文章编号:1006-9348(2005)01-0100-03

自适应蚁群算法在序列比对中的应用

梁 栋,霍红卫
(西安电子科技大学计算机学院,陕西 西安 710071)

摘要:序列比对是生物信息学的重要研究工具。蚁群算法是一种新型的模拟进化算法,并被成功地应用于旅行商问题(TSP)等组合优化问题中。该文将蚁群算法应用于序列比对,并提出基于自适应调整信息素的改进算法。仿真结果表明这种新的比对算法是有效的,而它的改进算法的效果更为理想。

关键词:蚁群算法;序列比对;信息素
中图分类号:TP301.6 **文献标识码**:A

An Adaptive Ant Colony Optimization Algorithm and Its Application to Sequence Alignment

LIANG Dong, HUO Hong-wei
(School of Computer Science, Xidian University, Xi'an Shanxi 710071, China)

ABSTRACT: Sequence alignment is one of the most important tools in bioinformatics research. Ant colony optimization is a novel simulated evolutionary algorithm successfully used in combinatorial optimization problems such as TSP. In this paper, ACO is applied in sequence alignment and improved by adaptive adjusting the pheromone. The simulation results demonstrate that the new approach is efficient and the improved algorithm is more efficient.

KEYWORDS: Ant colony optimization; Sequence alignment; Pheromone

1 引言

序列比对(Sequence Alignment)是一种重要的生物信息处理技术,通过比对中获得的序列信息可以推断基因的结构、功能和进化关系,是生物信息学的重要研究工具。

蚁群算法(ACO)是一种新型的模拟进化算法,通过由候选解组成的群体在不断进化的过程中寻求最优解[2]。它是在对自然界蚁群的寻径方式进行研究和模拟的基础上,由Dorigo M.等人首先提出的[1]。目前该算法在旅行商问题(TSP)等组合优化问题中有较多的研究[2]。

本文将蚁群算法应用于序列比对,并在此基础上,针对蚁群算法易于陷入局部最优解的特点,提出动态自适应调整信息素的改进算法,以求扩大搜索空间,提高收敛速度。仿真实验证明这种新的比对算法是有效的,而其改进算法的效果更为理想。

2 序列比对问题描述

双序列比对是多序列比对和序列数据库搜索的基础。Needleman 和 Wunsch 提出的比对方法属于动态规划范畴[3]。

对于一个序列 S,$|S|$ 是 S 中的字符个数,$S[i]$ 表示序列的第 i 个字符。$S[1..i]$ 表示序列的前 i 个字符组成的子序列。S 中的字符由某个有限字符集合 Ω 确定(如 *DNA* 由4种核糖核酸 A,T,C,G 确定)。基因序列在突变中的变化包括替代、插入和删除,我们用"-"来表示插入和删除所产生的空位。对于 $x,y\in\Omega\cup\{-\}$,定义 $\sigma(x,y)$ 为计分函数,表示 x,y 比较时的得分,以下是最简单的一种:

$$\sigma(x,y)=\begin{cases}2 & x=y\in\Omega\\ -1 & x\neq y\in\Omega\\ -1 & x='-'\ or\ y='-'\end{cases}\qquad(1)$$

S 和 T 的一个比对 A 用序列和中字符的一一对应表示,其中①$|S'|=|T'|$,②$|S'|$,$|T'|$ 去掉空格就是 S 和 T。A 的得分为:

$$Score(A)=\sum_{i=1}^{|S'|}\sigma(S'[i],T'[j])\qquad(2)$$

取得最大值的比对就是最优比对。

3 蚁群算法应用于序列比对

3.1 蚁群算法的原理

人工蚁群算法是受到真实蚁群觅食行为的启发提出的。

收稿日期:2003-10-14

蚂蚁这种群居昆虫，虽然单个个体的行为十分简单，但是由其组成的群体却能完成较为复杂的行动，比如可以找到巢穴到食物源的最短路径。仿生学家经过研究发现，每只蚂蚁觅食时在走过的路线上会留下一种称为信息素(pheromone)的物质，蚂蚁之间就是靠感知这种物质的浓度进行信息传递的。蚂蚁倾向于朝浓度高的方向移动，而且找到食物后会顺原路返回。这样就形成了一种正反馈机制：由于外激素随着时间而挥发，距离短的路径因蚂蚁来回用时少，信息素浓度将高于距离长的路径，后来的蚂蚁选择距离短的路径，然后又留下新的信息素，随着时间的推移，距离短的路径上的信息素越来越多，选择它的蚂蚁也逐渐增多，最后整个蚁群聚集到最短路径上。

