文章编号：2095-2163(2020)01-0022-06 中图分类号：TP3-0 文献标志码：A

关系数据模型中函数查询的结构特征

吴文莉[1]，刘国华[1,2]
（1 东华大学 计算机科学与技术学院，上海 201620；2 上海市数据科学重点实验室(复旦大学)，上海 201203）

摘 要：函数查询是大数据分析应用中重要的一种操作，如何准确、高效地得到查询结果是大数据环境下函数查询亟待解决的问题，函数查询的结构特征是解决该问题的基础。目前，大数据中可执行函数查询的数据模型以关系模型为主导，因此，本文重点研究关系数据模型中函数查询的结构特征。首先，对经典关系数据库中属性进行了扩充定义，在此基础上，给出函数查询的形式化定义，并用一阶语言描述函数查询，最后，证明了函数查询具有一阶查询层次结构特征。

关键词：大数据；关系数据模型；函数查询；查询结构

Structural features of functional queries in relational data model

WU Wenli[1], LIU Guohua[1,2]
(1 School of Computer Science and Technology, Donghua University, Shanghai 201620, China;
2 Shanghai Key Laboratory of Data Science(Fudan University), Shanghai 201203, China)

【Abstract】 Functional query is an important operation in big data analysis application. How to get the query results accurately and efficiently is an urgent problem to be solved in the big data environment. The structural features of functional query is the basis for solving this problem. At present, the data model of executable functional query in big data is dominated by relational model. Therefore, this paper focuses on the structural features research of functional query in relational data model. Firstly, the attributes of the classic relational database are extended. On this basis, the formal definition of the functional query is given, and the functional query is described by the first-order language. Finally, it is proved that the functional query has the feature of first-order query hierarchy.

【Key words】 big data; relational data model; functional query; query structure

0 引 言

查询(即，由数据库到关系的函数)和查询语言(即，用于表示这一函数的语言)[1]一直以来都备受人们关注。Codd 于 1970 年提出关系数据模型后，关系模型的查询及查询语言成为研究热点，出现了一批具有影响力的研究成果，如一阶关系演算以及关系代数[2-3]、合取查询[4]、表查询[5]、函数查询语言[6]等。

近年来，随着大数据地位的提升，如何在大数据环境下准确、高效地得到查询结果是函数查询亟待解决的问题，其中如何解决查询解答问题一直是人们关注的重点问题。大数据为查询解答带来了挑战，大数据查询的计算复杂性不再类似传统查询[7]。文献[7]对查询解答问题难易程度的判定提出了形式化的方法，并对预处理问题进行了研究，提出了数据驱动的预处理和查询驱动的预处理两种方法。文献[8]明确了大数据环境下查询解答问题难易程度的划分标准，对什么查询在大数据上是易处理的、如何求解大数据查询的复杂性等问题给出了一系列预处理方案，对查询解答问题的近似求解算法进行了探讨。

在大数据应用环境下，函数查询成为主要操作，如函数查询语言可以用于定义大数据上的分析查询，用其结果定义各种执行任务[9]。为了便于表示大数据上的函数查询，Nicholas 等人[9]提出一种高级函数查询语言(HIFUN)，但没有给出函数查询的形式化定义，也没有对函数查询的结构特征及复杂性问题进行研究。函数查询的结构特征是分析查询解答复杂性的基础，本文以关系数据模型为对象，对经典的关系数据库进行了扩充定义。在此基础上，给出函数查询的形式化定义，分析函数查询的可计算性，用一阶语言描述函数查询，并证明函数查询的结构是一阶查询层次结构。

1 扩充数据库及函数查询

文献[10]对数据库的结构特征进行了详细分析并且给出了经典结论，但在其数据库的定义中忽

基金项目：上海市数据科学重点实验室(复旦大学)开放课题资助。

作者简介：吴文莉(1995-)，女，硕士研究生，主要研究方向：数据库理论；刘国华(1966-)，男，博士，教授，主要研究方向：数据库、外包数据库、隐私保护。

收稿日期：2019-10-15

略了属性的描述，因此该定义不适用于函数查询的结构特征研究。本文针对文献[8]中数据库的定义进行扩充，文献[10]关于数据库的定义如下。

定义 1　数据库　令 U 是某个可数论域。数据库是元组 $B=(D, R_1, R_2, \cdots, R_k)$，其中 $D \subset U$，D 是有限的。对于每一个 $1 \leqslant i \leqslant k$，当 $a_i \geqslant 0$ 时，$R_i \subset D^{a_i}$。a_i 为 R_i 的秩，B 的类型可以看作 $\bar{a}=(a_1, a_2, \cdots, a_k)$。将向量 $R_1, R_2, \cdots, R_k$ 简写成 $\bar{R}$，将数据库写成 $B=(D, \bar{R})$。

