

自由曲面的平面映射原理及其算法

梁伟文 宾鸿赞

(华中科技大学机械学院 武汉 430074)

摘要 论述空间曲面的平面映射定义及其变形;分析了空间曲面的几种近似映射方法,并同时分析了各种映射方法的优缺点;针对在空间曲面应用最广泛的 NURBS 曲面,提出了 NURBS 曲面的混合映射算法;讨论了小曲率 NURBS 曲面平面映射的算法,并将该算法成功地应用于花纹鞋模的设计中。

关键词 自由曲面 平面映射 NURBS 曲面

AN ALGORITHM FOR PLANAR MAPPING OF CURVED SURFACE

Liang Weiwen Bin Hongzan

(School of Mechanical Engineering, Huazhong University of Science and Technology, Wuhan 430074)

Abstract Development of a curved surface is a new tool for CAD/CAGD(Computer Assisted Geometric Design). The definition and the deformations of planar mapping of a curved surface are presented in this paper. Then several planar mapping methods of curved surface as well as their advantages and disadvantages are analyzed. For NURBS surface is widely used in CAGD, a new planar hybrid mapping algorithm is put forward. By using this algorithm, distortions of the development are reduced. At last we give an example to validate this algorithm.

Keywords Curved surface Planar mapping NURBS

1 引言

根据曲面理论,三维曲面可分为平面、直纹面和空间曲面;根据曲面的可展性又可分为可展曲面和不可展曲面。空间曲面在工程中应用是相当广泛的。如飞机机翼、叶轮机叶片和流型表面等。但空间曲面几乎是不可展,这为我们的工程设计增加了复杂性。为了简化这种复杂性,将空间曲面映射为参数平面。如在鞋底花纹设计、轮胎花纹设计和瓶模花纹设计时,在空间曲面上设计花纹将是一件很复杂的事情。为此,我们将空间曲面上雕刻花纹的设计降维到平面上进行。首先将空间曲面映射为平面,在平面上进行花纹设计;然后将设计好的花纹纹理映射到自由曲面,根据纹理编辑 3D 花纹。在设计过程中,空间曲面的映射为平面是一个复杂过程。许多学者对此作了一些工作,文献[1]提出的直纹面三角网格分割法,文献[2]、[3]使用有限元计算曲面展开,文献[4]提出三角平面网格法,文献[5]利用曲面自身的几何特性——测地曲率展开曲面,其基本算法是将曲面划分为等参数网格,沿着一个方向,根据曲线的测地曲率,将这些曲线展开在一张平面上。以上各种展开方法各有优缺点,可以完成某种类型曲面的展开。由于 NURBS 方法为初等曲线、曲面和空间型曲面的精确表示和设计提供了一个公共数学形式。本文提出一种基于 NURBS 曲面混合映射方法。

空间曲面的平面映射与微分几何映射的概念密切相关。

微分几何映射定义:设 A 、 B 是两个集合,如果按照某种对应法则 F ,对于集合 A 中的任何一个元素,在集合 B 中都有唯一元素和它对应,这样的对应叫做从集合 A 到 B 的映射,记作 $F: A \rightarrow B$ 。对应曲面的平面映射,其中 A 代表空间曲面, B 代表映射平面, F 是映射算法。研究映射的各种算法,一方面保证映射的一一对应关系;另一方面保证一定的映射精度,即映射变形。以下是对空间曲面映射变形的分析。

2 曲面映射变形分析

空间曲面映射到平面时,其上点之间的相互位置总是要变动,因此曲面上的几何特性——曲线的长度(L)、曲线的方向(D)、曲面的面积(S)受到破坏。其映射所引起的主要变形为:长度变形、面积变形、角度变形。例如:

图 1 是实体球面映射到平面正方形的三种不同映射方法,从图中可以看出,球面上的某一点在三个正方形中的相对位置是不同的。在球面上所有经线的长度相等,而纬线根据不同纬度而长短不一。从长度变形方面分析:从图 1(a)可以看出,各条纬线长度、经线长度相等。这说明各条纬线不是

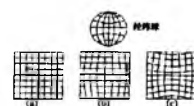


图 1 球面网格化后映射到平面

收稿日期:2002-04-28。梁伟文,博士生,主研领域:计算机辅助设计与制造。

按照同一比例缩小的,而经线却是按同一比例缩小的;图 1(b)中,各条纬线长度都相等,而经线长度不相等。这说明各条纬线不是按照同一比例缩小的,经线也不是按同一比例缩小的;图 1(c)中,各条纬线长度都不相等,经线长度也不相等。这说明各条纬线不是按照同一比例缩小的,而经线却是按同一比例缩小的。从面积变形方面分析:球面上,同一纬度带内,经度差相同的梯形网格面积相等,同一经度带内,纬度愈高,梯形面积愈小。从图 1(a)可以看出,所有网格面积相等,所以其映射规律是不等面积映射。图 1(b)、图 1(c)中,网格面积也有变化,所以其映射规律也是不等面积映射。所谓的角度变形,指曲面上两相交曲线构成的角度,相对于该两曲线映射到平面后所构成角度的变形。图 1(a)中,经纬线相互垂直,其映射规律:等角度映射。图 1(b)、图 1(c)为变角度映射。

根据以上分析,空间曲面映射成平面的精度可以通过以下几个指标来衡量:相对面积误差(ds);相对角度误差(dD);相对长度误差(dL)。因此映射大致可以分为以下几类:①等角映射;②等积(面积)映射;③任意映射;④保持特殊点映射;⑤等距映射。

3 空间曲面的几种近似展开方法

3.1 单圆锥映射法

由于圆锥面是可展曲面。圆锥映射是假定以圆锥面作为映射面,将空间曲面映射到圆锥面上,然后把圆锥面沿一条母线剪开展成平面。圆锥映射的网格线中,纬线是同心圆弧,而经线是同心圆弧半径,且经纬线是相互垂直的,所以经纬线的长度比就是最大、最小长度比。

