物化视图选择中权限因子的设计

王伟皓 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手段,物化视图的选择一直是数据仓库领域的研究热点。通过研究和实验,提出在物化视图选择中加入权限因子,将各候选视图的权限值纳入算法评价函数的计算中,使最终得到的物化视图集既能面向企业基层提供 OLAP查询,又能保证企业决策层 OLAP查询的速度。

关键词 数据仓库 物化视图选择 权限因子 遗传算法

DESIGN OF PRIVILEGE FACTOR IN MATERIALIZED VIEWS SELECTION

Wang Weihao Zheng Ning
(*School of Computer, Hangzhou Dianzi University, Hangzhou 310018, Zhejiang, China*)

Abstract Materialized view is an effective method for improving the efficiency of enquires in data warehouse and the selection of materialized views is one of the important issues in data warehouse domain. Privilege factor is added to algorithm of materialized views selection in order to deal with privilege value during the calculation process of evaluation function. In this way, the final selected collection of materialized views can not only support the OLAP enquires for grass roots of enterprise but also guarantee the response speed of the OLAP enquires for decision-layer of enterprises.

Keywords Data warehouse Materialized views selection Privilege factor Genetic algorithm

0 引 言

数据仓库是面向主题的、集成的、非易失的且随时间变化的数据集合,用来支持管理人员的决策[1]。数据仓库中存储有海量的用于查询和分析的集成数据,为了提高查询和分析的速度,数据仓库将一部分查询视图预先进行计算并加以存储,这便是物化视图技术。在设计数据仓库过程中,一个最重要的步骤就是选择哪些视图进行物化,使其在磁盘空间或(和)维护代价限制条件下,对 OLAP查询的响应时间最小。这需要考虑多方面因素,如查询处理代价、物化视图的维护代价,以及物化视图的存储空间,并且需要在这几方面进行权衡[2]。

物化视图选择中研究的算法有贪心算法[4-6]、遗传算法[3,5]等,这些算法能解决存储空间或更新代价约束下物化视图的选择问题。但随着数据仓库的不断深入应用,OLAP查询的使用不再局限于企业的决策层,更多的普通员工也能在工作中使用 OLAP系统。这便给物化视图选择算法提出了新的要求,因为决策层对查询和分析的时效性要求较高,所以其查询对应的视图被物化的概率应该相对较高,而普通员工查询对应的视图被物化的概率应该相对较低。然而实际情况往往相反,由于企业的组织结构呈金字塔型,普通员工视图的查询频率较高,将其物化会较显著地降低整体查询代价,所以它们被物化的概率便比决策层视图被物化的概率高。为了弥补这方面的缺陷,使物化视图集既能加快决策层的查询速度,又能保证普通员工查询速度的基本需求,本文提出在物化视图选择算法中加入权限因子,将权限值纳入算法评价函数的考量之中,较好地处理了决策层查询速度和查询频率之间的矛盾。

1 星型模型及其维组合之间的偏序图

数据仓库中常用的数据模型有星型模型和雪花模型。本文中将采用星型模型,它由一张事实表和多张维表组成。事实表和维表之间以外键的方式连接,事实表由用户关心的度量值和每个维表的主键组成。为了更好地论述问题,下面举一个简单的例子。

例 1:一个事实表 Sales,三个维表 Part、Customer、Supplier组成的星型模型,表示零件从供应商处购买,销售给客户。关系模式如下:

Sales(PartID, SupplierID, CustomerID, Dollar Sold) /事实表,其中 Dollar Sold是度量值
Part(PartID, Name, Category) /零件维
Customer(CusotmerID, Name, Address) /客户维
Supplier(SupplierID, Name, City) /供应商维

用户在使用数据仓库时,要获得不同程度的综合数据,即按照维对事实表的数据进行分组汇总。一个有 n 维的星型模型,总共有 2^n 种分组方式,每一种分组方式可以对应一类查询(具有相同 Group By子句的查询)。由此可知,例 1中有 8种分组

