

数据仓库中物化视图选择的一种混合算法

徐海涛, 郑 宁

(杭州电子科技大学 计算机学院, 浙江 杭州 310018)

摘 要: 物化视图是数据仓库中提高查询效率的有效方法, 物化视图选择问题是数据仓库设计时期最重要的决定之一。通过研究和实验, 提出了一种结合遗传算法和模拟退火算法的混合算法, 用于解决物化视图的选择。理论分析和实验结果表明, 该混合算法的搜索性能优于传统的遗传算法, 能够提供更高质量的解。

关键词: 物化视图选择; 遗传算法; 模拟退火算法

中图分类号: TP311.13 文献标识码: A 文章编号: 1000-7024 (2005) 10-2752-04

Hybrid algorithm of materialized views selection in data warehouse

XU Hai-tao, ZHENG Ning

(School of Computer, Hangzhou Dianzi University, Hangzhou 310018, China)

Abstract: Materialized view is an effective method for improving the efficiency of queries in data warehouse system, and the problem of materialized view selection is one of the most important decisions. Through study and experiment, a hybrid algorithm was offered that combined genetic algorithm and simulated annealing algorithm to solve the problem of materialized view selection. Theory analysis and experiment result prove that the hybrid algorithm could provide the solution with higher quality and its search ability excels the traditional one.

Key words: materialized view selection; genetic algorithm; simulated annealing algorithm

1 引 言

数据仓库和联机分析处理(OLAP)是现代决策支持系统的核心部分, 已经成为数据库领域的热点问题。OLAP 是数据仓库系统的主要应用, 主要面向决策人员和高级管理人员。数据仓库里数据的海量性和 OLAP 系统查询的复杂性, 使得查询往往需要很长的时间去执行^[5]。一个普遍采用的提高 OLAP 系统响应速度的技术便是视图物化技术。它是一种将视图所对应数据加以实际物理存储的技术。其目的是通过预计算来加快 OLAP 系统的响应速度。然而把视图进行物化存储既需要占用可观的磁盘空间, 又需要耗费大量的系统资源以对其进行维护, 所以如何选择一组合适的视图集加以物化, 使其在满足存储空间限制的条件下, 各查询的综合响应时间最小, 是一个极为重要的问题。这就是物化视图选择问题。在设计数据仓库时, 其中一个最重要的步骤就是选择哪些视图进行物化, 使其在满足磁盘空间限制或者(和)维护代价限制条件下, OLAP 系统查询的综合响应时间最小^[2,4]。

物化视图的选择一直是数据仓库领域的研究热点, 学术界和工业界都对其进行了大量的研究工作, 很多算法被应用到物化视图选择中。遗传算法由于其健壮性和强大的全局搜索能力自然获得了广泛的应用, 如文献[2,4]所述。但是传统

遗传算法最为严重的问题是“过早收敛”问题^[3]。由于群是有限的, 传统的繁殖(reproduction)-交叉(crossover)-变异(mutation)机制和按适应性比例选择的生存策略, 使得适应性高的个体在下一代中获得较多的取样, 而包含有用模式的低适应性的个体则被过早淘汰, 随着遗传代数的增加, 一旦某些模式在群体中占有优势, 传统遗传算法就会强化这种优势, 从而使搜索范围迅速变窄。迅速收敛的群达到的未必是全局最优, 这就产生了过早收敛。解决过早收敛问题的方法很多, 其中一个比较有效的方法便是将传统遗传算法和模拟退火算法结合, 在遗传算法的生存策略中引入模拟退火算法的 Metropolis 接受准则: 设新个体的适应性为 f , 变动的阈值为 f^* , 当 $f < f^*$ 时, 接受新个体; 否则, 以一定概率 $P = \exp((f^* - f)/T)$ 接受新个体(其中 T 是控制参数, 相当于热力学中的温度), 以使群体保持“有用的多样性”。本文开创性地将遗传算法和模拟退火算法的结合引入到物化视图的选择中, 设计实现了 GA_SA 算法。

2 物化视图选择问题描述

数据仓库的星型模型由一个装载大量数据的事实表和多个维表构成。事实表和维表之间以外键的方式相连接, 事实表由用户关心的度量值和每个维表的主键组成。举一个简单的例子, 零件从供应商处买来, 销售给客户, 构成一个 3 维的