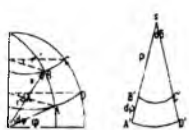


图 2 圆锥映射

由图 2 可以看出,球面上圆弧长 $AB = R d\varphi$, 圆弧 $AD = r d\lambda = R \cos \varphi d\lambda$ 。在映射面上,微分线段 $A'B' = -d\varphi$, $A'D' = \rho d\lambda$, 根据长度比的定义,可得:

$$m = \frac{A'B'}{AB} = \frac{-d\varphi}{R d\varphi}$$

$$n = \frac{A'D'}{AD} = \frac{\rho d\lambda}{R \cos \varphi d\lambda} = \frac{\rho}{R \cos \varphi}$$

因而面积比: $P = mn$, 最大角度变形: $\sin \frac{\varphi}{2} = \left| \frac{m-n}{m+n} \right|$ 。由以上分析可知,单圆锥映射的各种变形都是角度 φ 的函数,与角度 λ 无关。因此,近似映射空间曲面时,在精度允许的条件下,应根据曲面曲率大小,合理选择映射圆锥,以便最终平面映射误差最小。

3.2 多圆锥逼近法

通过以上对单圆锥映射的分析,可以得知 φ 愈大,则变形也愈大。为了减小变形,我们提出多圆锥逼近法,就是用变锥角的多个圆锥来逼近被展开曲面。圆锥映射同样存在长度、面积和角度变形。在此,我们还是用典型不可展的二次曲面球来分析该方法。空间曲面经过多圆锥映射到平面后,形成不连续的多个面片,也就是说产生了裂缝或重叠。但是从变形方面来考虑,它有利于减少变形。当采用的圆锥越多,平面映射的精度越高。对于有裂缝或重叠的平面映射图形,对

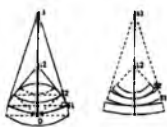


图 3 多圆锥映射

它进行修正,使裂缝或重叠消失。最后用样条曲线拟合各个面片的外轮廓边界线,形成连续的边界,从而形成连续平面。外轮廓线的修正可以采用文献[6]中提出的等距线法或局部面积不变法。

3.3 圆柱映射法

圆柱映射是假定以圆柱面作为映射面,把空间曲面映射到圆柱面,然后再展开圆柱面形成平面。圆柱面映射在工程实际应用中,大多以多个圆柱面片来逼近被展开曲面。根据曲面波动的大小和曲率半径大小,选择不同的圆柱面片逼近。此方法称为多圆柱面片逼近法。

对于圆柱面逼近法,已经做了很多研究工作。Ferris 曾经研究过此问题。Hoschek 和 Schneider 提出了一种空间曲面的近似展开法:通过一序列给定分布点,得到一系列离散线集,把这些线集作为最初的母线,然后通过解非线性最优问题,最后构造了可展曲面。Thomas Randrup 提出的圆柱逼近曲面的平面映射,是基于曲面的 Gaussian 映象,得到一个投影平面。在投影平面上得到参考曲线,最后构造圆柱。他的研究成果应用于船体的设计。

4 自由曲面平面映射

4.1 NURBS 曲面的平面映射

近年来,NURBS 曲线、曲面在工程中应用越来越广泛。STEP 产品数据表达和交换国际标准选用 NURBS 作为几何描述的主要方法。主要因为 NURBS 曲面对标准曲线、曲面和自由曲线、曲面提供了统一的数学参数表示;而且可以通过权因子和控制点来灵活地改变曲面的形状;个别控制顶点和权因子的调整只影响曲面的局部。利用这些特点,可以用它构造各种复杂的曲面造型和表面特殊的效果。由双参数变量分段有理多项式定义 NURBS 曲面:

$$S(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n W_{ij} P_{ij} N_{i,p}(u) N_{j,q}(v)}{\sum_{i=0}^m \sum_{j=0}^n W_{ij} N_{i,p}(u) N_{j,q}(v)} \quad u, v \in [0, 1]$$

其中 p_{ij} 是矩形域上特征网格控制点列, W_{ij} 是相应控制点的权因子, $N_{i,p}(u)$ 和 $N_{j,q}(v)$ 是 p 阶和 q 阶的 B 样条基函数,它们是在节点矢量 $S \{s_0, s_1, \Lambda, s_{m+p-1}\} T \{t_0, t_1, \Lambda, t_{n+q-1}\}$ 上定义的。若在非均匀参数轴上定义的节点矢量 S, T 具有下述形式:

$$S = \{0, 0, \Lambda, 0, s_{p+1}, \Lambda, s_{m+p-1}, 1, 1, \Lambda, 1\}$$

$$T = \{0, 0, \Lambda, 0, t_{q+1}, \Lambda, t_{n+q-1}, 1, 1, \Lambda, 1\}$$

则由 S, T 定义的曲面是非均匀非周期的有理 B 样条曲面,我们简称 NURBS 曲面。

NURBS 曲面是不可展曲面。为了将其映射到平面上,需要将自由曲面首先离散成多个面片,即对曲面进行网格划分。离散后的曲面通过一定的映射算法映射到平面。在文献[7]中, Bennis 等人以保持曲面上等参数线的采样点处的弧长和测地曲率不变将曲面展开。测地曲率是根据测地线得到的。而测地线是曲面任意两点间连线最短的曲线。当曲面可展时,平面上的测地线对应于平面上的直线。基于这个理论, Bennis 等人将参数曲面上的某条等参数线以保持弧长和测地曲率的方式展开为平面曲线,然后以此参数线为基线用相同的方式将整个曲面展开,这时得到一个初步展开结果。然后用松弛迭代法分散上述过程的累积误差。这个算法存在两个不足:一是如何选取

