文章编号:1006-2467(2009)06-0906-04

一种新型的自适应蚁群聚类算法

张 蕾[1], 曹其新[1], 李 杰[2]
(1. 上海交通大学 机器人研究所, 上海 200240; 2. 辛辛那提大学 NSF I/UCR 中心 辛辛那提 OH 45221)

摘 要: 提出了一种新型的自适应蚂蚁聚类算法. 该算法将每个待聚类模式看作一只蚂蚁,采用蚂蚁移动模型实现模式的聚类. 为了改善蚂蚁移动的随机性,提高运行效率,提出了一种局部最近邻运动原则来指导蚂蚁的移动;并且提出了一种自适应调整蚂蚁移动阈值的方法以简化参数的选取. 通过数据的聚类对该算法和已有算法进行了比较. 结果表明,该算法具有运行效率高、参数选取简单及自适应性等优点.

关键词: 群体智能; 聚类; 蚁群算法
中图分类号: TP 18 **文献标识码**: A

A New Self-Adaptive Clustering Algorithm Based on Ant Swarm

ZHANG Lei[1], *CAO Qi-xin*[1], *LEE Jay*[2]
(1. Research Institute of Robotics, Shanghai Jiaotong Univeristy, Shanghai 200240, China;
2. NSF I/UCR Center for Intelligent Maintenance Systems, University of Cincinnati, OH 45221, USA)

Abstract: To reduce the randomness of ant's movement and improve the efficiency, a principle moving to the nearest neighbor in the local environment was suggested to guide the ant's movement. Moreover, a method to self-adaptively adjust the threshold of ant's movement was presented, which simplified the parameter's selection. The algorithm was compared with others through the application of data clustering. The results show the proposed algorithm has high efficiency, simple parameters and self adaptivity.
Key words: swarm intelligence; clustering analysis; ant algorithm

群体智能(Swarm Intelligence, SI)是指简单个体通过交互和协作而表现出来的一种复杂智能行为. 其中蚁群算法是研究最为广泛和成熟的,它是基于蚂蚁觅食行为的模拟进化算法,目前已在旅行商问题、二次分配问题、网络路由问题等离散组合优化问题中得到了成功的应用[1].

群体智能聚类(Swarm Intelligence Clustering, SIC)模型来源于对蚁群打扫蚁穴行为的观察. Marco 等[2]利用不同的蚂蚁类型进行了实验,发现某些类型的蚂蚁能够将分散在蚁穴各处的蚂蚁尸体分拣成大小不同的堆. 越大的堆会吸引蚂蚁把更多的尸体堆放在这一堆上,这种现象的本质是一种正反馈机制. Deneubourg 等[3]提出了解释这种聚类的基本模型,称为 BM 模型. Lumer 等[4]将 BM 模型应用到数据的聚类分析,设计了 LF 算法,主要思想是将待聚类模式随机分布在一个二维网格上,然后由蚂蚁测量当前对象在局部环境内的群体相似度,并将这种群体相似度通过概率转换函数转换成拾起或放

收稿日期:2008-07-27
基金项目:国家自然科学基金资助项目(50705054);上海交通大学青年教师启动基金资助项目(A2846B)
作者简介:张 蕾(1975-),女,山东潍坊市人,讲师,研究方向:智能维护、智能机器人。电话(Tel.):021-34206547;
E-mail:zhanglei@sjtu.edu.cn.

下的概率,通过群体之间的相互作用,经多个循环后即可实现模式的聚类. Handl 等[5]将 LF 算法应用于文本的聚类. 吴斌等[6]在 LF 算法的基础上,采用比基本模型更简单的概率转换函数,用于客户行为的分析.