例 1　举出一个数据库 B 的实例。论域 $U=\{2,3,4,5\}$，数据库 $B=(D, R_1, R_2)$，$D=\{2, 3, 4, 5\}$，$R_1=\{(2, 5), (4, 2), (4, 3), (5, 2)\} \subset D \times D$，$R_2=\{4, 2\} \subset D$，见表 1、表 2。$k=2$，$a_1=2$，$a_2=1$。

表 1　例 1 中的关系 R_1

Tab. 1　Relation R_1 of example 1

关系	D	D
R_1	2	5
	4	2
	4	3
	5	2

表 2　例 1 中的关系 R_2

Tab. 2　Relation R_2 of example 1

关系	D	D
R_2	4	2

以上定义的不足之处是没有给出属性的描述，为了扩充数据库的定义，把属性看作函数。给出属性的形式化定义如下。

定义 2　属性　数据库 $B=(D, \bar{R})$ 与定义 1 含义相同，B 中关系 R_i 中的每个属性都是函数。$Att=\{Att_1, Att_2, \cdots, Att_k\}$ 是个集族，Att_j 是个属性集，Att_j 中属性的个数与 R_i 的秩相同。Att_j 中每个属性(即简单属性) 表示如下：

$$A_i=\cup_{l=1}^{n}\{t_i(l,i)\}, \tag{1}$$

其中，l 表示行号，i 表示列号，t_i 是一个如下形式的函数：

$$t_i: RO \times CO \rightarrow D,$$

其中，RO 表示行号的集合，CO 表示列号的集合。由函数 t_i 确定 Att_j 中每个属性 A_i 的属性值。

$EAtt=\{EA_1, EA_2, \cdots, EA_k\}$ 是扩充属性集，扩充属性 EA_i 对应于 $\bar{R}$ 中的扩充属性，扩充属性的个数与关系 R_i 的个数相同。扩充属性表示如下：

$$EA_i=\cup_{l=1}^{n}\{et_i(D^c)\}, \tag{2}$$

其中，l 表示行号，i 表示列号，et_i 是一个如下形式的函数：

$$et_i: D^c \rightarrow U,$$

其中，D^c 表示 Att_j 中某些属性的属性值的笛卡尔积。由函数 et_i 确定属性 EA_i 的属性值。

扩充数据库的定义如下。

定义 3　扩充数据库　令 U 是某个可数论域。扩充数据库是元组 $B_f=(D, R_1, R_2, \cdots, R_k, S_1, S_2, \cdots, S_k)$，其中 $D \subset U$，D 是有限的。对于每一个 $1 \leqslant i \leqslant k$，函数集合 S_i 对应关系 R_i 中的属性的集合，当 $(a_i+1) \geqslant 1$ 时 $R_i \subset U^{a_i+1}$。(a_i+1) 为 R_i 的秩，亦是 S_i 中函数的个数。其中，前 a_i 个函数 S_{a_i} 为简单函数(即简单属性)，第 (a_i+1) 个函数 S_{a_i+1} 为复杂函数(即扩充属性)。B_f 的类型可以看作 $(\bar{a}+1)=((a_1+1), (a_2+1), \cdots, (a_k+1))$。将向量 $R_1, R_2, \cdots, R_k$ 简写成 $\bar{R}$，将向量 $S_1, S_2, \cdots, S_k$ 简写成 $\bar{S}$，将扩充数据库写成 $B_f=(D, \bar{R}, \bar{S})$。假设扩充数据库每个关系中只有一个扩充属性。

例 2　举一个扩充数据库 B_f 的实例。论域 $U=\{2, 3, 4, 5, 6, 7\}$，扩充数据库 $B_f=(D, R_1, R_2, S_1, S_2)$，其中 $D=\{2, 3, 4, 5\}$，$R_1=\{(2, 5, 7), (4, 2, 6), (4, 3, 7), (5, 2, 7)\} \subset U \times U \times U$，$R_2=\{(4, 6), (2, 3)\} \subset U \times U$，关系 R_1、R_2 分别见表 3、表 4。$S_1=\{A_1, A_2, EA_1\}$，$S_2=\{A_3, EA_2\}$。$k=2$，$(a_1+1)=(2+1)$，$(a_2+1)=(1+1)$。其中扩充属性集 $EAtt=\{EA_1, EA_2\}$。

