

1983年全美计算机会议论文集

中国科学院成都计算机应用研究所情报室译

AFIPS

73.87083
144
:1983
:2

1983年全美计算机会议论文集

下 卷

中国科学院成都计算机应用研究所

目 录

决 策 系 统

对数据库中存在相关性的新展望.....	(1)
关系模型管理系统的设计问题.....	(8)
DSS效用的焦点.....	(13)
集中决策的信息资源管理.....	(17)
管理系统的战略剖面.....	(21)
DSS 研制系统.....	(27)
用模糊语言和图示数据库设计决策系统.....	(43)
适于使用数据库的决策系统.....	(64)

硬 件

大学与高性能计算技术的未来.....	(79)
图形显示用的动态RAM体系结构.....	(89)
iRAM——微处理机的一种新型存贮器.....	(97)
继续发展MULTIBUS来应付VLSI革命所带来的问题.....	(106)
M6809指令系统的分析.....	(110)
从试用的足迹谈起：视频电报在美国的进展.....	(123)
在商业/用户市场的视频电报和电文广播.....	(128)
多用户/多任务应用的温彻斯特.....	(132)
Intel iAPX 432—容错计算机用的超大规模集成电路积木式部件.....	(139)
MP/C 的性能评价.....	(146)
具有重复内容共用存贮器的多处理机.....	(166)
VLSI处理阵列的可重组体系结构.....	(174)
相联数据文件的无冲突存贮器地址分配.....	(189)
可重新配置的容错多机网络.....	(207)

长途通信/应用

开放系统互连 (OSI) 的标准会话协议	(226)
智能外设接口在系统体系结构中的作用	(233)
OSI 参考模型网络层的进展	(242)
数字语言技术: 压缩, 编辑和存贮	(249)
自动语言识别的统计建模	(251)
大规模集成电路技术对于语言处理的含意	(252)
网络的安全性和脆弱性	(253)
IBM 信息网络的性能和有效性测量	(261)
为网络增长而设计和管理SNA网络	(269)
IBM 信息网络的备用设备和恢复	(277)
SNA网络中逻辑问题的确定	(281)
在Bell系统内规划高速数字服务	(288)
用来改进大型长途计算机通信网络中集中路径选择的三种试探法	(292)
一种新的用于包交换计算机网络的概率性的路径选择算法	(307)
办公室通信用的光导无线调制解调器	(324)
高吞吐量的互联结构	(333)
对计算机合同的新观点	(337)
一种用于开发信息系统的信息系统	(344)
估计质量的度量	(354)
软件生产率的测量方法	(357)
NTT的电气通信实验室自动化系统	(362)
数字光学磁盘在图书馆资料保存和文献检索中的应用	(370)

个人计算机

• 软件维护目标	(373)
----------	---------

对数据库中存在相关性的新展望

T. C. CHIANG

摘要

为了保证数据的一致性，在更新记录时必须保持记录之间存在的相关性。在设计系统时必须识别这种相关性，才能正确地执行更新操作。

本文介绍对数据库中存在相关性的新展望。它等于一套基本更新规则，它可以结合数据库管理系统 (DBMS) 来保持记录之间的存在相关性。文中表明由其他现有商业上能买到的 DBMS 支持的存在相关性可以用基本规则精确定义。而且，这些基本规则的组合捕捉到新的存在相关性语义，这些语义是不由市场上买到的系统处理的。本文还讨论到与记录密切关联的“干扰”问题解决办法。这些干扰问题对系统操作虽然重要，但以前的文献中从来没有讨论过。

引言

在一个与许多实体有着复杂关系的企业中，一个实体的存在往往依赖于其他实体的存在。由于一个数据库代表一个真实的世界，那个世界中的存在相关性在数据库上反映为数据记录中的存在相关性。在需要频繁删改和插入记录的应用中，数据库的存在相关性必须正确处理，以避免数据的不一致性。

在数据库系统中处理**存在相关性**的适当位置是在数据库管理系统 (DBMS) 中 (与在用户程序中相反)。像 IBM/IMS 这样的商业性 DBMS 或其他以 CODASYL 为基础的系统处理少数几种存在相关性。例如在 IMS 中删除常驻段的事例将触发常驻段下面所有派生段事例的删除。在以 CODASYL 为基础的

系统中，在一个 CODASYL 集合中成员资格说明有 AUTOMATIC 和 MANUAL，和 MANDATORY 及 OPTINAL，它定义记录中某种相关性。例如，一个成员插入一个集合，它不能被删除而独立于集合之外。这种机理不够丰满到足以捕捉大多数数据库中其他种类的存在相关性。

