

# 引入竞争机制的连续 Hopfield 神经网络矢量 量化算法设计与实现<sup>\*</sup>)

余春东<sup>1,2</sup> 李 磊<sup>2</sup> 孙世新<sup>1</sup> 王茂芝<sup>1,3</sup> 邓洪勤<sup>4</sup>

(电子科技大学计算机科学与工程学院 成都610054)<sup>1</sup> (中国科学院软件研究所 北京100080)<sup>2</sup>

(成都理工大学信息管理学院 成都610059)<sup>3</sup> (西昌卫星发射中心 西昌615000)<sup>4</sup>

**摘 要** 本文对照经典的矢量量化算法的不足,讨论了基于竞争机制的连续 Hopfield 神经网络矢量量化算法的设计与实现。详细描述了网络映射过程、网络能量函数的刻画和神经元状态转换方程。实验结果表明,与经典的 LBG 算法相比,本文所提算法具有更好的性能和强大的并行处理能力以及更优良的全局优化能力。

**关键词** 矢量量化,码书,竞争机制,Hopfield 神经网络,能量函数

## Algorithm Design and Implementation of Vector Quantization Using Competitive Continuous Hopfield Neural Network

SHE Chun-Dong<sup>1,2</sup> LI Lei<sup>2</sup> SUN Shi-Xin<sup>1</sup> WANG Mao-Zhi<sup>1,3</sup> DENG Hong-Qin<sup>4</sup>

(College of Computer Science and Engineering, UESTC, Chengdu 610054)<sup>1</sup>

(Institute of Software, Chinese Academy of Sciences, Beijing 100080)<sup>2</sup>

(College of Information Managing, Chengdu University of Technology, Chengdu 610059)<sup>3</sup> (Xichang Satellite Emit Center, Xichang 615000)<sup>4</sup>

**Abstract** This paper discusses the implementation of vector quantization algorithm based on competitive continuous Hopfield neural network contrasting to the defects of traditional VQ algorithm. At the same time, the details on network mapping, energy function constructing and neuro state changing equation have been described. The results of the experiment indicate the performance of the algorithm proposed in this paper is more efficient with powerful parallel ability and workable global optimization effect contrasting to LBG.

**Keywords** Vector quantization, Codebook, Competitive mechanism, Hopfield neural network, Energy function

## 1 引言

自从 Y. Linde, A. Buzo 与 R. M. Gray 在1980年给出矢量量化码书设计的 LBG<sup>[1]</sup>算法以来,矢量量化作为图像压缩编码中的一种重要的技术得到广泛的关注和研究。从数学的观点看,矢量量化中的码书设计实质上就是把系统的率失真函数看成目标函数的一个在高维空间使率失真函数最小的全局优化问题。LBG 算法在给定初始码书的前提下,采用动态聚类的思想,通过不断对码书进行迭代,直到收敛,并把收敛时的码书取为最终码书。这样就导致 LBG 算法不可避免地具有如下两个缺陷:一是对初始码书敏感;二是容易陷入局部极小点。

本文从全局寻优的角度,利用连续 Hopfield 网络的优化计算能力,给出了基于竞争机制的连续 Hopfield 神经网络矢量量化算法设计和实现过程,详细介绍了算法操作过程中的网络映射和能量函数构造以及神经元状态转换方程等细节。

## 2 矢量量化

矢量量化可以定义为从  $k$  维欧几里得空间  $R^k$  到其一个有限子集  $C$  的映射,即  $Q: R^k \rightarrow C$ , 其中  $C = \{C_1, C_2, \dots, C_N \mid C_i \in R^k\}$  称为码书,  $C_i$  称为码字。该映射满足:  $Q(V \mid V \in R^k, V = (v_1, v_2, \dots, v_k)) = C_i$ , 其中  $C_i = (C_{i1}, C_{i2}, \dots, C_{ik})$  为码书  $C$  中