等参数线;另一个是松弛迭代法的收敛问题。为此,设计了一种混合映射算法,该混合算法用等参数 NURBS 曲线将曲面离散。根据矩形域上特征网格控制点列,可以构造 $m+1$ 和 $n+1$ 条 NURBS 曲线。把这些网格线交点作为文献[7]的采样点。同时这些 NURBS 曲线,把曲面被划分为 $m \times n$ 个面片,通过比较面片的面积与平面映射的小区域的面积,判断映射误差,优化映射算法。

4.2 NURBS 曲面混合平面映射算法

步骤:① 在 (u, v) 参数域,分别在 u 方向和 v 方向,等参数 $u = i/m, v = j/n (i=0, 1, \Lambda, m; j=0, 1, \Lambda, n)$ 。对曲面 $S(u, v)$ 离散化。

在 u 方向生成一系列 NURBS 曲线:

$$C(t) = \sum_{i=0}^n W(i) p(i) N_i^k(t) / \sum_{i=0}^n W(i) N_i^k(t)$$

$$N_i^k(t) = \frac{(1-T(i))N_i^k(t)}{T(i+k-1)-T(i)} + \frac{(T(i+k)-t)N_{i+1}^k(t)}{T(i+k)-T(i+1)}$$

其中 k 是曲线的阶次, T 是节点顺序, w 是权因子, p 是控制顶点, N 是基函数:

$$N_i^k(t) = \begin{cases} 1/\Delta & T(i) \leq t \leq T(i+1) \\ 0 & \text{其他} \end{cases}$$

同理在 v 方向生成一系列 NURBS 曲线。

离散后的曲面为四边形面片,分别表示为: $\{P_{ij}\}, i=0, 1, \Lambda, m; j=0, 1, \Lambda, n$, 其中 n, m 由设计者指定。理论上 m, n 越大,映射精度越高,但所计算的时间也越长。

② 根据生成的曲面网格,用 Bennis 方法展开 $\{P_{ij}\}$ 。展开时,基点的选择对展开的累积误差影响很大。在此我们根据文献[6]中,最佳展开基点选择的分析。根据对称展开误差较小,选用参数中点为展开基点,初始等参数线为: $u = m/2, v = n/2$ 。记平面上的映射面片为 $\{D_{ij}\}$ 。

③ 根据曲面面积和平面面积的误差,计算面积变形相对误差。如果误差超出允许的最大范围,则 $m = m+1, n = n+1$,重复映射过程,直到符合要求为止。

④ 最后对映射面进行一些优化、修整,得到我们所要求的展开平面。

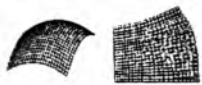


图4 自由曲面的展开 该算法中用离散后的局部四边形面片 P_{ij} 逼近 NURBS 曲面。它是一种离散形式表达的曲面的重新参数化。当曲面是可展曲面时,这种参数化类似于曲线上的弧长参数化,即展开平面上的线段长度等于曲面上对应的曲线的弧长;当曲面是不可展曲面时,这种保长性质在局部范围成立。在算法中,基点的选择是很重要的。不同的基点其展开的平面图形可能是不一样的。在基点选择时,要综合考虑展开的各种变形,而且我们习惯上从一端顺次展开时带来的误差最大,应引起展开时注意。

5 结 论

本文从集合映射的概念引出自由曲面平面映射的定义;以单位球的平面映射为例,分析了曲面平面映射的变形,为曲面展开时精度评价提供了准则;总结了自由曲面的几种传统展开方法及其优缺点;重点介绍了 NURBS 曲面的混合平面映射方法,给出了映射算法步骤。该方法有别于传统平面映射方法,

在局部映射过程中,使用两个精度(面积和长度精度),控制了局部曲面的展开误差,防止了展开误差的积累。同时采用 NURBS 曲线作为曲面的网格线,便于生成离散网格。但对于复杂曲面,其展开图形有缝隙或重叠,并且随着所取展开方向不一样,同一曲面可能有不同的平面展开。

参 考 文 献

- [1] 席平,“三维曲面的几何展开”,《计算机学报》,1997.
- [2] Shimada T, Tada Y. Approximate transformation of an arbitrary curved surface into a plane using dynamic programming, Computer - Aided Design, 1991, 23:153 ~ 159.
- [3] Shimada T, Tama Y. Development of curved surface using finite element method. In: Computer - Aided Optimum Design of Structures. Recent Advances, Springer Verlag 1989, 23 30.
- [4] Phillip N. Azariadis, Nilos A. Aspragathos, On using planar developments to perform texture mapping on arbitrarily curved surfaces, Computers & Graphics 24(2000)539 ~ 554.
- [5] Azariadis P. N., Aspragathos N. A., Geodesic curvature preservation in surface flattening through constrained global optimization, Computer Aided Design, 2001, 33:581 ~ 591.
- [6] 毛昕,“曲面片的可展度及最佳展开基点”,《工程图学报》,1998.
- [7] Bennis Chakib, Jean - Marc Veien, Gerard Iglesias, Piecewise surface flattening for non - distorted texture mapping. ACM Computer Graphics, 1991, 25:237 ~ 246.

