

基于局部进化的 Hopfield 神经网络的 优化计算方法

黎明 杨小芹 周琳霞

(南昌航空工业学院测试与控制工程系, 南昌 330034)

摘要 提出一种基于局部进化的 Hopfield 神经网络优化计算方法, 该方法将遗传算法和 Hopfield 神经网络结合在一起, 克服了 Hopfield 神经网络易收敛到局部最优值的缺点, 以及遗传算法收敛速度慢的缺点。该方法首先由 Hopfield 神经网络进行状态方程的迭代计算降低网络能量, 收敛后的 Hopfield 神经网络在局部范围内进行遗传算法寻优, 以跳出可能的局部最优值陷阱, 再由 Hopfield 神经网络进一步迭代优化。这种局部进化的 Hopfield 神经网络优化计算方法尤其适合于大规模的优化问题, 对图像分割问题和规模较大的 200 城市旅行商问题的优化计算结果表明, 其全局收敛率和收敛速度明显提高。

关键词 遗传算法 Hopfield 网络 优化计算 旅行商问题

中图分类号: TP301.6 TP391.41 文献标识码: A 文章编号: 1006-8961(2004)02-0207-07

Optimization with Partially Evolved Hopfield Neural Networks

LI Ming, YANG Xiao-qin, ZHOU Lin-xia

(Dept. of Test & Control Engineering, Nanchang Institute of Aeronautical Technology, Nanchang 330034)

Abstract A novel optimization method using partially evolved Hopfield neural network is proposed in this paper. The method uses Hopfield neural networks and a genetic algorithm on a local area of Hopfield neural networks to compensate each other for defects. The defect of the Hopfield neural network is captured by locally optimal solutions. The defect of genetic algorithms is the lower convergence speed when it optimizes large scale problems. In the proposed method, the Hopfield neural network and a genetic algorithm are used alternately. Solutions obtained with the converged Hopfield neural network are applied to the genetic algorithm to escape from locally optimal solutions. The genetic algorithm is only carried out on some local areas of Hopfield neural network so as to effectively save the computational consumption. The method is evaluated by investigating two large scale optimization problems; image segmentation and 200 cities TSP problem. Experiments show that the local minima of large scale networks can be greatly improved by the partially evolved Hopfield network and the convergence speed is obviously enhanced.

Keywords genetic algorithms, hopfield neural networks, optimization computation, TSP problem

1 引言

自 Hopfield 神经网络被成功应用于属于 NP 完全问题的旅行商(TSP)问题^[1,2], Hopfield 神经网络就得到了越来越广泛的应用, 例如 Hopfield 网络用

于图像边界检测^[3]、图像分割^[4,5]等。但是 Hopfield 网络存在不稳健性^[6], 这主要是由于计算能量函数在能量下降时可能陷入局部最小值, 当 Hopfield 网络规模变大、能量函数变复杂时, 这种问题变得更为严峻。影响能量函数陷入局部最小值的因素主要有: 神经元初值的选择、能量函数的控制参数选择、神经元的输入输出激活函数选择。提高 Hopfield 神经网络

络在优化计算时的稳健性的方法,目前主要有4类:

(1) 选择合适的能量函数简化优化计算的复杂性^[7~9]。但这种方法一般只能对特定的优化计算问题有效,仅适用于小规模 Hopfield 网络优化计算问题。

(2) 为 Hopfield 网络加入陷阱逃脱算法,典型的方法有为收敛的 Hopfield 网络加入扰动的 LEM 方法^[10],如果扰动后的网络能量得到降低,就可逃出目前的局部最小陷阱。文献^[10]还提出了局部扰动算法用于解决大规模 Hopfield 网络局部收敛问题。但是,如果在 Hopfield 网络状态空间中,全局能量最小点远离局部最小陷阱,则扰动算法逃离局部最小陷阱的能力非常有限。

