersecting the load line $\mathbf{I}_R = -G\mathbf{V}_R + C_1\mathbf{I}$ with the DP characteristic $\mathbf{I}_R = f(\mathbf{V}_R)$ of the nonlinear resistor N_R, as shown in Fig. 6(b).

Whether there is a shunt DC current or not, as long as the equilibrium points are located on the same piecewise-linear segment of the nonlinear resistor, the Jacobian matrices $\mathbf{J}_{\mathbf{F_a}}$ or $\mathbf{J}_{\mathbf{F_b}}$ in both cases of $\mathbf{I}=0$ and $\mathbf{I}\neq 0$ are entirely identical, i. e. ,

$$
\mathbf{J}_{\mathbf{F_a}} = \begin{bmatrix} 0 & -\frac{1}{L} & 0 \\ \frac{1}{C_2} & -\frac{G}{C_2} & \frac{G}{C_2} \\ 0 & \frac{G}{C_1} & -\frac{G'_a}{C_1} \end{bmatrix} \quad \text{if } |\mathbf{x}| \le \mathbf{E},
$$

$$
\mathbf{J}_{\mathbf{F_b}} = \begin{bmatrix} 0 & -\frac{1}{L} & 0 \\ \frac{1}{C_2} & -\frac{G}{C_2} & \frac{G}{C_2} \\ 0 & \frac{G}{C_1} & -\frac{G'_b}{C_1} \end{bmatrix} \quad \text{if } |\mathbf{x}| > \mathbf{E}. \tag{7}
$$

Correspondingly, the eigenvalues, the eigenvectors, the stabilities of the equilibrium points and the dynamical behaviours nearby the equilibrium points lying on the same piecewise-linear segment of nonlinear resistor are not influenced by the shunt DC bias current. The above discussion about the bifurcation parameter R in the case $\mathbf{I}=0$ are all suitable for the case $\mathbf{I}\neq 0$.

By moving load line, the practical equilibrium point for the case of $\mathbf{I}\neq 0$ could be changed into a virtual equilibrium point as long as the DC bias current is large enough, as in Fig. 6 the P^+ and P^0 changed into the virtual points P_I^+ and P_I^0, respectively. It implies that even under the same bifurcation parameter R, a stable equilibrium point in the case of $\mathbf{I}=0$ may disappear in the case of $\mathbf{I}\neq 0$. Because of virtual point P_I^+ the trajectory enters into the region D_0 from D_{+1}, then due to virtual point P_I^0 it returns to the region D_{+1} from D_0 again. Therefore, the corresponding multi-period limit cycle attractor or a strange attractor might happen. On the other hand, however, it is noticed that the moving load line can keep P_I^- being a practical stable equilibrium point as P^-.

For given parameters $G_a, G, \mathbf{E}$ and C_1, the critical value $\mathbf{I}^*$ of the bias CCS can be derived from $\mathbf{E}G_a = -G\mathbf{E} + C_1\mathbf{I}^*$, namely,

$$
\mathbf{I}^* = \begin{cases} (G_a+G)\mathbf{E}/C_1 & \text{for } P^+ \text{ and } P_I^+, \\ -(G_a+G)\mathbf{E}/C_1 & \text{for } P^- \text{ and } P_I^-. \end{cases} \tag{8}
$$

Therefore, if $|I| \le |I^*|$ and under the same parameter R, the dynamic behaviours nearby P_I^-, P_I^0 and P_I^+ are entirely similar to P^-, P^0 and P^+, respectively. If $|I| > |I^*|$, although the parameter R unchanged, a new kind of attractors different from the original attractor will arise. In this sence the bias DC current $\mathbf{I}$ is also referred to as a bifurcation parameter.

When $G \to \infty$, the limit-cycle attractor P^c, yielded by the unstable eqilibrium point P^0, the virtual eqilibria P_0^- and P_0^+, are not affected by the bias current $\mathbf{I}$.

C. Synchronization Mechanism

As for the synchronizations between the cell's states and the transmissions of the cost values and heuristic knowledge values, they are resulted from the following derivation. The state equations for the two cells in Fig. 3, that are intercoupled via a voltage buffer, are as follows:

$$
\begin{aligned}
\dot{\mathbf{x}}' &= \alpha'(\mathbf{y}' - \mathbf{x}' - f'(\mathbf{x}')) + \mathbf{i}'_a, \\
\dot{\mathbf{y}}' &= \mathbf{x}' - \mathbf{y}' - \mathbf{z}', \\
\dot{\mathbf{z}}' &= -\beta'\mathbf{y}', \\
\dot{\mathbf{x}} &= \alpha(\mathbf{y} - \mathbf{x} - f(\mathbf{x})) + \mathbf{i}_a, \\
\dot{\mathbf{y}} &= \mathbf{x} - \mathbf{y} - \mathbf{z}, \\
\dot{\mathbf{z}} &= -\beta\mathbf{y}.
\end{aligned} \tag{9}
$$

