

多 Agent 并行遗传算法在地震勘探属性优化中的应用

刘其成^{1,2} 郑纬民¹

(清华大学计算机科学与技术系 北京 100084)¹ (烟台大学计算机学院 烟台 264005)²

摘 要 研究了基于多 Agent 的并行遗传算法,并将其应用于石油勘探的属性优化。针对常规遗传算法的不足,采用 Agent 构建多 Agent 系统实现了基于粗粒度的并行遗传算法,该算法能从进化环境中获取表征当前进化状态的有用信息,智能地监控调度 GA 的进化操作,在避免早熟的同时加快全局寻优,提高遗传算法搜索的效率,同时具有通讯开销小的特点。将该方法用于地震勘探属性优化,取得了良好的效果。

关键词 Agent,属性优化,遗传算法,并行

中图分类号 TP302 **文献标识码** A

Seismic Exploration Attribute Optimization Based on Multi-Agent Parallel Genetic Algorithm

LIU Qi-cheng^{1,2} ZHENG Wei-min¹

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)¹

(School of Computer Science and Technology, Yantai University, Yantai 264005, China)²

Abstract The method of design and implementation for parallel genetic algorithm is based on thick grain, with the multi-Agent combined with genetic evolution technology, to optimize the seismic attribution selection. A multi-Agent system includes two kinds of Agents: N-Agent and M-Agent, they can exchange the useful information which can represent the current situation of evolution. The methods are benefit to improve the performance of parallel genetic algorithm, and to raise the searching efficiency of the parallel genetic algorithm. The projecting feature in the parallel module is less communicating overhead. It was used in the optimization of oil exploration attribute selection and the result is good.

Keywords Agent, Attribute optimization, Genetic algorithm, Parallel

在运用地震勘探方法预测和评价石油储层时,通常使用与储层预测有关的各种地震属性。以各种方法提取的一系列地震属性包含着丰富的地质信息,但有些属性可能彼此相关,这就造成了信息的重复和冗余,属性的无限增加也会给储层预测带来不利的影响。针对具体问题,从全体地震属性中挑选出最佳的地震属性子集是非常必要的。地震属性优化问题就是从众多地震属性中挑选出与研究目标关系最密切、反应最敏感的少数属性,再利用优化后的地震属性进行目标层储层参数(如孔隙率、泥质含量和储层厚度等)反演。地震属性优化方法主要有两种:一种是根据专家的知识挑选那些对储层预测最有影响的属性,但所解问题与地震属性间关系复杂时,难于凭借经验直观地选取;另一种是用数学的方法进行筛选比较,找出带有最多储层信息的属性^[1]。

地震属性优化方法中衡量“最优”的标准就是使误识率或预测误差最小。要想得到最优属性组合,可以采用枚举法将各种属性组合的误识率或预测误差进行比较,选出最优属性组合,此即组合优化问题。但是当组合数目大时,不宜用枚举法求解。因此,只好寻找计算量较小的次优解法,常用的有属

性比较法、顺序前进法、顺序后退法等。地震属性优化的新方法有遗传算法等,但是常规遗传算法仍存在进化缓慢、易于早熟、搜索效率不高等缺点,难以在合理的时间内找到满意解。尤其当问题的复杂性及难度增加时,这些弊病就更明显。

Agent 技术是计算机技术的最新发展。作为一种主体,它能从环境中获取信息并对环境进行操作,智能地完成相应的活动。本文拟建由多个 Agent (Multi-Agent) 组成的地震属性优化系统,其中存入遗传算法经多年发展所积累的共识(经验知识),并使其能获取各代繁衍的信息,全程监控、指导演化进程,及时调整进化参数,避免早熟,加速收敛,以期高效地实现全局寻优。

同时,随着高分辨率地震勘探和四维地震(时移三维地震)勘探等新技术的推广,地震数据野外的采集量将变得更加庞大。所以,地震中的高性能计算变得越来越重要,分布式、并行处理的需求更加迫切。现有的遗传算法并行模型均存在通讯代价大、系统效率低以及硬件环境要求高等不足。

采用基于 Agent 的并行遗传算法,把每个个体视作一个 Agent,每个子群体进行中的优秀 Agent 可以与其它子群体

到稿日期:2009-05-27 返修日期:2009-08-26 本文受 973 国家重点基础研究规划计划(2007CB310903),国家自然科学基金(60703055, 60673144)资助。