(3) 在 Hopfield 网络能量下降过程中加入热力学系统^[11],例如模拟退火策略通过使得 Hopfield 网络有时能接受能量上升,从而有可能跳出局部最小陷阱。但模拟退火策略也有可能使得 Hopfield 网络跳出一个局部最小陷阱后再陷入另外的局部最小陷阱,因此当 Hopfield 网络的规模较大时,对局部最小陷阱的多次逃逸使得收敛到全局最小的速度大大降低。

(4) 采用遗传算法进化 Hopfield 网络^[12~14]。在这类方法中 Hopfield 网络的神经元输入输出状态方程的迭代过程与遗传算法对神经元状态的进化过程交替进行,利用遗传算法全局寻优能力强的优点,提高 Hopfield 网络收敛到全局最优状态的能力。目前研究的遗传算法和 Hopfield 网络的方法都是遗传进化整个 Hopfield 网络,因此当 Hopfield 网络规模很大时,进化计算量非常大,造成收敛速度较慢。

本文提出了一种局部进化的 Hopfield 网络,首先进行 Hopfield 网络神经元输入输出状态方程的迭代过程,使得 Hopfield 网络的能量函数值逐渐降低,直到收敛;然后在 Hopfield 网络中随机选择一些局部区域,构成一些新的小规模 Hopfield 网络,再采用遗传算法优化这些小规模 Hopfield 网络,如果其能量函数值在进化过程中得到降低,则将这些小规模 Hopfield 网络的输入和输出状态代回到整个 Hopfield 网络,再次进行神经网络的状态方程迭代,使整个神经网络的能量函数值降低直到收敛, Hopfield 网络的状态迭代方程与遗传算法如此交替运行,以提高 Hopfield 网络收敛到全局最优的能力。这种局部进化的 Hopfield 网络由于只对小规模 Hopfield 网络的能量函数进行优化计算,因此运算

量大大降低,使得整个算法收敛速度较快。

2 Hopfield 神经网络及其应用

Hopfield 网络是一种反馈型的神经网络^[6],神经元之间具有对称的连接,神经元的输入和输出函数一般采用 sigmoidal 函数。设 Hopfield 网络由二维神经元阵列组成 $\{(i, j); i \in [1, N]; j \in [1, M]\}$, 其中 N 和 M 分别表示横坐标和纵坐标方向的神经元个数,设 V_{ij} 和 U_{ij} 分别表示神经元 (i, j) 的输出和输入状态, $W_{ij,kl}$ 表示从神经元 (i, j) 到神经元 (k, l) 的连接权重系数。神经元 (i, j) 的输出由 Hopfield 网络中其他神经元的输出和一个偏移量 B_{ij} 所决定,神经元 (i, j) 输入、输出的状态迭代方程为

$$C_{ij} \frac{dU_{ij}}{dt} = -\frac{U_{ij}}{R_{ij}} + \sum_{k=1}^N \sum_{l=1}^M W_{ij,kl} V_{kl} + B_{ij} \quad (1)$$

其中, C_{ij} 和 R_{ij} 分别是预定义的参数,并且

$$V_{ij} = g\left(\frac{U_{ij}}{U_0}\right) = \frac{1}{2} \left(1 + \tanh \frac{U_{ij}}{U_0}\right) \quad (2)$$

其中, U_0 是一个常数。Hopfield 网络的能量函数可表示为

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^M \sum_{k=1}^N \sum_{l=1}^M W_{ij,kl} V_{ij} V_{kl} - \sum_{i=1}^N \sum_{j=1}^M V_{ij} B_{ij} + \sum_{i=1}^N \sum_{j=1}^M \int_0^{V_{ij}} g^{-1}(V) dV \quad (3)$$

当 $C_{ij} > 0$ 和 $W_{ij,kl} = W_{kl,ij} \forall i, k \in [1, N] \& j, l \in [1, M]$ 时, $\frac{dE}{dt} \leq 0$, Hopfield 网络是一种稳定的系统,各神经元输入和输出将随状态方程(1)和(2)的迭代收敛在一个稳定的状态。