(上接第 21 页)

式中的 d 表示干信比值, E_i 表示分析窗内所有谱线的能量, E_{si} 表示分析窗内六根谱线的能量。分析窗指的是频谱图中信号能量比较集中的两个区域,是重点分析的对象,见图 3(b)中的两个矩形框。

$$d = (\sum E_i - \sum E_{si}) / \sum E_i$$

5 结束语

VHDL 是一种针对硬件设计的语言, MATLAB 是一种数学分析工具,使用 MATLAB 来为硬件设计进行分析和测试扩展了 VHDL 仿真系统的功能。本文所举的例子采用 MATLAB 对硬件系统的输出数据进行频域分析,更加直观、明确的显示了 VHDL 仿真系统无法力及的频域参数。实际上本文所描述的方法还比较原始,但切实可行。现在 VHDL 和 MATLAB 之间的接口问题已经有所研究。比如 Xilinx 新近推出的 System Gen 软件系统能在 MATLAB 中进行建模,然后编译生产 VHDL 代码。可以预见,一种 VHDL 仿真系统和 MATLAB 分析系统结合的产物将会出现,并会给设计者们带来巨大的便利^[5]。

参 考 文 献

- [1] 卢毅、赖杰, VHDL 与数字电路设计, 北京: 科学出版社, 2001.
- [2] 黄文梅、熊桂林、杨勇, 信号分析与处理 - MATLAB 语言及应用, 长沙: 国防科技大学出版社, 2000.
- [3] 林瑜筠, 新型移频自动闭塞, 北京: 中国铁路出版社, 2001.
- [4] 申普兵, 数字通信, 长沙: 国防科技大学出版社, 2001.
- [5] Malay Haldar, Optimized Hardware Synthesis for FPGAs - A Dissertation for the Degree of Ph. D., USA: Northwestern University, 2001. http://www.ece.nwu.edu/~malay/htmls/phd/malay_phd.pdf

一种新的半模糊化聚类方法

闫德勤

迟忠先

(辽宁师范大学计算机系 大连 116029) (大连理工大学计算机系 大连 116026)

摘要 本文对模糊聚类方法进行了分析,讨论了截集参数与 FCM 方法中参数 m 的关系及参数选取的依据。提出了一种上下截集半模糊聚类方法(UD-SFCM),给出了该方法的应用与比较实验结果。

关键词 聚类 FCM UD-SFCM

AN APPROACH TO DETERMINATION OF THRESHOLDS FOR SEMI-FUZZY C-MEAN CLUSTERING

Yan Deqin

(Department of Computers, Liaoning Normal University, Dalian 116029)

Chi Zhongxian

(Department of Computers, Dalian University of Technology, Dalian 116026)

Abstract In this paper, Fuzzy C-Mean algorithm is studied. The relationship between parameter m used in Fuzzy C-Mean algorithm and the threshold values used as in Semi-Fuzzy C-Mean is discussed. Besides, a new algorithm for Semi-Fuzzy C-Mean clustering is set forth. Experimental results show the new algorithm is more efficient and reliable.

Keywords Data clustering FCM UD-SFCM

1 引言

聚类分析领域中的 FCM(Fuzzy-C-Mean)方法由于其较传统的硬聚类(HCM)方法更能客观地适应现实应用而成为聚类研究的热点^[1-4]。为提高计算效率, Selim 等人在[5]中首次提出了半模糊聚类(Semi-Fuzzy-C-Mean 简称为 SFCM)方法。之后, Kamel 等^[6]对 SFCM 方法进行了改进。对半模糊聚类方法而言,重要的是如何处理模糊关系矩阵 U 的元素,归结为截集参数的选取。在半模糊聚类方法中,从已有的工作看,截集参数的选取带有直观性与实验性。因此,截集参数的如何选取及其理论依据是一个尚待继续研究的问题。本文在这方面做了初步的探索。在文中我们对 FCM 方法进行了分析,讨论了截集参数的选取与 FCM 方法中参数 m 的关系以及对聚类的影响。在此基础上提出了一种新的半模糊聚类方法:上下截集法(UD-SFCM)。实验结果证明该方法有效快捷。

2 FCM 聚类方法

设 $X = \{x_1, x_2, \dots, x_n\}$ 为 n 元数据集合, $x_i \in R^s$ 。FCM 聚类方法就是把 X 划分为 c 个模糊子集 $S_1, S_2, \dots, S_c$ 。若用 $z_1, z_2, \dots, z_c$ 表示这 c 个模糊子集的聚类中心, u_{ij} 表示元素 x_j 对 z_i 的隶属度,则 FCM 算法为优化目标函数:

$$J_m(U, Z) = \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m d_{ij}^2$$

约束条件为:

$$\sum_{j=1}^n u_{ij} = 1 \quad 1 \leq i \leq c \quad (1)$$

$$u_{ij} \geq 0 \quad 1 \leq i \leq c \quad 1 \leq j \leq n \quad (2)$$

其中, $U = \{u_{ij}\}$ 为 $n \times c$ 矩阵, $Z = [z_1, z_2, \dots, z_c]$ 为 $s \times c$ 矩阵。 d_{ij} 为 x_i 与 z_j 的距离: $d_{ij}^2 = \|x_i - z_j\|^2 = (x_i - z_j)^T (x_i - z_j)$ 。 m 是一个大于 1 的参数。

通过迭代算法,找到目标函数 J_m 的极小点 (U^*, Z^*) :

$$z_i^* = \frac{\sum_{k=1}^n (u_{ik})^m x_k}{\sum_{k=1}^n (u_{ik})^m} \quad (3)$$

$$u_{ik}^* = \left(\frac{1}{d_{ik}^2} \right)^{\frac{1}{m-1}} / \sum_{j=1}^c \left(\frac{1}{d_{jk}^2} \right)^{\frac{1}{m-1}} \quad (4)$$

3 UD-SFCM 方法

FCM 算法中隶属度矩阵 U 的元素由(4)式得到。(4)式可写为:

$$u_{ik} = \frac{1}{1 + \sum_{j \neq i}^c \left(\frac{d_{jk}^2}{d_{ik}^2} \right)^{\frac{1}{m-1}}}$$

收稿日期:2002-04-15。闫德勤,副教授,主研领域:模式识别,自动机理论,图像处理。

$$= \left(1 + \sum_{j=1}^c \left(\frac{d_{ki}}{d_{kj}} \right)^p \right)^{-1} \quad (5)$$