虽然 LF 算法得到了不断改进,但在实际应用时仍有 2 个比较突出问题:一是蚂蚁移动的随机性使得算法很多时间都浪费在了无效移动上;另一个是参数的设置和选取对算法有很大影响,缺乏合理的指导原则. 徐晓华和 Chen 等[7,8]给出了一种新的蚂蚁运动模型,称为 AM 模型. AM 模型将每个聚类的数据看作一个蚂蚁,同样采用相似度和概率转换形成聚类. AM 模型在一定程度上解决了 BM 模型时间成本较高的问题. 本文在采用 AM 模型的基础上,提出了一种局部最近邻运动原则来改善蚂蚁移动的随机性,蚂蚁的相似度衡量和运动判定准则均采用一种简单的计算方法以简化参数选取,并提出了一种自适应调整蚂蚁运动阈值的方法,使得蚂蚁可以根据当前的聚类情况不断调整阈值,达到最好的聚类结果.

1　AM 模型和聚类算法

在 BM 模型中,模式被随机投影到一个二维平面上,然后虚拟的蚂蚁被放到该平面上,对模式进行拾起或者放下. AM 模型与 BM 模型最大的区别在于 AM 模型将待聚类的模式看作一个个蚂蚁,并模拟蚂蚁在生存环境中寻找一个舒适的位置来休息的行为,每只蚂蚁的状态很简单——动或者不动,如果判断周围环境不适合休息,蚂蚁就继续移动;如果适合则停止不动. 蚂蚁判断周围环境的准则可以和 BM 模型中完全相同. 由此可见,AM 模型只不过是将原来模式被动地被虚拟蚂蚁捡起而变成模式主动地移动.

AM 模型中,模式仍然是投影到二维的网格空间,蚂蚁在网格中的移动采用 8 个邻居随机选择的方式. 本文只借鉴 AM 中所提出的将模式看作蚂蚁的思想,而蚂蚁移动或静止、移动的方向、移动阈值的设定等均采用本文提出的方法. 这种新型的自适应蚂蚁聚类(New Ant Clustering, NAC)算法的实现过程可简单描述为:将待聚类模式(蚂蚁)投影于一个二维网格上,计算每只蚂蚁当前在局部环境中的群体相似度,然后根据阈值判断是移动还是静止,如果移动,则按照局部最近邻原则,移向局部最近的一个蚂蚁;如果静止,则保持不动. 如此对所有模式进行多次循环而形成聚类.

2　自适应蚂蚁聚类算法的实现

2.1　模式投影

AM、BM 模型中,模式均被随机投影到一个二维平面,这种随机投影的方式使得在聚类的初始阶段,每个模式与其局部环境中其他模式的相似度比较低,导致模式很容易被捡起却不容易被放下,从聚类开始到各模式与局部环境内的其他模式,初步具有一定的相似性需要耗费大量的时间. 因此,本文提出首先对各模式进行主成分分析,然后对前 2 个主成分进行一定的处理并作为投影坐标. 由于前 2 个主成分能保留模式的大部分信息,因此相近的模式投影后会挨得比较近. 这种投影方式的优点主要体现在:聚类初始时,各模式与局部环境内的模式具有很大的相似性,相当于原算法已经执行了很多次循环后的结果,因此算法的运行效率会大大提高;此外,这种投影方式所保证的局部相似性为蚂蚁采用局部最近邻移动方式提供了基础.

利用主成分分析对模式进行预处理及投影的详细过程可参见文献[9].

2.2　群体相似度的计算

群体相似度是蚂蚁(模式)与其所在局部环境中其他蚂蚁(模式)的综合相似度. 文献[5]中给出了基本的群体相似度计算式:

$$f(O_i) = \sum_{O_j \in L(O_i, r)} \left[1 - \frac{d(O_i, O_j)}{\alpha} \right] \qquad (1)$$

式中:$L(O_i, r)$ 为模式 O_i 的局部环境,为以 r 为半径的圆形区域;$d(O_i, O_j)$ 为模式 O_i、O_j 之间的距离,通常为欧式距离;α 为群体相似系数,其值对聚类中心的个数以及算法的收敛速度具有重要的影响. 如何调整 α 需依据个人经验,缺乏理论指导而难以掌握.