Assuming that all circuit components of the coupled cells are matched exactly and considering, owing to the voltage buffer between the cells, that $\mathbf{x}' = \mathbf{x}$, we obtain

$$
\begin{aligned}
\tfrac{d(\mathbf{y}'-\mathbf{y})}{dt} &= -(\mathbf{y}'-\mathbf{y}) - (\mathbf{z}'-\mathbf{z}), \\
\tfrac{d(\mathbf{z}'-\mathbf{z})}{dt} &= -\beta(\mathbf{y}'-\mathbf{y}).
\end{aligned} \tag{10}
$$

Therefore, we have $(\mathbf{y}'(\mathbf{t}) - \mathbf{y}(\mathbf{t})) \to 0$ for $t \to \infty$. This implies also

$$
\begin{aligned}
&\mathbf{i}_m(t) \to \mathbf{i}'_m(t), \\
&\mathbf{i}'_i = \mathbf{i}'_m + \mathbf{i}_{(m,i)},
\end{aligned} \tag{11}
$$

where $\mathbf{i}_{(m,i)}$ is the estimated cost value of the edge (m,i) of the implicit graph, including the heuristic knowledge value concerning the problem m; and $\mathbf{i}'_m + \mathbf{i}_{(m,i)}$ is assumed to be the minimum current among all the input currents of the rectifier unit I.

IV. Implementation Of Search Algorithms $\mathcal{P}^*_{AO}$

A. State-Encoding Attractors

For the hyper-distributed hyper-parallel algorithms, $\mathcal{P}^*, \mathcal{P}^*_h$ and $\mathcal{P}^*_{AO}$, of searching any implicit graph, [1] defines the following states of a node: ready state $\bigtriangledown$, activated state $\oplus$, cut state $\otimes$, and hidden state $\odot$ used for

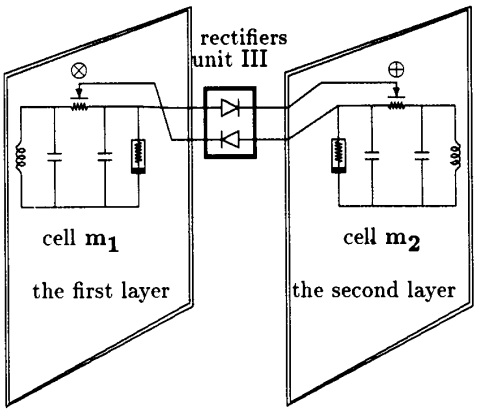


Figure 5: The synchronization and intercoupling between the layer I and the layer II of CNN

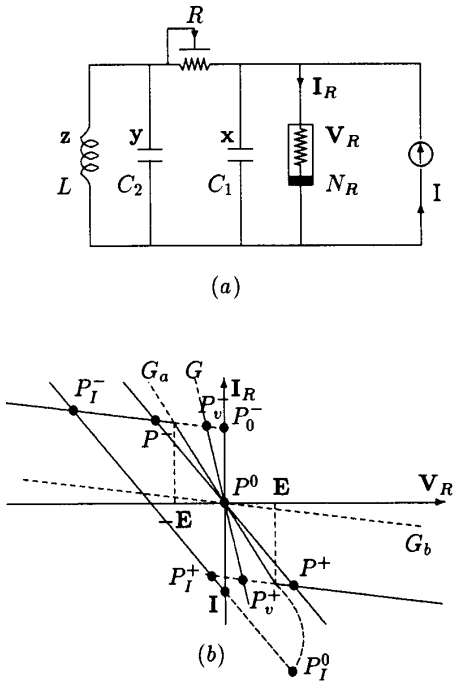


Figure 6: The Chua's circuit cell with a shunt DC current source I and the equilibrium points

top-down search; ready state $\triangle$, activated state $\uplus$, cut state $\otimes$, and hidden state $\oslash$ used for bottom-up search. Moreover, any edge has the following states: ready state $\sim$, flow state $\rightarrow$, blocked state $\bowtie$, and hidden state $\odot$ for top-down search; ready state $\approx$, flow state $\leftrightarrow$, blocked state $\bowtie$, and hidden state $\oslash$ for bottom-up search.

Except that cut state $\otimes$ for nodes and blocked state $\bowtie$ for edges are regarded as the common states for both top-down search and bottom-up search, the others of top-down and bottom-up search states are independent with each other[1].

The states of a node or an edge of an implicit AND/OR graph can be encoded in terms of different attractors existing for different bifurcation parameter values in the corresponding Chua's cell. Because the hidden states represent that a node or an edge has not been discovered so far, it is the existence of the corresponding Chua's cell that means the nonhidden state of the node or the edge, whereby encoding the hidden state by a specified attractor of a Chua's cell is not necessary. Furthermore, the first layer and the second layer of chaotic arrays are used for top-down and bottom-up search, respectively, and a node, an edge and a hyper-edge correspond to different Chua's cell, therefore the following four attractors of a Chua's cell are sufficient for encoding all the states of a node and a edge.