的码字,并满足  $d(V, C_i) = \min_{1 \leq j \leq N} (d(V, C_j))$ 。其中,  $d(V, C_j)$  为矢量  $V$  与码字  $C_j$  之间的失真测度。常见的失真测度为均方误差。定义为

$$d(V, C_j) = \|V - C_j\|^2 = \sum_{i=1}^k (v_i - c_{ij})^2 \quad (1)$$

码书设计过程即是要寻找子集  $C$ , 使得

$$J(C) = \min_C \left\{ \sum_{V \in R^k} \min_{C_i \in C} d(V, C_i) \right\}, C_i \in R^k \quad (2)$$

## 3 基于竞争机制的连续 Hopfield 神经网络矢量量化算法设计与实现

### 3.1 连续 Hopfield 神经网络

连续 Hopfield 神经网络是由美国物理学家 J. Hopfield 于 1984 年提出的。由于连续 Hopfield 网络成功地应用于 TSP 问题的求解,从而使得它在组合优化问题计算中得到广泛应用<sup>[2~8]</sup>。组合优化问题就是在给定的约束条件下,求出使目标函数极小(或极大)的变量组合问题。将 Hopfield 网络应用于求解组合优化问题,就是将优化的目标函数转化为网络的能量函数,把问题的变量对应于网络的状态,然后让网络按照状态方程进行转化,当网络收敛时,其能量函数达到极小值,而网络的状态就对应于问题的最优解。

令  $V_{x,i}$  表示神经元  $(x, i)$  的输出,  $W_{x,i,y,j}$  表示神经元  $(x, i)$

<sup>\*</sup>) 本文得到中国科学院知识创新工程方向性研究项目基金(名称:大型数字对象应用环境及其并行模拟,批准号:KG CX2-JG-09)的支持。余春东 博士研究生,主要研究方向为并行计算与数据库技术、图像压缩编码技术。李 磊 研究员,博士生导师,主要研究方向为数据库和图像压缩编码技术。孙世新 教授,博士生导师,主要研究方向为并行计算技术。王茂芝 硕士,讲师,主要研究方向为图像压缩技术。邓洪勤 高级工程师,主要研究方向为航天技术。

和神经元  $(y, j)$  之间的连接权,  $\theta_{x,i}$  表示神经元  $(x, i)$  的偏置。对于二维的全互连连续 Hopfield 网络, 其能量函数和神经元  $(x, i)$  的状态转换方程分别为<sup>[9]</sup>:

$$E = -\frac{1}{2} \sum_x \sum_y \sum_i \sum_j W_{x,i,y,j} V_{x,i} V_{y,j} + \sum_x \sum_i \frac{1}{R_{x,i}} \int_0^{V_{x,i}} g^{-1}(\zeta) d\zeta - \sum_x \sum_i \theta_{x,i} V_{x,i} \quad (3)$$

$$C_{x,i} \frac{dU_{x,i}}{dt} = -\frac{U_{x,i}}{R_{x,i}} + \sum_y \sum_j W_{x,i,y,j} V_{y,j} + \theta_{x,i} \quad (4)$$

其中  $g$  为 S 形函数, 一般定义为  $g(x) = 1/(1 + e^{-\lambda x})$ ,  $\lambda$  为一个调节参数,  $R_{x,i}$  和  $C_{x,i}$  为网络的性能参数, 其含义可参考文[9]。  $U_{x,i}$  为神经元  $(x, i)$  的状态值。