收稿日期:2005-10-08. 王伟皓,硕士生,主研领域:数据仓库,信息处理。

方式, 分别为: (P, S, C)、(P, C)、(P, S)、(S, C)、(P)、(S)、(C)、(none), 其中(none)表示不带有 Group By子句的查询, P、C、S 分别是零件维、客户维、供应商维主键的缩写。

由于每个查询可以对应一个视图, 所以下文中如无特殊说明, 将不再区分查询和视图的概念。

定义 **1**　视图间依赖形成的偏序关系

给定两个视图 V_i 和 V_j, 如果仅用 V_j 能回答视图 V_i, 便称 V_i 依赖于 V_j, 即视图 V_i 和 V_j 之间存在依赖关系(Dependency)。那么, 它们之间的偏序关系表示为 $V_i \preceq V_j$。

定义 **2**　视图及其依赖关系的偏序图

可以用一张有向无循环的偏序图来描述视图结点和其间存在的依赖关系: $G=(V, E)$, 其中 V代表数据仓库中 OLAP查询对应的视图, 图中用结点表示, E 代表视图间的依赖关系, 图中用有向边表示。如果图中存在一条 V_i 至 V_j的有向边, 即 $V_i \rightarrow V_j \in E$, 则表示 $V_j \preceq V_i$, 且$\neg \exists V_k$, 使得 $V_j \preceq V_k \wedge V_k \preceq V_i$, 其中 $V_j \neq V_k \neq V_i$。本文使用 $V(G)$和 $E(G)$来分别表示图 G中结点和边的集合。

例 1中 8组视图之间存在的依赖关系所形成的偏序图如图 1所示。例如图中(P)表示每一个零件销售额的视图, (P, S)表示每个供货商下每一个零件销售额的视图。从结点(P, S)到(P)存在一条边, 表示视图(P)可由视图 (P, S)求出。

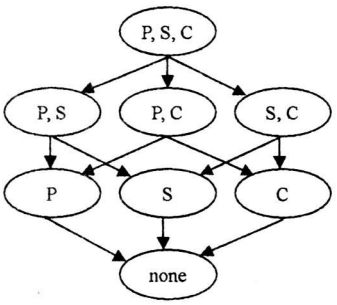

图 1　视图形成的偏序图

2　一种带权限因子的物化视图选择问题描述

在定义物化视图选择的代价模型之前, 首先需要引入一些和代价模型相关的权定义。在偏序图 $G=(V, E)$中存在以下两类权定义, 分别与图中的结点及边相关。

· 对于结点(视图) $v \in V(G)$存在四个权定义

r_v: 结点 v的读取代价;

p_v: 结点 v的权限值;

f_v: 结点 v的查询频率;

g_v: 结点 v的更新频率。

· 对于边(v, u) $\in E(G)$存在两个权定义

$q(v, u)$: 利用结点 v对结点 u进行查询的代价;

$m(v, u)$: 利用结点 v对结点 u进行更新的代价。

本文在结点的权定义中加入了代表权限因子的权, 对于每个物化候选视图, 因访问者不同, 即有些视图是企业决策层查询所要使用的, 而另一些视图是普通员工查询所要使用的, 会使得不同的物化候选视图所拥有的权限值不同。

文中假定视图的访问者权限越高, 其权限值 p_v越大, 访问者权限越低, 其权限值 p_v 越小。同时, 对于一些企业决策层和普通员工都要使用的视图, 它们的权限值大于单类访问者所使用视图的权限值。

定义 **3**　视图 v的查询代价 $q(v, M)$

对于已选择的物化视图集 M, $q(v, M)$表示使用物化视图集 M来回答视图 v的最小代价, 即 $q(v, M) = \mathrm{Min}(q(v, v_m)), v_m \in M)$, 其中 $q(v, v_m)$的值等于物化视图 v_m 的读取代价加上从结点 v_m 到 v的路径上各边相关的查询代价之和。通常从物化视图集 M到视图 v存在多条路径, 那么 $q(v, M)$便是其中的最短路径。