刘其成(1970-),男,博士,副教授,主要研究领域为高性能计算等,E-mail:ytliuqc@163.com;郑纬民(1946-),男,教授,博士生导师,主要研究领域为高性能计算、网格等。

交互,从而实现通讯代价小而又能确保最终求取最优解的目标。

本文研究基于多 Agent 的并行遗传算法,并将此方法应用于石油勘探的属性优化,取得了良好的效果。该方法在石油勘探属性优化中的应用目前尚未见相关报道。

1 地震勘探属性优化

由于众多的地震属性间并不是相互独立的,有些属性所反映的地质信息是类似的,且各种属性对储层参数的影响也是不同的,因此,在进行地震属性参数与储层参数的相关性研究,并进一步由地震属性参数预测储层参数时,必须分析地震属性参数间的相关性,优选出能够反映储层参数本质特征的、相互之间独立的地震属性参数。在进行储层预测和评价时,通常使用与储层预测相关的各种地震属性。地震属性的引入通常要经过一个从少到多,又从多到少的过程。从少到多是指在设计预测方案的初期阶段应该尽量多地提取各种与储层预测有关的地震属性。这样可以充分利用各种有用的信息,吸收各方面的地质信息,以改善储层预测的效果。从多到少是指针对具体研究对象,从全体地震属性中挑选出最佳的地震属性子集,即地震属性优化问题。

地震属性优化应遵循的基本准则包括:优选后属性整体与研究对象具有某种相关性,能够对样本进行有效分类;达到属性结构的最优化,以尽可能相互独立的变量组成尽可能低维的变量空间;使有用信息损失为最小,剔除起干扰作用的属性。

地震属性优化的工作包括:1)根据研究对象或所解问题,结合由模型理论分析与实际经验所建立的基于储层特征的地震属性分类表及地震、地质学家对具体工作区的经验,对地震属性进行初选;2)根据实际工区地质特征建立的地震地质模型对初选地震属性做进一步的分析和筛选;3)进行井旁道地震属性计算,并利用交绘图等方法,了解所提取属性的总体异常特征分布规律,对与储层特征有明显对应关系的属性进行必要的预处理,形成初选地震属性集;4)运用先进的数学方法对初选地震属性集进行优选,达到最优特征组合,衡量“最优”的标准是在满足上述属性选择基本准则的基础上使得检验样本分类误差最小,即达到有效分类;5)如不能达到有效分类,重复以上步骤。

在用遗传算法解决地震属性的优选问题时,首先应该对每个解都进行编码。假设属性组合与遗传算法中的二进制编码的对应关系为:染色体长度与地震属性个数相同,每个属性分别与每个基因相对应。每个基因用二进制表示(0或1),因此,当某个基因为1时,表示其对应的属性被选用;否则,不参与属性组合。很显然,对任何一种特征组合,存在唯一的一个字符串与之对应。

根据上述方法原理可以对地震勘探属性进行优化处理,利用优化后的地震属性并综合考虑研究区内的地质信息,可以实现储层参数预测。

2 基于 Agent 的并行遗传算法

2.1 并行遗传算法中 Agent 的结构

系统中有多处理机,每个处理机上有一个子群体参与遗传操作。把群体中的每个个体设计成一个普通 Agent(N-

Agent);每个子群体设置一个管理 Agent(M-Agent),起到监视器的作用。每个 N-Agent 分别代表一个参与进化的个体,N-Agent 的个数为染色体的种群容量,它们是遗传算法演化进程的主要承担者;M-Agent 只有一个,负责全程监控,并调度交叉、变异、选择等各种遗传操作。

普通 Agent 的描述如下:

$\langle N\text{-Agent} \rangle ::= \{ \langle \langle \text{染色体} \rangle, \langle \text{适应值} \rangle, \langle \text{年龄} \rangle, \langle \text{生存期限} \rangle, \langle \text{年龄增加标志} \rangle, \langle \text{交叉率} \rangle, \langle \text{变异率} \rangle, \langle \text{交叉标志} \rangle, \langle \text{变异标志} \rangle, \langle \langle \text{设置交叉标志} \rangle, \langle \text{设置变异标志} \rangle, \langle \text{年龄增加} \rangle, \langle \text{消亡} \rangle \} \}$