### 3.2 矢量量化码书设计的 Hopfield 网络映射

若把一幅图像分成  $n$  块, 每块的大小和码书中码字的维

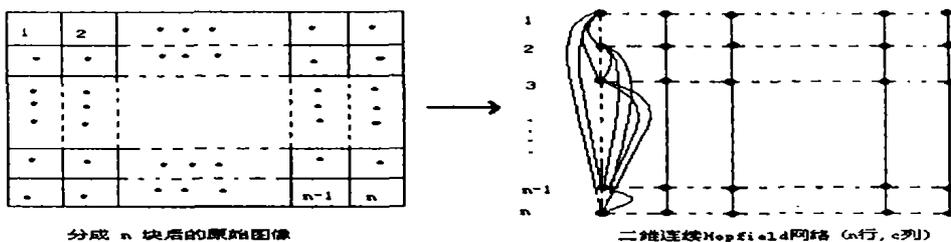


图1 码书设计映射成二维连续 Hopfield 网络

### 3.3 矢量量化码书设计的 Hopfield 网络能量函数刻画和状态转换方程的描述

由于在聚类的过程中, 追求的是同类元素之间“距离”最小, 或不同类元素之间“距离”最大。在此, 采用同类元素之间“距离”最小的策略。又因为在映射后的二维连续 Hopfield 网络中列的含义为类, 行则对应为训练矢量。为了把优化的目标——同类元素之间的“距离”最小——转化为网络的能量函数, 需要对 Hopfield 网络全互连方式进行调整, 即只对相同列的神经元进行互连, 这样就可以把同类元素之间的“距离”转化为网络的能量函数。图1示出了第一列神经元的互连情况。

3.3.1 能量函数的刻画和状态转换方程的描述 在上述互连方式下, 二维连续 Hopfield 网络能量函数的定义由(3)式修改为如下方式:

$$E = -\frac{1}{2} \sum_{x=1}^n \sum_{y=1}^n \sum_{i=1}^c W_{x,i,y,i} V_{x,i} V_{y,i} + \sum_{x=1}^n \sum_{i=1}^c \frac{1}{R_{x,i}} \int_0^{V_{x,i}} g^{-1}(\xi) d\xi - \sum_{x=1}^n \sum_{i=1}^c \theta_{x,i} V_{x,i} \quad (5)$$

其中  $n$  为训练矢量个数,  $c$  为码书中类的数目。改变之处在于把下标  $(y, j)$  修改为  $(y, i)$ , 从而实现对相同列(即同类元素)计算“距离”的约束。其它量的含义和(3)式相同。同时, 神经元的动态转换方程则由(4)式修正为如下形式:

$$C_{x,i} \frac{dU_{x,i}}{dt} = -\frac{U_{x,i}}{R_{x,i}} + \sum_{y=1}^n W_{x,i,y,i} V_{y,i} + \theta_{x,i} \quad (6)$$

神经元  $(x, i)$  的状态迭代按下式进行:

$$U_{x,i}(t+1) = U_{x,i}(t) + \frac{dU_{x,i}(t)}{dt} \quad (7)$$

由于  $W$  是对称的, 即  $W_{x,i,y,i} = W_{y,i,x,i}$ , 所以能量函数  $E$  关于输出  $V_{x,i}$  的偏微分由(5)式得:

$$\frac{\partial E}{\partial V_{x,i}} = -\sum_{y=1}^n W_{x,i,y,i} V_{y,i} + \frac{1}{R_{x,i}} U_{x,i} - \theta_{x,i} = -C_{x,i} \frac{dU_{x,i}}{dt} \quad (8)$$

在基于矢量量化的图像压缩过程中, 目标函数是同类元素(训

数一致, 设为  $k=l \times l$ 。假定码书的大小为  $c$ 。那么对训练矢量规模为  $n$ , 码书大小为  $c$  的图像进行矢量量化, 转化为二维连续 Hopfield 神经网络来实现时, 对应的神经元个数为  $n \times c$ 。这是由于矢量量化的过程其实就是把  $n$  个训练矢量分别归入码书中  $c$  类的某一类的过程。在二维连续 Hopfield 神经网络中, 以列代表码书中的类, 行代表训练矢量。若限定二维连续 Hopfield 网络神经元的状态只能为 0 和 1, 为 0 表示某个训练矢量不属于某一类, 为 1 则意味着训练矢量归入相应的类。同时, 限定每一行所有神经元最终只能有一个神经元的状态值为 1, 即某个训练矢量只能归入某一类中。这样就完成了矢量量化的二维连续 Hopfield 神经网络映射过程。如图 1 所示。映射后网络的连线方式在 3.3 小节中介绍。

