基于免疫遗传算法的关系型数据库查询优化技术

The relation database query optimization technique based on immune genetic algorithm

王 力 王成良 （重庆大学 软件学院 重庆 400044）

摘 要： 本文引入结合免疫系统原理和遗传算子自适应调整的算法，即免疫遗传算法。该算法具有可防止未成熟收敛和保证种群的多样性等优点。在使用此算法搜索最优解时，可防止陷入局部寻优情况的出现。经过实验计算，免疫遗传算法对多连接查询优化有很好的效果，优化后的查询代价较遗传算法有很大的降低。

关键词： 查询优化 免疫遗传算法 遗传算法 查询优化器

1 引言

用户向数据库管理系统（DBMS）提出的查询请求（Q），查询优化器[1]（Query Optimizer）的目标就是选择最有效的查询执行计划（Query Execution Plan 简称 QEP），以存取相关数据和回答查询。假定 S 是适合查询（Q）的所有策略的集合，S 中的每一个元素 s 都有一个相关联的代价 C(s)，查询优化算法的目标是找到 $s_o \in S$，满足：

$$C(s_o) = \min C(s) \quad s \in S$$

实际上，查询优化的最终目的就是提高查询效率，缩短查询请求的响应时间，花费大量的时间在优化上以找到一个最佳的 QEP，使得优化时间与最佳的 QEP 的执行时间之和非常大，也就是响应时间非常长的做法是并不明智的。因此，在 QEP 空间很大的情况下，优化目标并不是找到一个最佳的查询执行计划，而是找到一个相对较好的查询执行计划，使得“优化时间 + 找到的 QEP 的执行时间”最小即可。但是，随着 Q 中包含的关系数的增加和查询执行引擎提供的物理操作的增加，QEP 空间变得很大，查询优化器面对巨大的搜索空间，必须使用一定的搜索算法，来搜索所谓的“最佳”QEP。由于多连接查询优化问题实质上就是组合优化问题[2]，因此搜索算法可以借鉴最优化算法来完成。

国内的韩梅[1]和任美睿[3]以及国外的 Bennett[4]和 Anderson[5]等人曾利用遗传算法求解数据库查询优化问题，取得较好成果。但是遗传算法仍旧存在难以克服的问题。例如：遗传算法在初始时很快向最优值逼近，但是在最优化附近收敛较慢，而对于多峰函数的优化问题，它往往会出现“早熟”，即收敛于局部极值局，导致解的质量不够高。其原因主要有两个方面，一是遗传算法采用固定的交叉概率和变异概率，这样容易造成适应度高的个体过度早熟，使遗传算法陷入局部最优；二是如果遗传算法陷入局部最优，则无法通过其它方法有效地跳过局部最优点，向全局最优收敛。因此本文引入免疫遗传算法，来更好的解决搜索空间的最优化搜索问题。

2 免疫遗传算法

2.1 免疫遗传算法概述

免疫遗传算法[6]（Immune Genetic Algorithm 简称 IGA）是基于免疫算法（Immune Algorithm 简称 IA）和遗传算法（Genetic Algorithm 简称 GA）的优化算法，在免疫系统的抗体多样性的维持机制中引入遗传算法，使 IGA 的性能比标准 GA 更进了一步。这样在工程优

化设计中，既利用免疫算法可以有效地使用多种机制求解多目标函数最优解的复杂自适应特性，又保留了遗传算法的搜索特性，克服遗传算法在局部搜索效率较差的缺点，又在很大程度上避免未成熟收敛，改善了算法的收敛性，尤其是在复杂约束领域内。对于工程优化设计中遇到的问题，不仅在局部层次提供了相当出色的自适应处理模型，而且在全局层次也具有许多有用的性能，是解决优化设计的一种智能方法。

2.2 免疫遗传算法的工作流程

免疫遗传算法将待求解的工程优化设计问题作为抗原(Antigen)，将问题的解作为抗体(Antibody)。通过抗原和抗体的亲和力(Affinity)、自身抗体浓度以及遗传算法对抗体的复制、交叉和变异的计算以求得目标函数的最优解。