收稿日期: 2004-09-10。

作者简介: 徐海涛 (1978-), 男, 山东青岛人, 硕士, 研究方向为数据仓库和信息处理; 郑宁 (1961-), 男, 浙江杭州人, 研究员, 博士生导师, 研究方向为计算机应用。

数据立方体,这3个维分别是:零件(Part)、供应商(Supplier)、客户(Customer),所关心的度量值(Measure)是销售额(Total_Sale),记录在一张名为Sale的事实表中。各维表的属性和星型模型如图1所示。

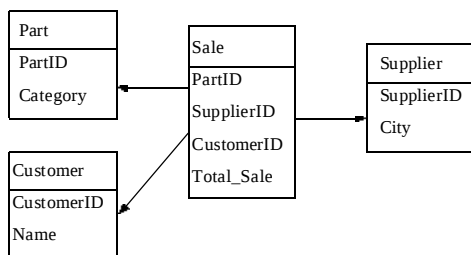


图1 星型模型的一个实例

对于一个有 m 个维表,第 i 维有 m_i 个属性的多维数据仓库来说,存在 $11^{m_i}(2^{m_i}+1)$ 个可能的 OLAP 查询(group-by SQL 查询)^[2]。在这些查询中,有些存在着依赖关系。由于每个查询对应一个视图,下文中如无特殊说明,我们不再区分视图和查询的概念。

定义 1

查询间的依赖关系(Dependency)。给定 2 个查询 Q_1 、 Q_2 ,如果仅仅利用 Q_1 就能回答查询 Q_2 的话,就称 Q_2 依赖于 Q_1 。

如果在一个图中,每个查询对应一个节点, Q_2 依赖于 Q_1 的话,则从 Q_1 到 Q_2 画一条有向边。这样我们就可以画出网格框架图。(网格框架图的详细论述见文献[5])。图2就是一个简单的网格框架图,图中 p 、 s 、 c 分别代表上面例子中的 PartID、SupplierID、CustomerID。为了清晰起见,我们用每个维表的主键代表这个维表而省略了维表中的细节。网格框架图中的每个视图 v 都有一个三元组 (f_v, g_v, r_v) 与之对应。其中 f_v 指 v 的查询频率, g_v 指 v 的修改频率, r_v 指 v 的读取代价。每条边 (v, u) 有 2 个值与之对应:视图 u 相对于视图 v 的查询代价 $q(u, v)$ (即利用 v 来回答 u 的代价)和视图 u 相对于视图 v 的维护代价 $uc(u, v)$ (即利用 v 来更新 u 的代价)。

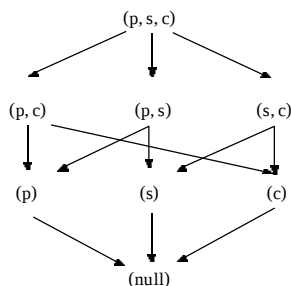


图2 一个简单的网格框架

定义 2

视图 v 的查询代价 $Q(v, M)$:当视图集 M 被物化时,视图 v 相对于 M 的查询代价 $Q(v, M)$ 等于视图 v 相对于 M 中的所有视图的查询代价的最小值,即

$$Q(v, M) = \text{Minimum}(q(v, v_m) | v_m \in M)$$

定义 3

视图 v 的维护代价 $UC(v, M)$:当视图集 M 被物化时,视图

v 相对于 M 的维护代价 $UC(v, M)$ 等于视图 v 相对于 M 中的所有视图的维护代价的最小值,即

$$UC(v, M) = \text{Minimum}(uc(v, v_m) | v_m \in M)$$

定义 4

总查询代价 $Q(M)$:当视图集 M 被物化时,视图 v_i 的查询频率为 f_{v_i} ,则总查询代价等于 L 中每个视图的查询代价加权的和(L 为网格框架图中的所有视图集)。即

$$Q(M) = \sum_{v_i \in L} f_{v_i} * Q(v_i, M)$$

定义 5

物化视图集 M 的维护代价 $UC(M)$ 。当数据源表发生变化时,物化视图也要做相应的修改。这个代价我们称之为物化视图的维护代价,则物化视图集 M 的维护代价 $UC(M)$ 为 M 中每个视图的维护代价加权和。即

$$UC(M) = \sum_{v \in M} g_v * UC(v, M)$$

其中 g_v 为物化视图 v 的修改频率。

这样,物化视图选择问题就可以描述为:寻找一个物化视图集 M ,使其在满足一定限制条件下,总查询代价 $Q(M)$ 最小。这个限制条件可以是 M 占用磁盘空间限制或者(和) M 的维护代价限制条件。随着磁盘价格的不断降低,维护代价限制(即物化视图集的更新时间)越来越成为物化视图选择问题的限制条件。本文将讨论维护代价限制条件下的物化视图选择问题。