管理 Agent 的描述如下:

$\langle M\text{-Agent} \rangle ::= \{ \langle \langle \text{最优适应值} \rangle, \langle \text{最优适应值的染色体} \rangle, \langle \text{种群大小} \rangle, \langle \text{年龄} \rangle, \langle \text{生存期限} \rangle, \langle \text{进化代数} \rangle, \langle \langle \text{状态更新} \rangle, \langle \text{新建 N-Agent 监控} \rangle, \langle \text{进化终止监控} \rangle, \langle \text{信息统计} \rangle, \langle \text{选择} \rangle, \langle \text{基因交叉} \rangle, \langle \text{基因突变} \rangle \} \}$

2.2 子种群功能

(1)子种群初始化

按照用户输入的相关参数创建 M-Agent,对 M-Agent 的状态进行初始化。由用户输入的参数包括两类 Agent 的生命种群容量的初值、染色体长度、期限、交叉控制参数 c_1 和 c_2 、变异控制参数 m_1 和 m_2 。再由 M-Agent 根据用户输入参数创建 N-Agent 群,并对 M-Agent 的状态进行初始化。已知的状态填入具体值,其它状态填入 0;染色体采用二进制编码,随机地产生基因值。

(2)进化进程监控

M-Agent 执行 $\langle \text{信息统计} \rangle$,获取 N-Agent 群的状态信息;再执行 $\langle \text{状态更新} \rangle$,记录进化到此处的最优适应值及最优适应值对应的染色体;将 N-Agent 群中最优的适应值作为自身的适应值,并对 M-Agent 自身的年龄进行调整;随后执行 $\langle \text{进化终止监控} \rangle$,由年龄决定是否终止进化过程;当决定不终止进化过程时,执行 $\langle \text{新建 N-Agent 监控} \rangle$,继续进化进程。N-Agent 的适应度由自己计算。

(3)选择操作

选择操作包括两种:①子群体内部的选择操作。M-Agent 应用轮盘赌方法或其它方法将按适者生存原则根据 N-Agent 的适应度从 N-Agent 群中选择染色体作为下一代,被选中的 N-Agent 的年龄增加标志被赋值为 1。同时在上一代的优秀 Agent 中,选出一定比例的优秀 Agent 直接作为下一代。选择出的优秀 Agent 还可以与其它子群体进行交互,或者被其它子群体选择。②引进其它子群体中的优秀 Agent。经评价被确定为优秀的其它子群体中的 Agent,如果具有本子群体中优秀 Agent 更大的适应度,则将它引进加入本子群体,淘汰本子群体中最差的 Agent。然后,N-Agent 执行 $\langle \text{年龄增加} \rangle$ 和 $\langle \text{消亡监控} \rangle$,淘汰年龄过高的染色体,从而避免超级个体逐渐占据种群而造成早熟。

(4)交叉率和变异率的调控

交叉率和变异率是遗传算法的两个重要参数,它们将影响进化的走向,这里采用式(1)、式(2)所示的准则调控交叉率

和变异率^[2]。

$$P_c = \begin{cases} c_1 \cdot \frac{(f_{\max} - f)}{(f_{\max} - \bar{f})}, & f \geq \bar{f} \\ c_2, & f < \bar{f} \end{cases} \quad (1)$$

$$P_m = \begin{cases} m_1 \cdot \frac{(f_{\max} - f)}{(f_{\max} - \bar{f})}, & f \geq \bar{f} \\ m_2, & f < \bar{f} \end{cases} \quad (2)$$

其中, f_{\max} , \bar{f} 分别表示当前 N-Agent 群中染色体的最大适应值和平均适应值, f 为相关染色体的自身的适应值, c_1, c_2 与 m_1, m_2 为 $[0, 1]$ 间的常数。

(5) 交叉操作

设置交叉标志的具体操作如下: 根据环境信息, 由式(1)计算出自身的 P_c , 随后产生一个 $(0, 1)$ 间的随机数。当它大于自身的 P_c 时, 将其交叉标志赋值为 1, 否则赋值为 0。由 N-Agent 自身执行〈设立交叉标志〉, 决定是否设置其交叉标志, 从而决定该 N-Agent 是否需要参加交叉或变异。M-Agent 会将设置了交叉标志的 N-Agent 随机配对执行〈基因交叉〉。完成之后, 清除所有标志, 即赋值为 0。