在科学界中对存在相关性的理论很少注意。例如在关系数据模型的早期工作中，对功能相关性和规格化非常重视。**更新异常**这个术语被用来定义与存在相关性有关的所有问题。然而在事实上，许多异常的，不过是期望的存在相关性，并且应当照此处理。近几年来，生产相关性问题已经引起重视。陈定义了实体中捕捉存在相关性的某些语义的正规和薄弱实体概念。在以后的一篇论文中，Dogac 和陈提出更新传播的概念。

同理, Smith和Smith谈到涉及关系记录自动更新的触发更新概念。Keller和Wiederhold表列出许多在保持数据库一致时被认为是很重要的存在相关性。这些论文对存在相关性问题的论述都不够广泛,不足以捕捉大多数的相关性语义。钱和Bergeron介绍了处理存在相关性集合的系统。但是没有提到其他与存在相关性有关问题的细节。Navathe和Schkolnick谈到意图表达的结构中的更新规则, Date定义了一种丰富的语言, 可以把存在相关性描述为关系数据模型的参考完整性。然而这些论文都没有把存在相关性的语义分解成原子个体, 或者提到干扰问题。

在本文中广泛讨论到存在相关性及其有关问题。存在相关性被看作是实体中间关系的性质。**耦合因素**被定义为处理实体之间的存在相关性的一组更新规则。本文说明由商业系统(例如IMS和CODASYL系统)和其他系统支持的更新规则可以用耦合因素来描述。此外, 对存在相关性问题的新看法使我们能够去发现新的相关性语义。DBMS将加强处理存在相关性的规则, 从而使记录删除和插入将不会把数据库变成一种非一致状态。当在同一组实体中存在两种关系和关系的存在相关性互相干扰的时候就会出现干扰问题。本文以后的几节将定义不同种类存在相关性的集合和他们的干扰, 并将介绍一种检测干扰的算法。

本文中介绍的成果实际上在美国贝尔电话公司供电话营业之用的DBMS中已经实现了。

存在相关性

一种扩展的实体关系(E—R)数据模型被用作处理存在相关性的基础。大体上说, 一个基本E—R模型可以检验由文件组成的数据库, 而这些文件则是记录的集合和记录

中的关系。在扩展的E—R模型中, 一组关系被叫做结合。结合有一个**耦合因素**作为它的性质之一。耦合因素就是一组存在相关性。为了简化表达, 本文的其余部分只讨论二进制结合的存在相关性。所以今后术语**耦合**将指二进制结合而言。还有, 结合将被看作是像它的范围一样, 是与所涉及到的两个文件的二进制关系。

虽然可能存在多种多样的存在相关性, 有四种基本的存在相关性应当指出: 定义如下。假定E1和E2是两个文件, A是E1和E2之间的结合。假定e1是E1中的任意记录, e2是通过A与e1相联的E2类型中的记录, 则e1和e2被叫做A—相联的。基本存在相关性可以定义为下列更新规则:

1. 一个e2不能插入, 除非已经存在一个与A相联的e1。e2的插入意味着在e1和e2之间建立起关系。
2. 删除一个e1意味着删除所有与A相联的e2。
3. 如果存在着一个与A相联的e1, 则不能删除一个e2。
4. 不能删除关系, 除非e1或e2被删除掉。

应当指出, 删除e1和e2意味着e1和e2之间的关系被删除。只有e1和e2都存在时才允许插入关系。还有, 由规则确定的相关性是有方向性和可传送的。举例说, 在以上规则的描述中, E2依赖于E1。而且, 如果E2依赖于E1和E1依赖于E3, 则意味着E2依赖于E3。

一组耦合因素可以定义为基本规则的幂集。因此, 可能有16种耦合因素, 其中包括空耦合因素, 它并没有上述的相关性。按照钱和Bergeron的意见, “非常紧密”的耦合因素的定义是一种具有全部四种相关性的耦合因素。其他耦合因素代表记录之间各种不同的相关性。耦合因素是没有方向性

