Neurocomputing 130 (2014) 28–35

Contents lists available at ScienceDirect

Neurocomputing

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

An improved competitive Hopfield network with inhibitive competitive activation mechanism for maximum clique problem

Gang Yang [a], Nan Yang [a,*], Junyan Yi [b,c], Zheng Tang [d]

[a] *Information School, Renmin University of China, Beijing, China*
[b] *Zhejiang University of Technology, Hangzhou, China*
[c] *Beijing University of Civil Engineering and Architecture, Beijing, China*
[d] *Faculty of Engineering, University of Toyama, Toyama, Japan*

ARTICLE INFO

Article history:
Received 19 June 2012
Received in revised form
3 October 2012
Accepted 9 November 2012
Available online 20 August 2013

Keywords:
Competitive Hopfield network
Maximum clique problem
Inhibitive competitive mechanism
Community detection

ABSTRACT

In this paper, we analyze the formula of weights definition in the discrete competitive Hopfield network (DCHOM) and point out its flaw when using it to solve some special instances of maximum clique problem (MCP). Based on the analysis, we propose an improved competitive Hopfield network algorithm (ICHN). In ICHN, we introduce a flexible weight definition method which excites the competitive dynamics, and we also present an initial values setting strategy which efficiently increases the probability of finding optimal solutions. Furthermore, an inhibitive competitive activation mechanism is introduced to form a new input updating rule which reduces significantly the number of neurons with an intermediate level of activations. Our algorithm effectively overcomes the flaw of the DCHOM, and exhibits powerful solving ability for the MCP. Experiments on the benchmark problems and practical applications verify the validity of our algorithm.

1. Introduction

Maximum clique problem (MCP) is a representative problem of combinatorial optimal problem and it is also a well-studied NP-Hard problem [13,4].This problem is computationally intractable even when it is used to approximate with certain absolute performance bounds. MCP has many practical applications in diverse fields such as computer vision, information retrieval, cluster analysis, fault tolerance. Moreover many important problems such as maximum independent set, minimum vertex covering, quadratic zero-one problem can be easily reduced to MCP [20]. Hence, it is becoming more and more important to find the optimal and near-optimal solutions to MCP in practical applications and theory researches [19]. In 1992, a maximum Hopfield network was successfully proposed by Takefuji et al. [20] and Lee et al. [14] to handle a class of NP-complete optimization problems, including MCP, which was often hard to solve by neural networks. Then researchers Funabiki et al. summarized several basic algorithms that can solve the MCP effectively [17]. In 2003, Galán et al. modeled competitive Hopfield-type neural networks such as maximum neural network to solve MCP, and illustrated maximum neural network using a completely synchronous model that cannot guarantee energy decrease and sometimes generates inaccurate results and oscillatory behaviors in the convergence process [9]. In 2005, Wang et al. proposed a discrete competitive Hopfield neural network which contains stochastic dynamics to escape from local minima [22]. They applied their improved competitive Hopfield neural network in some other practical problems soon after, and proposed further improvements [21]. These newly proposed methods validate the superior performance of competitive mechanism in neural networks substantially through solving some well-known benchmark problems. Sequently, Yi et al. proposed a shrinking chaotic maximum neural network, which also applied the flexibility of competitive property of maximum neural network and chaotic dynamics sufficiently, to solve MCP [24]. Most of the conventional neural networks fundamentally utilize gradient descent dynamics, while Chen and Aihara proposed a transient chaotic neural network (TCNN) [7,8] based on the continuous Hopfield neural network (CHNN) for combinational optimization problems [6]. TCNN is a better algorithm than the other algorithms only with gradient descent dynamics, which was analyzed and improved to solve MCP in 2009 [23]. But it is difficult to balance the chaotic dynamics and gradient descent dynamics to converge to a stable equilibrium point corresponding with an acceptably near-optimal solution [11]. Compared with chaotic dynamics, competitive mechanism displays more effective intrinsic characteristics to help neural network escape from local minima. In 2008, a neuro-GA approach using a maximum neural network, along with the chaotic mutation capability of genetic algorithms, was proposed to solve the reduced maximum clique

* Corresponding author. Tel.: +86 82500902.
E-mail address: Yangnan@ruc.edu.cn (N. Yang).

0925-2312/$ - see front matter © 2013 Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.neucom.2012.11.055

problem [1]. Sequently, the neuro-GA algorithm was used to solve maximum fuzzy clique problem [3]. Moreover, evolutional algorithm is an important kind of powerful method to solve combinatorial optimization problems. Recently, a reactive evolutionary algorithm with guided mutation (EA/G), named R-EVO, is proposed as a typical evolutional method to solve MCP [5].

In this paper, based on the analysis of the weights definition in discrete competitive Hopfield network (DCHOM), an intrinsic flaw is pointed out when we are using it to solve some special instances of the MCP. Furthermore, we propose an improved competitive Hopfield network algorithm (ICHN), and introduce a flexible weight definition method to excite the competitive dynamics of the ICHN. For increasing the successful probability of finding optimal solutions, an initial values setting strategy is introduced into the ICHN, and an inhibitive competitive activation mechanism is embedded into neuron updating functions to reduce significantly the number of neurons with intermediate level of activations. The ICHN effectively overcomes the flaw of original maximum neural network, and exhibits powerful solving ability for MCP. Simulation results show that the ICHN has superior ability for the MCP through solving well-known benchmark problems and practical problems (MCP in community detection) within a reasonable number of iterations.