(6) 变异操作

设置变异标志的具体操作如下: 根据环境信息, 由式(2)计算出自身的 P_m , 随后产生多个 $(0, 1)$ 间的随机数, 其个数等于染色体长度。当第 i 个随机数大于其自身的 P_m 时, 将染色体第 i 个的变异标志位赋值为 1, 否则赋值为 0。由 N-Agent 自身执行〈设立变异标志〉, 决定是否设置其变异标志, 从而决定该 N-Agent 是否需要参加变异。M-Agent 会将变异标志位为 1 的对应的染色体位进行变异, 即执行〈基因变异〉。完成之后, 清除所有标志, 即赋值为 0。

(7) 进化的终止

由 M-Agent 执行〈进化终止监控〉, 在满足相关条件时结束进程, 否则将继续进化。

2.3 多 Agent 交互与通信

在子种群中, 各 N-Agent 从进化环境中获取信息, 并据此调整自身的状态——参数, 执行设置交叉或变异标志、消亡等相应的动作; M-Agent 与所有的 N-Agent 交流并获取信息, 经汇总计算得出演化进程当前的状态, 并向环境发布, 对 N-Agent 的动作产生影响; N-Agent 根据演化进程当前的状态采取〈新建 N-Agent 监控〉、〈状态更新〉等相应的动作。M-Agent 和 N-Agent 都有年龄和生存期限, 当年龄到达生存期限时它们将消亡。M-Agent 的消亡表示整个遗传算法运行终止。

各子群体间的通讯通常有两种方式。这里的模型主要采用共享存储, 使用黑板模型作为各子群体间交互信息的主要工具^[3]。

黑板上记录整个群体中最优秀的个体(B-Agent), 它是一个公共数据区; 同时, 子群体之间不直接交互, 而是与黑板上的 B-Agent 进行交互。通讯过程如下: 一个子群体的 M-Agent 在适当的时机引进黑板上的 B-Agent, 并进行评估选择。如果该 B-Agent 优于本子群体中的最优个体, 则将该 B-Agent 加入自己的子群体中; 如果自己的最优个体优于被引进的 B-Agent, 则将自己的最优个体作为 B-Agent 传递到黑板中。黑板在接收新的来自不同子群体的 B-Agent 时, 首先将它们与现有的 B-Agent 进行比较, 只将最优的 B-Agent 保存在黑板中。对应黑板有一个接收队列——缓冲区, 用于解决

接收时的冲突。即同一时刻若有多个子群体传递它们自己的 B-Agent, 则需要按照某种策略解决这种冲突。

另外, 黑板可以传递搜索终止信息。如果一个子群体从黑板上接收到的 B-Agent 与自己的 B-Agent 一样, 则向黑板发送终止搜索的信息。如果绝大多数子群体都要求终止搜索过程, 则在黑板上发出终止信息, 各子群体在接收终止信息时即结束运行。需要注意的是, 如果某一时刻子群体从黑板上接受的 B-Agent 与自己的 B-Agent 一样, 而没有接收到终止信息, 但随后从黑板接受到的 B-Agent 比自己的 B-Agent 更好时, 需要向黑板发出撤消终止的信息。

2.4 多 Agent 协调和协作

在多 Agent 系统中, 每个 Agent 都具有自主性和理性。在求解过程中会按自己的目的、知识和能力进行活动, 所以常会出现矛盾或冲突。当出现矛盾或冲突时就需要协调, 协调方法分显式和隐式两类。这里考虑采用隐式协调^[4], 即每个子群体中的 Agent 均遵循社会规则, 在社会规则的约束下, 各 Agent 进行有效的合作, 搜索共同的最优目标^[5,6]。

此处的社会规则主要是有关冲突问题的。在多 Agent 系统中, 冲突的类型大致上可以分为 3 大类, 即资源冲突、目标冲突和结果冲突。这里多 Agent 系统中各 Agent 之间不会存在资源冲突和目标冲突, 但结果冲突的情况普遍。正因为各子群体存在着结果不一致, 才致使它们相互协作, 朝着一致结果的方向努力, 一旦这种冲突解决了, 即各子群体得出了统一一致的结果, 那么这个结果就是最优解, 搜索过程也就结束了。因此, 多 Agent 系统中所存在的结果冲突是群体中各 Agent 协作求解的驱动力。