免疫遗传算法计算过程如下：

(1) 输入抗原及设定参数。输入目标函数及约束条件，作为抗原，设定种群规模、交叉率、变异率；

(2) 产生初始抗体识别抗原。根据应答的情况，初次应答则初始抗体全部随机产生；再次应答则初始抗体部分随机产生、部分通过激活免疫记忆细胞，在包含最优抗体的免疫记忆细胞库中产生；

(3) 产生初始抗体(可行解)，在优化问题的设计约束控制下，随机产生 n 组初始设计向量，作为免疫系统的初始抗体种群 N；

(4) 计算抗体 v 和抗原的亲和力，抗体 v 和 w 之间的亲和力，并对亲和力排序，计算抗体浓度，并对浓度排序；

(5) 根据抗体、抗原亲和力计算结果，选择亲和力高于阀值 Tc 的抗体进入下一轮迭代，并使浓度高的部分抗体淘汰；

(6) 群体更新。按照基于浓度调节的适应度函数方法，根据预先设定的淘汰率，选择生存期望值更高的个体进入下一代. 在此基础上，按照交叉率和变异率对选择的抗体进行交叉、变异；

(7) 判别终止条件，若满足，计算停止，否则返回第 3 步；

(8) 取抗体种群中与抗原亲和力最大的抗体，作为优化设计的最优解。

其工作流程图如图 1 所示。

2.3 IGA 与 GA 比较

IGA 与 GA 相比，具有如下显著特点[7]：

(1) 产生多样性抗体的能力。通过细胞的分化，免疫系统可产生大量的不同抗体来抵御各种抗原。利用该特征可维持进化过程中个体的多样性，提高遗传算法全局搜索能力，避免陷入局部最优解；

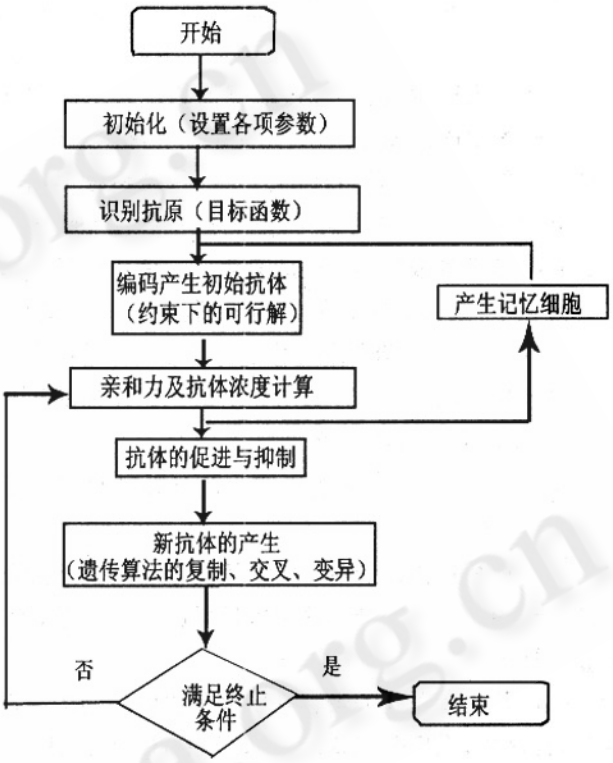

图 1　免疫遗传算法的工作流程图

(2) 自我调节机构。免疫系统的平衡机制通过对抗体的促进和抑制作用，能自我调节产生适当数量的必要抗体。这一功能对应于遗传算法中个体浓度的抑制和促进，可提高遗传算法的局部搜索能力；

(3) 免疫记忆功能。产生抗体的部分细胞会作为记忆细胞而被保存下来，对于今后侵入的同类抗原，相应的记忆细胞会迅速激发而产生大量的抗体。在遗传算法中利用这种抗原记忆识别功能，可加快搜索速度，提高遗传算法的总体搜索能力。