表 3　例 2 中的关系 R_1

Tab. 3　Relation R_1 of example 2

A_1	A_2	EA_1
2	5	7
4	2	6
4	3	7
5	2	7

表 4　例 2 中的关系 R_2

Tab. 4　Relation R_2 of example 2

A_3	EA_2
4	6
2	3

下面给出函数查询的形式化定义。

定义 4　函数查询　类型为 $(\bar{a}+1) \rightarrow b$ 的函数查询是部分函数：

$Q_f: \{B_f \mid B_f$ 是类型为 $(\bar{a}+1)$ 的数据库$\} \rightarrow 2^{U^b}$，满足以下条件：

(1) Q_f 满足部分递归。

(2) 如果 $Q_f(B_f)$ 有定义，$Q_f(B_f) \subset U^b$ 且 Q_f

(B_f) 是有限的。

(3) 如果函数查询满足:函数查询是部分递归并且满足一致性条件:如果 $B_f \rightarrow^h B_f'$,那么 $Q_f(B_f') = h(Q_f(B_f))$,那么函数查询是可计算的。

补操作与组合操作是查询中的 2 个基本操作,现给出函数查询中补操作和组合操作的定义。

定义 5 补操作 Q_f 是类型为$(\bar{a}+1)$的函数查询,补函数查询 $\neg Q_f$ 与 Q_f 类型相同,定义如下:

$$\neg Q_f(D, \bar{R}, \bar{S}) = U^b - Q_f(D, \bar{R}, \bar{S}), \quad (3)$$

且当 $Q_f(D, \bar{R}, \bar{S})$ 无定义时,$\neg Q_f(D, \bar{R}, \bar{S})$ 是无定义的。对于函数查询集合 C,有:

$$\neg C = \{\neg Q_f \mid Q_f \in C\}, \quad (4)$$

例 3 已知有例 2 所示的扩充数据库 $B_f = (D, R_1, R_2, S_1, S_2)$,以及扩充数据库 B_f 上的函数查询 Q_f,如果查询结果的秩 $b = 2$,那么函数查询 Q_f 可以表示为:

$$Q_f: \{B_f = (D, R_1, R_2, S_1, S_2)\} \rightarrow 2^{(U \times U)}$$

函数查询 Q_f 的查询结果见表 5。

表 5 Q_f的查询结果

Tab. 5 The result of query Q_f

EA_1	A_3
7	4
6	2
7	2

$Q_f(B_f) = \{(7, 4), (6, 2), (7, 2)\}$

$\neg Q_f(B_f) = U^2 - Q_f(B_f) = \{(2, 2), (2, 3), (2, 4), (2, 5), (2, 6), (2, 7), (3, 2), (3, 3), (3, 4), (3, 5), (3, 6), (3, 7), (4, 2), (4, 3), (4, 4), (4, 5), (4, 6), (4, 7), (5, 2), (5, 3), (5, 4), (5, 5), (5, 6), (5, 7), (6, 3), (6, 4), (6, 5), (6, 6), (6, 7), (7, 3), (7, 5), (7, 6), (7, 7)\}$

定义 6 组合操作 $\overline{Q}_f = (Q_{f1}, Q_{f2}, \dots, Q_{fn})$ 是一组类型为$(\bar{a}+1) \rightarrow b_1, (\bar{a}+1) \rightarrow b_2, \dots, (\bar{a}+1) \rightarrow b_n$ 的函数查询序列,其中 $\bar{a} = (a_1, a_2, \dots, a_k)$。如果 Q_f 是类型为$(\bar{b}+1) = ((b_1+1), (b_2+1), \dots, (b_k+1)) \rightarrow c$ 的函数查询,那么函数查询 Q_f、$\overline{Q}_f$ 的组合 $Q_f \circ \overline{Q}_f$ 是类型为$(\bar{a}+1) \rightarrow c$ 的函数查询,定义如下:

$$Q_f \circ \overline{Q}_f(D, \bar{R}, \bar{S}) = Q_f(D, \overline{Q}_f(D, \bar{R}, \bar{S}), \bar{S}) = Q_f(D, Q_{f1}(D, \bar{R}, \bar{S}), Q_{f2}(D, \bar{R}, \bar{S}), \dots, Q_{fn}(D, \bar{R}, \bar{S}), \bar{S}). \quad (5)$$