人工蚁群算法模拟了这一过程。每只蚂蚁在解空间独立地搜索可行解，解越好留下的信息素越多，随着算法推进，较优解路径上的信息素增多，选择它的蚂蚁也随之增多，最终收敛到最优或近似最优的解上[2]。

3.2 蚁群比对算法的设计

对序列 S = CAGGA 和 T = CGGTTA，仿照动态规划法那样阵建立矩阵(如图 1)。

蚂蚁从矩阵左上角出发选择一条路线到达右下角，就形成一个比对，我们规定在水平或垂直方向上移动一格，表示在相应的序列中插入一个空位，沿对角线移动一格表示到达的新位置对应字符的匹配。图 1 中的路线表示了如下比对结果：

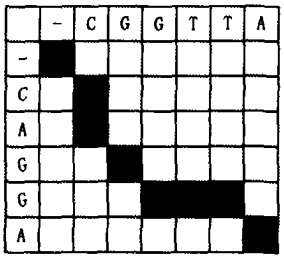

图 1 单个蚂蚁的行走路线

序列 S′：CAGG - - A

序列 T′：C - GGTTA

根据式(1)和(2)得分为 5。在找到的所有路线中得分最高的比对就是最优比对。

与 TSP 问题不同，蚂蚁在每个位置选择移动方向的数目是固定的，总是向右，向下，沿对角线向右下三个方向，序列比对对加入的空位数量也有一定要求。所以蚁群比对算法的设计与 TSP 问题有所不同。

我们用 $\tau_{ijk}(t)$ 表示 t 时刻图 1 中 (i,j) 位置上第 k 个方向的路径 R_{ijk} 上的信息素浓度。其中 $i = 0,1,2\ldots|S|, j = 0,1,2\ldots|T|$，而 $k = 0,1,2$ 分别表示向右，向下，即向右下方向。初始时刻，设定 $\tau_{ijk}(0) = \tau_0$(τ_0 为常数)。

蚂蚁 $z(z = 1,2\ldots m)$ 从矩阵左上角出发，每走一步，根据当前位置上各条路径上的信息素浓度决定向某一路径转移。为 (i,j) 位置上蚂蚁 z 选择第 k 个方向的路径定义一个得分：

$$SCORE^z_{ijk} = \tau_{ijk}(t)^\alpha M^\beta d^\gamma \tag{3}$$

$\tau_{ijk}(t)$ 的意义如前面所述。M 为启发信息，可以根据式(1)来确定 M 的值。当 $k = 2$ 时，即向右下方移动，所到达的新位置上，序列 S,T 对应位置上的字符如果相同，M 的值取匹配得分，式(1)中为 2，不相同则 M 值取不匹配罚分的绝对值的倒数，为 1。$k = 0$ 或 1 时，即向右或向下移动，必然在序列中产生空位，M 的值取空位罚分的绝对值的倒数，为 1。当然计分方式不同，M 的取值方法会不同。

为防止空位过多，应尽量使蚂蚁大致沿着对角线方向行走。我们采用 d 来调整蚂蚁的选择概率，如果蚂蚁的位置为 (i,j)，当 i 和 j 之差的绝对值在某个正数 H 以内，三个方向的 d 值相同，这样三个方向等可能被选择，超过 H 时，如果 $i > j$，我们分配给向右和右下方向更大的 d 值，增大蚂蚁选择这两个方向的可能性，$i < j$ 的情况类同。H 的大小对连续插入空位的数目起到了限制作用。