图像分割可以转化为图像分割阈值平面的优化问题^[4,5],并由 Hopfield 网络求解。图像的两维像素阵列中像素一一对应于 Hopfield 网络的两维神经元阵列中神经元,图像边界像素点对应的神经元的初始输出值等于该像素的归一化灰度值,并始终保持不变。Rout 定义的 Hopfield 网络能量函数通过对图像边界点之间的像素点灰度值进行插值,对图像分割阈值平面进行优化计算^[5]。当 Hopfield 网络收敛时,其神经元输出值就是对应像素点的图像分割阈值。由 Rout 的能量函数可得 Hopfield 网络神经元之间的权重系数为

$$W_{ij,(i+n)(j+m)} = \begin{cases} -Mask(n+2, m+2) & -2 \leq n, m \leq 2 \\ 0 & \text{其他} \end{cases} \quad (4)$$

$$Mask = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 2-2\alpha & -8-2\alpha & 2-2\alpha & 0 \\ 1 & -8-2\alpha & 20+16\alpha & -8-2\alpha & 1 \\ 0 & 2-2\alpha & -8-2\alpha & 2-2\alpha & 0 \\ 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad (5)$$

其中, α 是一常数, $Mask$ 为掩模算子。由于图像的像素点很多, 因此对应的 Hopfield 网络规模非常大。

由 Hopfield 网络求解 TSP 问题时采用换位矩阵表征一条访问路径, 对 N 个城市的 TSP 问题, 设城市 x 和城市 y 之间的距离为 d_{xy} , 则神经元 (x, i) 的输入输出状态方程^[6]为

$$\frac{dU_{xi}}{dt} = -\frac{U_{xi}}{\tau} - A \sum_j V_{xj} - B \sum_y V_{yi} - C \left(\sum_x \sum_j V_{xj} - N \right) - D \sum_y d_{xy} (V_{y,i+1} + V_{y,i-1}) \quad (6)$$

其中, τ 、 A 、 B 、 C 和 D 是预定义常数。此时神经元 (x, i) 和 (y, j) 之间的连接权重系数为

$$W_{xi,yj} = -A\delta_{xy}(1-\delta_{ij}) - B\delta_{ji}(1-\delta_{xy}) - C - Dd_{xy}(\delta_{j,i+1} + \delta_{j,i-1}) \quad (7)$$

$$\delta_{ij} = \begin{cases} 1 & \text{如果 } i = j \\ 0 & \text{其他} \end{cases} \quad (8)$$

而神经元 (x, i) 的偏量为

$$B_{xi} = CN \quad (9)$$

Hopfield 网络的能量函数为

$$E = \frac{A}{2} \sum_x \sum_i \sum_{j \neq i} V_{xi} V_{xj} + \frac{B}{2} \sum_i \sum_x \sum_{y \neq x} V_{xi} V_{yi} + \frac{C}{2} \left(\sum_x \sum_i V_{xi} - N \right)^2 + \frac{D}{2} \sum_x \sum_{y \neq x} \sum_i d_{xy} V_{xi} (V_{y,i+1} + V_{y,i-1}) \quad (10)$$

3 局部进化的 Hopfield 神经网络

当 Hopfield 网络规模的扩大和能量函数的复杂度增加时, 其神经元的状态陷落在局部最优点的可能性也随之增加。如前所述, 将 Hopfield 网络与遗传算法相结合是一种提高 Hopfield 网络的全局收敛率的方法。遗传算法是基于自然进化原理的一种自适应寻优方法, 遗传群体中每个个体是寻优搜索空间的一种可能解, 它被编码为基因形式。初始遗传群体通常由随机方法产生, 也可根据特定问题的先验知识确定。每个个体的适应度反映其优劣程度, 并决定其进入下一代进化的概率。进化过程中的基因交叉、基因变异和个体选择运算使得遗传群体的