本文采用了一种更简单的相似度衡量方式. 由式(1)可知,相似度衡量的本质由模式间的距离决定,因此直接采用 d 来表示相似度. 模式 O_i 与其局部环境中其他模式的相似度为

$$f(O_i) = \frac{1}{n} \sum_{O_j \in L(O_i, r)} d(O_i, O_j) \qquad (2)$$

式中,n 为模式 O_i 的局部环境内其他模式的个数. 式(2)表明,$f(O_i)$ 的值越大,说明模式 O_i 的群体相似度越小.

此外,本文算法不再采用概率转换函数,而是直接设定相似度阈值 F,$f(O_i)$ 与阈值进行比较可以看出蚂蚁是移动还是静止. 这种方法简单易行,而且避免了概率转换函数中参数取值对算法的影响.

2.3 蚂蚁的局部最近邻移动原则

在 SIC 算法中,蚂蚁的移动是随机的,蚂蚁捡起(放下)一个模式后,都是随机地选择一处坐标,如果在此坐标处没有放下(拾起)模式,则再随机选择. 如果阈值没能及时调整,更会导致蚂蚁长时间负载一个模式放不下,或者长时间空闲. 文献[7,8]中,蚂蚁的移动采取在 8 个邻居随机选择其一的方式,这种随机移动使得算法的很多时间都浪费在了蚂蚁的无效移动上. 因此,本文提出一种局部最近邻移动原则来改善蚂蚁的移动.

一只蚂蚁 O_i 的局部最近邻就是在其局部环境中,与该蚂蚁最相似的蚂蚁,记为 O_k,数学表达为

$$d(O_i,O_k)=\min\{d(O_i,O_j)\},\quad O_j\in L(O_i,r) \tag{3}$$

如果蚂蚁根据阈值判断局部环境不适合休息,则蚂蚁移向自己的局部最近邻. 这种原则的直接含义是蚂蚁向离自己局部最近的伙伴靠近,这符合物以类聚的常理. 蚂蚁聚类的最终目标是找到全局最近的伙伴,那么,在局部环境下,先寻找最相似的伙伴完全合理. 在这种原则指导下,蚂蚁移动的下一个坐标定义为

$$\left.\begin{aligned} xo_i(t+1)&=xo_k(t)\ \pm1\\ yo_i(t+1)&=yo_k(t)\ \pm1\end{aligned}\right\} \tag{4}$$

式中:t 为循环次数;可随机选择"+"、"-",目的是使蚂蚁靠近当前最相近的蚂蚁,但两者的坐标不完全重合. 相比原来的随机移动,这种移动方式相当于局部寻优,尤其在已经聚类了一段时间,且各蚂蚁和局部环境中的其他蚂蚁已经具备一定的相似性时,这种移动方式更具合理性.

2.4 蚂蚁移动阈值的自适应调整

本文不再采用概率转换函数,而是直接设定相似度阈值 F,将 $f(O_i)$ 与 F 进行比较来决定蚂蚁是移动还是静止. F 必须随算法的进行适当调整,如果 F 过大,会导致蚂蚁老是静止不动;过小,会导致蚂蚁一直在动,找不到合适的地方停下来休息. 因此本文提出一种自适应调整阈值的方法.

在每次循环中,蚂蚁的状态变化有 4 种形式:①移动-移动;②移动-静止;③静止-移动;④静止-静止. 假设这 4 种形式的蚂蚁数量分别为:m_1、m_2、m_3、m_4,蚂蚁数量的总和为

$$N=m_1+m_2+m_3+m_4$$

即 N 也为聚类模式的个数. 在该 4 种形式中,如果形式②、③的蚂蚁数量比较多,说明阈值设置合适,蚂蚁在频繁地改变自己的状态;如果形式①中蚂蚁的数量比较多(本文定为 80 %),则阈值设置过小,应适当加大;如果形式④中蚂蚁数量较多(本文定为 80 %),则阈值设置过大,应适当变小. 因此应着重计算形式①、④中蚂蚁的比例数,且采用多个循环取平均的方式,避免一次循环数的波动性.