定义 **4**　物化视图 v的维护代价 $m(v, M)$

对于已选择的物化视图集 $M(\subseteq V(G))$, $m(v, M)$表示使用物化视图集 $M-\{v\}$来更新物化视图 v的最小代价。即 $m(v, M) = \mathrm{Min}(m(v, v_m), v_m \in M-\{v\})$, 其中 $m(v, v_m)$的值等于从结点 v_m 到 v的路径上各边相关的更新代价之和。通常从物化视图集 $M-\{v\}$到物化视图 v存在多条路径, 那么 $m(v, M)$便是其中的最短路径。

定义 **5**　总查询代价 $r(G, M)$

当视图集 M 被物化后, 视图 v的查询频率为 f_v, 则总查询代价等于 $V(G)$中每个视图的查询代价的加权和, 即 $r(G, M) = \sum_{v \in V(G)} f_v \times q(v, M)$。

定义 **6**　物化视图集 M的维护代价 $U(M)$

当数据源表发生变化时, 物化视图也要作相应的修改, 这个代价称为物化视图的维护代价。物化视图集 M的维护代价 $U(M)$为 M中每个物化视图的维护代价的加权和, 即 $U(M) = \sum_{v \in M} g_v \times m(v, M)$。

定义 **7**　物化视图集 M的总权限值 $P(M)$

当视图集 M 被物化后, 由于偏序图的结点中引入了与结点相关的权限值, 所以便可定义物化视图集 M 的总权限值 $P(M)$等于各物化视图权限值之和, 即 $P(M) = \sum_{v \in M} p_v$。

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, 使其在维护代价限制条件下, 总查询代价 $r(G, M)$最小, 同时需要计算该物化视图集 M 的总权限值 $P(M)$对 $r(G, M)$的影响。

3　带权限因子的物化视图选择算法

一般的遗传算法中, 处理受限的组合优化问题时需要将评价函数和惩罚函数结合起来, 但在实践中结合两者的惩罚系数往往靠经验来制定, 会造成一定的误差, 所以, 本文在物化视图选择的遗传算法中使用了随机排序算法[5,7,8], 使得在处理受限的组合优化问题时不需要结合评价函数和惩罚函数, 而是用排序概率来权衡使用评价函数还是惩罚函数。

3.1　染色体编码及产生初始群体

算法首先按照一定的图遍历算法对偏序图中的结点进行排序, 根据排序结果创建二进制字符串。算法中称该字符串为染色体, 每条染色体代表一种物化视图选择的候选解决方案。染色体中每一位对应偏序图中某个视图, 其长度为偏序图中视图的数量, 0表示图中对应视图不被物化, 1表示图中对应视图被物化。

算法然后会随机产生一组染色体, 作为染色体初始群体。

3.2　评价函数和惩罚函数

在传统的维护代价限制下物化视图选择算法中, 评价函数

定义为[5]:

$$f(x)=r(G,\Phi)-r(G,M_x),\ \text{满足}\ U(M_x)\leqslant S$$

其中, $r(G,\Phi)$代表不存在物化视图集情况下的总查询代价, 即最大总查询代价, $r(G,M_x)$代表存在物化视图集 M_x情况下的总查询代价。算法最终要选择一组物化视图集 M_x, 使函数 $f(x)$的值最大化, 同时满足其维护代价 $U(M_x)$不超过规定的限制值 S。

由于在维护代价限制下物化视图选择算法中考虑了权限值, 所以加入权限因子后, 评价函数定义为:

$$f(x)=r(G,\Phi)-r(G,M_x)+\lambda P(M_x),\ \text{满足}\ U(M_x)\leqslant S$$

这里 λ为权限系数, 用于调节物化视图集总权限值 $P(M_x)$在评价函数中的比重。加入权限因子后, 如果物化视图集 M的总权限值高, 即该物化视图集较多地为企业决策层所关注, 那么由它得出的评价函数值便会得到相应的增长, 该物化视图集作为最终选择结果的概率也会相应提高。