训练矢量)之间的“距离”最小, 从而映射后的二维连续 Hopfield 网络能量函数为:

$$E = \frac{A}{2} \sum_{x=1}^n \sum_{y=1}^n \sum_{i=1}^c \frac{1}{\sum_{h=1}^c V_{h,i}} V_{x,i} D_{x,y} V_{y,i} + \frac{B}{2} \sum_{x=1}^n \sum_{i=1}^c \sum_{j=1}^c V_{x,i} V_{x,j} + \frac{C}{2} \left[ \left( \sum_{x=1}^n \sum_{i=1}^c V_{x,i} \right) - n \right]^2 \quad (9)$$

$$\left\{ \begin{aligned} \left( \sum_{x=1}^n \sum_{i=1}^c V_{x,i} - n \right) &\rightarrow 0 & (9.1) \\ \left( \sum_{i=1}^c \sum_{j=1}^c V_{x,i} V_{x,j} \right) &\rightarrow 0 & (9.2) \\ \left( \sum_{x=1}^n \sum_{i=1}^c \sum_{j=1}^c V_{x,i} V_{y,j} \right) &\rightarrow 0 & (9.3) \end{aligned} \right.$$

其中  $\sum_{h=1}^c V_{h,i}$  表示属于码书中第  $i$  类的训练矢量的个数,  $D_{x,y}$  表示训练矢量  $X$  和  $Y$  的平均失真, 表示为:

$$D_{x,y} = E[d(X, Y)] = \frac{1}{n} \sum_{i=1}^n d(X_i, Y_i) \quad (10)$$

(10)式中  $d(X, Y)$  的定义和公式(1)一致。

在(9)式中, 第一项表示码书中  $c$  类的总平均失真。第二项表示在最后的分类中, 每个训练矢量只能属于某一类中。它分两步来实现: 第一步保证某个(如第  $x$  个)训练矢量归入  $c$  个类的某一类中, 它是通过第  $x$  ( $x=1, 2, \dots, n$ ) 行(代表第  $x$  个训练矢量)的所有元素两两相乘之和应为 0 的约束来实现的(见约束 9.2 式); 第二步是保证每个训练矢量都只归入某一类中, 它是通过  $n$  个行(对应  $n$  个训练矢量)所有元素两两相乘之和也应为 0 来实现的(见约束 9.3 式)。第三项保证所有的  $n$  个训练矢量只能归入码书中的  $c$  个类中, 也就是说在网络中最后只有  $n$  个神经元的值为 1, 它是通过约束(9.1)式来保证的, 取平方是为了符合能量表达形式, 也体现了对不符合约束条件时的一种惩罚, 同时也是计算上保证能量函数非负性的一种手段。综合地讲, 第一项是使所有类中训练矢量的平均失真最小, 这是借助于对每一类的总平均失真求和后最小来实

现的,第二和第三项分别是惩罚项,是通过拉格朗日乘子法来转化为能量函数(9)式的。

对于二维连续 Hopfield 网络的能量函数(9)式,按照状态方程(6)和(7)进行迭代直到收敛,其收敛结果和参数  $A, B, C$  的取值有很大关系,而寻找参数  $A, B, C$  的最优权值是费时和困难的,文[10]对此进行了探讨。在此,通过引入竞争机制把能量函数中的惩罚项简化,从而很好地解决这一问题。

3.3.2 竞争机制的引入 处于相同行的所有神经元对于同一输入训练矢量相互竞争,竞争获胜的神经元采取胜者全取(WTA: Winner-Take-All)的策略使其输出为1,相同行中其它神经元输出都为0。即对于相同行的神经元输入输出关系如下式:

$$V_{x,j}(t+1) = \begin{cases} 1, & \text{if } j=m=\arg\{\max_i(U_{x,i}(t))\} \\ 0, & \text{if } j \neq m \end{cases} \quad (11)$$

通过竞争机制的引入,可以保证每行只有一个神经元的输出为1,即竞争获胜的神经元;同时,它还可以保证  $n$  个训练矢量都将归入  $c$  类中。而且,可以使网络快速收敛到能量函数的极小值。下面分析引入竞争机制后对网络能量函数的影响,以及如何实现网络的快速收敛。