当 $m \rightarrow 1$ 时, $p \rightarrow \infty$ 。因此,若有一 s 使 $\frac{d_{ki}}{d_{ks}} > 1$, 当减小 m 时,即减小 u_{ki} 的值,从而减低 x_k 对 z_i 的隶属度。由于隶属度较小的元素被拒识,因而减小 m 相当于取一截集参数 tu , 使得当 $u_{ki} < tu$ 时令其为 0。同理,若对所有的 $j(1 \leq j \leq c)$ 有 $\frac{d_{ki}}{d_{kj}} < 1$, 当减小 m 时, $u_{ki} \rightarrow 1$, 从而增加 x_k 对 z_i 的隶属度。相当于取一截集参数 td , 使得当 $u_{ki} > td$ 时令其为 1。由此我们得出如下结论: 改变 m 值的大小相当于对隶属度矩阵元素取截集时改变截集参数的大小, 反之亦然。

我们称由截集参数 tu 决定的隶属度矩阵元素集合为上截集, 由截集参数 td 决定的隶属度矩阵元素集合为下截集。由此可知, 无论对隶属度矩阵元素做上截集处理还是对隶属度矩阵元素做下截集处理从聚类实质上讲都相当于对 m 值进行调整, 也提高了运算速度。为此, 我们提出一种新的半模糊聚类方法: 上下截集法 (UD-SFCM)。

算法如下:

step1 选择初始聚类中心 $z_j, 1 \leq j \leq c$ 。选择上下截集参数 tu, td 。

step2 计算距离矩阵 $\{d_{ij}\}, d_{ij}^2 = \|x_i - z_j\|^2, 1 \leq i \leq n, 1 \leq j \leq c$ 。

step3 计算隶属度矩阵 U 的元素 u_{ij} 。

若对某个 r 有 $d_{ir} = 0$, 则 $u_{ir} = 1$, 并且 $u_{ij} = 0 (j \neq r)$, 否则计算: $b = 1 / \sum_{k=1}^c (d_{ij} / d_{ik})^{2(m-1)}$ 。此时, 若 $td < b < tu$, 则 $u_{ij} = b$ 。否则

$$u_{ij} = \begin{cases} 0 & (b \leq td) \\ 1 & (b \geq tu) \end{cases}$$

step4 隶属度矩阵元素归一化:

$$u_{ij} = u_{ij} / \sum_{j=1}^c u_{ij} \quad 1 \leq j \leq c$$

step5 计算新的聚类中心:

$$z_j^* = \sum_{i=1}^n u_{ij}^m x_i / \sum_{i=1}^n u_{ij}^m$$

step6 若 $\|z_j^* - z_j\|^2 < \varepsilon$, 结束。否则, 转向 **step2** ($1 \leq j \leq c$)。

4 实验结果与结论

分别用 FCM, SFCM 和 UD-SFCM 三种方法对 Iris Data 进行分类实验, UD-SFCM 方法得到了较好的结果。对于文献[6]中提到的数据用 UD-SFCM 方法则显示出明显的优势。该数据为一维 6 元数组: $\{-5, -4, 1, 2, 5, 6\}$ 。若分为两类, 即 $c = 2$, 则两个聚类中心应为 -4.5 和 3.5。表 1 给出了上下截集变化时用 UD-SFCM 算法所得到的聚类中心以及 TW 变化时用 SFCM 算法所得到的聚类中心。表 2 给出了 m 变化时用 FCM 算法所得到的聚类中心。表 2 的数据变化证明了截集的变化相当于 m 的变化, 这正是本文第 3 节讨论的结论。另一点要说明的是由于采用上下截集算法其计算量少于 SFCM 算法, 因而更加提高了计算效率。

表 1 用 UD-SFCM 与 SFCM 算法计算的聚类中心结果

UD-SFCM 算法聚类中心	SFCM 算法聚类中心	tu	td	TW
-4.3838, 3.756	-4.3838, 3.756	1	0.2	0.1
-4.3996, 3.692	-4.35, 3.76	0.7	0	0
-4.5, 3.5	-4.3838, 3.756	0.7	0.2	0.2

表 2 用 FCM 算法所得到的聚类中心结果

m 取值	$m = 2$	$m = 1.5$	$m = 1.1$
聚类中心	-4.35, 3.75	-4.47, 3.54	-4.5, 3.5

参 考 文 献

- [1] J. C. Dunn, A Fuzzy Relative of ISODATA Process and Its Use in Detecting Compact, Wellseparated Clusters, J. Cybernet, 1974, 3: 32 ~ 57.
- [2] J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum Press, New York, 1981.
- [3] J. C. Bezdek, R. Hathaway, M. Sabin and W. Tucker, Convergence Theory for Fuzzy C-Means: Counterexamples and Repairs, IEEE PAMI, 1987, 17(5): 873 ~ 877.
- [4] J. M. Jolion et al., Robust Clustering with Applications in Computer Vision. IEEE PAMI, 1991, 8(13): 791 ~ 801.
- [5] S. Z. Selim, M. A. Ismail, Soft Clustering of Multidimensional Data: A Semi-Fuzzy Approach, Pattern Recognition, 1984, 17(5): 559 ~ 568.
- [6] M. S. Kamel, S. Z. Selim, A thresholded Fuzzy c-Mean Algorithm for Semi-Fuzzy Clustering, Pattern Recognition, 1991, 24(9): 825 ~ 833.