算法执行过程中, 有些物化视图集的维护代价可能会超过限制值 S, 所以需要定义惩罚函数 $\Phi(x)$, 其定义如下:

$$\Phi(x)=\max\{U(M_x)-S,\ 0\}$$

算法执行时会对两个物化视图集进行比较排序, 如果其中任一物化视图集的维护代价超过限制值, 即其 $\Phi(x)\neq 0$且产生的随机数大于排序概率 P_f, 那么排序将依据两者的惩罚函数值进行判断, 否则排序将依据两者的评价函数值进行判断。

3.3 遗传算法实现

针对以上在评价函数中加入权限因子的分析, 运用遗传算法来实现物化视图的选择, 能更好地平衡企业决策层查询速度和普通员工查询速度之间的比例。算法的基本框架如下:

```
参数: 群的大小为 P
begin
随机产生初始群 G(0)
repeat
  t=t+1;
  G1(t)=Crossover(G(t-1));
                          /使用杂交算法, 产生新的群
  G2(t)=Mutation(G1(t));
          /使用变异算法, 对群中每个染色体的每一位进行变异
  S=StochastidRanking(G(t-1)∪G2(t));
/使用随机排序算法, 对 G(t-1)∪G2(t))进行排序, S的大小为 2×P
  G(t)=S的前 P个染色体;
until满足终止条件
end
```

基于框架中用到了杂交[5]、变异[5]和随机排序[5,7,8]三种操作。

1) 杂交

杂交是遗传算法中产生新染色体的主要方法。它通过从父代中任选两个染色体, 以杂交概率 P_c随机选择一个或多个交叉点, 交换双亲染色体交叉点右边的部分, 来得到两个新的染色体。文中采用单点交叉, 下面给出一个杂交操作的实例。

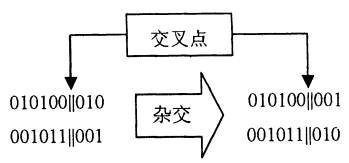

图 2 杂交操作实例

2) 变异

变异是遗传算法中产生新染色体的另一种方法, 它以变异概率 P_m随机改变染色体上每一位的值, 使 0变为 1, 或将 1变为 0。如果没有变异操作, 则无法在染色体初始组合以外的空间进行搜索, 使遗传过程在早期就陷入局部解而中止, 从而影响解的质量。通过变异操作, 可确保群体中染色体类型的多样性, 扩大解的搜索空间, 从而获得质量较高的优化解。

3) 随机排序

在通过杂交和变异操作生成新的染色体群之后, 需使用随机排序算法对群中染色体进行排序。随机排序操作使遗传过程中生成的每一代染色体群不断向好的方向进化, 逐渐向最优解逼近。它类似于冒泡排序, 通过比较两个相邻染色体的评价函数值或惩罚函数值, 来决定两者是否进行对换。算法会用到一个排序概率 P_f, 来权衡排序时是使用评价函数还是惩罚函数。

4 实验结果

为了进一步考察在评价函数中加入权限因子的作用, 实验通过调整 $f(x)$中的权限系数 λ来观察决策层和普通员工平均查询时间的变化, 并对不加权限因子和加入权限因子的物化视图选择算法的收益进行了比较。

实验中遗传算法的参数选择如下: 变异概率 P_m为 0.01, 杂交概率 P_c为 0.1, 权衡参数 P_f为 0.4, 群大小为 100, 数据立方体维数为 8, 每个维的属性个数为 2到 4个。