平局适应度逐步提高, 直至收敛。因此采用遗传算法进行优化计算需要决定: 编码方法、初始群体产生方法、适应度函数定义、遗传算子选择, 以及遗传群体规模、交叉变异率等的控制参数。

目前研究的遗传算法和 Hopfield 网络相结合的方法都是遗传进化整个 Hopfield 网络^[12~14], 即对整个神经网络进行编码、遗传交叉和变异运算, 因此当 Hopfield 网络规模很大时, 遗传个体的基因编码长度庞大, 相应的进化计算量非常大, 造成收敛速度较慢, 跳出局部最优陷阱的能力降低。例如利用 Hopfield 网络和对整个神经网络运算的遗传算法相结合的方法优化图像分割阈值平面时, 对一幅 256×256 像素、灰度级的图像, Hopfield 网络神经元个数为 256×256 , 对应的遗传算法搜索空间包括 $256^{256 \times 256}$ 个可能的解。对于如此巨大的搜索空间, 遗传算法帮助 Hopfield 网络跳出局部最优陷阱的能力几乎丧失。再例如, 对于大规模的 TSP 问题求解时, 由于搜索空间同样巨大, 遗传算法所能提供的帮助也十分有限。

针对上述问题, 提出局部进化的 Hopfield 网络, 其原理框图如图 1 所示, 算法包括 Hopfield 网络的状态方程迭代和遗传算法对 Hopfield 网络进行局部进化计算两大部分。在 Hopfield 网络的状态方程迭代过程中, 首先初始化 Hopfield 网络各个神经元的输出状态, 由式(1)计算各个神经元的输入, 再由神经网络状态式(2)计算神经元在下一时刻的输出状态, 然后返回到计算各个神经元的输入步骤, 直到收敛。收敛后转入遗传算法对 Hopfield 网络进行局部进化计算。

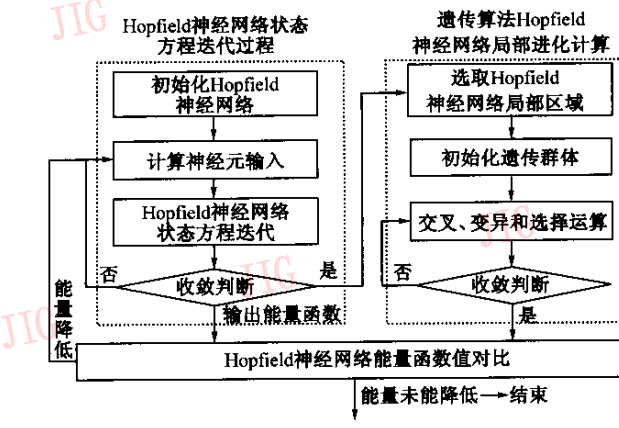


图 1 局部进化的 Hopfield 神经网络

在遗传算法对 Hopfield 网络进行局部进化计算过程中, 选择 Hopfield 网络的某些不同的局部区域分别进行进化优化计算, 这些局部区域的进化遵

守三个原则: (1) 在整个神经网络中随机选择局部区域; (2) 对多个局部区域分批进化, 每次只对一个局部区域进行进化计算; (3) 对每一个局部区域进化计算时, 保持局部区域以外的神经元的输入输出状态不变, 以使得在局部区域的能量降低的情况下, 整个神经网络的能量降低。局部区域的形状根据具体优化问题来确定, 例如对图像分割问题可采用矩形, 而对 TSP 问题需采用交叉的条形, 图 2 和图 3 分别是矩形和交叉条形的示意图。交叉条形局部区域的选择可以保证满足求解 TSP 问题时对换位矩阵的限制条件^[6]。