且当函数查询 Q_f 或者函数查询 $\overline{Q}_f$ 无定义时,$Q_f \circ \overline{Q}_f$ 是无定义的。如果 C_1, C_2 表示函数查询集合,那么:

$C_1 \circ C_2 = \{Q_f \circ \overline{Q}_f \mid Q_f \in C_1$,且如果 $\overline{Q}_f = (Q_{f1}, Q_{f2}, \dots, Q_{fn})$,那么 $Q_{fi} \in C_2\}$

例 4 举出一个扩充数据库 B_f 的实例。论域 $U = \{2, 3, 4, 5, 6, 7, 12, 15\}$,扩充数据库 $B_f = (D, R_1, R_2, R_3, S_1, S_2, S_3)$,其中 $D = \{2, 3, 4, 5, 6\}$,$R_1 = \{(2, 5, 7), (4, 2, 6), (4, 3, 7), (5, 2, 7)\} \subset U \times U \times U$,$R_2 = \{(2, 4, 6), (4, 2, 3)\} \subset U \times U \times U$,$R_3 = \{(3, 5, 15), (2, 6, 12)\} \subset U \times U \times U$,关系 R_1、R_2、R_3 分别见表 6 ~ 表 8。$S_1 = \{A_1, A_2, EA_1\}$,$S_2 = \{A_1, A_3, EA_2\}$,$S_3 = \{A_2, A_4, EA_3\}$。即 $k = 3$,$(a_1 + 1) = (a_2 + 1) = (a_3 + 1) = (2 + 1)$。

表 6 例 4 中的关系 R_1

Tab. 6 Relation R_1 of example 4

A_1	A_2	EA_1
2	5	7
4	2	6
4	3	7
5	2	7

表 7 例 4 中的关系 R_2

Tab. 7 Relation R_2 of example 4

A_1	A_3	EA_2
2	4	6
4	2	3

表 8 例 4 中的关系 R_3

Tab. 8 Relation R_3 of example 4

A_2	A_4	EA_3
3	5	15
2	6	12

已知有如上扩充数据库 B_f, B_f 上的函数查询 Q_f, Q_{f1}, Q_{f2},其类型分别为$(1,1) \rightarrow 2$,$(2+1) \rightarrow 1$,$(2+1) \rightarrow 1$,$\overline{Q}_f = (Q_{f1}, Q_{f2})$,则 $Q_f \circ \overline{Q}_f = \{(4, 15), (2, 12)\}$,$Q_f \circ \overline{Q}_f$ 的查询结果见表 9。

表 9 $Q_f \circ \overline{Q}_f$的查询结果

Tab. 9 The result of query $Q_f \circ \overline{Q}_f$

A_3	EA_3
4	15
2	12

2 函数查询层次结构

查询语言是用来描述查询的工具[11],为了从理论上研究查询问题,人们通常使用一阶语言描述查询[10]。本文的研究也是基于一阶语言,下面给出描

述函数查询的一阶查询语言的定义。

定义 7　一阶语言　L 是没有函数符号，具有等式的一阶语言，$R_1, R_2, \cdots$ 作为关系符号，其中 R_i 是具有扩充属性的关系，使用符号 R_i 表示关系及作为关系本身，关系 R_i 的元数隐含在上下文中。FO 表示由以下表达式组成的语言：

$$\bar{x}.\bar{R}.\Psi, \tag{6}$$

其中，Ψ 是 L 中的公式；$\bar{x}$ 是不同的变量向量，包括所有在 Ψ 中出现的自由变量；$\bar{R}$ 是不同的谓词符号，包括所有出现在 Ψ 中的关系。

当 $|\bar{x}| = b, R_i$ 的秩为 $(a_i + 1)$ 时，FO 中的表达式 $\bar{x}.(R_1, R_2, \cdots, R_k).\Psi(\bar{x})$ 表示 $(\bar{a} + 1) \to b$ 类型的函数查询 Q_f。函数查询 Q_f 定义为 $Q_f(D, \bar{R}, \bar{S}) = \{\bar{d} \in U^b \mid \Psi(\bar{d})$ 在 $Q_f(D, \bar{R}, \bar{S})$ 中为真$\}$。