把(11)式代入网络能量函数(5)和(9)式中,网络能量函数可以简化为如下形式:

$$E = \frac{1}{2} \sum_{x=1}^n \sum_{y=1}^n \sum_{i=1}^c \frac{1}{\sum_{l=1}^n V_{l,i}} V_{x,i} D_{x,y} V_{y,i} \quad (12)$$

连接权值简化为:

$$W_{x,i,y,i} = \frac{1}{\sum_{l=1}^n V_{l,i}} D_{x,y} \quad (13)$$

把(12)式代入(8)式,得到:

$$\frac{dU_{x,i}}{dt} = -\frac{1}{2} \sum_{y=1}^n \frac{1}{\sum_{l=1}^n V_{l,i}} D_{x,y} V_{y,i} \quad (14)$$

所以,通过引入竞争机制,使得二维连续 Hopfield 网络的能量函数转化为(12)式,而且网络的状态迭代方程按照(7)和(14)式进行。这样,大大简化了网络能量函数的复杂性,减少了计算代价,使得网络收敛速度加快。

### 3.4 算法描述

综合上述分析和推导,基于竞争机制的连续 Hopfield 神经网络码书设计算法可描述如下:

(1)初始化。随机赋予网络中  $n \times c$  个神经元的初始输入值,并随机给定初始码书。

(2)利用(11)式的输入-输出关系,计算每一个神经元新的输出。

(3)根据网络动态状态转换方程(7)和(14),迭代计算网络神经元的新的状态值。

(4)返回算法第(2)步,直到网络收敛。

(5)利用网络  $c$  列中每一列神经元的输出对应的训练矢量计算最终码书。

### 3.5 算法实现

3.5.1 初始化的实现 神经元的初始状态实现:给定一个  $U_0$  的值,按  $U_{x,i} = U_{00} + \delta v_{x,i}$  取各神经元的初始状态,式中  $U_{00} = \frac{1}{2} U_0 \ln(nc-1)$ ,  $\delta v_{x,i}$  为  $(-1, 1)$  之间的随机值。

初始码书的实现:按等间隔方式在  $k$  维训练矢量空间中选取。对于分块大小为  $4 \times 4$  的情况,其中间隔长度为8。这是由

于对于原始的  $512 \times 512$  图像,分成  $4 \times 4$  的块后,转化为  $128 \times 128$  的  $k$  维训练矢量图像矩阵。由于码书大小为256的  $k$  维矢量,而  $256 = 16 \times 16$ ,从而可以从  $128 \times 128$  的训练矢量中以行、列步长都为8来选取。对于其它分块大小,同样处理。由于图像在空间域具有相关性,这种选取方式在一定程度上考虑了这种相关性,是一种除分裂法之外的较好方法。

3.5.2 网络收敛性判断 可以通过多种途径实现。一是可以通过网络中神经元的状态是否再改变来实现;二是可以通过网络能量函数是否收敛来判断。本文采用第二种方式。即取定一个判定能量函数是否收敛的收敛阈值  $\epsilon = 0.0001$ ,看能量函数收敛序列的相邻差值是否小于所给定的收敛阈值。小于则认为收敛,否则继续迭代。

3.5.3 最终码书的计算 网络收敛后,把每一列(对应码书中的每一类)中状态为1的神经元所对应的训练矢量全部加起来,然后求平均即得到最终码书。

## 4 实验结果

本文利用256个灰度等级(即8位量化)的  $512 \times 512$  标准 Lena 图像进行实验。实验方案设计为在不同子块大小和不同码书大小的情况下,对比本文所提算法和矢量量化码书设计经典算法 LBG 算法的相关性能。具体地讲,图像子块大小取为  $4 \times 4$  和  $8 \times 8$ ,也就是矢量的维数  $k=16$  和  $k=64$ ,从而导致训练矢量大小分别为  $n=16384$  和  $n=4096$ ;码书  $c$  则分别取为64、128和256。本文主要利用峰值信噪比进行对比。峰值信噪比(PSNR)定义<sup>[11]</sup>如下:

$$PSNR = 10 \lg \frac{255^2 MN}{\sum_{m=1}^M \sum_{n=1}^N [f(m,n) - f'(m,n)]^2} \quad (\text{dB})$$