α,β,γ 是分配给信息素，启发信息和 d 的权值，体现了它们对决策的影响力大小，在实验中可以进行合理地调整。

我们采用了这样的选择策略：首先设定 $q_0 \in (0,1)$，蚂蚁在选择路径时产生一个(0,1)之间的随机数 p，当 $p \leqslant q_0$ 时，蚂蚁 z 选择 $SCORE^z_{ijk}(k = 0,1,2)$ 之中分值最大的方向 k。当 $p > q_0$ 时，蚂蚁 z 按照式(4)决定三个方向被选择的概率。

$$p^z_{ijk}(t) = \frac{SCORE^z_{ijk}}{\sum_{k=0}^{2} SCORE^z_{ijk}} \tag{4}$$

式(4)采用轮盘赌的方式实现。如果 $q_0 = 0.3$，则蚂蚁以 0.3 的概率选择分值最大的方向，以 0.7 的概率按式(4)决策。q_0 选得小，有利于增加解的搜索空间，但不容易收敛。q_0 选得大则反之[4]。

经过一代进化，当所有的蚂蚁通过不同的路线到达矩阵右下角，得到一组比对结果，就完成了寻找最优路线的一次循环。这时要对每条路径的信息素进行全局更新，新的信息素要加进来，旧的信息素要挥发。定义 $q_g \in (0,1)$ 为衰减系数，则 $1 - q_g$ 反映了信息素的挥发程度。更新公式如下：

$$\tau_{ijk}(t+1) = q_g \times \tau_{ijk}(t) + \Delta\tau_{ijk} \tag{5}$$

$$\Delta\tau_{ijk} = \sum_{Z=1}^{m} \Delta\tau^Z_{ijk} \tag{6}$$

$\Delta\tau_{ijk}$ 为本次循环中路径 R_{ijk} 中的信息素增量。初始时刻 $\Delta\tau_{ijk} = 0$。$\Delta\tau^Z_{ijk}$ 为第 z 只蚂蚁在路径 R_{ijk} 上留下的信息素。

$$\Delta\tau^Z_{ijk} = \begin{cases} Q \times A_z (\text{蚂蚁 } z \text{ 经过 } R_{ijk}) \\ 0 (\text{蚂蚁 } z \text{ 未经过 } R_{ijk}) \end{cases} \tag{7}$$

Q 为常数，A_z 与蚂蚁 z 所走路线代表的序列比对的分值相关。在 TSP 问题中，用 L_k 表示路线的长度，它总是大于零的。但序列比对分值可正可负，所以要将其映射到正值再分配给 A_z，比对分值越高，A_z 越大。

由于蚂蚁选择不同的路线，经过的格子数可能不相等，即不是同时到达矩阵右下角。为每只蚂蚁增加一个初始值为 0 标志，当到达右下角时置为 1。当所有的蚂蚁都到达右下角，完成信息素全局更新后，所有的蚂蚁被重置到矩阵左上角，置标志为 0，开始下一轮循环。算法在达到最大进化代数或最

优解连续 10 代没有变化时终止。

为了加快搜索，记录每代取得的最好的解，和历代最好的解比较获得最优解。

4 自适应调整信息素的改进算法

蚁群算法通过正反馈机制来强化较好的解，但容易出现停滞，陷入局部最优解[5]。针对这个问题，提出自适应调整信息素的方法，根据解的搜索情况，动态地调整信息素的分配。

采用式(8)的时变函数 $Q(t)$ 来代替式(7)中的常数 Q。进化初期为了增大搜索空间，$Q(t)$ 取较小的值，随着算法的推进取值逐步增大，强化较好的解。在算法的仿真中，我们采用 $Q1 = 0.0001$，$Q2 = 0.0005$，$Q3 = 0.001$ 以及 $T1 = 30$，$T1 = 60$。

$$Q(t) = \begin{cases} Q1(0 < t \leqslant T1) \\ Q2(T1 < t \leqslant T2) \\ Q3(T2 < t) \end{cases} \tag{8}$$

在陷入局部最优解时，某条路径上的信息素在数量上占绝对优势[5]，因此我们对信息素的最大值和最小值进行了限制，如规定 $\tau_{ijk}t)_{\min} = 0.05$，$\tau_{ijk}(t)_{\max} = 30.0$。限制最大值可以防止某条路线的信息素浓度过大，限制最小值可以防止搜索后期没走过的路径信息素浓度过低，使较差的路线被强化。

为了鼓励解质量的改善，又不减小搜索空间，在进化一定代数以后，采用式(9)根据解的情况动态地调整信息素的分配。若路线 R_{ijk} 上取得的解(即比对得分)为 $Score$，较目前得到的最优解 $Score_{\max}$ 有所改善，则增大路线 R_{ijk} 上的信息素增量的分配，并更新 $Score_{\max}$ 的值，若低于最优解，则减小信息素增量的分配。

$$\Delta\tau_{ijk} = \frac{Score}{Score_{\max}} \times \Delta\tau_{ijk} \tag{9}$$