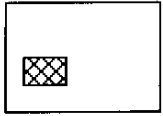


图 2 矩形局部区域

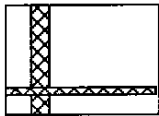


图 3 交叉条形局部区域

针对某个局部区域, 将区域内所有神经元的输出状态进行串行编码构成遗传个体的基因。随机产生遗传群体中的初始个体后, 经过对基因的交叉和变异运算, 以及根据个体适应度的选择运算, 使得局部区域的能量逐渐降低直到遗传算法收敛。

接下来进行 Hopfield 网络能量函数值对比, 即比较某个局部区域在 Hopfield 网络状态方程迭代收敛后的能量函数值和此局部区域经重新初始化和进化计算收敛后的能量函数值。如果进化计算后局部区域的能量函数值更低, 则将对应的遗传个体基因解码为神经元的输出状态, 并代回 Hopfield 网络, 使整个神经网络能量函数值降低, 再进入下一轮的 Hopfield 网络的状态方程迭代过程。如果所有的局部区域经进化计算后能量函数值未能降低, 则整个算法收敛结束。矩形局部区域的进化和代回 Hopfield 网络过程如图 4 所示。

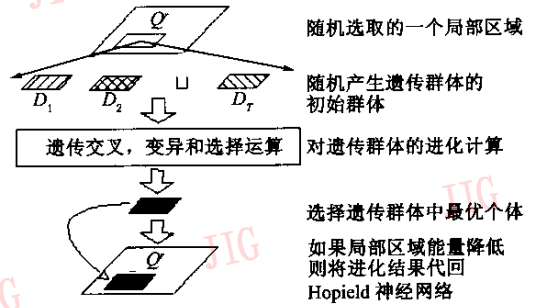


图 4 矩形局部区域的进化和代回 Hopfield 网络的过程

4 Hopfield 神经网络局部区域能量函数分析

设 Hopfield 网络为一 $N \times M$ 的矩形神经元阵列 Q , 神经元之间权重和神经元输入偏移量分别为 $W_{ij,kl}$ 和 $B_{ij}; (i, j) \in Q$ 。它可划分为 K 个局部区域 $\{Q^{(1)}, Q^{(2)}, \dots, Q^{(K)}\}$, 并满足条件

$$Q^{(r)} \cap Q^{(s)} = \emptyset, r \neq s; r, s = 1, 2, \dots, K \quad (11)$$

$$Q^{(1)} \cup Q^{(2)} \cup \dots \cup Q^{(K)} = Q \quad (12)$$

对任一局部区域 $Q^{(r)}$, 定义其神经元之间权重和神经元输入偏移量分别为 $W_{ij,kl}^{(r)}$ 和 $B_{ij}^{(r)}; (i, j) \in Q^{(r)}$

$$W_{ij,kl}^{(r)} = W_{ij,kl}, (i, j), (k, l) \in Q^{(r)} \quad (13)$$

$$B_{ij}^{(r)} = B_{ij} + \sum_{(k,l) \in (Q-Q^{(r)})} W_{ij,kl} V_{kl} \quad (i, j) \in Q^{(r)} \quad (14)$$

其中, $Q-Q^{(r)}$ 表示属于 Q 但不属于 $Q^{(r)}$ 的区域。根据式(3), 区域 $Q^{(r)}$ 形成的 Hopfield 子网络的能量函数为

$$E^{(r)} = -\frac{1}{2} \sum_{(i,j) \in Q^{(r)}} \sum_{(k,l) \in Q^{(r)}} W_{ij,kl}^{(r)} V_{ij} V_{kl} - \sum_{(i,j) \in Q^{(r)}} V_{ij} B_{ij}^{(r)} + \sum_{(i,j) \in Q^{(r)}} \int_0^{V_{ij}} g^{-1}(V) dV \quad (15)$$