的，即所有耦合因素的规则的方向都相同，它们的方向就定义为耦合因素的方向。耦合因素也是可传送的。耦合因素可传送的性质对更新的传播极为关键。

利用耦合因素的这些定义，我们可以看出得到其他规则支持的更新规则能够用耦合因素来描述。

IMS

IMS的更新规则由它的分级结构定义。

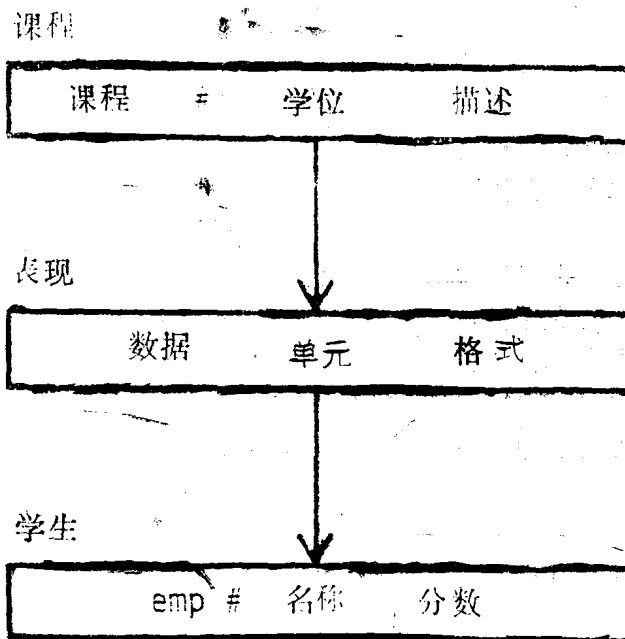


图1 教育数据库结构

因此，删除层次结构中的一个节点意味着删除它的全部子节点。在层次结构中插入一个节点需要有它的母节点存在。

IMS更新规则可以看作是由上述规则1，规则2和规则4组成的耦合因素。例如，研究一下由Date提出的一个教育数据库子集。图1示出教育数据库的结构。

只有提供课程，学生段才能插入（即规则1），删除一个课程提供，意味着删除与提供相联系的全部学生段（即规则2）。而

且，IMS不允许分解层次关系（即规则4）。必须指出，IMS不支持规则3。

CODASYL系统

在CODASYL系统中，成员资格层在一个集合中，它可以用来描述更新规则。假空物主（O）和成员（M）是两种记录类型，OM是O和M之间定义的集合。OM中M的m成员资格被认为是**强制性的**，假如m不能由集合中消掉，而物主O的删除又意味着删除全部成员的话。假如成员资格是**可以选择的**，那么m可以从OM中消掉而不要从数据库中删除掉m。假如OM中M的成员资

格是**自动的**，那么当M的一个m插入时，DBMS将自动地在OM中确定成员资格。如果成员资格是**手动的**，那么程序员必须显式地确定成员资格。成员资格有四种：（1）强制性—自动的，（2）可选择—自动的，（3）强制性—手动的，（4）可选择—手动的。我们可以使用耦合因素来准确地描述这些成员资格。强制性—自动的是包括规则2，4和1的耦合因素；可选择—自动的是包括规则2和1的耦合因素；强制性—手动的是包括规则2和4的耦合因素；而可选择—手动的是规则2。

相关系统

在相关系统中没有类似的更新规则。存在相关性被当作是规格化的前后关系中的异态。举例说,研究下面这个供货厂商的关系

$S'(S\#, SNAME, STATUS, CITY)$

S' 是第二种标准形式,因为所有的非关键属性依赖于关键属性,而STATUS依赖于CITY。这意味着在数据库中有某些城市的供货厂商以前,有关STATUS的信息不能插入,在一个CITY中删除唯一的供货厂商将删除全部STATUS和CITY信息。这些都被Date认为是异态。但是,对某些应用来说,他们可能是符合事实的存在相关性。在这种情况下,耦合因素包括规则1和D节中描述的规则。

为了讲清楚与相关系统中存在相关性有联系的问题,让我们来研究另一个例子。我们研究一个由三个关系组成的数据库:(1)供货厂商(SUPP),(2)另件(PART),和(3)装船(SHIP)。每个关系有一组属性,一组元如下:

SUPP(S# SNAME, STATUS, CITY)。

s1	adams	20	Chicago
s2	baker	10	Newark
s3	chang	30	L. A.
.	.	.	.
.	.	.	.
.	.	.	.

PART(P#, PNAME, COLOR)

p1	chip	red
p2	LED	green
p3	tan	red
.	.	.
.	.	.
.	.	.

SHIP(S#, P#, QTY)

s1	p1	10
s1	p3	5
s2	p2	20
s2	p3	8
.	.	.
.	.	.
.	.	.

假定我们要从数据库中删除所有的红色另件(从PART和SHIP中)。利用伪相关语言,我们就可以表达删除操作如下:

1. 在SHIP·p# = PART·P#时,删除SHIP元
2. PART.color = "red"时,删除PART元。

现在,如果我们颠倒顺序,首先做第2步,然后做第1步,那么删除SHIP元是不可能的,因为关于红色另件的信息在那里不能用。通常,没有相关系统能阻止按错误顺序进行删除。如果我们把SHIP看作是PART和SUPP之间的结合,则删除红色另件将触发SHIP关系元的删除。

我们已经讲过,耦合因素的概念可以用来描述由某些现有系统支持的存在相关性,并可以用来捕捉其他没有被现有系统识别的相关性语义。耦合因素有许多其他重要的性质:其中之一将在下节中讨论。下节中讨论的问题从来还没有由现有系统处理过。

干 扰

当两个相邻结合的耦合因素互相干扰时就会发生干扰现象。举例说,假如我们的数据库由部门文件,雇员文件和项目文件组成,数据库的更新规则如下:

1. 如果没有雇员为之工作的部门, 那么就不能插入雇员记录
2. 如果能处理项目的部门里没有雇员, 那么就不能插入项目
3. 如果没有部门的项目, 那么就不能插入部门记录

如果这三条更新规则是强制性的, 那么没有记录可以插入数据库中。在支持记录之间更加复杂关系的系统中, 两个耦合因素之间的干扰成为一个问题。在本节中, 我们将讲到这种干扰。我们将用B节中提到的前三种耦合因素来说明问题。更详尽的定义如下:

相邻——如果只有A和B共有一个领域, 则结合A可以说成是与另一个结合B相邻, 反之亦然。

并行相邻——如果只有A和B的两个大致上明显的领域集合相同, 则结合A和B被说成是并行相邻。

串行相邻——如果只有A和B精确地共有一个领域, 则结合A和B被说成是串行相邻。

干扰状态——如果没有记录能够插入两个相邻结合的任何领域, 或者从中删除记录, 那么这两个相邻结合被说成是在干扰状态之中。

干扰耦合因素——如果把两个耦合因素赋予两个相邻结合将导致干扰状态, 那么这两个耦合因素被说成是互相干扰。

现在我们准备来看看所有由于八种可能的耦合因素配对而可能产生的干扰状态。

并行干扰

当两个并行相邻结合有干扰的耦合因素时就可能产生并行干扰。只有当两个耦合因素的方向相反时它才能够发生。因此, 本节中的全部定理都假定耦合因素的方向相反。 e_1 和 e_2 用以代表一个结合的两个领域 E_1 和

E_2 中的记录。规则是给两个不同结合规定的。

定理 1。规则 1 干扰规则 1。

证。由于 e_1 的插入依赖 e_2 的存在, 反之亦然, 结合将处于干扰状态。

定理 2。规则 3 干扰规则 3。

证。由于不首先删除 e_2 , 就不能删除 e_1 , 反之亦然, 所以结合将处于干扰状态。

显然, 如果两个耦合因素含有一个干扰规则, 这两个耦合因素要互相干扰。

为了辨别所有各种可能出现的干扰状态, 我们在一耦合因素中分别用三个布尔变量X, Y和Z表示规则 3, 2和1的存在。如果规则存在, 布尔变量的数值为 1。因此, 八个耦合因素可以表示成

XYZ
000
001
010
011
100
101
110
111

在图 2 上示出表示所有可能出现的状态的矩阵, 图中的X表示干扰状态。没有包含规则的空耦合因素不会干扰任何别的耦合因素。从图 2 可以看出, 总共有 28 种干扰状态, 这些干扰状态很容易检查; 画出与所研究的耦合因素的交线; 如果得到的集合包含规则 1 或规则 3, 就有干扰。考虑到每个耦合因素可以用三个布尔变量来表示, 我们甚至可能制成一台硬件的机器, 这台机器可以实现图 2 上那个矩阵所代表的布尔功能。

000 001 010 011 100 101 110 111

000							
001	×	×	×	×			
010							
011	×	×	×	×	×		
100			×	×	×	×	
101	×	×	×	×	×	×	
110			×	×	×	×	
111	×	×	×	×	×	×	

图2 干扰状态

线路干扰

将并行干扰的概念加以推广，可以产生出其他干扰的概念。将关于存在相关性的 E-R图定义为有标号的定向图，我们可以用节点来代表文件，用矢线代表结合。矢线上的标号代表结合名称（它唯一地标志矢线）和耦合因素。图3示出传送相关性信息的 E-R图解实例，图中的 E1 和 E2 是代表文件的节点。A 是结合，C 是 A 的耦合因素。矢

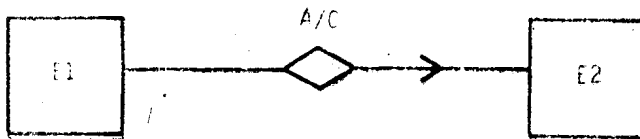


图3 相关性的E-R图解

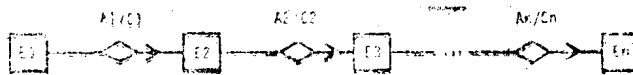


图4 路径的例子

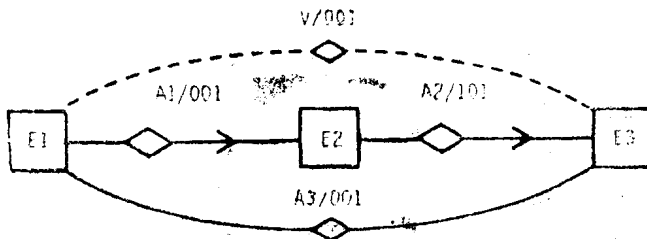


图5 线路干扰

线上的箭头代表耦合因素的方向。由于箭头是从 E1 指向 E2，所以 E1 通过 A/C 依赖于 E2。E1 称为始节点，而 E2 则称为终节点。

E-R图解中的路径是串行相邻边的顺序，所以一边的终节点是另一边的始节点。图4示出一个路径的例子，图中 A1, A2, ..., An 是结合的名称。线路就是路径 A1, A2, ..., An，在这个线路中 A1 的始节点是 An 的终节点。