如果最优解在几代内没有改善，则可以适当减小要添加的信息素，以求摆脱局部最优解。

5 仿真结果

采用 $NC = 100$(最大进化代数)，$m = 10$(蚂蚁总数)，$\tau_0 = 20.0$，$\alpha = 4.0$，$\beta = 3.0$，$\gamma = 2.0$，$q_0 = 0.3$，$q_g = 0.8$ 作为蚁群算法的参数进行仿真实验。

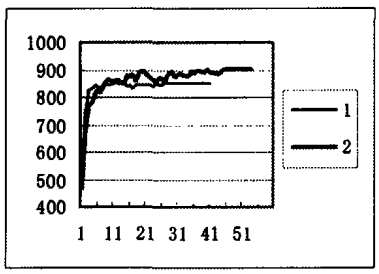

图 2 两种算法的进化曲线

选择平均长度为 520 的两条 *DNA* 序列进行比对，采用式(1)作为计分函数，最优解事先通过动态规划算法求得为 902。图 2 显示了基本蚁群算法和自适应蚁群算法的进化过程。横坐标轴表示进化的代数，纵坐标轴表示某一代取得的最优解。曲线 1 和 2 分别代表基本算法和自适应算法的进化曲线。

从图 2 中看出基本算法在第 31 代达到 854，之后就陷入了局部解 850 中。自适应算法增大了解空间，开始阶段解质量低于基本算法，但搜索到更优秀解的概率较大，随着算法的推进，在第 45 代收敛到最优解 902，连续 10 代最优解不变后算法终止。

表 1，表 2 显示了在不同长度的几组序列上，基本蚁群算法和自适应改进算法的比对结果。每组序列进行 20 次比对。从表中可以看出改进算法的寻优能力明显优于基本算法，在长序列的比对中更为突出。

表 1 基本蚁群算法比对结果

编号	序列平均长度	最好结果	最坏结果	平均值	已知最优解
1	126	234	230	232.40	234
3	430	768	712	736.65	768
5	1150	2070	1864	1936.75	2075

表 2 自适应蚁群算法比对结果

编号	序列平均长度	最好结果	最坏结果	平均值	已知最优解
1	126	234	230	233.35	234
2	430	768	755	764.00	768
4	1150	2075	2005	2053.95	2075

表 3 显示了蚁群算法对氨基酸序列的比对效果。我们选用“皮质铁氧还蛋白氧化还原酶”编号为 P08165 和 Q61578，平均长度为 492 的两条序列，共进行 20 次比对。计分方法采用 PAM250 替换矩阵和仿射空位罚分，“空位设置罚分”为 - 10，“空位扩展罚分”为 - 0.5，匹配得分为 5，不匹配罚分为 - 4，与“欧洲生物信息研究所(EBI)”提供的双序列比对程序“EMBOSS - Align”一致。最优解也通过该程序计算求得。

表 3 氨基酸序列比对结果

采用算法	最好结果	最坏结果	平均值	已知最优解
基本算法	2194.5	2081	2142.95	2221.5
自适应算法	2221.5	2181	2196.55	2221.5

从平均值看出自适应算法的结果更接近最优解。PAM250 替换矩阵计分方法比较复杂，除了精确匹配外还要计算相似得分，因此结果偶然性大，解的精确性有所下降。

6 结论

仿真实验表明，蚁群算法在序列比对中的应用取得了较好的效果，而动态自适应调整信息素的改进算法可以明显地提高蚁群算法的寻优能力。相信经过进一步研究，蚁群算法在序列比对中的应用前景会更广阔。

(下转第 106 页)

2	0.95	0.58	29	不收敛***		
3	0.6	0.48	30	不收敛***		
4	0.85	0.5	40	收敛*	8	
5	0.88	0.48	45	收敛**	10	
6	0.9	0.38	90	收敛**	5	
7	0.96	0.8	120	不收敛*		μ、α过大

说明:'*'代表化时3小时以下;'**'代表化时3至7小时;'***'代表化时12小时以上

策略2:

初始染色体随机生成4条7位十进制串;

迭代时间:45分钟;

优化后的BP网络参数 $\mu=0.91$ $\alpha=0.43$ $n=35$;

随后的预测准确次数37;

参考文献:

[1] W Schiffmann, M Joost, R Werner. Optimization of the Backpropagation Algorithm for Trining Multilayer Perceptrons Technical Report[R]. University of Koblenz, Institute of Physics, 1993.