例 5　如下表达式：

$(x, s_2)(R_1, R_2)(\exists y)(R_1(x, y, s_1) \wedge R_2(y, s_2))$

表示类型为 $((2+1), (1+1)) \to 2$ 的查询。

定义 8　一阶查询　Q_{fM} 记为由 M 表示的函数查询，$M \in FO, Q_{fW} = \{Q_{fM} \mid M \in W\}, W \subset FO$。集合 Q_{FO} 是一阶查询的集合，并用 F 表示。

定义 9　否定操作　FO 中的形如 $\bar{x}.\bar{R}.\Psi$ 的表达式 M，其否定 $\neg M$ 记为 $\bar{x}.\bar{R}.\neg \Psi$；$\neg W = \{\neg M \mid M \in W\}, W \subset FO$。

例 6　令 M 为 $(x, s_2)(R_1, R_2)(\exists y)(R_1(x, y, s_1) \wedge R_2(y, s_2))$，则 $\neg M$ 为 $(x, s_2)(R_1, R_2)(\forall y) \neg (R_1(x, y, s_1) \vee R_2(y, s_2))$。

引理 1　对于任意 $M \in FO$，有 $Q_{f(\neg M)} = \neg Q_{fM}$，对于任意 $W \subset FO$，有 $Q_{f(\neg W)} = \neg Q_{fW}$。

同样地，定义替换操作类比函数查询中的组合操作。

定义 10　替换操作　$M = \bar{x}.(T_1, T_2, \cdots, T_k).\Psi$，其中 T_i 的秩为 $(a_i + 1)$。$\bar{N} = (N_1, N_2, \cdots, N_n)$，其中 $N_i = y_i.(R_1, R_2, \cdots, R_k).\Phi_i, R_i$ 的秩为 $(a_i + 1), |\bar{y}_i| = (a_i + 1)$。$M \circ \bar{N}$ 表示表达式 $\bar{x}.(R_1, R_2, \cdots, R_k).\Psi'$，其中同时用 Φ_i 替换 Ψ 中的 T_i 得到 Ψ'，通过重命名受限变量或相等变量，将 y_i 与 T_i 中出现的参数进行匹配。

例 7　令 $M = s_1.(T_1, T_2).(\exists y)(T_1(x, y, s_1) \wedge T_2(y, s_2)), N_1 = (y, s_4, z).(R_1, R_2).(\forall w)(R_1(y, z, s_3) \vee R_2(w, z, s_4)), N_2 = (u, s_6).(R_1, R_2).(\exists y)(R_1(u, u, s_5) \wedge R_2(u, y, s_6))$，那么 $M \circ \bar{N} = s_1.(R_1, R_2).(\exists y)((\forall w)(R_1(s_1, s_1, s_3) \vee R_2(w, s_1, y)) \wedge (\exists z)(R_1(s_1, s_1, s_5) \wedge R_2(s_1, z, y)))$。

定义 11　对于集合 $W, V \subset FO$，其组合操作记为 $W \circ V = \{M \circ (N_1, N_2, \cdots, N_n) \mid M \in W, N_i \in V, 1 \leqslant i \leqslant k\}$。

易证得出研究引理，详见如下。

引理 2　对于任意 $M, (N_1, N_2, \cdots, N_n) \in FO$，有 $Q_{f(M \circ (N_1, \cdots, N_n))} = Q_{fM} \circ Q_{f(N_1, \cdots, N_n)}$。对于任意集合 $W, V \subset FO$，有 $Q_{f(W \circ V)} = Q_{fW} \circ Q_{fV}$。

定义 12　存在查询　EX 表示如下形式的表达式集合：

$$\bar{x}.\bar{R}.(\exists \bar{y}).\Psi,$$

其中，$\bar{R}$ 是具有扩充属性的关系，Ψ 是无量词的，$E = Q_{EX}$ 表示存在性函数查询的集合。

引理 3　对于任意函数查询集合 C，有：

$$C \cup \neg C \subset E \circ C = E \circ \neg C = E \circ (C \cup \neg C)$$

证明　函数查询是函数。令 C_1 为集合 C 中的任一函数查询，其定义域为 D_1，值域为 Rn_1，则 $\neg C_1$ 的定义域为 D_1，值域为 $\neg Rn_1$，令 E_1 为集合 E 中的任一存在性函数查询，其定义域为 D_2，值域为 Rn_2。那么有：

