基于多 Agent 的汉字签名认证系统的任务分配策略研究*

邵 斌[1]，严智敏[2]，王国钧[1]

(1. 湖州师范学院 信息工程学院，浙江 湖州 313000；2. 湖州师范学院 教育科学与技术学院，浙江 湖州 313000)

摘 要：针对基于多 Agent 的汉字签名认证系统的任务分配策略进行了讨论，并结合遗传算法和模拟退火算法给出了一种新的基于多 Agent 的汉字签名认证系统的任务分配策略。

关键词：签名认证；任务分配；遗传算法；模拟退火算法

中图法分类号：TP391.41 **文献标识码**：A **文章编号**：1001-3695(2004)11-0124-03

Study of Task Assignment on Signature Verification of Chinese Characters Based on Multi-Agent

SHAO Bin[1], YAN Zhi-min[2], WANG Guo-jun[1]

(1. *School of Information Engineering, Huzhou Normal College, Huzhou Zhejiang* 313000, *China*; 2. *School of Education Science & Technology, Huzhou Normal College , Huzhou Zhejiang* 313000, *China*)

Abstract: Discusses the task assignment on signature verification of Chinese characters based on multi-agent, and presents a new algorithm for task assignment on signature verification of Chinese characters based on multi-agent integrating genetic algorithm and simulated annealing.

Key words: Signature Verification; Task Assignment; Genetic Algorithm; Simulated Annealing

1 引言

作为一种身份认证手段，签名在人类社会活动，特别是商务活动是一种普遍采用的形式。随着电子商务技术的发展，电子签名认证技术也越来越引起人们的重视，并成为电子商务技术和安全认证技术中一个重要的前沿性研究课题。

一般电子签名认证与手写文字的机器识别技术密切相关，不同在于目标要求上的差异。文字识别强调不同字类的辨识，而电子签名更加强调书写风格的确认。因此在书写特征的提取方面往往有很大的区别。在我国汉字签名研究中，不像汉字识别过多强调结构特征，而是更加注重形态统计特征以及风格上细微的差异特征。

目前，电子签名认证研究主要分为脱机和联机两大类。纵观已有的电子签名认证研究，可以看出对签名的识别和认证均强调一种特定匹配方法的应用，还缺乏多方法系统综合运用的研究，对于签名细节的辨认也没有引起重视。显然，这对于强调书写风格为目的的签名认证是难以取得好的效果。鉴于此，我们通过引入多 Agent 系统的技术方法，来构建一种多方法相互协调认证的中文签名识别计算方法，以期更好地解决中文电子签名的认证问题。但是在多 Agent 系统中任务的分配是一个 NP 完全问题。假设系统中有 n 个 Agent 和 m 个任务，在 m，n 较小时，可以用遍历搜索获得最佳分配方案，其执行时间为 $O(n^m)$。当 m，n 较大时，这种方法实现起来显然不太现实。所以在这种情况下，人们所寻求的只是满意解或近似解。目前，在这方面已经形成了一些有代表性的算法，如基于图论的分配算法，0～1 程序设计方法、启发式算法以及智能任务分配算法和专家系统方法等。近年来，科学家们对程序设计思想的孜孜探求取得了丰硕的成果，出现了诸如遗传算法和模拟退火算法等一系列令人耳目一新且又具有实用价值的算法。这些新算法的出现对任务分配这一传统问题提供了新的思路，从而出现了基于遗传算法和模拟退火算法来解决这一问题的一些新方法，对基于多 Agent 的汉字签名认证提供了可能。

收稿日期：2003-11-07

基金项目：国家自然科学基金资助项目(60275023)；浙江省自然科学基金资助项目(M603169)；浙江省教育厅资助科研项目(20010140)

2 遗传算法的基本思想[2]