The attractor P^- or P_I^-, a stable equilibrium point, represents the ready state $\triangledown$ and $\triangle$ of a node, ready state $\sim$ and $\approx$ of an edge, and is taken as the initial state of any existing Chua's cell. The stable equilibrium point, P^+ or P_I^+, represents the activated state $\oplus$ and $\uplus$ of a node, the flow state $\rightarrow$ and $\leftrightarrow$ of an edge. The strange attractor P^s represents the cut state $\otimes$ of a node and the blocked state $\bowtie$ of an edge. When the control conditions of bifurcation parameter R of a cell are not met, then $R \rightarrow 0$, i. e. , $G \rightarrow \infty$ makes the cell turn into the limit-cycle attractor P^c around the unstable equilibrium point P^0.

B. Implementation of Concurrent Waves Propagations

The concurrent propagations of $\oplus$ or $\rightarrow$, and $\uplus$ or $\leftrightarrow$ states waves for algorithm $\mathcal{P}_{AO}^*$ are embodied in the concurrent propagations of attractors P_I^+ along the first and the second layer of the CNNS, respectively. The attractor P^+ or P_I^+ of a Chua's cell can be propagated to the adjacent cells in the same layer by means of synchronization mechanisms (11).

By means of the bifurcation effect of the bias DC current **I**, the attractor P^s of a Chua's cell in the second layer can be propagated to an adjacent cell of the second layer, as long as all the input currents of the rectifier II between the two cells are beyond the critical value I^*. Then through the changes of bifurcation parameters R of Chua's cells, the corresponding cells of the first layer are concurrently forced into attractor P^c. Consequently, the concurrent propagations of $\otimes$ and $\bowtie$ states waves are

realized in this way.

On the other hand, however, P^- is set as the cell's initial state, so the propagations of P^- between the cells are not taken into account, Because of different circuit parameters, there are no propagations between the cell with P^c and the cells with other attractors.

For the example of Fig. 2, when the cell $\mathbf{j_1}$ of the first layer of the CNNS becomes P^+ state and the j is found by the GS to be a primitive unsolvable problem node, thus cell $\mathbf{j_2}$, at first, changes its own state from P^c into P^- on account of the bifurcation condition $\oplus\mathbf{j_1}$ of cell $\mathbf{j_2}$ through the intercoupling between layers as shown in Fig. 4, 5, and next the DC current i'_j enough large beyond threshold $\mathbf{I^*}$ will be injected into the cell $\mathbf{j_2}$ via the bias DC shunt, whereby finally cell $\mathbf{j_2}$ is further transformed from P^- into P^s. Moreover, due to the P^s of $\mathbf{j_2}$, the $\mathbf{j_1}$ of the first layer is coerced into the attractor P^c from P^+, and through the cell $[(\mathbf{j_2}), \mathbf{s_2}]$ expressing hyper-edge (s, j) of implicit graph Fig. 2 the cell $\mathbf{s_2}$ and $\mathbf{s_1}$ are synchronized into P^s, P^c from P^-, P^+, respectively. The cells $(\mathbf{m_1}, \mathbf{i_1})$ and $(\mathbf{m_1}, \mathbf{k_1})$ also change into P^c on account of P^c state of $\mathbf{j_1}$ belonging to the same hyper-edge. While for the case of primitive solvable node v, cell $\mathbf{v_1}$ is activated to P^+ from P^- and $\mathbf{v_2}$ behaves from P^c to P^+ via P^- with the bias $\mathbf{I} = i'_v = 0$.

C. Implementation of Concurrent Waves Competitions

The rectifier units I and II of the CNNS implement the competitive activations required by the hyper-distributed hyper-parallel heuristic algorithms. For examples, the rectifier unit I between cells $\mathbf{i_1}, (\mathbf{m_1}, \mathbf{i_1})$ and $(\mathbf{a_1}, \mathbf{i_1})$ selects and makes the minimum current of $i_m + i_{(m,i)}$ and $i_a + i_{(a,i)}$ flow into $\mathbf{i_1}$, so as to synchronize $\mathbf{i_1}$ with P^+ and to transmit the accumulated cost values and the local heuristic knowledge values to $\mathbf{i_1}$.

