Information Sciences 181 (2011) 92–105

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Mutation Hopfield neural network and its applications

Laihong Hu [a,b,*], Fuchun Sun [a], Hualong Xu [b], Huaping Liu [a], Xuejun Zhang [c]

[a] *Department of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China*
[b] *Xi'an Research Institute of High-tech, Xi'an, Shannxi 710025, PR China*
[c] *School of Instrumentation Science and Optoelectronics Engineering, Beihang University, Beijing 100191, PR China*

ARTICLE INFO

Article history:
Received 11 July 2008
Received in revised form 15 July 2010
Accepted 10 August 2010

Keywords:
Mutation
Hopfield neural network
Max-cut problem
Aircraft landing scheduling

ABSTRACT

In this paper, a new operator is proposed to optimize the traditional Hopfield neural network (HNN). The key idea is to incorporate the global search capability of the Estimation of Distribution Algorithms (EDAs) into the HNN, which typically has a powerful local search capability and fast operation. On account of this property of the EDA, our proposed algorithm also exhibits a powerful global search capability. In addition, the possible infeasible solutions generated during the re-sampling period of the EDA are eliminated by the HNN. Therefore, the merits of both these methods are combined in a unified framework. The proposed model is tested on a numerical example, the max-cut problem. The new and optimized model yielded a better performance than certain traditional intelligent optimization methods, such as HNN, genetic algorithm (GA). The proposed mutation Hopfield neural network (MHNN) is also used to solve a practical problem, aircraft landing scheduling (ALS). Compared with first-come-first-served sequence, MHNN sequence reduces both total landing time and total delay.

1. Introduction

Although many approaches (such as genetic algorithms) have been proposed for global optimization, serious problems associated with real-time operation persist. On the other hand, the neural network approach [38] (especially that for hardware implementations, such as the field-programmable gate array (FPGA) chip) provides feasible solutions to complex optimization problems within a very short time (i.e., in real-time) [5]. In the past two decades, neural networks have been widely used to solve different problems such as mathematical programming, pattern recognition, group classification, series prediction, data mining. The Hopfield neural network (HNN), which was originally proposed in [21], exhibits the best performance in solving optimization problems out of all existing neural networks, and has therefore attracted considerable attention [36]. Technically, the HNN approach to optimization involves handling a dynamic system in which the behavior of the network and the problems to be solved can be characterized by the energy function, or the Lyapunov function. This architecture can be realized by using an electronic circuit, and used as an online solution with a parallel-distributed process, making it particularly suitable for real-time optimization [36]. However, the energy function of the HNN decreases rapidly only during the first few iterations. Thereafter, the network oscillates between neuron states with the same energy, and ultimately gets trapped in local minima [2]. The local minima problem is caused by the gradient descent dynamics of the binary HNN. Recently, Wang [35] proposed an HNN combined with an Estimation of Distribution Algorithm (EDA), a population-based evolutionary algorithm. In the proposed algorithm, once the network is trapped in local minima, the perturbation based on EDA can generate a new starting point for the HNN. Generating a new starting point for the HNN leads to further research,

* Corresponding author at: Department of Computer Science and Technology, Tsinghua University, Beijing 100084, PR China.
E-mail address: hulh07@gmail.com (L. Hu).

0020-0255/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.ins.2010.08.007

thereby allowing the network to escape local minima. However, the proposed approach has been used only in solving the two-page crossing number problem; its extension to other applications, therefore, is imperative.

In this paper, we propose a new mutation Hopfield neural network (MHNN) to solve practical optimization problems. The key idea is to incorporate the global search ability of EDA into the HNN, which typically has a powerful local search capability. The proposed new optimization model is tested on a numerical example, the max-cut problem, and exhibits better performance. Finally, we use the proposed MHNN to solve a practical problem in the form of aircraft landing scheduling (ALS), in which better results were yielded than that with first-come-first-served (FCFS) sequence.