其中,  $f(m,n)$  为原始图像数据(像素值);  $f'(m,n)$  为经过算法复原图像数据;  $M, N$  为图像的总行数和总列数。性能对比如表1所示。

实验环境:东芝笔记本电脑—TOSHIBA Genuine Intel CPU 600MHz; 64MRAM; WIN98 OS; VC++ 6.0。

表1 竞争连续 Hopfield 算法和 LBG 算法对比

分块大小	算法	峰值信噪比(PSNR: dB)		
		码书大小(c)		
		64	128	256
4×4	竞争连续 Hopfield	31.0144	32.2436	33.9675
	LBG	29.6516	30.5612	31.6784
8×8	竞争连续 Hopfield	28.3576	29.4820	30.5310
	LBG	26.4540	27.0128	28.1905

图2示出了码书  $c=256$  时,不同分块大小的压缩后还原效果图。

结论 本文给出了一种基于竞争机制的连续 Hopfield 神经网络矢量量化算法设计与实现,给出了算法具体操作过程中的网络映射过程以及网络能量函数和神经元状态转换方程的详细描述,最后通过实验对比了所提算法和经典的 LBG 算法在不同码书大小应用于图像压缩的矢量量化码书设计实际时的实验性能。如在码书大小为256时,对于分块大小为  $4 \times 4$  时,所提算法 PSNR 比 LBG 提高2.2891dB;在分块大小为  $8 \times 8$  时,所提算法 PSNR 比 LBG 算法提高2.4405dB。由此也验证了 Hopfield 网络能够有效地回避 LBG 算法的局部收敛性,具有更优良的全局优化能力,取得了令人满意的结果。实

验结果表明,基于竞争机制的连续 Hopfield 神经网络矢量量

化算法具有更好的性能以及强大的并行处理能力。



(a) 分块大小为  $4 \times 4$  的 LBG 算法还原图 (PSNR=31.6784)  
 (b) 分块大小为  $8 \times 8$  的 LBG 算法还原图 (PSNR=28.1905)  
 (c): 分块大小为  $4 \times 4$  的竞争连续 Hopfield 算法还原图 (PSNR=33.9675)  
 (d): 分块大小为  $8 \times 8$  的竞争连续 Hopfield 算法还原图 (PSNR=30.5310)

图2 码书  $c=256$  时两种算法压缩还原图对比

### 参考文献

- 1 Linde Y, Buzo A, Gray R M. An algorithm for vector quantization design. IEEE Trans on Communications, 1980, COM-28(1): 84~95
- 2 Suganthan P N, Teoh E K, Mital D P. Optimal mapping of graph homomorphism onto self organizing Hopfield network [J]. Image and Vision Computing, 1997, 15: 679~694
- 3 Suganthan P N, Teoh E K, Mital D P. Pattern recognition by homomorphic graph matching using Hopfield neural networks [J]. Image and Vision Computing, 1995, 13: 45~60
- 4 Shen Dinggang, Horace H S Ip. A Hopfield neural network for adaptive image segmentation: An active surface paradigm [J]. Pattern Recognition Letters, 1997, 18: 37~48
- 5 Campadelli P, Medici D, Schettini R. Color image segmentation using Hopfield networks [J]. Image and Vision Computing, 1997, 15: 161~166
- 6 Zhou Changhe, Liu Liren. Vector-product Hopfield model [J]. Optics Communications, 1999, 168: 445~455
- 7 Chang C-Y, Chung P-C. Medical image segmentation using a contextual - constraint - based Hopfield neural cube [J]. Image and Vision Computing, 2001, 19: 669~678
- 8 Chen R-M, Huang Y-M. Competitive neural network to solve scheduling problems [J]. Neurocomputing, 2001, 37: 177~196
- 9 Kurita N, Funahashi K-I. On the Hopfield Neural Networks and Mean Field Theory [J]. Neural Networks, 1996, 9(9): 1531~1540
- 10 Talavan P M, Javier. Parameter setting of the Hopfield network applied to TSP [J]. Neural Networks, 2002, 15: 363~373
- 11 Nasrabadi N M, King R A. Image Coding Using Vector Quantization: a Review. IEEE Trans on Commun., 1988, 36(8): 957~971