3 物化视图选择 GA_SA 算法

3.1 染色体编码

我们采用二进制的 0/1 字符编码。首先,按照一定的图遍历策略将网格框架图中的节点进行排序。然后,根据排序结果创建一个二值字符串,0 表示相应节点不进行物化,1 表示对应节点进行物化。该字符串称为染色体对应物化选择问题的一个方案。

当数据立方体的维数较大时,对应的视图数有成千上万个,记录这么长的 0/1 串是不现实的。这时可以只记录那些被选择了的视图编号。

3.2 初始群体的形成

同传统遗传算法一样,我们采用由计算机随机选择一些视图作为物化视图的方法来生成初始群体。

3.3 适应度函数

因为物化视图选择问题的目标是寻找总查询代价最小的视图集。而一般遗传算法中采用的适应度函数需要最大值,所以采用如下转化:

$$f(x) = \begin{cases} Q_{\max} - Q(x), & \text{当 } Q(x) < Q_{\max} \\ 0, & \text{其它} \end{cases}$$

其中 $Q(x)$ 是总查询代价函数, $f(x)$ 是适应度函数。 $Q_{\max}$ 是一个设定的最大值。

3.4 进化操作设计

我们采用的进化操作有 3 种:交叉、变异、选择。

3.4.1 交叉

交叉是遗传算法中产生新个体的主要方法。它模拟生物进化过程中的繁殖现象,通过 2 个个体的交叉组合,产生新的优良品种,即:从父代中任选 2 个个体(染色体),随机选择 1

个或多个交叉点,交换双亲染色体交叉点右边的部分,就可得到2个新的染色体。在本文中,我们采取单点交叉。下面给出一个交叉操作的实例。

```
010011||001      010011||100
001100||100      001100||001
```

3.4.2 变异

变异是遗传算法中产生新个体的另一种方法,它以变异概率 p_m 随机地改变遗传基因的值,使0变为1,或将1变为0。若没有变异操作,则无法在初始基因组合以外的空间进行搜索,使进化过程在早期就陷入局部解而中止进化进程,从而使解的质量受到很大限制。通过变异操作,可确保群体中遗传基因类型的多样性,以使搜索能在尽可能大的空间进行,从而更可能获得质量较高的优化解。

3.4.3 选择

我们采用的混合算法与传统遗传算法的最主要区别便在于选择操作的设计。与传统遗传算法只是接受优化的解(即后代的适应度函数值比祖先的大)不同,混和算法在传统遗传算法的基础上,引入模拟退火算法的Metropolis接受准则。除接受优化的解外,也会以一定的概率接受非优化的解。随着温度 T 的降低,系统逐渐地不接受非优化解,并最终在温度趋向于0的时候收敛到解状态。具体步骤为:对每对父本 P_1 、 P_2 经过交叉、变异操作后,生成子代 C_1 、 C_2 ,如果 $f(C_i) > f(P_i)$,其中 $i=1, 2$,则用 C_i 替换 P_i ;否则,以概率 $\exp((f(C_i) - f(P_i))/T)$ 保持 P_i 。

3.5 调整交叉、变异操作生成的新个体

前面已经提到,物化视图选择方案必须满足维护代价限制,才是一个可行的解,但是上文描述的生成新个体的交叉操作和变异操作并没有考虑生成个体的维护代价是否超出限制,导致所得到的子代个体很可能根本不是可行解。解决这一问题的方法是:每次进行交叉和变异操作后,对生成的不满足维护代价限制条件的个体进行调整,使之成为一个可行解。本文采取的调整过程如下:若个体所选择的物化视图的维护代价超出限制条件,则随机选择一个未选择的视图代替一个已选择的视图,直到满足维护代价限制条件为止。