图 3显示了随着权限系数 λ的变化, 决策层平均查询时间和普通员工平均查询时间的变化。横轴表示评价函数中权限系数 λ的变化, 纵轴表示平均查询时间, 单位是毫秒, 实验中总查询次数设为 20次。

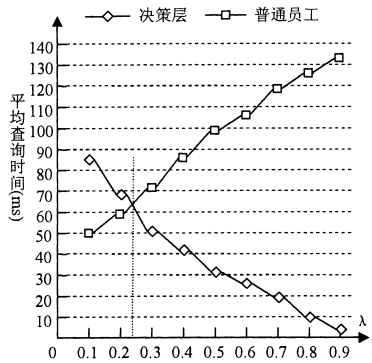

图 3 决策层平均查询时间与普通员工平均查询时间

从图 3可以看出, 决策层平均查询时间曲线和普通员工平均查询时间曲线交汇在权限系数 λ约等于 0.23之上。为了保证决策层的查询时间小于普通员工的查询时间, 就需要设置 λ的值大于 0.23。从实验结果可以得出, 一般情况下可将 λ的值设为 0.4或 0.5, 这时决策层的查询时间约为普通员工查询时间的 1/2或 1/3。

图 4对不加权限因子和加入权限因子的物化视图选择算法在收益上进行了比较, 横轴表示评价函数中权限系数 λ的变化, 不加权限因子的物化视图选择算法不受其影响, 纵轴表示两种算法解的收益比, 即 $f(x_{\text{加入权限因子}})\setminus f(x_{\text{不加权限因子}})$, 其中 $f(x)=r(G,\Phi)-r(G,M_x)$。

从图 4可以看出, 总体上两种算法解的收益(即最终选取

(下转第 175页)

$a_0(z)=5$, $a_1(z)=-0.2$, $a_2(z)=0.44$, $c(z)=100$, $c_1=c_2=1$. 取 $m=16$的 Haar小波基, 可得仿真曲线如图 1~图 3所示。

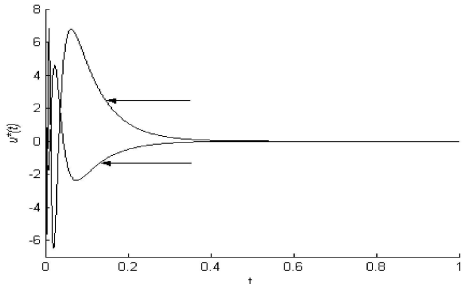

图 1　最优控制曲线 $u_1^*(t)$, $u_2^*(t)$

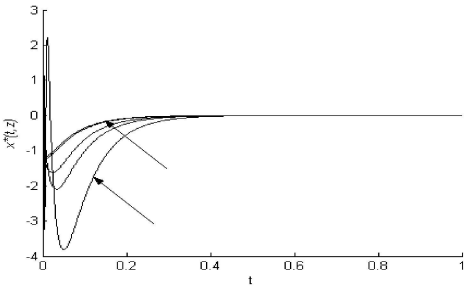

图 2　最优状态变量 $x^*(t,z)(z=0.1-0.5)$逼近曲线

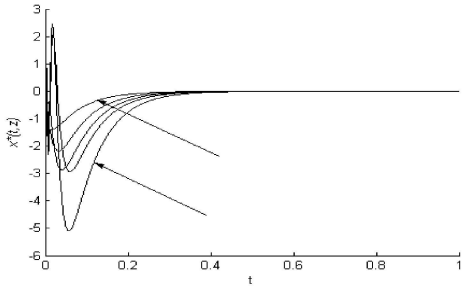

图 3　最优状态变量 $x^*(t,z)(z=0.1-0.5)$逼近曲线

4　结束语

提出的基于小波变换的分布参数系统的最优边界控制算法, 通过小波变换及其相应的运算矩阵性质的应用, 将分布参数系统的最优边界控制问题转化为集总参数系统的最优控制问题, 有效地解决了分布参数系统的最优边界控制问题。通过以上对于分布参数系统问题的描述和基于 Haar小波的逼近算法的推导以及仿真结果, 可以看出利用 Haar小波逼近方法来解决分布参数系统最优边界控制问题, 是一种有效的方法。该方法算法简单、计算量小、逼近精度较高, 避免了直接求解的困难, 为解决分布参数系统最优边界控制问题提供了一种新的途径。

参　考　文　献

[1] 高桂革.基于小波分析的分布参数系统控制的若干问题研究.上海:华东理工大学硕士学位论文, 2002.