遗传算法是一种模仿自然选择和遗传机制的优化方法。它将问题的每一个可能解看作是群体中的一个个体，并将每个个体编码为字符串的形式，根据预先确定的目标函数对每个个体进行评价，给出一个适应值。在算法开始时，随机选择多个可能解构成初始群体，根据每个个体的适应值利用遗传算子(选择、交配、变异)对群体中的个体进行操作，得到一个新群体。交配和变异有可能产生出更好的个体，重复的选择、交配和变异导致连续的进化，平均而言，群体中的个体表现出越来越好的性能。

遗传算法有三种最基本的遗传算子, 它们分别是:

(1) 交配。将两个个体的相应部分进行交换。交配一般分两步进行: ①对新群体中的个体随机配对; ②对每一对个体在其串长度范围内随机地产生一个或多个整数作为交配点, 按照交配概率对配对的个体在交配点上进行交配运算。

(2) 变异。它以较小的变异概率随机地改变新群体中某些个体的某些位。变异本身是一种随机搜索, 它能恢复计算过程中丢失的某些重要信息, 以保持群体的多样性, 从而保证了遗传算法的有效性。

(3) 选择。它是根据每个个体的适应值将个体复制到新群体的过程。适应值越大, 复制的可能性越大, 其繁殖的子孙就越多。选择算子模仿了生物的自然选择现象。

一般来说, 解决一个具体问题的遗传算法包括下面五个步骤: ①对问题的可能解进行编码; ②创建初始群体; ③确定适应值函数; ④进行遗传操作; ⑤指定控制算法的参数、变量及终止条件。

3　模拟退火原理[3]

模拟退火算法最早见于 IBM 托马斯·J·沃森研究中心的 S. Kirkpatrick 等人的文章, 它从某个初始解出发, 经过大量解的变化后, 可以求得给定控制参数值时组合优化问题的相对最优解, 然后减小控制参数 Temp 的值, 重复执行该算法, 就可以在控制参数 Temp 趋于 0 时, 最终求得组合优化问题的整体最优解。控制参数的值必须缓慢衰减, 才能确保模拟退火算法最终趋于组合优化问题的整体最优解。

为了便于问题的分析, 需要设计一个基于问题配置的目标函数 Cost, 于是对配置的优化过程就转换为对 Cost 的极小化过程。Cost 的极小化过程模拟自然界的退火过程, 由一个逐步冷却的温度 Temp 来控制。在每一个温度值, 将尝试一定的极小步骤。在每一个极小化步骤中, 随机选择一个新的配置并计算 Cost 函数。这时, 如果 Cost 小于以前的 Cost, 则选定新的配置方案; 如果 Cost 大于以前的 Cost, 则计算概率 $p = e^{\frac{\Delta cost}{temp}}$, 然后在(0, 1) 上产生随机数 r, 如果 $r \leq p$, 则选择新的配置方案, 如果 $r > p$, 仍保留原配置方案。也就是说, 模拟退火算法根据法则接收新解, 除接收优化解外, 还在一个限定范围内接收恶化解, 这正是模拟退火算法与局部搜索法的本质区别所在。开始时 Temp 值大, 可能接收较差的恶化解; 随着 s 值的减小, 只能接收较好的恶化解; 最后在 Temp 值趋于 0 时, 就不再接收恶化解, 这就使模拟退火算法既可以从局部优化的"陷阱"中跳出, 又有可能求得组合优化问题的整体最优解。