(上接第 2 页)

//隐藏文本条调用用户对象函数, 调整控件大小和位置
this.uf.ResizePannels(tv_1, dw_1, ii_barthickness, ii_windowborder, li_workspace, this.x)

3) 按下鼠标左键的用户自定义事件: mousedown

用户自定义事件名为 mousedown, event_id 为 pbm_lbutton-down, 当按下鼠标左键表示拖动即将开始时将静态文本条的背景色变为黑色。其脚本如下:

this.BackColor = 0 //使文本条的颜色变为黑色, 显示拖动轨迹

4 小 结

本文介绍了在 PB 中通过鼠标拖动静态文本条实现窗口中控件大小自动调整的基本原理, 结合实例通过使用用户自定义对象和用户事件给出了一种通用的实现方法及过程。上述实例只是从实现功能上来说明的, 因此只使用了树视图和数据窗口两类控件。如果要实现其它类型的控件大小调整, 可以仿照类似的方法对程序做一点改动就可以实现。以上代码均调试通过, 从实际使用效果来看, 此法方便、可靠。

参 考 文 献

- [1] 吴洁明, PowerBuilder 6.0 应用与开发 [M], 北京: 清华大学出版社, 1998.
- [2] 沃高工作室, PowerBuilder 6.0 应用开发指南 [M], 北京: 北京人民出版社, 1998.
- [3] 刘红岩, PowerBuilder 原理与应用指南 [M], 北京: 电子工业出版社, 1999.

基于粒子群优化的 BP 网络学习算法

王岁花 冯乃勤

李爱国

(河南师范大学计算机科学系 新乡 453002) (西安科技学院计算机科学系 西安 710054)

摘要 本文提出一种新颖的基于粒子群优化的 BP 网络学习算法,该算法是一种全局随机优化算法。用 Iris 分类问题,将所提出的算法与 BP 算法作了对比实验。实验结果表明:所提出的算法性能优于 BP 算法,而且具有良好的收敛性。

关键词 神经网络 粒子群优化 BP 算法

A BP NETWORKS LEARNING ALGORITHM BASED ON PSO

Wang Suihua¹ Feng Naiqin¹ Li Aiguo²

¹(Department of Computer Science, Henan Normal University, Xinxiang 453002)

²(Department of Computer Science, Xi'an University of Science and Technology, Xi'an 710054)

Abstract A novel BP neural networks learning algorithm based on particle swarm optimizers (PSO) is proposed in this paper. A stochastic global optimization technique is used in proposed algorithm. Comparing experiments that classify Iris problem are made between proposed algorithm and BP algorithm. The experiment results show that proposed algorithm does not only superior to BP algorithm, but also has fast congruence speed.

Keywords Neural network Particle swarm optimizer BP algorithm

1 引言

尽管 BP 网络在诸多应用领域取得了巨大的成功^[1],不过仍然存在着一些困扰^[2]:1)学习算法的收敛速度慢;2)局部极小问题,BP 学习过程是一个非线性优化过程,所以不可避免地会遇到优化过程中常见的局部极小问题,令学习结果不满意;3)“过学习”问题,即有时看起来学习结果不错,但实际上泛化能力很差。因此,研究 BP 网络的新的替代学习算法一直是研究者们努力研究的问题。

粒子群优化(Particle Swarm Optimizer, PSO)算法^[3-7]是一类新兴的随机全局优化技术,因此是一类有着潜在竞争力的神经网络学习算法。文献[8]研究了用 PSO 算法训练积单元神经网络取得了良好的效果。但是,将 PSO 算法用于训练其它结构的神经网络的研究尚不多见。

本文提出一种基于 PSO 的 BP 网络学习算法,并用 Iris 分类问题,与 BP 算法作了对比实验。实验结果令人满意,说明 PSO 算法适合训练 BP 网络,并且也有希望应用于其它种类的前向神经网络的训练。

2 PSO 算法^[3]

D 维搜索空间中,有 m 个粒子,其中第 i 个粒子的位置是 $\vec{x}_i = (x_{i1}, x_{i2}, \dots, x_{iD})$, $i = 1, 2, \dots, m$, 其速度为 $\vec{v}_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ 。

$\dots, v_{iD}$)。将 $\vec{x}_i$ 代入目标函数可计算出其适应值。记第 i 个粒子搜索到的最优位置为 $\vec{p}_i = (p_{i1}, p_{i2}, \dots, p_{iD})$, 整个粒子群搜索到的最优位置为 $\vec{p}_g = (p_{g1}, p_{g2}, \dots, p_{gD})$ 。粒子状态更新操作如下:

$$v_{id} = wv_{id} + c_1 r_1 (p_{id} - x_{id}) + c_2 r_2 (p_{gd} - x_{id}) \quad (1)$$

$$x_{id} = x_{id} + v_{id} \quad (2)$$

其中, $i = 1, \dots, m$, $d = 1, \dots, D$; w 是非负常数,称为惯性因子。 w 也可以随着迭代线性地减小^[5];学习因子 c_1 和 c_2 是非负常数; r_1 和 r_2 是介于 $[0, 1]$ 之间的随机数; $v_{id} \in [-v_{max}, v_{max}]$; v_{max} 是常数。

迭代中止条件一般选为最大迭代次数和粒子群迄今为止搜索到的最优位置满足适应阈值。

3 基于 PSO 的 BP 网络学习算法

BP 网络是一种多层结构的前向网络,其隐层神经元的激活函数为 S 型函数,而输出神经元的激活函数可以是 S 型函数,也可以是线性函数,典型的 S 型函数如下:

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

其中 x 是神经元的加权输入函数。

收稿日期:2002-10-31。王岁花,硕士,主研领域:软件理论,算法设计与分析。

用 PSO 算法训练 BP 网络时,定义粒子群的位置向量 $\vec{x}$ 的元素是 BP 网络的全体连接权和阈值。首先初始化位置向量 $\vec{x}$,然后用 PSO 算法搜索最优位置,使如下均方误差指标(适应值)达到最小

$$J = \frac{1}{N} \sum_{i=1}^N \sum_{j=1}^C (y_{j,i}^d - y_{j,i})^2 \quad (4)$$

其中, N 是训练集的样本数; $y_{j,i}^d$ 是第 i 个样本的第 j 个网络输出节点的理想输出值; $y_{j,i}$ 是第 i 个样本的第 j 个网络输出节点的实际输出值; C 是网络输出神经元的个数。