3 基于免疫遗传算法的数据库多连接查询优化问题

3.1 查询优化步骤

本文主要研究的问题定义为基于左线性空间的数

据库的多连接查询优化，目标函数作为抗原，抗体对应于优化问题解，其步骤为：

(1) 编码并形成初始抗体群。在对多连接查询优化问题进行编码时一定要依据关系之间的位置及个数，这样有利于在遗传算子、免疫算子和初始化过程中采用启发式信息以提高搜索效率，其次是使免疫遗传算法的搜索空间尽可能小且可能包含足够多的近似最优解。

由于在进行查询优化时，最重要的是关系的顺序而不是关系之间的相邻关系，因此采用有序串编码(浮点数)形式是一个比较合适的方法。首先将查询所包含的 n 个关系从 1 开始编号，然后按左线性 join 树从左到右叶结点的编号顺序构成染色体。使用这种方法的好处在于：该编码方式的结构清晰，便于进行交叉、变异等操作。每个 QEP 由几位有序串组成，是在实验时由多连接查询的关系个数决定。编码时还需随机产生 M 组抗体 $Yi = [P'_1, P'_2, \cdots, P'_N]$, $i = 1, 2, \cdots, M$. 并用试凑法[8]使随机产生的抗体的分量满足 $P_{Imin} \leqslant Pi \leqslant P_{Imax}$。

(2) 计算抗体适应度。抗体和抗原之间的适应度可由目标函数变换得到。这里取目标函数的倒数，即：$A_{bgw} = 1/T_o$。

抗体和抗体之间的亲和度反映了抗体之间的相似程度，在这里，采用公式 $B_{w,v} = 1/(1 + H_{w,v})$ 计算。其中，$H_{w,v}$为抗体 w 和抗体 v 的 Euclidean 距离。

(3) 基于抗体浓度的群体更新。首先在群体中再随机产生 N 个抗体，使抗体总数为 M + N。然后计算抗体浓度和适应度。根据事先确定的抗体间 Euclidean 距离的极限值，判断出抗体间的相似度，将相似度高的抗体看作同一种抗体，则抗体 w 的抗体浓度 Cw 的计算式如下：

$$C_w = \sum_{i=1}^{M+N} ai/(M+N)$$

式中：当 a_i 大于为抗体 w 和抗体 v 的亲和度阀值时取 1，否则为 0；M + N 为抗体总数。

最后进行群体更新。首先进行生存选择。抗体 w 的生存期望值为 $E_w = A_{bgw}/C_w$，显然与抗原适应度大的抗体以及浓度低的抗体成活到下一代的能力较强，这既保存优秀抗体，又保证抗体的多样性，使算法不至于早熟收敛。在生存选择时，抗体 w 都要与抗体群中任意抽取的若干个抗体的生存期望值进行比较。生存期望值越高的抗体，根据预先设定的淘汰率 P_e 比较后进入下一代的机会越大。之后的遗传交叉和变异操作采用一点交叉算子和单点变异策略，按照事先设定的交叉率 P_c 和 P_m 变异率进行。通过多样性判别和约束条件判别，对产生的解大量筛选，以保证上述操作后解的可行性。

(4) 记忆细胞分化、更新记忆库。当抗体 w 的浓度 C_w 超过浓度阀值 T_c 时，表明这种抗体在抗体群中占有优势，将分化成记忆细胞。但由于记忆细胞的总数是有限的，所以当分化的记忆细胞达到上限时，由新加入的与原有抗体有最大相似度的抗体代替原抗体。

(5) 判断终止条件。判断是否达到进化截止代数，判断抗体的平均浓度时都达到稳定。条件满足程序结束，否则回到步骤 2)。

3.2 遗传算子的确定

免疫遗传算法能使抗体保持多样性并且最终能够收敛到最优解的主要操作，就是在算法中有交叉、变异算子的存在，从而使整个抗体群沿着适应度较好的方向搜索。

① 选择算子。选择的目的是为了从当前抗体群中选出优良的个体，使它们有机会作为父代为下一代繁殖子孙。进行选择的原则是期望繁殖率大的个体为下一代贡献一个或多个后代的概率大，选择实现了达尔文的适者生存原则。