此处的协作方式属水平协作, 即在协作过程中, 每一个 Agent 都具备独立解决问题的能力^[5,7]。但如果各 Agent 相互作用, 共同努力, 将会更有效地求解问题, 并提高解的可信度。

3 应用实例与算法效率分析

本文中所使用的地震资料是某地区的两次三维地震采集的地震资料。我们选取了其中某一区块, 通过对目的层段的记录做自相关、功率谱、傅里叶谱、自回归及其他统计特征分析。在地震剖面上按道或按照体提取地震属性, 这些地震属性体现了储层流体的变化因素。对上述地震属性采用本文的方法进行属性优化处理, 然后预测储层流体的变化。用于优化的地震属性有互相关均值、最大相关、最小相关、最大振幅值、振幅扭度、平均振幅值、最大波谷值、均方根振幅、绝对振幅、瞬时振幅、瞬时频率、瞬时相位等。优化后的地震属性组合体为最大波谷值、平均振幅值、振幅扭度、绝对振幅值、瞬时振幅的方差、瞬时振幅。

基于 Agent 的并行遗传算法属于粗粒度的并行模型, 其最大的特点是通讯代价小, 只存在于子群体之间 B-Agent 的交互, 而且这种交互是有限的。每个子群体在一定时间间隔的某个时刻从公共存储区——黑板获得的信息是当前整个群体的 B-Agent, 各子群体所引进的 B-Agent 都优于其自己的 B-Agent, 因而被引进的 B-Agent 有利于加速该子群体的收敛速度。

设有 h 个子群体, 每个子群体的规模为 i , 每个子群体均进化 k 代。经典粗粒度孤岛并行遗传算法中, 各子群体进化

j 代($j < k$)后进行一次分配和收集操作。每次分配操作的通讯代价为 h_i , 每次收集操作的通讯代价也为 h_i 。当群体进化 k 代时的通讯总代价为 $2h_i k/j$ 。基于 Agent 的并行遗传算法中, 每个子群体独立进化 j 代($j < k$)后进行一次通讯, 每一代均产生了比黑板中 B-Agent 更优的 B-Agent。每个子群体的每一次通讯都接收一个来自黑板的当前全局 B-Agent, 并向黑板发送一个自己的 B-Agent。若传送一个进化个体为 1 个单位的通讯代价, 则每个子群体同时进化 k 代时通讯代价总和为 $2hk/j$ 。由此可见, 同样情况下, 基于 Agent 的并行遗传算法的通讯代价只有经典粗粒度孤岛并行遗传算法的 $1/i$ 。

另外, 一个子群体从上一次访问公共存储区中的 B-Agent 到下一次访问公共存储区中的 B-Agent 之间, 很可能公共存储区中的 B-Agent 被更新了多次, 子群体实际只访问了上一次访问时公共存储区中的 B-Agent 和下一次访问时公共存储区中的 B-Agent, 跳过了中间被更新了多次的公共存储区中的 B-Agent 的访问, 因而避免了一些不必要的通讯代价。因此, 基于 Agent 的并行遗传算法模型的通讯代价比其它粗粒度并行模型要小, 比细粒度并行模型的通讯代价更小。

结束语 在遗传算法中引入 Agent 之后, 可充分发挥遗传算法的群体搜索策略的优势。运用多 Agent 系统智能地实施遗传算法, 能从随机的遗传过程中获取表征进化状态的信息, 智能地调度遗传操作, 较好地克服早熟问题, 在个体空间中有效地实现全局最优搜索。基于多 Agent 的并行遗传算法的最大特点是增强了子群体的自主性, 极大地减少了子群体之间的通讯开销。各子群体的 Agent 之间交换的信息始终是当前整个群体中的最优个体, 一个子群体所引进的最优个体有利于改善其新一代个体的特质, 有利于该子群体加快收敛

(上接第 233 页)

单, 为避免信息量损失, 把信息熵引入处理过程, 满足离散化的标准, 处理过程中每一步都使用最简单高效的处理算法来实现。通过实验也证明该算法对数据量与类别数多的数据有较好的性能及较高的实用价值。