$$l_1(t)=\frac{1}{p}\sum_{q=t-p+1}^{t}\frac{m_1(q)}{N} \tag{5}$$

$$l_4(t)=\frac{1}{p}\sum_{q=t-p+1}^{t}\frac{m_4(q)}{N} \tag{6}$$

式中,$l_1(t)$,$l_4(t)$ 分别为每 p 次循环后形式①、④中蚂蚁的比例数. 本文取 $p=50$. 即每循环 50 次计算一次 $l_1(t)$ 和 $l_4(t)$. 阈值自适应调整为

$$F(t+1)=\begin{cases}1.1F(t), & \text{若 } l_1(t)>0.8,\text{且 } \mathrm{mod}(t,50)=0\\ 0.9F(t), & \text{若 } l_4(t)>0.8,\text{且 } \mathrm{mod}(t,50)=0\\ F(t), & \text{其他}\end{cases} \tag{7}$$

上述阈值调整方法只需在每次循环时,利用 1 个变量表示每个蚂蚁的状态变化情况,方便易行.

2.5 NAC 算法的实现步骤

(1) 初始化各参数:最大循环次数 T;蚂蚁个数 N;半径 r(无量纲);相似度阈值初始值 $F(1)$;p.

(2) 对待聚类模式进行主成分分析,保留前 2 个主成分并进行处理;

(3) 将模式投影到二维网格平面,每个模式对应一个蚂蚁,给蚂蚁赋初始坐标.

(4) for $t=1:T$

for $i=1:N$

以蚂蚁 O_i 的当前坐标为中心,r 为半径,利用式(2)计算此蚂蚁在局部环境中的相似度 f. 比较 f 与阈值 $F(t)$ 的大小,如果 $f\le F(t)$,蚂蚁静止不动;否则,按局部最近邻移动原则(式(4)),赋给蚂蚁新的坐标值.

End (蚂蚁个数循环结束)

计算本次循环的 $l_1(t)$、$l_4(t)$

每 p 次循环后按式(7)调整相似度阈值 $F(t)$

End(达到最大循环次数)

(5) 根据聚类结果计算每类的聚类中心,输出各聚类的模式.

3 算法应用

本文利用 NAC 算法与文献[8]中的 SIC 算法及文献[7]中的 AAC 算法进行比较分析,实验数据来自 UCI 机器学习数据库[10],选用鸢草(iris)数据集,共有 150 条记录,每条 4 个属性,类别数为 3. 3 种算法的参数如表 1 所示. 表中:A 为投影面积(无量纲);η 为分类正确率.

表 1　3 种算法的参数比较

Tab. 1　Comparison of three algorithms

算法	T	N	A	r	η %
SIC	60 000	6	480 ×450	20	90.0
AAC	5 000	150	12 ×12	≈1	97.1
NAC	5 000	150	80 ×80	5	96.8

NAC 算法中各模式初始时刻的投影如图 1 所示. 图中,x、y 为横、纵坐标,无量纲. 采用不同的图示表示不同的类别. 由图 1 可见,NAC 算法采用主成分分析后的投影方式,各模式投影后具有了一定的分别,这使得在算法的初始时刻,各模式与其局部环境中的其他模式具有了很大的相似性,因此大大节省了聚类时间. 而且,这种投影方式也为局部最近邻移动提供了基础,如果初始时刻是随机投影,各模式之间凌乱无序,采用局部最近邻移动不一定能取得好的效果.

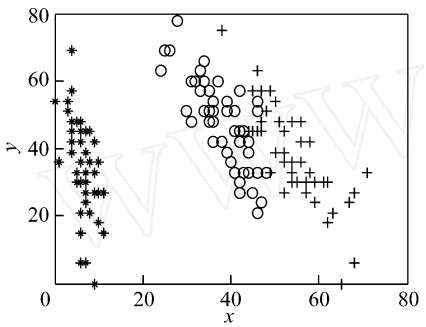

图 1　NAC 算法初始时刻模式的投影图

Fig. 1　Projections of the patterns in NAC algorithm

经反复实验,NAC 一般在经历 4 000～5 000 次循环后,就可以形成较好的聚类结果,聚类的结果如图 2 所示.