本文将父代个体按选择概率进行排序，从父代种群中选择排序靠前的 5% 个体直接复制到子代中，剩下个体的选择采用轮盘赌式选择策略。

② 交叉算子。交叉(Crossover)算子是将选择操作中选取的双亲基因进行重新组合。通过交叉操作可以得到新一代个体，新个体组合了其父辈个体的特性。交叉体现了信息交换的思想。交叉操作通过交换两个父个体的一部分来产生新的子个体，该操作按照一定的交叉概率 P_c 来进行。

③ 变异算子。变异(Mutation)算子保证了生物继续进化。为了维持解群体的多样性，变异首先在群体中随机选择一个个体，对于选中的个体以一定的概率随机地改变个体中某个位的值。变异操作在染色体上自发地产生随机的变化，以提供初始群体中不存在的基因或找回选择过程中丢失的基因，为群体提供新的内容。变异算子将个体的基因链的各位按概率 P_m 进

行变异，同生物界一样，免疫算法中变异发生的概率很低，通常取值在 0.001 -0.2 之间。

对交叉后的群体，以某一概率改变某一个或某一些基因位上的基因值为其它的等位基因。变异本身是一种局部随机搜索，与选择算子结合在一起，保证了免疫遗传算法的有效性，使免疫遗传算法具有局部的随机搜索能力，同时使得免疫遗传算法保持种群的多样性，以防止出现未成熟收敛。

4 仿真实验

为了检验免疫遗传算法的正确性和其优化效率，作者进行了遗传算法、免疫遗传算法查询优化的仿真对比实验。

免疫遗传算法所采用的参数：抗体总数 M +N = 20，其中初始抗体数量 M = 15，追加抗体数量 N = 5；淘汰率 P_e = 0.15；交叉率 P_c = 0.7；变异率 P_m = 0.02；浓度阀值 T_c = 0.5；亲和度阀值 T_v =0.95。

为了便于比较，遗传算法采用的参数：初始种群数 P = 20；进化代数控制在 50 代；交叉率 P_c = 0.7；变异率 P_m = 0.02。变异时基因换位次数选 5。在不同关系数下的查询时间取 3 次查询的平均值，并把结果输出到屏幕上。实验结果如表 1 和表 2 所示：

表 1　基于遗传算法的查询优化结果

关系数	4	8	12	16	20
所需时间(ms)	157	223	318	1986	2387

表 2　基于免疫遗传算法的查询优化结果

关系数	4	8	12	16	20
所需时间(ms)	153	186	255	974	1243

5 结论

针对遗传算法的局部搜索能力不强、未成熟收敛等缺陷，本文在数据库多连接查询优化中引入了免疫遗传算法。免疫遗传算法在遗传算法的基础上引人了抗体相似度调节机制，能有效保持抗体的多样性，避免局部收敛，提高全局搜索能力和收敛速度。从实验结果可以看出：随着关系的个数的增加，免疫遗传算法所生成的计划比遗传算法的明显要好，克服了遗传算法局部寻优能力不足的缺陷，显示了较好的寻优性能。

参考文献

1 韩梅，数据库管理系统查询优化技术研究，解放军信息工程大学，2004.

2 周冬平，关系数据库查询优化技术的研究与实现[D]，南京：南京航空航天大学，2000.

3 任美睿、李建中、李金宝，基于遗传算法的关系数据库查询优化策略，黑龙江大学自然科学学报，2004，(03).

4 K. Bennett, M. C. Ferris, and Y. Ioannidis. A genetic algorithm for database of optimization. In Proc. 4th Int. Conference on Genetic Algorithm s[C]. San Diego, A, 1991. 400 - 407.

5 E. j. Anderson and M. C. Ferris . A genetic algorithm for the assembly line balancing problem . In Proc. of the integer programming combinatorial Optimization Conf. Waterloo [C]. Canada, 1990, University of Waterloo Press.

6 王卫荣、金鹏、黄康，免疫遗传算法及其在多目标优化设计中的应用，机械工程与自动化，2006.

7 缪红萍，免疫遗传算法及应用研究，北京化工大学，2005.

8 王煦法、张显俊、曹先彬，一种基于免疫原理的遗传算法[J]，小型微型计算机系统，1999，20 (2)：117 - 120.