参 考 文 献

[1] Hussain F, Liu H, Tan C L, et al. Discretization: an Enabling Technique[J]. *Data Mining and Knowledge Discovery*, 2002, 6(4):393-423

[2] Dougherty J, Kohavi R, Sahami M. Supervised and Unsupervised Discretization of continuous feature [C] // *Machine learning: Proc. 12th Int'l Conf.* 1995:194-202

[3] 孟庆生. 信息论[M]. 西安: 西安交通大学出版社, 1986: 18-30

[4] 王东锴. 决策表中的知识发现研究[D]. 合肥: 中国科学技术大学, 2002

[5] 王国胤. *Rough 集理论和知识获取*[M]. 西安: 西安交通大学出版社, 2001

[6] Ribeiro M X, Ferreira M R P, et al. Data pre-processing: a new algorithm for feature selection and data discretization[C] // *Conference on Soft computing as Transdisciplinary Science and Technology: Proc. 5th (CSTST' 2008)*. Cergy-Pontoise, France, 2008: 252-257

[7] 高原, 耿国华, 周明全, 等. 一种改进的快速数据离散化算法[J].

速度。石油勘探属性优化的应用实例表明, 它具有更好的性能, 能高效地求解地震勘探数据, 处理最优化问题。

参 考 文 献

[1] 杨光正, 等. 一种薄层砂岩的分类方法[J]. *模式识别与人工智能*, 1994, 7(4): 312-316

[2] Srinivas M, Patnaik L M. Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms[J]. *IEEE Transaction on System, Man and Cybernetics*, 1994, 24(4): 656-667

[3] Hartvigsen G, Johnsan D. Corperation in a distributed artificial intelligence environment the stormcast application[J]. *Engineering Apply of Artificial Intelligence*, 1990, 3(3): 229-237

[4] Onn S, Tennenholtz M. Determination of social laws for multi-Agent mobilization[J]. *Artificial Intelligence*, 1997, 95(1): 155-167

[5] Chaib-draa B, Millot P. A framework for cooperative work: An approach based on the intentionlity[J]. *AI in Engineering*, 1990(4): 199-205

[6] Findler N V. Multiagent coordination and cooperation in a distributed dynamic environment with limited resources[J]. *Artificial Intelligence in Engineering*, 1995, 9: 229-238

[7] Zhang C. Cooperation under uncertainty in distributed expert system[J]. *AI*, 1992, 56: 21-69

[8] 丁明勇, 代春艳, 杨永斌. 一种基于 Multi-Agent 的工程项目智能评标模型[J]. *计算机科学*, 2008, 35(1): 224-226

[9] 梁军, 程毅毅. 基于混合蚁群遗传算法的 Agent 联盟求解[J]. *计算机科学*, 2009, 36(4): 227-231

小型微型计算机系统, 2009, 30(2): 279-282

[8] Nguyen S H. Discretization of Real Value Attributes: A Boolean reasoning approaches[D]. Warsaw: Warsaw University, Poland, 1997

[9] Su Chao-Ton, Hsu Jyh-Hwa. An Extended Chi2 Algorithm for Discretization of Real Value Attributes[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2005, 17(3): 437-441

[10] 苗夺谦, 王珏. 粗糙集理论中概念与运算的信息表示[J]. *软件学报*, 1999, 10(2): 113-116

[11] 王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. *计算机学报*, 2002, 25(7): 1-8

[12] 谢宏, 程浩忠, 牛东晓. 基于信息熵的粗糙集连续属性离散化算法[J]. *计算机学报*, 2005, 28(9): 1570-1574

[13] Hettich S, Bay S D. The UCI KDD Archive [DB/OL]. <http://kdd.ics.uci.edu/>, 1999

[14] Kurgan L A, Cios K J. CAIM Discretization Algorithm[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2004, 16(2): 145-153

[15] 李国正, 王猛, 等. *支持向量机导论*[M]. 北京: 电子工业出版社, 2000

[16] 刘国平. *支持向量机若干问题以及数据挖掘平台的研究*[D]. 上海: 上海交通大学, 2005

[17] 赵静娟, 倪春鹏, 等. 一种高效的连续属性离散化算法[J]. *系统工程与电子技术*, 2009, 31(1): 195-199