[2] 顾幸生, 蒋慰孙.线性分布参数系统的最优边界控制.浙江大学学报, 1996 增刊(2): 52-57.

[3] 高桂革, 顾幸生, 曾宪文. Haar小波运算矩阵与性质在分布参数系统控制中的应用.华东理工大学学报, 2004 30(4): 458-461.

[4] 顾幸生, 蒋慰孙.基于正交函数逼近变换的分布参数系统可控性与可观性.华东理工大学学报, 1997 23(5): 596-602

[5] Gu J S, Jiang W S. The Haar wavelets operational matrix of integration. Int. J. of Systems Sci., 1996, 27(7): 623-6.

(上接第 **106**页)

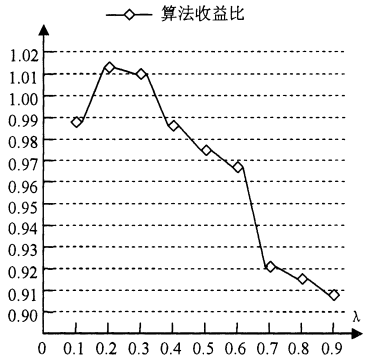

图 4　加入权限因了与不加权限因子解的收益比较

的物化视图集的评价函数值)非常接近, 比值介于 0.9至 1.02之间。虽然随着 λ值的增大, 加入权限因子的物化视图选择算法更多地考虑了权限值大的视图, 使它们被物化的概率增大, 这会影响到最终选取的物化视图集的收益, 导致两种算法的收益比降低。但是在收益非常接近的情况下, 加入权限因子的物化视图选择算法得到的视图集能给企业决策层带来更高的查询效率。

5　结束语

在物化视图选择算法中加入权限因子, 意味着对候选视图进行分级, 算法会根据视图权限值的不同来决定它们入选物化视图集的概率, 权限值高的视图被选中的概率大。由于这些视图通常对应企业决策者的查询, 因此在数据仓库应用范围不断扩大的情况下, 决策层的查询分析速度能得到保证。

参　考　文　献

[1] Inmon W H. 数据仓库[M]. 北京:机械工业出版社, 2002

[2] Lee M, Hammer J. Speeding UP Warehouse Physical Design Using A Randomized Algorithm[C]. Int. J. Cooperative Information Systems, 2001, 10(3): 327-353.

[3] Harinarayan V, Rajaraman A, Ullman J D. Implementing Data Cubes Efficiently[C]. ACM SIGMOD International Conference of Management of Data, Montreal, Canada, June 1996, 205-216.

[4] Yu J X, Yao X, Chi-Hon Choi, Gou G. Materialized View Selection as Constrained Evolutiongary Optimization[C]. IEEE Transactions on Systems, Man and Cybernetics, Part C, 2003, 458-476.

[5] Gupta H, Mumick I S. Selection of Views to Materialized Under a Maintenance Cost Constraint[C]. International Conference on Database Theory, Jerusalem, Israel, January 1999, 453-470.

[6] T.P. Runarsson, Yao X. Stochastic Ranking for Constrained Evolutionary Optimization[C]. IEEE Transactions on Evolutionary Computation, 2000, 284-294.

[7] Zhang C, Yao X. An Evolutionary Approach to Materialized Views Selection in a Data Warehouse Environment[C]. IEEE Transactions on Systems, Man and Cybernetics, Part C, 2001, 284-294.