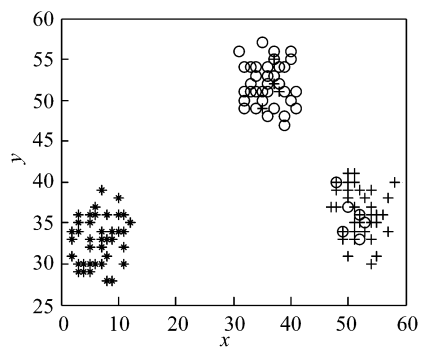

图 2　NAC 算法的聚类结果

Fig. 2　The clustering results of NAC algorithm

比较表 1 中的各参数可知,由于 AM 模型的采用,使得 NAC 在聚类效率和正确率上明显优于 SIC. NAC 和 AAC 的 T、η 比较接近,由于 AAC 的每个蚂蚁被设计成 Agent 的形式,蚂蚁移动采用的邻域类型和本文不同,因此无法进行精确的比较. 但本文采用比 AAC 更简单的相似度计算和自适应的阈值调整,简化了参数的选取,提高了算法的实用性. 阈值自适应调整的曲线如图 3 所示.

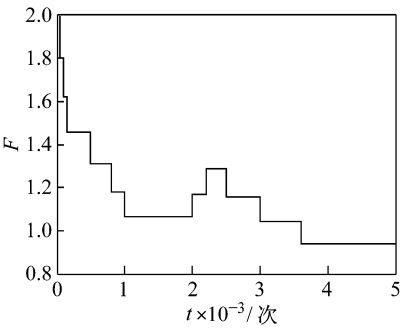

图 3　蚂蚁移动阈值的自适应调整

Fig. 3　The self-adaptively adjusting of ant movement threshold

4　结　语

本文提出了一种基于蚂蚁运动模型新型聚类算法,该聚类算法具有参数选取简单、聚类效率高、自适应性等特点. 并且本文给出了算法的详细实现过程,通过对数据的聚类分析和其他几种算法的比较,证明了本文算法的有效性. 蚁群聚类算法在实际应用中仍存在很多待改进之处,例如本文的局部最近邻移动原则在改善蚂蚁移动随机性的同时,也在一定程度上限制了蚂蚁移动的步幅. 因此,如何适当调节局部环境的半径 r,如何根据蚂蚁的密度指导蚂蚁的移动,是下一步待研究的问题.

参考文献:

[1] Mitra S, Hayashi Y. Bioinformatics with soft computing [J]. **IEEE Transactions on Syatems, Man, and Cybernetics,** Part C: Applications and Revies, 2006, 36(5):616-635.

[2] Marco D, Eric B, Guy T. Ant algorithms and stigmercy [J]. **Future Generation Computer System,** 2000,16(9): 851-871.

[3] Deneubourg J L, Goss S, Frank N, *et al*. The dynamics of collective sorting: robot-like ants and ant-like robots [C]// **Proceedings of the 1st International Conference on Simulation of Adaptive Behavior: From Animals to Animats.** Cambridge, MA: MIT Press/ Bradford Books,1991:356-363.

[4] Lumer E, Faieta B. Diversity and adaptation in populations of clustering ants [C]// **Proceedings of the 3rd International Conference on Simulation of Adaptive Behavior: From Animals to Animats.** Cambridge, MA: MIT Press/ Bradford Books,1994: 501-508.

(下转第 913 页)

本没有冲击信号,只是在低频部分存在着周期信号,也没有周期冲击信号的存在;故障齿轮运转的时频图如图 3(b)所示. 由图可见,匹配追踪方法能够提取出啮合频率及其谐波和冲击信号,时频图上的冲击信号呈准周期分布,间隔时间大约为 0.1 s,正好与 f_o = 10 Hz 相吻合,这预示齿轮的齿尚存在着局部损伤. 因此,匹配追踪算法能够有效、准确地检测出齿轮故障.