线路干扰的定义是由结合线路的干扰耦合因素造成的干扰状态。利用第1节的算法和相关性规则的传送性，我们可以把线路干扰问题减小为并行干扰问题。例如，图5表示出线路干扰。

为了检查这种状态，我们可以采取以下步骤：

1. 检查线路。
2. 从线路节点开始。将两条串行相邻边和公共节点减少成一条“虚拟”边 V，它的耦合因素是两个原有结合之交。
3. 重复第2步，直到获得两个并行相邻结合为止。
4. 用前面并行干扰那一节的算法检验并行干扰。

将来的工作

有许多有关存在相关性的重大问题。

1. 制定出更多的存在相关性规则。例如，规则2的变化可概述如下：如果e1是e2的唯一与A相联的e1，则删除e1意味着删除全部与A相联的e2。这个规则捕捉到新的相关性语义，当它与别的规则配对时将产生更多的干扰状态。
2. 把耦合因素的定义扩展到第n级结合。
3. 将耦合因素概念普遍应用到更新相关性。存在相关性是一种更新相关性。另外一种**修改相关性**。修改相关性发生在数据项目的修改依赖于其他数据项目的修改或存在的情

况下。举例说，删除一个雇员记录可能触发公司里全部雇员数目的修改。

结论

我们把存在相关性看作是一组数据语义，它的重要性与功能相关性在相关理论中的重要性是相同的。我们已经介绍了数据库中存在相关性的问题。我们已经制定出一组存在相关性的规则，从中可以定义结合的耦合因素。我们已经介绍过某些耦合因素的重要特性，还规定出许多干扰状态及检验它们的算法。我们深信还有许多存在相关性的重大问题尚有待于去发现。

马德清 译
姚广校

关系模型管理系统的设计问题

ROBERT W. BLANNING

摘要

一部分有关决策支持系统中模型管理的文献提出数据管理的关系结构，这些数据管理可以扩展至决策模型管理。本文探讨关系模型管理中的一个重要设计问题：关系模型组的组织，模型询问语言的相关完整性和系统实现。

引言

在决策支持系统中（DSS），决策模型的使用不断增多，加以数据库管理系统（DBMS）在实现中取得的成功，引起了以下的一些意见：（1）决策模型和数据一样，它是重要的信息源，应当有效地进行管理；（2）为了帮助DSS用户更为有效地利用模型，需要研究模型管理的规范；（3）为了实现重要的模型管理设想要研究模型管理系统。这种模型的例子是供调度和分配调度之用的数学编程模型，运输网络的 Monte Carlo 仿真和企业财务结构的确定性仿真。虽然现在还没有模型管理系统出现，也可以争辩说，如像 EMPIRE 和 IFPS 那样支持财务计划语言的系统是模型管理系统的先驱，模型管理系统最终是要实现的，正像六十年代的文件处理系统是 DBMS 的先驱者一样。

提供模型管理之用的结构之一是关系结构；这种结构的想法是把模型组看做是一组

有输入属性—输出属性和它们之间存在功能相关性的（虚拟）关系，正好像数据库可以被看作是一组有键和容量属性的关系一样。这种看法有两个好处。第一个是由于用模型执行的某些操作在相关数据管理中有配对操作，所以在后面范畴内的重要设想可以在前面的范畴使用。举例说，模型的执行类似于数据关系的选择或限制。同理，无论何时有几个模型用于响应询问，使得一些模型的输出是送到别的模型的输入时，执行类似于接合的操作。显然，在系统实现和程序执行级上，相关数据和模型管理之间有很大的区别，但是我们在这里只涉及到模型的用户观点，而不涉及执行模型的进程。第二个可能的好处是最终可能综合数据和模型管理的关系结构，从而产生 DSS 中信息管理的单独关系结构。

关系模型管理的三个主要问题如下：

1. **模型组的组织。**模型组应当被 DSS 用户看作是单独的“通用”模型，或是一组单独的模型？如果是看成后者，

那么什么是划分通用模型的准则呢？
关系模型组有像关系数据库的标准形式吗？

2. 模型询问语言的相关完整性。什么操作应当由关系模型管理的询问语言规定？他们如何不同于相关数据管理的操作？什么是关系完整模型的语言特性（即在Chomsky层的位置）？