(上接第171页)

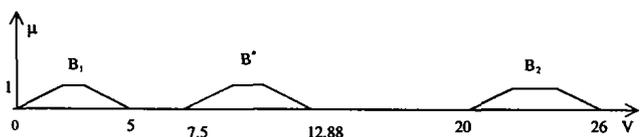


图7 改进的 KH 推理方法的推理结果

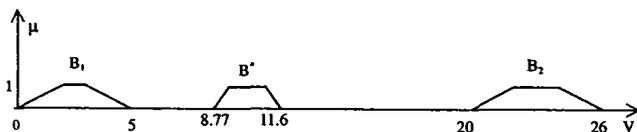


图8 基于几何相似的推理方法的推理结果

通过对以上推理结果图形的对照比较,我们的基于几何相似的推理方法的良好推理效果是显而易见的。从输入事实和推理结果与规则间图形的关系,可以明显看出基于几何相似的推理方法的推理结果远优于改进的 KH 插值推理方法的推理结果和 KH 线性插值推理方法所获得的推理结果,从图中可看出,基于几何相似的推理方法的推理结果  $B^*$  与  $A^*$  的关系和推理规则中  $B_1$  与  $A_1$ ,  $B_2$  与  $A_2$  的关系一致;而且  $B^*$  与  $B_1, B_2$  的关系和  $A^*$  与  $A_1, A_2$  的关系也完全一致。但改进的 KH 插值推理方法却没有这种关系的一致性。这也表明我们的基于几何相似的推理方法更能得到满意的推理结果。

**结论** Koczy 和 Hirota 提出的 KH 线性插值推理方法的推理结果在许多情况下是非凸的,甚至不是模糊集。而改进的 KH 插值推理方法的推理结果不够满意。本文在相似性原理的基础上,提出了基于几何相似的插值推理方法,推理结果具有凸性和正规性,该方法既具有计算方法上的简单与高效,又有满意的推理结果,而且插值方法具有直观的物理意义解释。这为智能系统在稀疏规则条件下进行模糊推理提供了一个十分有用的工具。

### 参考文献

- 1 李凡. 模糊信息处理系统. 北京: 北京大学出版社, 1998
- 2 Zadeh L A. The Concept of a linguistic variable and its application to approximate reasoning Part I, II, III. Information Science, 1975, 8: 199~251, 301~357, 1975, 9: 43~80
- 3 李洪兴. 模糊控制的插值机理. 中国科学(E 辑), 1998, 28(3): 259~267
- 4 Koczy L T, Hirota K. Interpolative reasoning with insufficient evidence in sparse fuzzy rule bases. Information Sciences, 1993, 71(1, 2): 169~201
- 5 Koczy L T, Hirota K. Approximate reasoning by linear rule interpolation and general approximation. International Journal of Approximate Reasoning, 1993, 9(3): 197~225
- 6 Baranyi P, Tikk D, Yam Y, Koczy L T, Nadai L. A new method for avoiding abnormal conclusion for  $\alpha$ -cut based rule interpolation. IEEE International Fuzzy Systems Conference Proceedings, 1999, 8: 383~388
- 7 Kawase S, Chen Q. On fuzzy reasoning by Koczy's linear rule interpolation. [Technical report]. Teikyo Heisei University, Ichihara, 1996
- 8 Shi Y, Mizumoto M. Some considerations on Koczy's interpolative reasoning method. Journal SOFT, 1996, 8: 147~157