4　结　语

本文研究了匹配追踪方法的原理和算法,并将该方法运用于齿轮的仿真信号和实验信号,分解结果与短时傅里叶变换进行对比,发现得到了具有高分辨率的自适应时频分布. 结果表明,该方法具有较高的时频分辨率,并且能够准确确定齿轮的运行状态,为识别齿轮缺陷提供了一种有效手段.

参考文献:

[1] 陈　进. 机械设备振动监测与故障诊断[M]. 上海:上海交通大学出版社, 1999.

[2] Mallat S, Zhang Z. Matching pursuit with time-frequency dictionaries [J]. **IEEE Transaction on Signal Processing**, 1993, 41(12): 3397-3415.

[3] Qian S, Chen D. Signal representation via adaptive normalized Gaussian function [J]. **Signal Processing**, 1994, 36(1), 1-11.

[4] Friedman J H, Stuetzle W. Projection pursuit regression [J]. **Journal of the American Statistical Association**, 1981, 76:817-823.

[5] Gersho A, Gray R M. Vector quantization and signal compression [M]. Boston: Kluwer Academic Publisher, 1992.

[6] Ebrahimi-Moghadam A, Shirani S. Matching pursuit-based region-of-interest image coding[J]. **IEEE Transactions on Image Processing**, 2007, 16(2): 406-415.

[7] Rankine L, Mesbah M. Boashash B. A matching pursuit-based signal complexity measure for the analysis of newborn EEG [J]. **Medical and Biological Engineering and Computing**, 2007, 45(3): 251-260.

[8] Mallat S. A wavelet tour of signal processing [M]. Beijing: China Machine Press, 2003.

[9] Badaoui M E, Antoni J, Guillet F. Use of the moving cepstrum integral to detect and localize tooth spalls in gears [J]. **Mechanical Systems and Signal Processing**, 2001, 15(5): 873-885.

[10] 何　俊,陈　进,毕　果,等. 调循环平稳度解调频原理分析及其在齿轮故障诊断中的应用[J]. 上海交通大学学报, 2007, 41 (11) :1862-1866.
HE Jun, CHEN Jin, BI Guo, *et al*. Frequency demodulation analysis of degree of cylcostationary and its application to gear defect detection [J]. **Journal of Shanghai Jiaotong University**, 2007, 41 (11): 1862-1866.

(上接第 909 页)

[5] Handl J, Meyer B. Improved ant-based clustering and sorting in a document retrieval interface [C]// Merelo Guervcs J J. **Parallel Problem Solving from Nature - PPSN VII, Lecture Notes on Computer Science.** Berlin/ Heidelberg: Spriger-Verlag, 2002: 913-923.

[6] 吴　斌,郑　毅,傅伟鹏. 一种基于群体智能的客户行为分析算法[J]. 计算机学报, 2003, 26 (8) : 913-918.
WU Bin, ZHENG Yi, FU Wei-peng. A customer behavior analysis algorithm based on swarm intelligence [J]. **Chinese Journal of Computers,** 2003, 26(8): 913-918.

[7] 徐晓华,陈　叔. 一种自适应的蚂蚁聚类算法[J]. 软件学报, 2006, 17(9) :1884-1889.
XU Xiao-hua, CHEN Ling, A adaptive ant clustering algorithm [J]. **Journal of Software**, 2006, 17 (9): 1884-1889.

[8] Chen Jian-bin, Sun Jie, Chen Yun-fei. A new ant-based clustering algorithm on high dimensional data space [C]// **14th ISPE International Conference on Concurrent Engineering.** Brazil: ISPE, 2007: 605-611.

[9] 张　蕾,曹其新,李　杰,一种基于群体智能聚类的设备性能横向比较算法[J]. 上海交通大学学报, 2006, 40(3): 439-443.
ZHANG Lei, CAO Qi-xin, LEE Jay. An algorithm for comparing machine performance based on swarm intelligence clustering [J]. **Journal of Shanghai Jiaotong University,** 2006, 40(3) :439-443.

[10] Fisher R A. Iris plants database [EB/OL]. (1988-09-21) [2008-06-27]. http://www.ics.uci.edu/～mlearn/ ML Repository.html.