3. 实现问题。有关关系模型管理的安全性、完整性、询问优化和接合的实现有什么问题？对于后者来说，假如一个模型的输出是另一个模型的输出，或者相反，那么所求模型组的一致解如何？

这三个问题将在以下三节中探讨，然后我们将讨论DSS中更广泛的模型管理问题。

模型组的组织

在模型管理文献中，已经提出了几个互相关操作（如像准备销售预测，然后模拟由此引起销售量对生产过程的影响）是否应当由一个模型或几个模型来执行的问题。有的文献提出根据组织功能（财务、销售等），组织级别（策略级、战术级或操作级），模型解法（仿真，线性编程等），或者这些的组合来分解模型。

另外一种方法与相关数据管理中所采用的方法相似，它首先识别所用模型中可能出现的异常，然后求出由于消除这些异常而得到的标准形式。在相关数据管理中引人注意的异常是更新异常。在关系中增加、删除或改变元时就可能产生这个问题。这种异常不会在相关模型管理中出现，因为模型关系中的元不以存贮方式存在；确切地说，他们是由于响应用户询问的需要才产生的。但是，其他叫做处理异常的异常将在模型管理中发现，他们的出现与相关数据管理的相似，但并不相等。

处理异常有三种类型，从它们可以得出三种类型的标准形式，叫做阿尔法型，贝塔型和舍格麻型。这三种异常是：

1. 输入异常。当用户为了需要输出被计算的模型，他必须至少输入一个输出计算并不需要的输入。输入异常类似于（但不等于）在相关数据管理中导致第二种标准形式的异常。

2. 查找异常。当在关系中有传送相关性时，例如有 { 价格 $\Rightarrow$ 销售量, 销售量 $\Rightarrow$ 生产成本 } 的形式时，这种形式需要那些想确定从假定销售量中得出的生产成本的用户输入不同的价格数值，直到销售量已经了解为止，由此可以确定相应的生产成本。把关系投影在两个关系 { 价格 $\Rightarrow$ 销售量 } 和 { 销售量 $\Rightarrow$ 生产成本 } 之中，这个投影等于相关数据管理中的第三种标准形式。

3. 输出异常。输出异常是对用户询问的不确定响应。不确定性的一个原因是不同模型中存在两个或两个以上相同的输出属性（例如，在两个模型中，两个都计算生产成本）。消除这种异常可以得到全部模型的输出属性都是两两分开的标准形式。这种标准形式不可能在相关数据管理中实现，因为某些像城市名称那样的输出特性可能与供货厂商，顾客，地区办公室等有关系。不确切性的另外一个原因是不能用投影到标准形式的办法解决的，它是在接合的实现中出现的。这个原因将说明如下。

像相关数据管理标准形式的标准形式往往在模型组设计中提供有用的指导，但是在有的情况中最好是违反这些指导。举例说，总财务仿真计算出由于实现总财务或销售策略而得到的收支范畴的数值。这些仿真包含

许多传送相关性在内，其形式有列的总数，行的总数和子总数，财务比率之类。还不明确的是用户应当把这种模型看作是二个模型，其中之一计算详细的数据，而另一个则计算总和及财务统计，总模型肯定不能用这样的方式编程。因此，如果按照像消除某种类型异常那样的一组准则构成模型组，就不应当忽略其他重要准则，如像处理效率或心理学的和美学的准则。

模型询问语言的相关完整性

在模型管理文献中都认识到适当用户接口的需要。已经提出了实现与DSS的数据和模型管理系统分开来的会话管理系统可以用来管理DSS和其用户之间的全部交互作用。但是，我们这里只涉及模型管理中相关完整性的准则，和相关完整模型询问语言的特性。

在模型管理中也有用处的相关数据管理中只有一种操作，就是选择。投影和接合的其他相关操作都不需要。不需要投影是因为唯一由模型执行的投影是其中的每一输入属性，或者是包括在投影之中，或者是由以前的选择操作规定的，因为DSS用户不愿意在不了解全部输入的情况下了解模型的输出。利用一个选择算符和一个只打印那些重要属性的报表写程序就能够完成这种投影。不需要接合操作是因为在相关模型管理中，能够由模型管理系统实现接合，因之他们对用户是透明的。在下节中还要解释这个问题。不需要设定和维持操作，因为模型关系不是联合兼容的（即模型组不包含有同一输入和输出属性的二个模型）。

除了选择（在相关模型管理中叫做“执行”）之外，还有二个相关完整性的准则。第一个是优化准则：用户标识出关系的输入属性，单独的输出属性和极大或极小标志符的无空子集。这个操作的结果是一个关系，

关系的元包含以下两个属性的优值：（1）标志出的输入属性，（2）所有剩余输入属性值的输出属性。第二个准则是灵敏性分析。这不是一种相关操作，也就是说，输出不是相关操作，而是一组输出属性相对于输入特性的灵敏性测量（例如，从另外一个属性的增量得到的一个属性的偏导数或增量）。还有更复杂的灵敏性测量，并且已有文献介绍过。