4　基于遗传算法和模拟退火算法的任务分配算法

4.1　算法说明

设在一个基于多 Agent 的汉字签名认证系统中, 由 n 个 Agent $A = \{a_1, a_2, \cdots, a_n\}$ 组成的一个多 Agent 系统, 采用 k 种汉字签名认证方法 $M = \{m_1, m_2, \cdots, m_k\}$, 一般 $n < k$。为了便于分析问题, 可把此问题进行形式化定义: $\sum = (M, <, Q, C, X)$, 其中 $M = \{m_1, m_2, \cdots, m_k\}$ 是任务的集合; "<"是 M 上的任务优先关系, $m_i < m_j (1 \leq i \leq k$ 且 $1 \leq j \leq k)$ 表示任务 m_i 必须在任务 m_j 执行之前完成; Q 是一个 $k \times n$ 矩阵, 其元素 q_{ij} 表示任务 m_i 在 Agent a_j 上的运行时间(假定每个任务的运行时间预知); C 是一个 $k \times k$ 矩阵, C_{ij} 表示任务 m_i 与 m_j 之间的通信开销; X 是一个 $k \times n$ 的任务分配矩阵: 若 m_i 分配到 a_j 上执行则 $x_{ij} = 1$, 否则 $x_{ij} = 0$。为了实现选择, 还必须设计一个目标函数 Cost。Cost 是一个包含多种因素折中的函数, 它应能体现出设计者对系统的性能要求, 本文的 Cost 为:

$$cost = \sum_{i=1}^{k} \sum_{p=1}^{n} (q_{ip} \cdot x_{ip} + w \cdot \sum_{r=1}^{p-1} \sum_{j=1}^{i-1} C_{pr} \cdot x_{ip} \cdot x_{jr})$$

其中, 常数 w 用来调节通信开销和执行开销之间的差异。在实际设计过程中, 可以根据不同的需要选取 w 的值或者其他的目标函数。

4.2　算法描述及简单分析

4.2.1　初始化

(1) 随机产生一个任务分配矩阵集

先设定一个全 0 的任务分配矩阵 X, 然后在 X 中的每一列由系统随机选定一个元素为 1, 如表 1 所示, 在三个 Agent 和七个任务的情况下, m_1, m_2, m_3 被分配给 a_1; m_4, m_5 被分配给 a_2; m_6, m_7 被分配给 a_3。用同样的方法, 产生多个任务分配矩阵。

表 1　任务分配矩阵

	m_1	m_2	m_3	m_4	m_5	m_6	m_7
A_1	1	1	1	0	0	0	0
A_2	0	0	0	1	1	0	0
A_3	0	0	0	0	0	1	1

(2) 描述和确定初始任务分配方案集

对一个给定的任务分配方案, 可用一个数据结构 TA 来描述: TA = { S, R[1..n] }, 其中 S 是一个 $[\log_2 n] \times k$ 位二进制串, 每 $[\log_2 n]$ 位称为一节, 从左到右第 i 节表示任务 m_i 所在的 Agent 情况, 这样表 1 中的任务分配矩阵可表示为 S = 000000001011010, 因为有三个 Agent, 所以两位为一节, 00, 01 和 10 分别表示任务被分配到 a_1, a_2 和 a_3 上执行, 11 是无效编码。R 是一个 n 元链表数组, R[i] 表示 Agent a_i 上的任务执行顺序, 仍以表 1 为例, 若任务之间满足"<"优先关系 { $< m_1, m_2 >$, $< m_4, m_5 >$, $< m_6, m_7 >$ }, 则可确定三种任务分配方案:

①a_1 上执行顺序为 $m_1 m_2 m_3$; a_2 上为 $m_4 m_5$; a_3 上为 $m_6 m_7$。

简记为 R[1]: 1→2→3; R[2]: 4→5; R[3]: 6→7。

②R[1]: 1→3→2; R[2]: 4→5; R[3]: 6→7。

③R[1]: 3→1→2, R[2]: 4→5; R[3]: 6→7。

由于分配到同一 Agent 上的任务之间有多种排列, 所以从一个任务分配矩阵可产生多种分配方案。在这些方案中, 有一些是违背"<"优先关系的, 称为无效分配方案, 否则, 称为有效分配方案。上面所列出的就是三种有效分配方案。从每一个任务分配矩阵所产生的分配方案中各选取一种或几种有效分配方案, 便形成初始任务分配方案集。