3.6 物化视图混合选择算法 GA_SA

GA_SA 算法可以描述如下:

input: 初始温度 T_0 , 群体大小 $pcount$, 物化视图维护代价限制, 网格框架图

output: 满足维护代价限制的物化选择所对应的状态MinS

begin

$m:=0$;

$T:=T_0$;

随机产生初始化群 $G(0)$, 计算群的适应度;

$MinS :=$ 群 $G(0)$ 中适应度函数值最大的个体;

While not($(T < 1)$ and (连续5个温度 T , $MinS$ 的值没有改变)) do

Begin

For $i:=1$ to $pcount/2$ do

Begin

从群 $G(m)$ 中选择2个个体 P_{i1} 、 P_{i2} 作为父本;

对 P_{i1} 、 P_{i2} 执行交叉, 变异操作, 调整生成的新个体使之满足维护代价限制, 生成子代 C_{i1} 、 C_{i2} 并计算 C_{i1} 、 C_{i2} 的适应度函数值;

for $j:=1$ to 2 do

begin

if $f(C_{ij}) > f(P_{ij})$ then 用 C_{ij} 替换 P_{ij}

else

begin

用随机发生器产生一个 $[0, 1]$ 区间的随机数

ξ ;

if $\xi < \exp((f(C_{ij}) - f(P_{ij}))/T)$ then 用 C_{ij} 替换 P_{ij}

else 舍去 C_{ij} , 保留 P_{ij} ;

end;

if $f(C_{ij}) > f(\text{MinS})$ then $\text{MinS} := C_{ij}$;

end;

End;

$T := 0.95 * T$;

$m := m + 1$;

End;

Return(minS);

end;

4 实验结果

为了进一步考察本文所设计的混合遗传算法的实际运行效果,我们在PIV 2.4G+256M内存+Windows2000 Server的机器上对算法的效果进行了实验,把混合算法与传统的遗传算法进行了对比。

(1)遗传参数选择:传统遗传算法和混合遗传算法设置相同的有:变异概率为0.1,群体大小为100。混合遗传算法的参数还有:初始温度为500,温度衰减系数为0.95。

(2)数据立方体:在实验中,我们选择的数据立方体维数为3到12,每个维的属性个数为2到4个。

实验结果如图3所示:图中的y轴表示相对适应度,即传统遗传算法与混合遗传算法解的适应度值之比。

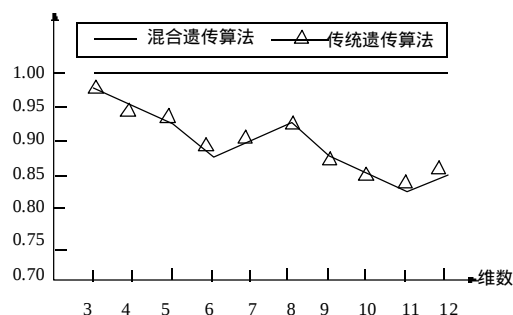


图3 实验结果

从图3可看出,混合遗传算法可得到比传统遗传算法适应度更高,即总查询代价更小的解,证实了混合遗传算法在解决物化视图选择问题时的确优于传统遗传算法。这主要是因为混合遗传算法使用了改进的选择再生的方式,保持了群的多样性,使搜索能有效地被引向含最优解期望值较高的区域。

5 结 论

遗传算法和模拟退火算法都起源于自然界的某些规律，是按自然法则计算的两大分支。将它们结合起来研究，符合自然科学及其分支相互影响、相互渗透的基本规律。本文首先对数据仓库中的物化视图选择问题进行了介绍，然后针对传统遗传算法的缺点，提出了一种结合遗传算法和模拟退火算法思想的混合算法 GA_SA，最后通过实验表明，GA_SA 算法有效的解决了传统遗传算法“过早收敛”的通病，为物化视图选择问题提供了一种新的有效方法。

参考文献：

- [1] Gupta H, Mumick I. Selection of views to materialize under a maintenance cost constraint[A]. Proceedings of the international conference on database theory [C], Jerusalem, Israel, 1999-01. 453-470.