整个 Hopfield 网络的能量函数可表示为

$$\begin{aligned} E &= -\frac{1}{2} \sum_{(i,j) \in Q} \sum_{(k,l) \in Q} W_{ij,kl} V_{ij} V_{kl} - \sum_{(i,j) \in Q} V_{ij} B_{ij} + \sum_{(i,j) \in Q} \int_0^{V_{ij}} g^{-1}(V) dV \\ &= -\frac{1}{2} \left(\sum_{(i,j) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \right) \left(\sum_{(k,l) \in Q^{(r)}} + \sum_{(k,l) \in (Q-Q^{(r)})} \right) W_{ij,kl} V_{ij} V_{kl} - \\ &\quad \left(\sum_{(i,j) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \right) V_{ij} B_{ij} + \left(\sum_{(i,j) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \right) \int_0^{V_{ij}} g^{-1}(V) dV \\ &= -\frac{1}{2} \left(\sum_{(i,j) \in Q^{(r)}} \sum_{(k,l) \in Q^{(r)}} + \sum_{(i,j) \in Q^{(r)}} \sum_{(k,l) \in (Q-Q^{(r)})} + \sum_{(i,j) \in (Q-Q^{(r)})} \sum_{(k,l) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \sum_{(k,l) \in (Q-Q^{(r)})} \right) W_{ij,kl} V_{ij} V_{kl} - \\ &\quad \left(\sum_{(i,j) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \right) V_{ij} B_{ij} + \left(\sum_{(i,j) \in Q^{(r)}} + \sum_{(i,j) \in (Q-Q^{(r)})} \right) \int_0^{V_{ij}} g^{-1}(V) dV \end{aligned} \quad (16)$$

由于 $\sum_{(i,j) \in Q^{(r)}} \sum_{(k,l) \in (Q-Q^{(r)})} W_{ij,kl} V_{ij} V_{kl} = \sum_{(i,j) \in (Q-Q^{(r)})} \sum_{(k,l) \in Q^{(r)}} W_{ij,kl} V_{ij} V_{kl}$, 将式(14)和式(15)代入式(16), 可得

$$E = E^{(r)} - \frac{1}{2} \sum_{(i,j) \in (Q-Q^{(r)})} \sum_{(k,l) \in (Q-Q^{(r)})} W_{ij,kl} V_{ij} V_{kl} - \sum_{(i,j) \in (Q-Q^{(r)})} V_{ij} B_{ij} + \sum_{(i,j) \in (Q-Q^{(r)})} \int_0^{V_{ij}} g^{-1}(V) dV \quad (17)$$

式(17)显示,当区域 $(Q-Q^{(r)})$ 的神经元输出状态不变时,局部区域 $Q^{(r)}$ 内的 Hopfield 子网络的能量降低将使得整个 Hopfield 网络的能量降低。因此对 Hopfield 网络局部区域的进化计算同样可以使得整个 Hopfield 网络跳出可能的局部最优陷阱,而局部区域的进化计算所需的计算量大大减少。

5 试验结果

5.1 图像分割阈值平面优化计算

进行图像分割阈值平面优化计算时采用的是联

想 P N 4800/2.6G/256MAM 微型计算机,程序用 VC++ 编写。图 5 是原始图像,尺寸为 256×256 像素、像素的灰度等级为 $[0,255]$ 。为了验证本文算法的性能,实验中分别采用了 4 种方法对图 5 进行了优化计算,这 4 种方法分别是:遗传算法、Hopfield 网络、Hopfield 网络与整体进化交替算法^[12]、本文提出的局部进化的 Hopfield 网络。



图 5 原始图像

表 1 列出的是各种算法进行 100 次图像分割的平均实验结果。

表 1 各种算法 100 次对图 5 进行图像分割运算的平均结果

	遗传算法	Hopfield 神经网络	Hopfield 神经网络与整体进化交替算法	局部进化的 Hopfield 神经网络
全局收敛率(%)	100	92	100	100
分割算法迭代次数	32 174 代遗传运算	208 代状态方程运算	135 代状态方程运算和 3 356 代整体遗传运算	285 代状态方程运算和 8 023 代局部遗传运算
收敛速度(s)	2 056.86	2.24	216.53	8.41
分割算法所需内存容量(Mb)	>25	<1	>25	<1