(3) 设置模拟退火算法中的初始温度 $temp_0$ 和收敛率 α

温度 $temp_{i+1} = \alpha\ temp_i$ 逐步降低, 这里 $0 < \alpha < 1$, 下标 i 既表示第 i 次迭代, 同时也指称遗传算法中的第 i 代个体。在任务数较多的情况下, 选取 $temp_0 = 1000$, $temp_\infty = 1$, $\alpha = 0.9$, 可取得较好结果。

4.2.2　循环

直到 temp = 1 或者连续多代未产生更好的分配方案为止。在

循环过程中若落入局部最优的"陷阱"时要采用静态爬山方法。

(1) 交配。从整个任务分配方案集中以一定的百分比 P_C 随机选择出一个供交叉的子集,利用它进行交叉繁殖。具体实现方式是对 TA.S 中的某些节进行重组,仍以三个 Agent 和七个任务为例,设有两个父代分配方案 TA1, TA2,其中 TA1.S = 00000001011010, TA1.R[1]:1→2→3, TA1.R[2]:4→5, TA1.R[3]:6→7; TA2.S = 00011000100001, TA2.R[1]:1→4→6, TA2.R[2]:2→7, TA2.R[3]:3→5。在 TA1.S 中随机选定几节替换到 TA2.S 中就形成了一个新的二进制串 S',假如选定 TA1.S 中的第 1、第 4、第 5 节(从左到右),替换后 S' = 00011001010001,同时,为保持数组 R 的一致性,从 TA1.R 作相应调整,调整有几种方案,从中选出一种或几种有效分配方案,就得到下一代的一组分配方案。

(2) 变异。除了交叉之外,还应对任务分配方案集以某个较小的比例 P_k 选出一个子集进行变异。变异的方式有多种,可以是对 TA.S 中的某些节求反,也可以是互换一方案中几个 Agent 所处理的任务,还可以是其他方式。由于变异不会造成个体数量变化,所以每次变异后对一种原方案只保留一种有效变异方案。

(3) 复制。对于没有进行交叉和变异的方案,直接加入到新的任务分配方案集中。

(4) 选择。交叉和变异产生的方案称为新方案。计算新方案的目标函数 cost',并令 Δcost = cost - cost',若 Δcost < 0,表明新方案优于原方案,将新方案加入到新任务分配方案集中;若 Δcost≥0 表明原方案优于新方案,此时计算概率值 $p = e^{\frac{\Delta cost}{temp}}$,temp 是当前温度。由系统产生一个(0,1)上的随机数 r,若 p≤r,则放弃新方案,若 p > r,则将新方案加入到新任务分配方案集中。我们可以对 p 算式作一个简单分析:在一定温度进行方案选择时,temp 也可看作常数,所以此时 p 只与 Δcost 有关,而且随 Δcost 递增而递减。Δcost 越小,表明新方案虽次于原方案,但接近原方案,因而是较"好"的方案,而此时 p 也就越大,p > r 成立的可能性也越大,越容易被接受;反之同理。

可以看出,选择过程选取了两类新方案:①优于原方案的新方案;②次于原方案但目标函数比较接近原方案的新方案。从而使新方案集的整体水平优于原方案集。

5 结束语

本文综合了遗传算法和模拟退火算法,提出在基于多 Agent 的汉字签名认证系统的任务分配新算法。由于遗传算法和模拟退火算法的独特性,使得本文所给出的任务分配算法有区别于传统方法的新特点。但是目标函数的选择、初始任务分配方案集选多大为宜是值得进一步研究的问题。

参考文献:

[1] erhard Weiss. Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence[M]. MIT Prus, 1999.

[2] H Holland. Adaptation in Natural and Artificial System[M]. Michigan: University of Michigan Press, Ann Ardor, 1975.

[3] Kirkpatrick, C D Gellatt, Jr M P Vecchi. Optimization by Simulated Annealing[J]. Science, 1983, (5).

[4] Fitoussi, M Fermenhalts. Choosing Social Laws for Multi-Agent Systems, Minimulity and Simplicity[J]. Artificial Intelligence, 2000, 119(1-2):61-101.