D. Global Performance

When a cell, e. g. , $\mathbf{m_1}$, of the first layer of the CNNS becomes P^+, the GS retrieves the information about whether the corresponding node $\mathbf{m}$ of the implicit graph is a primitive unsolvable or a primitive solvable, If the case is neither primitive unsolvable nor primitive solvable, then the GS finds out all the son nodes of $\mathbf{m}$ of the implicit graph, and, for the first retrieved son node, makes the CNNS yield an corresponding new Chua's cell in both the first and the second layer array. For the son node $\mathbf{i}$ of $\mathbf{m}$, the intercoupling patterns between the cells, $\mathbf{m_1}, \mathbf{i_1}$ and $(\mathbf{m_1}, \mathbf{i_1})$, of the CNNS's first layer will be made up, as shown in Fig. 3. For the AND son nodes $\mathbf{i}, \mathbf{j}, \mathbf{k}$, of $\mathbf{m}$, the connection patterns between cells, $\mathbf{m_2}, \mathbf{i_2}, \mathbf{j_2}$ and $\mathbf{k_2}$, via the intermediate cell $[(\mathbf{i_2}, \mathbf{j_2}, \mathbf{k_2}), \mathbf{m_2}]$, of the CNNS's second layer, are shown in Fig. 4.

At first, the first layer's cell ζ_1 corresponding to the starting node ζ of an implicit graph is compelled into the attractor P^+ from P^- with the injected bias current $i_\zeta = 0$. Then, as mentioned above, the attractor P^+ will be concurrently propagated to the adjacent Chua's cells of the first layer with the bias currents expressing the necessary cost values and heuristic knowledge values. Once such a cell of the first layer that expresses a primitive solvable or a primitive unsolvable problem node becomes P^+, the concurrent propagations of the P^+ and P^s will take place in the second layer of the CNN, as stated above. Once the cell ζ_2 of the second layer is set to the attractor P^+ from P^-, the search process finishes with success. Consequently, the optimal solutions of the given implicit AND/OR graph are indicated by the cascades of the second layer's cells with attractor P^+ and the intermediate cells whose bias DC currents in the shunt are allowed to flow through the next rectifier unit II and to transmit to the successive cells of the second layer of the CNNS.

REFERENCES

[1] Dianxun Shuai and Yoichiro Watanabe, "New heuristic distributed parallel searching algorithms and neural network implementations," in *Proc. IEEE Int'l Symp. Nonlinear Theory and Its Appli.* (Hawaii, U. S. A.), pp. 759-762, Dec. 1993

[2] Dianxun Shuai and Yoichiro Watanabe, " A hierarchy neural network approach to symbolic logic algorithms of problem solving," in *Proc. IEEE Int'l Conf. Neural Networks* (Nagoya,Japan), pp. 1606-1610, Sep. 1993

[3] Dianxun Shuai, " An approach to solving the longest common subsequences, based on string-coding functional and neural network," in *Proc. IEEE Int'l Conf. System Engineering* (Kobe, Japan), pp. 564-567, Sep. 1992

[4] Dianxun Shuai, " A homomorphic harmony network approach to solving the unification problem," in *Proc. IEEE Int'l Conf. Neural Networks* (Beijing, China), pp. I-564-571, Oct. 1992

[5] Eugene Charniak, Drew McDermott, *Introduction to Artificial Intelligence*, Addison_Wesley publishing Company, California, 1986.

[6] Montanari, U. , " Heuristically guided search and chromosome matching, " *Artif. Intell.* , vol. 1, no. 4, pp. 227-245, 1974

[7] Gasching, J. , " Performance measurement and analysis of certain search algorithms, " *Report CMU-CS-79-124*, Carnegie-mellon University, may 1979

[8] Goldberg, D. E. , *Genetic algorithms in search, optimization, and machine learning*, Addison-Wesley Publishing Company, 1989

[9] Pearl, J. , " Some recent results in heuristic search theory, " *IEEE Trans. PAMI-6*, no. 1, pp. 1-12, 1984

[10] Chua L. O. and Yang L. , " Cellular neural networks theory, " *IEEE Trans. Circuits Syst.* , vol. 35, pp. 1257-1272, 1988

[11] Whitely, D. and Hanson, T. , " Optimizing neural networks using faster, more accurate genetic search, " in *Proc. of ICGA-89*,1989

[12] Ogorzalek M. J. , " Taming Chaos - Part I: Synchronization, " "Part II: Control, " *IEEE Tran. circuits and systems*, Vol. 40, no. 10, pp. 693-706, 1993

[13] Krinsky, V. I. , Biktashev, V. N. and Efimof, V. I. , "Autowaves principles for parallel image processing, " *Physica* 49D, pp. 247-253, 1991

[14] Chua, L. O. , Yang, L. and Kneg, K. R. , " Singnal processing using cellular neural networks," *J. VLSI Singnal Processing*, vol. 3, pp. 25-51, 1991

[15] Peter Kennedy, " Three Steps to Chaos : Part I: Evolution," "Part II: A Chua's Circuit Primer," *IEEE Trans. on circuits and systems*, Vol. 40, no. 10, pp. 640-674, 1993