式(5)中的参数 $\alpha=1.0$,各种算法的其他控制参数分别为:

(1) 遗传算法 仅采用遗传算法对整个 Hopfield 网络进化计算,群体规模为 100 个个体,交叉率和变异率分别为 0.1 和 0.02,并采用精英选择算子。

(2) Hopfield 网络 每个神经元的控制参数设置为 $B_{ij}=0.5, C_{ij}=1.414, R_{ij}=0.5$ 。式(2)的控制参数设置为 $U_0=50$ 。各个神经元的初始输入值在范围 $[-100,100]$ 内选择随机产生,然后可得各个神经元的初始输出值,但在边界点上的神经元输出始终不变,等于对应像素点的归一化灰度值。

(3) Hopfield 网络与整体进化交替算法 Hopfield 网络控制参数设置与方法 2 的中相同,遗传算法控制参数设置与方法 1 的中相同。

(4) 局部进化的 Hopfield 网络 Hopfield 网络

控制参数设置与方法 2 的中相同,遗传算法控制参数设置与方法 1 中的相同。Hopfield 网络的局部区域采用 25×25 大小的矩形形状,算法在 Hopfield 网络中随机选择了 10 个局部区域进行了进化计算。

在表 1 的实验结果中,单独采用 Hopfield 网络优化图像分割阈值平面时 100 次优化计算中有 8 次收敛在局部最优,Hopfield 网络分割图 5 时只陷到过 一个局部最优点,其能量函数值为 $E=-910546$,图像分割结果如图 6 所示,而图 7 显示的是全局收敛的分割结果,此时最低的能量函数值为 $E=-1142855$ 。采用其他 3 种方法都能 100% 收敛到全局最优。

表 1 的实验结果显示,遗传算法计算量远远超过其他 3 种算法,而本文提出的局部进化的 Hopfield 网络方法收敛速度较快,其计算量只比单独采用 Hopfield 网络方法计算量稍大。



图 6 分割图像收敛在局部 图 7 分割的图像收敛在全局

表 1 还列出了不同算法对计算机存储容量的需求量。由于遗传算法的群体规模为 100 个个体,而图像具有 256×256 像素,因此每个个体需存储 $256 \times 256 \times 4$ 比特信息,遗传算法共需要 $256 \times 256 \times 4 \times 100 / (1\,024 \times 1\,024) = 25\text{Mb}$ 的存储容量。同样 Hopfield 网络与整体进化交替算法也需要同样多的存储容量。当采用局部进化的 Hopfield 网络方法时,由于一次只对一个局部区域进行进化计算,局部进化计算仅需要 $25 \times 25 \times 4 \times 100 / (1\,024 \times 1\,024) = 0.24\text{Mb}$ 的存储容量。

5.2 大规模最佳旅行商 TSP 问题优化计算

采用 Hopfield 网络、基于模拟退火的 Hopfield 网络^[12]所提出的局部进化的 Hopfield 网络对 200 个城市的 TSP 问题优化计算结果如表 2 所示。

表 2 3 种算法求解 200 城市 TSP 的结果

	Hopfield 神经网络	基于模拟退火的 Hopfield 神经网络	局部进化的 Hopfield 神经网络
全局收敛率 (%)	0	7	11
状态方程迭代次数	1 264	4 451	2 094
收敛速度(s)	3 160	11 305	5 988

由于计算量非常大,用 VC++ 编写的程序是在 HPX4000 双 CPU(主频 1.8GHz、内存 4G)工作站上运行的。200 个城市 TSP 的数据来自于 <http://elib.zib.de/pub/Packages/mp-testdata/tsp/tsplib/> 中的 kroA200 数据库,就目前所知,他的全局最优路径长度为 29 368。表 2 显示的是 3 种方法对 kroA200 数据进行 100 次优化计算的统计平局结果。其中 Hopfield 网络控制参数选择为 $\tau = 1$ 、 $A = 500$ 、 $B = 500$ 、 $C = 200$ 、 $D = 500$ 、 $U_0 = 0.001$ 。在计算 Hopfield 网络的状态方程时选择 $U_{ij}(t) = U_{ij}(t-1) + K \frac{dU_{ij}(t-1)}{dt}$,控制参数 K 在第 1 次迭代状态方程时取 0.3,以后取