- [2] Xu Yu Jeffrey, Xin Yao, Chi-Hon Choi, et al. Materialized view selection as constrained evolutionary optimization [J]. IEEE Transactions on Systems, Man and Cybernetics-part, 2003, 33(4): 458-467.
- [3] 王雪梅, 王义和. 模拟退火算法与遗传算法的结合[J]. 计算机学报, 1997, 20(4): 381-384.
- [4] Zhang C, Yao X, Yang J. An evolutionary approach to materialized views selection in a data warehouse environment[J]. IEEE Transactions on Systems, Man and Cybernetics-part, 2001, 31(3): 282-294.
- [5] Harinarayan V, Rajaraman A, Ullman J. Implementing data cubes efficiently[A]. Proceedings of the acm sigmod international conference of management of data[C]. Montreal, Canada, 1996. 205-216.
- [6] Kalnis P, Mamoulis N, Papadias D. View selection using randomized search[J]. Data and Knowledge Engineering Journal, 2002, 42(1): 89-111.

(上接第 2745 页)

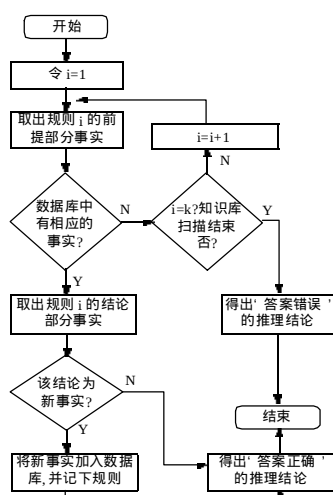


图 4 推理原理

(2) 推理机用这些事实与知识库中规则的前提事实进行匹配。为此首先应将知识库中的一条规则的前提事实取出来，而后看这些前提事实是否都在数据库中，若不全在，继续扫描知识库，进行下一条规则的匹配，若全在则这条规则匹配成功。

(3) 把匹配成功的规则、结论部分的事实作为新的事实加到数据库中去。

(4) 再用更新后的数据库中的所有事实，重复上述(2)、(3)两步，如此反复进行，直到得出结论或不再有新的事实加到数据库中为止。

为保证主观题的判卷准确性还需要有比较完整的知识库维护机制，包括事实库的建立、撤消、知识的插入、删除、检索、一致性检查以及安全维护等。

5 题库文件加密、解密算法设计思想

在算法设计中，充分利用了 VFP 有关文件操作方面的函

数：fopen()、fcreate()、fseek()、fwrite()、fread()、fclose()。具体加密算法实现方法是：利用 sys() 函数搜索考试文件夹下面的所有题库文件(.DBF)，利用 fopen() 函数打开题库文件，利用 fread() 将其读入内存，并将库文件所有字符的 ASCII 值加一个确定值后，再利用 fwrite() 写回文件，并且将题库文件扩展名改为.DAT；解密算法实现方法是：利用 sys() 函数搜索所有考试文件夹下面的加密的题库文件(.DAT)，利用 fopen() 函数打开题库文件，利用 fread() 读入内存，将题库文件有字符的 ASCII 值减去一个加密对应的确定值后，利用 fwrite() 写回文件，并将题库文件扩展名改为.DBF。

6 结 论

本考试系统的通用性非常强，功能比较齐全，界面比较好，授权教师可随时增加或修改题库，操作非常简单，完全可以应用于英语、政治、语文、数学等各种课程考试，目前已在湖南工业大学及一些中小学校得到了推广。

参考文献：

- [1] 李新国. 基于 VF6.0 的学校考试自动组卷系统[J]. 计算机工程与设计, 2003, 24(11): 66-69.
- [2] 尹朝庆, 尹皓. 人工智能与专家系统[M]. 北京: 中国水利水电出版社, 2002. 151-167.
- [3] 郑燕. 网络无纸化考试系统[J]. 安徽理工大学学报(自然科学版), 2003, 23(1): 39-43.
- [4] 王永庆. 人工智能原理与方法[M]. 西安: 西安交通大学出版社, 1998. 297-393.
- [5] 曾华军, 申瑞民, 申丽, 等. 远程考试系统中的数据安全[J]. 计算机工程, 2001, 27(3): 133-135.
- [6] 曲守宁, 董吉文, 程涛远. 计算机基础考试中 Office 操作自动评阅的研究[J]. 济南大学学报, 2001, 15(3): 201-203.