这些准则的清晰度已用于MQL的设计，但尚未实现。MQL是一种类似SEQUEL的相关完整模型询问语言。举例说，关系REL = (P, R, E · N)的输入属性是销售价格P和原料费R，而它的输出属性是获得的原料费E和纯收入N。如果原料价格是5，则计算使纯收入达到极大值的销售价格时，可写出。

```
MAXIMIZE N
OVER P
WITH (R, =, 5)
USING REL
PUT RPT
```

上面的RPT是表报编写人的名称。（在保留字下面划线。）为了求出每个R值的最优P值，我们取消了WITH语句。

更有意义的操作是嵌入优化。假定要访问这个模型（即顾客模型）的原料供货厂商想求出使E（顾客的原料费，因而也就是供货厂商的收入）达到极大值的R值，设想顾客以后将把它的销售价格定得使在那种原料价格时它的纯收入达到最大值。请求语句如下

```
MAXIMIZE E
OVER R
USING
MAXIMIZE N
OVER P
```

USING REL PUT RPT

这种类型的操作是相关映射, 由于它有自参照特性, 所以特别引人注目。这方面它类似于在相关数据管理中常常使用的非正式测试询问。“找出所有收入多于他们的管理员的雇员。”所适用的数据关系是它的元包括一个雇员标志符, 雇员的薪金和雇员管理人的标志符。这种类型的嵌入优化可以在相关数据管理中执行相似的功能, 即用作非正式测试询问来确定模型询问语言的使用是否舒适。

MQL的三种语言特性已经过鉴定: 它与上下文无关, 它是天上的明星, 虽然阶数无限, 语言的简单扩展(允许在一个单独的句子中请求语句的并置不受限制)是一阶的, 所以具有有限幂特性。这些特性表明MQL不是一种很难实现和使用的语言。

实现问题

在文献中没有详细谈到过实现问题。安全性, 完整性, 询问优化等充其量也只不过草率地提到而已。得到简短论述的一个问题是接合的实现。

在前节中, 我们用REL表示由MQL存取的关系, 这里暗中假设REL表示一个模型。但是, 它可以表示一组模型, 这组模型需要集体地响应询问。可以存取模型集合的办法有三个。第一个办法是如果模型是独立的, 即如果没有输入到任何模型, 而模型又是另一个模型的输出, 那么模型可以以任意顺序执行, 他们的输出属性值集合起来可以产生报表。第二个办法是如果至少有一个输入到一个模型, 而这个模型是另一个模型的输出, 但是由他们的输入/输出关系排序的模型形成一个部分定序的集合(即没有循环), 那么至少有一个顺序, 模型可以按照这个顺序

执行。在这两种情况中, 实现都是直截了当的: 模型依照适当的顺序执行, 而数据文件则可能用作通信的手段。

在第三种情况中需要响应询问的模型组不是部分定序的, 这可能遇到困难。举例说, 有两个经济模型, 一个是供应模型, 而一个是需求模型。供应模型的输入是供应量的向量(可能是几个地区内几种产品的向量), 而输出是价格向量, 供应厂商认为按这个价格供应这些数量是合算的, 需求模型的输入是地区内支付产品的价格向量, 而输出是以这些价格满足消费者需要的向量。一般说, 模型还有别的输入(例如影响供应的经济活动水平和/或对产品的需求), 也可能有别的输出。把一个或两个向量的假定数值(例如价格)输入适当的(需求)模型, 算出输出(需求向量), 并把这个输出输入到其他(供应)模型来计算他的输出(供货价格向量), 就可以获得产品价格和产量的一致集合。继续进行这个过程, 直到在过程结束时算出供货价格向量充分地接近于开动过程所用需求价格的集合为止。

在这种情况下, 有关接合的实现有四个问题要处理:

1. **周期识别。**第一个问题是确定在模型组中是否有周期, 如果有的话, 在求一致解时必须消去和使用什么模型属性(例如或者是价格, 或者是前例中的产量)。假如集合中只有几个模型, 可以从观察确定, 也有办法在更复杂的情况中作出决定。
2. **算法研究。**第二个问题是关于在迭代过程的每个阶段适当调整属性值的算法研究(在前面的例子中是调整被输入到需求模型中的价格的规则)。实际上, 这似乎是不困难的。上述类型的供应/需求模型配置实质上是非线性的, 利用简单的调整规则

已经求出覆盖六到十种迭代过程的模型配置。但是,有各种各样先进的定点算法,在简单规则失效时就可以使用这些算法。

3. **存在。**第三个问题是模型组一致解的存在(即定点的存在)。Brouwer 定点定理的总结说明由 R^N 的凸紧致(即闭合的和有界的)子集在它本身内的连续映射至少包含一个定点。遗憾的是这些条件在模型管理中往往不能满足。例如,生产成本可能是产量的不连续函数,布尔变量可以代表决定投资或不投资,而许多如像成本和收益这样的变量是无界的。像建立这些比较一般情况的存在条件方面做的工作不多。