0.05。模拟退火的温度控制为 $T(t) = 0.9T(t-1)$ 。局部进化 Hopfield 网络时局部区域采用十字交叉形状,局部区域大小为 $2 \times 10 \times 200$ 。

表 2 的计算结果显示,对 200 城市的 TSP 问题优化计算时,仅采用 Hopfield 网络方法在 100 次优化中没有一次能收敛到全局最优。局部进化的 Hopfield 网络比基于模拟退火的 Hopfield 网络的全局收敛率略高,而收敛速度快近 1 倍,体现了局部进化的 Hopfield 网络在计算速度上的明显优势。

参 考 文 献

- Hopfield J J, Tank D W. Neural computation of decisions in optimization problems[J]. Biological Cybernetics, 1985, 52(3): 141~152.
- Hopfield J J, Tank D W. Computing with neural circuits: a model[J]. Science, 1986, 233: 625~633.
- Campadelli P, Medici D, Schettini R. Color image segmentation using Hopfield networks[J]. Image and Vision Computing, 1997, 15(3): 161~166.
- Rout S, Seethalakshmy, Srivastava P, et al. Multi-modal image segmentation using a modified Hopfield neural network[J]. Pattern Recognition, 1998, 31(6): 743~750.
- Chao C H, Dhawan A P. Edge detection using a Hopfield neural network[J]. Optical Engineering, 1994, 33(11): 3739~3747.
- 靳蕃. 神经计算智能基础[M]. 成都: 西南交通大学出版社, 2000: 137~148.
- Bizzarri R. Convergence properties of a modified Hopfield-Tank model[J]. Biological Cybernetics, 1991, 64(4): 293~300.
- Abe S. Global convergence and suppression of spurious states of the Hopfield neural networks[J]. IEEE Transactions on Circuit and Systems-I: Fundamental Theory and Applications, 1993, 40(4): 246~257.
- Aiyer S V B, Niranjan M, Fallside F. A theoretical investigation into the performance of the Hopfield model[J]. IEEE Transactions on Neural Networks, 1990, 1(2): 204~215.
- Peng M, Gupta N K, Armitage A F. An investigation into the improvement of local minima of the Hopfield network[J]. Neural Networks, 1996, 9(7): 1241~1253.
- Wells D M. Solving degenerate optimization problems using networks of neural oscillators[J]. Neural Networks, 1992, 5(6): 949~959.
- Watanabe Y, Mizuguchi N, Fujii Y. Solving optimization problems by using a Hopfield neural network and genetic algorithm combination[J]. Systems and Computers in Japan, 1998, 29(10): 68~74.
- Huang J S, Liu H C. Object recognition using genetic algorithms with a Hopfield's neural model[J]. Expert Systems with Application, 1997, 13(3): 191~199.
- Imada A, Araki K. Application of an evolution strategy to the

Hopfield model of associative memory [A]. In: Proceeding of 1997 IEEE International Conference on Evolutionary computation[C]. San Diego, 1997: 679~683.

黎 明 1965 年生,教授,博士生导师,1997 年于南京航空航天大学获得测试技术与仪器专业博士学位。主要研究领域为进化计算、人工神经网络和模式识别等。



杨小芹 1964 年生,副教授,1984 年于西北工业大学电子工程系获学士学位。主要研究方向为智能化测试技术与图像处理等。



周琳霞 1972 年生,讲师,2002 年于南昌航空工业学院获得测试技术与仪器专业硕士学位。主要研究领域为图像处理和模式识别等。