基于 PSO 的 BP 网络学习算法流程如下:

1) 选定粒子数 m ; 适应值阈值 ε ; 最大允许迭代步数 $T_{\max}$; c_1 、 c_2 和 w ; 初始化 X 和 V 为 $(0,1)$ 间的随机数。

2) 迭代步数 $t=0$; $J_g = \infty$; $\vec{J}_p = (\infty, \dots, \infty)$

3) while($J_g > \varepsilon$ & $t < T_{\max}$)

4) for $i=1:1:m$

5) 根据 x_i 和训练机样本,按式(4)计算 J_i ;

6) if $J_i < J_p(i)$ $J_p(i) = J_i$; $p_i = x_i$; endif

7) if $J_i < J_g$ $J_g = J_i$; $p_g = x_i$; endif

8) end for.

9) for $i=1:1:m$

10) 按式(1)计算 v_i ;按式(2)计算 x_i ;

11) end for

12) $t = t + 1$;线性减小 w ;

13) end while

上述流程中, $X = (x_1, x_2, \dots, x_m)$, 其中 $x_i = (x_{i1}, x_{i2}, \dots, x_{iD})^T$ 是第 i 个粒子的位置; $V = (v_1, v_2, \dots, v_m)$, 其中 $v_i = (v_{i1}, \dots, v_{iD})^T$ 是第 i 个粒子的速度; $\vec{J}_i$ 是 m 个粒子迄今搜索到的最优适应值,其对应的粒子位置矩阵是 $p = (p_1, \dots, p_m)$; J_g 是粒子群迄今搜索到的最优适应值,对应的最优粒子位置是 p_g ; 线性减小 w 的策略有多种,下式是其中较好的一种:

$$w(t) = w_0 - w_1 \times (t/T_{\max}) \quad (5)$$

其中, w_0 和 w_1 是常数。 w_0 是 w 的最大值,而 $w_0 - w_1$ 则是 w 的最小值。

4 实验

Iris 分类问题(Iris 是一种鸢尾属植物)是用于测试分类器性能的一个著名的基准函数数据集。在数据记录中,每组数据包含 Iris 花的四种属性:萼片长度,萼片宽度,花瓣长度,和花瓣宽度,三种不同的花各有 50 组数据。这样总共有 150 组数据。将总共 150 组数据分为两个集合:训练集合与测试集合,每个集合 75 组数据。

实验用 3 层 BP 网络对 Iris 问题来做分类。网络有四个输入节点和三个输出节点;每个输入节点代表花的一个属性,而每个输出节点代表一类花。隐层节点的数目分别选为 6、12、18、24 和 30。隐层节点和输出节点的激活函数如(3)式。

在实验中,PSO 算法的参数固定取粒子数为 20; $c_1 = c_2 = 2$, w 随迭代次数由 0.7 线性地减小为 0.3;迭代次数取 100 次。分别用第 3 节的算法和 BP 算法在训练集上训练神经网络。而用

测试集的样本对训练好的神经网络进行测试。

用测试集作分类时,将某一样本施加到网络的输入节点,如果网络输出节点的输出值都满足下式:

$$|y_j^d - y_j| \leq \alpha, j=1, \dots, C \quad (6)$$

则认为分类正确,否则,分类错误; α 是阈值,在 $[0,1]$ 之间取值。该文的实验中,取 $\alpha = 0.5$ 。

评价网络分类效果的指标如下:

$$E = \frac{en}{N} \times 100\% \quad (7)$$

其中, en 表示错误分类的样本数目; N 表示总的样本数目。

为比较该文算法与 BP 算法^[1,2] 的计算效率,提出均方误差计算次数作为比较两种算法迭代次数的依据。对该文的算法为:均方误差计算次数 = 迭代次数 * 种群的粒子数;而对 BP 算法,均方误差计算次数 = 迭代次数。

表 1 是当均方误差计算次数确定为 2000,本文算法和 BP 算法在不同隐层节点数时分别对测试集的分类结果。实验中用的 BP 算法参见文献[2]。而图 1 则是本文算法训练 BP 网络时的 3 种典型情况的均方误差演化曲线。

由表 1 知,本文算法都优于 BP 算法。值得注意的是:尽管图 1 显示 3 种情况的最终的均方误差值相差并不十分显著,但是对照表 1 不难发现,这 3 种情况对测试集的分类结果相差却比较显著。这说明这 3 种网络拓扑结构的泛化能力是不同的。因此,均方误差指标并不能完全反映网络的分类能力,这是在使用 BP 网络时应该注意的问题。

观察图 1 不难看出,本文的算法具有良好的收敛性质,克服了 BP 算法存在的“假饱和”的问题,说明本文的算法不仅性能相对较优,而且收敛快。

表 1 两种算法对 Iris 问题的分类结果

网络拓扑	E(%)	
	本文算法	BP 算法
4-6-3	38.67	42
4-12-3	22.67	31.33
4-18-3	21.33	23.54
4-24-3	9.33	15.67
4-30-3	4	10.46

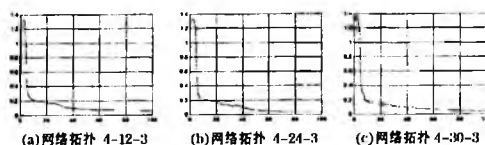


图 1 PSO 算法训练 BP 网络的误差演化曲线

纵坐标:均方误差;横坐标:迭代次数

5 结论

本文提出一种基于 PSO 的 BP 网络学习算法,并用 Iris 分类问题,与 BP 算法作了对比实验,实验结果说明:本文提出的算法性能优于 BP 算法,而且具有良好的收敛性。

参考文献

[1] 焦李成,神经网络系统理论[M],西安电子科技大学出版社,1990.