[5] 邵斌, 王国钧. 多 Agent 系统的性能评价[J]. 微电子学与计算机, 2003, 20(8):80-81.

[6] 周昌乐. 手写汉字的机器识别[M]. 北京: 科学出版社, 1997.

作者简介:

邵斌(1971-),男,浙江湖州人,系副主任,讲师,在读硕士,研究方向为人工智能、中文信息处理;严智敏(1971-),女,浙江湖州人,讲师,学士,研究方向为形式逻辑、算法设计;王国钧(1946-),男,浙江湖州人,副教授,研究方向为算法设计与分析、图像处理。

(上接第 61 页)为例,但其思想方法可用于其他优化问题。

然而,蚁群算法是一种新的模拟进化算法,其研究才开始不久,还没有形成系统分析的方法和坚实的数学基础,各种实验参数的确定也没有理论指导。目前国际上的诸多研究成果还都是基于实验分析,因此,还有许多理论问题有待进一步研究。但可以推断,随着研究的深入,蚁群算法也将与其他模拟进化算法一样,获得越来越多的应用。

参考文献:

[1] Colorni, M Dorigo, V Maniezzo. Distributed Optimization by Ant Colonies[C]. Proceedings of ECAL91 European Conference of Artificial Life, Paris, France, Elsevier Publishing, 1991. 134-144.

[2] Dortgo, Vittorio Maniezzo, Alberto Colorni. Ant System: Optimization by Ant Colonies Cooperating Agents[J]. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, 1996, 26(1):29-41.

[3] Dortgo, L M Gambardella. Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem[J]. IEEE Trans. Evolutionary Computation, 1997, 1(l):53-66.

[4] oshyar R, Jamali S H, Locus C. Ant Colony Algorithm for Finding Good Interleaving Pattern in Turbo Codes[J]. IEEE Proceedings Communications, 2000, 147(5):257-262.

[5] oung-Jae Jeon, Jae-Chul Kim. Application of Ant Colony Algorithm for Loss Minimization in Distribution Systems[J]. Transactions of the Korean Institute of Electrical Engineers, 2001, 50(41):88-96.

[6] ianni Di Caro, Marco Dorigo. Ant Net: Distributed Stigmergetic Control for Communications Networks[J]. Journal of Artifcial Intelligence Research, 1998, (9):317-355.

[7] origo M, Di Caro G. Ant Colony Optimization: A New Meta-heuristic[C]. Proceedings of the 1999 Congress on Evolutionary Computation, Washington, DC, USA, 1999. 1477.

[8] ambardella L M, Dorgio M. An Ant Colony System Hybridized with a New Local Search for the Sequential Ordering Problem[J]. INFORMS Journal on Computing, 2000, 12(3):55-237.

[9] Meuleau, M Dortgo. Ant Colony Optimization and Stochastic Gradient Descent[C]. Artif. Life, 2002, 8(2):103-121.

[10] uest Editorial. Special Section on Ant Colony Optimization[J]. IEEE Transactions on Evolutionary Computation, 2002, 6(4).

[11] Stutzle, H H Hoos. MAX-MIN Ant System[J]. Future Gener. Comput. Syst., 2001, 16(8):889-914.

[12] eung Gwan Lee, Tae Ung Jung, Tae Choong Chung. An Effective Dynamic Weighted Rule for Ant Colony System Optimization[C]. Proceedings of the 2001 Congress on Evolutionary Computation, NJ, USA, IEEE Press, 2001. 393-397.

[13] heng-Fa Tsai, et al. A New Approach for Solving Large Traveling Salesman Problem Using Evolution ant Rules[C]. Neural Networks, IJCNN '02, Proceedings of the 2002 International Joint Conference, Honolulu, HI, USA, IEEE Press, 2002. 1540-1545.

作者简介:

朱海梅(1973-),女,江苏扬州人,硕士研究生,主要研究方向为智能控制;朱庆保(1955-),男,江苏南京人,教授,硕士生导师,主要研究方向为智能控制。