2. Maximum neuron model for MCP

Let $G=(V,E)$ be an arbitrary undirected graph, where $V=\{1,\dots,n\}$ is the vertex set of G and $E\subseteq V\times V$ is the edge set of G. $A=(a_{ij})_{n\times n}$ is the adjacency matrix of G, where $a_{ij}=1$ if $(i,j)\in E$ and $a_{ij}=0$ if $(i,j)\notin E$. The complement graph of $G=(V,E)$ is the graph $\overline{G}=(V,\overline{E})$, where $(\overline{E})=\{(i,j)/i,j\in V,\ i\neq j \text{ and } (i,j)\notin E\}$. Given a subset $S\subseteq V$, we call $G(S)=(S,E\cap S\times S)$ the subgraph induced by S. A graph $G=(V,E)$ is complete if all its vertices are pairwise adjacent, that is, $\forall i,j\in V$, $(i,j)\in E$. A clique C is a subset of V such that $G(C)$ is complete. The MCP requires a clique that has the maximum cardinality.

Based on the previous researchers' works, [20,14] formulate the MCP problem as a global minimization of the function

$$E=\sum_{x=0}^{n}\sum_{y=0}^{n}\sum_{i=1}^{2} t_{xy}v_{xi}v_{yi} \tag{1}$$

subject to the constraints $\sum_{i=1}^{2}v_{xi}=1$ for $x=0,1,\dots,n$, where $v_{xi}\in\{0,1\}$, and $v_{01}=1$ determinately. The weight matrix is defined by

$$t_{0i}=t_{i0}=\frac{1}{4}\left(\sum_{j=1}^{n}\overline{a_{ij}}-1\right)$$
$$t_{ii}=0;\ t_{ij}=t_{ji}=\tfrac{1}{4}\overline{a_{ij}}\quad \forall i\neq j,\quad i,j=1,\dots,n. \tag{2}$$

In this formulation, the first step in solving the MCP is to construct a graph G_M by adding a vertex 0 to $\overline{G}$ (the complement graph of G). And t_{ij} represents the weight of an edge between vertices i,j. Fig. 1 shows a simple graph G with 6 vertices and 10 edges and the graph G_M. The maximum clique of G contains vertices #2, #3, #5 and #6. The second step is to minimize the summation of the weights whose vertices belong to the same partition set, where V_M is partitioned into $V^+=\{x/v_{x1}=1, v_{x2}=0\}$ corresponding to the vertices in the clique and $V^-=\{x/v_{x1}=0, v_{x2}=1\}$ corresponding to the vertices not included in the clique.

Galan et al. presented that maximum neural network using completely synchronous model cannot guarantee energy decrease and sometimes generates inaccurate results and oscillatory behaviors in the convergence process. However, they just illustrated the oscillatory behavior of parallel maximum neural network for MCP, the critical factor of weight setting was not revealed in detail. Moreover, due to lacking the influence of neurons' delicate adjustment in their discrete model, the initial values become a very important factor to affect the solution quality and convergence speed. Thus in our algorithm we consider these inferior situations generally and propose effective methods to overcome these disadvantages.

3. The proposed algorithm for MCP

In the competitive network model for the MCP, as the added vertex #0 connects with all the other vertices, it directly influences the whole network competitive process. In the convergence process, v_{01} is always set as 1, which drives those vertices connected with vertex #0 to get saturation to 1 tentatively and gradually. And v_{02} drives other vertices to saturate to 0 gradually. Similarly, the competition between v_{x1}, v_{x2}, …, and $v_{xm}(x=1,\dots,n)$ produces powerful searching ability for the neural network to solve some special combinatorial optimization problems such as N-Queen problem and channel assignment problem, especially when the competitive group $v_{xi}(i=1,\dots,m)$ has adaptive numbers of vertices.

The weight definition of vertex #0 is a crucial factor determining the competitive intensity of neural networks. However, for a large number of MCP instances, the original weight definition function in Eq. (2) cannot guarantee that neural networks obtain powerful competitive activation to search optimal solutions regrettably. Hence either in completely synchronous model or asynchronous model, the competitive Hopfield networks using Eq. (2) are easy to get convergence to local minima. Here we give a simple example to show the flaw of weight definition Eq. (2) specifically in competitive neural networks for the MCP. Let us consider two simple graphs in Fig. 2. The weight matrices gotten

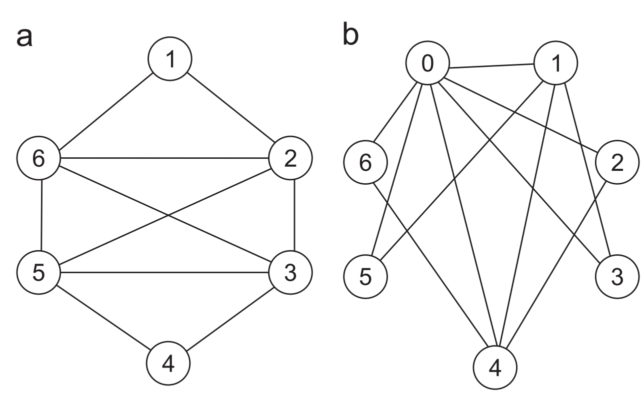

Fig. 1. (a) A graph G and (b) the constructed graph G_M.

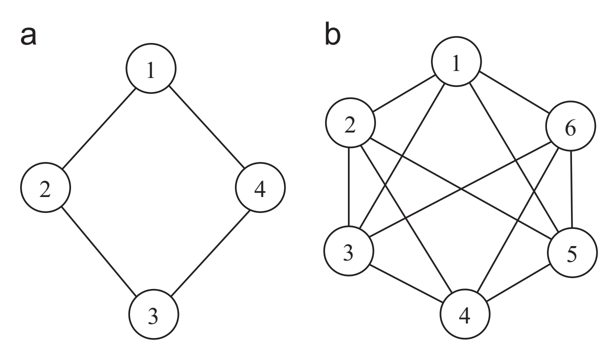

Fig. 2. (a) A graph with four vertices and (b) a graph with six vertices.