4. **唯一性。**最后的问题是模型集合一致解的唯一性(即定点唯一性)。这个问题很重要,因为缺少唯一性意味着一组确定模型可能是非确定性集合。DSS的用户应当认识到,他们的决定可能有一个以上的可能输出。大家都知道的收缩映射定理是一种经典唯一性结果,它在这里似乎没有用处。较新的结果以单调性条件和曲率条件为基础(在这个条件中某个变量呈现一致按比例上升或下降),这些结果是相关的,但是这个领域内的工作还远不够完善

相关模型管理系统实现技术的情况总结最恰当的说法应当是,需要作的许多事情不止是在实现这些系统的实践中,而且还在于指导实现的理论研究中。

DSS中的信息管理

在引言中已经提到过,把模型组看作是一组关系的一个理由是这种看法最后可以

和建立数据相关的看法结合起来,从而产生DSS中信息管理的统一结构,在DSS中重要的信息可以由模型求解进程算出,或者从文件中检索出来。关于信息的用户意图(与系统实现有明显差别)。数据和模型比较如下:

 1. **组织。**虽然数据管理和模型管理的标准形式不同,但十分相似。这是出乎意料之外的,因为构成数据关系标准形式的准则,也就是消除更新异常并不适用于模型管理。即使如此,数据关系中在更新元时遇到的异常与处理模型关系中遇到的异常没有很大的差别。

2. **相关完整性。**不出所料,模型管理中相关完整性的准则不是完全像数据管理中的一样。他们唯一公用的操作是选择。可以看出,相关映射出现在数据和模型管理中,但各有不同的原因。在数据管理中它用来执行接合,而在模型管理中则用以执行嵌入优化。

人们可能要问到,相关结构是不是一种最适当的结构,在这种结构中,可以完成数据和模型管理的综合,尤其是因为 CODASYL 结构也已经扩展来支持模型描述和操作。在模型管理中,更普遍地说是在信息管理中,相关结构的好处与在数据管理中是相似的:它是简单而精巧的。相关代数的核心数学原理是自然哲学家和科学家几世纪以来了解周围世界的成果,发现这种结构也可用于了解和控制这个世界的经理人员是不足为奇的。最终确定这种结构的用途必须等到模型管理系统的实现和操作,和他们与数据管理系统的结合,但是对 DSS 中信息管理的综合来说,似乎是一个良好的起点。

杜行果 译 侯玉山 校

DSS 效用的焦点

CARL HARRINGTON

摘要

本文着重对决策支持系统 (DSS) 作出自顶向下的观察, 看看 DSS 的出现如何改善管理效果。覆盖的范畴包括信息要求的研究, 使用信息时的一致性要求, 决策规则和对目标决策的综合。

从这些范畴中改进效果是在开发 DSS 中产生的信息要求所固有的。能够对决策中改进信息一致性和规则可以改进管理效果存在任何怀疑吗? 反之, 不管计算机对运算和显示数目在计算上的优点和效益, DSS 自动地导致问题的识别, 乞求解决那些问题, 因而为达到总的目标, 在选择行动计划和程序时作出更加一致而完善的决定。

通过 DSS 所获得的管理效用

为了能够说明问题, 我选择了我认为通过决策支持系统 (DSS) 能够改进管理效果的某些比较关键性的范畴。这些范畴是通过观察决策过程, 和 DSS 作为机构整体, 而不是作为单独功能决策范畴而识别的。用这种方法观察决策过程可以显示 DSS 的许多属性, 这些属性在 DSS 被实现之前, 或者不存在, 完者接于所无效。从这个观点看来, DSS 效用的焦点是

1. 研究决策所需要的信息
2. 更多经理使用信息时的一致性
3. 通过结构良好的过程作出决策的规则

则

4. 对总目标决策的综合

在评论每个这些焦点时, 将介绍全国 DSS 改进管理的效果, 并用几条结论作为结束。

信息需要的研究

首先, 通过把注意力集中到决策所需信息的研究上就可以改进管理效果。你可以满怀信心地说要有效地管理就是要有足够的信息来管理。这就是计算机, 计算机软件 and DSS 的设计能提供改进管理效果的基础的地方。

如果有用的信息能够以可以接受的价钱买到时, 当然要利用。在计算机支持系统可供使用之前, 在作策略性的或长期的计划时, 许多需要的信息实际上不能获得。实际上也不能制定许多可供选择的行动计划来作为评价潜在效果的基础。

利用 DSS 的倘若……将会怎样和寻的