[2] Hecht - Nielsen R. Theory of Backpropagation Neural Networks[C]. Proc. IJCNN - 89. 1989. 1 - 593.

[3] R P Lippmann. Pattern Classification using Neural Networks[J]. IEEE Comm, Magazine, Nov. 1989, 47 - 64.

[4] S Y Kung, J N Hwang. An Algebraic Projection Analysis for Optional Hidden Units Size and Learning Rates in Back - Propagation Learning [J]. IEEE, 1991, Ⅰ:363 - 370.

[5] S Y Kung, Hu Yu Hen. A Frobenius Approximation Reduction(FARM) for Determining Optimal Number of Hidden Units[J]. IEEE, 1991, Ⅱ: 163 - 168.

[6] M T Hagan, M B Mangis. Training Feed Forward Networks with Marquart Alogrithm[J]. IEEE Trans. on Neural Networks, 1994, 5(6):989 - 993.

[7] T P Vogl, J K Mangis, ect. Accelerated the Convergence of the Back Propagation Method[J]. Bio. Cybern, 1988, 59(9):256 - 264.

[8] 蔡国正,屈梁生.共轭梯度神经网络的研究[J].西安交通大学学报,1995,29(8):73 - 76.

[9] 叶东毅.BP网络的一个非单调学习算法[J].模式识别与人工智能,1997,10(3):221 - 225.

[10] 杨秋贵,张杰,张熹贞.基于拟牛顿法的前向神经元网络学习算法[J].控制与决策,1997,12(4):357 - 360.

[11] 袁曾任.人工神经网络及其应用[M].北京:清华大学出版社,2000

[12] J H Holland. Adapation in Natural and Artifical Systems. Ann Arbor, MI: University Michigan Press, 1975.

[13] 杨启文,等.遗传算法优化速度的改进[J].软件学报,2001,12(02):270 ~ 275.

[14] 唐飞,滕弘飞.十进制整数编码遗传算法的模式定理[J].计算机科学,1999,12(6):54 - 56.

[15] 潘正君,等.演化计算[M].北京,清华大学出版社,1999.

[作者简介]

闫河(1972 - 1),男(汉族),陕西勉县人,信息获取与处理专业在读博士生,讲师,主要研究方向:管理信息系统、人工智能系统、图象处理技术、模式识别;

成卫(1967 - 7),男(汉族),重庆市人,硕士,副教授,主要研究方向:数据库技术、人工智能系统、虚拟现实技术;

潘英俊(1960 - 9),男(汉族),重庆市人,教授、博导,主要研究方向:光电测控与传感技术、智能机器人触觉传感技术、信息光学理论与应用、微波治疗技术及仪器研制等;

何光敏(1973 - 6),女(汉族),四川平昌县人,本科,高级教师,主要研究方向:多媒体技术。

(上接第102页)

参考文献:

[1] M Dorigo, V Maniezzo and A Colorni. The Ant System: Optimization by a colony of cooperating agents[J]. IEEE Transactions on Systems, Man, and Cybernetics - part B, 1996, 26(1):1 - 13.

[2] 王颖,谢剑英.一种自适应蚁群算法及其仿真研究[J].系统仿真学报,2002 - 1,14,(1):31 - 33.

[3] S B Needleman and C D Wunsch. A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins[J]. J. Mol. Biol., 1970,48:443 - 453.

[4] 毕军,付梦印,张宇河.一种改进的蚁群算法求解最短路径问题[J].计算机工程与应用,2003,(3):107 - 109.

[5] 覃刚力,杨家本.自适应调整信息素的蚁群算法[J].信息与控制,2002 - 6,31(3):198 - 201.

[作者简介]

梁栋(1974 -),男(汉族),河北省行唐县人,硕士研究生,主要研究方向为算法设计、生物信息学等;

霍红卫(1963 -),女(汉族),陕西人,教授,博士,主要研究方向为分布与并行计算、算法设计、生物信息学等。