The rest of the paper is organized as follows. Section 2 provides a brief overview of HNNs. In Section 3, the details of the proposed model and an algorithm for general cases are given. In Section 4 (for the max-cut problem) and Section 5 (for the ALS problem), the application-specific implementation and performance evaluation of the proposed algorithm are presented. In Section 6, conclusions and suggestions for future work are put forward.

2. Brief overview of Hopfield neural networks

The use of neural networks for solving optimization problems was initiated by Hopfield and Tank [21]. They demonstrated the computational power of the neural network by applying their model to the traveling salesman problem. Since then, several investigators have adopted the Hopfield model to solve various optimization problems (e.g., [2,5,36]). In the conventional HNN, each neuron is modeled as a nonlinear device (i.e., operational amplifier) with a sigmoid, monotonically increasing function defined by the logistic function

$$V_i = f_i(U_i) = \frac{1}{1+e^{-\alpha_i U_i}}, \tag{1}$$

where U_i is the input of the ith neuron; V_i, with a value between 0 and 1, is the output of the ith neuron; and α_i denotes the gain of the sigmoid function.

Each neuron receives resistive connections (modeling the biological synaptic connection) from other neurons, and these connections can be fully described by the interconnection matrix $\mathbf{T} = [T_{ij}]$. Here, T_{ij} is the interconnection weight from the jth neuron to the ith neuron. Each neuron also receives an input bias current of I_i, the only user-adjustable parameter. In Fig. 1, the conventional structure of a discrete HNN [21] is illustrated.

A discrete HNN with n neurons can be represented by two $n \times n$ real matrices $W^0 = (w^0_{ij})_{n\times n}$, $W^1 = (w^1_{ij})_{n\times n}$, and an n-dimensional column vector $b = (b_1, \ldots, b_n)^T$, where w_{ij} denotes the connection strength between neuron i and neuron j, b_i represents the threshold of neuron i [7]. Denoting the input of neuron i at time k as $u_i(k)$, and the output at time k is denoted as $v_i(k)$, we get

$$u_i(k) = \sum_{j=1, j\neq i}^{n} w_{ij} v_j(k) + b_i, \tag{2}$$

$$v_i(k+1) = f(u_i(k)), \tag{3}$$

where the excitation function $f(\cdot)$ is usually selected as the symbolic function $sgn(\cdot)$. In this case, two values for the output of each neuron are possible: 1 or -1. Therefore, we have:

$$v_i(k+1) = \begin{cases} 1, & \sum\limits_{j=1,j\neq i}^{n} w_{ij} v_j(k) + b_i \geqslant 0, \\ -1, & \sum\limits_{j=1,j\neq i}^{n} w_{ij} v_j(k) + b_i < 0, \end{cases} \tag{4}$$

The energy function of the discrete HNN is defined as:

$$E = -\frac{1}{2} \sum_{i=1, i\neq j}^{n} \sum_{j=1, j\neq i}^{n} w_{ij} v_i v_j + \sum_{i=1}^{n} b_i v_i. \tag{5}$$

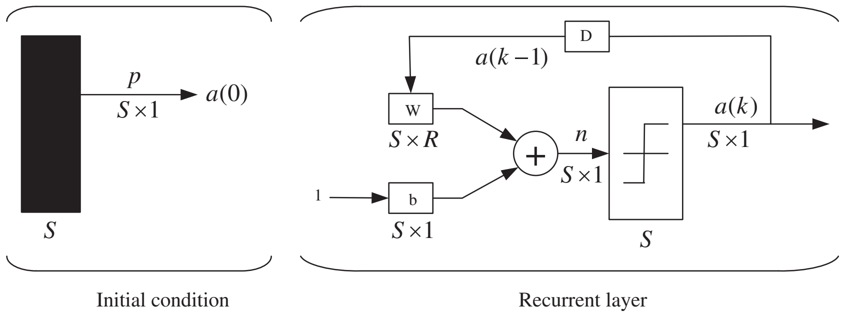

Fig. 1. Illustrative diagram of a discrete HNN.