- [2] 李孝安、张晓敏,神经网络与神经计算机导论[M],西北工业大学出版社,1995.
- [3] 李爱国、覃征、鲍复民、贺升平,“粒子群优化算法”,《计算机工程与应用》,2002,21:1~3.
- [4] Kennedy J, Eberhart R, Particle Swarm Optimization[C], IEEE Int'l Conf. on Neural Networks, 1995, 4:1942~1948.
- [5] Shi Y, Eberhart R, A modified particle swarm optimizer [C], IEEE World Congress on Computational Intelligence, 1998, 69~73.
- [6] Ciuprina G, Ioan D, Munteanu I, Use of Intelligent - Particle Swarm Optimization in Electromagnetics [J], IEEE Trans. on Magnetics, 2002, 38(2): 1037~1040.
- [7] Clerc M, Kennedy J, The Particle Swarm - Explosion, Stability, and Convergence in A Multidimensional Complex Space [J], IEEE Trans. on Evolutionary Computation, 2002, 6(1):58~73.
- [8] Van den Bergh F, Engelbrecht A P, Training Product Unit Networks Using Cooperative Particle Swarm Optimizers [C], Proc. of the third Genetic and Evolutionary Computation Conference, 2001.

~~~~~

(上接第35页)

```
i = i + 1
Wend
ado_record.close '关闭记录集
```

至此,已经实现了 Lotus Domino 系统与 ERP 之间的数据交互。

### 3.2 B-O 接口模型构建及其实现

Domino 服务器所支持的传统的 Notes 客户端同大多数 ERP 系统的客户端一样,不能提供向客户提供基于浏览器交互界面。但在处理与客户关系和电子商务有关的业务时,B/S 结构的系统往往是最佳的选择。

此外,尽管 Domino 服务器也提供了直接的 WEB 发布功能,但在实际应用中,它在浏览器上的逻辑实现和界面表现能力都相当薄弱。为此,我们在 Domino 的基础上构造了 B-O 接口模型(如图 2 所示)。

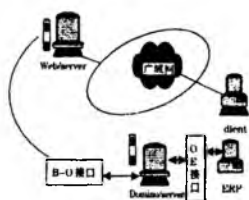


图2 B-O接口模型

在B-O接口中,采用动态链接库对需要完成的功能进行封装,以组件的形式在企业内部系统与WEB应用之间形成了一个中介。这样做的优势在于:首先,通过该中介可以在OA与ERP的协同工作系统(借助O-E模型实现)之上提供统一的WEB交互逻辑;其次,摆脱了Domino在WEB应用上的限制,将浏览器端的表示任务交给更加专业的WEB服务程序去完成;此外,比起WEBSphere等系统来说,这种方法的成本是相对较低的。

当用户直接通过Web来向Domino/server发出请求时,它必须经过一个中间过程,那就是组件。在组件里放置的是一个编译好的DLL,当DLL接受到请求时,就立即向Domino发出请求并取回数据交给用户,而在此过程中,最重要的是如何从Domino/server取出数据并放回。

示例程序如下:

```
以下是为此类定义了一个属性,名为"Name"
Public Name As String
以下是为此类定义了一种方法
```

```
Public Sub show()
Dim session As Object
Dim db As Object
Dim doc As Object
Dim dbusers As Object
Dim maildoc As Object
Dim collection As Object
Dim peopledoc As Object
Dim searchformula As String
Dim recipients(1 To 4) As String
Dim varr As Variant
Dim dbRQ As Variant
Dim m As Variant
Set session = CreateObject("Notes.NotesSession")
Set db = session.GetDatabase("oaserver", "某某电子公司 \ WrSp.nsf")
searchformula = "Form = ""原材料采购计划"" & bh = 1234213""
Set collection = db.Search(searchformula, Nothing, 0)
Set peopledoc = collection.getnthdocument(1)
If peopledoc Is Nothing Then
Exit Sub
Else
varr = peopledoc.getitemvalue("bh")
For Each m In varr
recipients(1) = recipients(1) + ";" + m
Next
End If
name = recipients(1)
End Sub
```

在调用上述DLL以后,就可以从Domino/server数据库中取出所需要的数据。同样,如果想把前台的数据返回到或存入Domino/server数据库中,可以用上面的逆过程。实际上,B-O接口的实现并不复杂,关键的技术难点在于对发布逻辑的封装。

## 4 结束语

在企业信息集成的过程中可以采用的技术路线很多。其中,基于Lotus Domino的接口技术从性能和成本的角度来考虑不失为一种有效的方法。

笔者在多家“江苏省中小企业信息化示范工程”项目的设计、开发过程中,借助B-O-E三级接口来完成异构系统的集成工作,均取得了较好的效果。

## 参考文献

- [1] 林昱、钱昆, Lotus Domino R5, 电子工业出版社, 2001, 6.
- [2] [美] Ash Rofail, Yasser Shohoud Com 与 Com +, 电子工业出版社, 2000, 4.
- [3] 罗琳、陈怡之, Notes 编程疑难详解, 人民邮电出版社, 2000, 10.
- [4] 宋汝良, “跨组织的工作流协同模型及协同规范”, 《计算机辅助工程》, 2000, 12.
- [5] 叶立新等, “基于工作流技术的OA系统模型”, 《计算机系统应用》, 1999, 10.
- [6] 王斌, “基于notes的oa系统公文流转中几个关键问题及解决方法”, 《信息工程学院学报》, 1998, 12.