$(E_1 \circ C_1): D_1 \to Rn_2$，

$(E_1 \circ \neg C_1): D_1 \to Rn_2$，

$(E_1 \circ (C_1 \cup \neg C_1)): D_1 \to Rn_2$，

综上所述，引理 3 成立。

本文将属性视为函数，对数据库重新定义，经过研究发现，函数查询的层次结构与多项式时间层次结构[12]、一阶查询层次结构[10]等已知层次结构相似。

下面给出表达式集合的层次结构的定义，由此引出函数查询集合的层次结构。

定义 13　表达式层次结构　表达式集合 FO 的集合 $\{\sum_i, \Gamma_i\}_{i<\omega}$ 定义如下：

(1) $\sum_0 = \{\bar{x}.\bar{R}.\Psi \mid \Psi$ 中无量词$\}$，

(2) $\sum_{i+1} = EX \circ \Gamma_i$，

(3) $\Gamma_i = \neg \sum_i$，

其中，关系 $\bar{R}$ 具有扩充的属性的关系。表达式集合 $\sum_i, \Gamma_i$ 对应的函数查询集合记为 $\sum_i^{Q_f}, \Gamma_i^{Q_f}$。

定义 14　一阶查询层次结构　函数查询集合的集合 $\{\sum_i^{Q_f}, \Gamma_i^{Q_f}\}_{i<\omega}$，称为一阶查询层次结构，其中 $\sum_i^{Q_f} = Q_{f\sum_i}, \Gamma_i^{Q_f} = Q_{f\Gamma_i}$。

由引理 1~3 可以得出：

定理 1　(1) $\Gamma_i^{Q_f} = \neg \sum_i^{Q_f}$，(2) $\sum_{i+1}^{Q} = E \circ \Gamma_i^{Q_f} = E \circ \sum_i^{Q_f}$，(3) $\sum_i^{Q_f} \cup \Gamma_i^{Q_f} \subset \sum_{i+1}^{Q_f} \cap \Gamma_{i+1}^{Q_f}$

引理 4 令 $\exists_0$(与 $\forall_0$ 相等)记为一阶语言 L 中无量词的表达式集合,$\exists_{i+1}$(分别为 $\forall_{i+1}$)记为 $(\exists\bar{x}).\Psi$(则对应于 $(\forall\bar{x}).\Psi$)形式的表达式集合,其中 Ψ 在 $\forall_i$(对应于 $\exists_i$)中且 $\bar{x}$ 在 Ψ 中是自由的,那么:

(1) 对于 $\exists_i$(分别为 $\forall_i$)中 Ψ 的,$\bar{x}.\bar{R}.\Psi$ 在 $\sum_i$(对应于 Γ_i)中。

(2) $\sum_i^{Q_f}$(分别为 $\Gamma_i^{Q_f}$)中的任意函数查询可以用 $\bar{x}.\bar{R}.\Psi$ 表示,其中 Ψ 在 $\exists_i$(对应于 $\forall_i$)中。

其中,关系 $\bar{R}$ 具有扩充的属性的关系,$\sum_i^{Q_f}$ 类(分别为 $\Gamma_i^{Q_f}$ 类)表示具有前束范式形式的一阶查询,伴随着 i 次量词交替,并且以存在量词(对应于任意量词)开头[10]。

证明 (1) 当 $i=0$ 时,$\exists_0$ 为一阶语言 L 中无量词的表达式集合,$\sum_0$ 为 $\{\bar{x}.\bar{R}.\Psi \mid \Psi$ 无量词$\}$,由定义可知 $i=0$ 时成立。令 Ψ 表示 $(\exists\bar{x}_1)(\forall\bar{x}_2)\cdots(\Theta\bar{x}_{i+1}).\Phi$($\Theta$ 表示$\exists$或者$\forall$),$\Psi\in\exists_{i+1}$。令 N 表示 $\bar{x}$,$\bar{x}_1.\bar{R}.(\forall\bar{x}_2)\cdots(\Theta\bar{x}_{i+1}).\Phi$,$\Phi\in\exists_i$。假设 N 在 Γ_i 中,令 M 表示 $\bar{x}.T.(\exists\bar{x}_1)T.(\bar{x},\bar{x}_1)$,则 M 在 EX 中,$M\circ N$ 在 $\bar{x}.\bar{R}.\Psi$ 中,由定义可知 $M\circ N$ 在 $\sum_{i+1}$ 中。由公式 $\forall_{i+1}$ 推导出 Γ_i 中的表达式的证明类似。

(2) 由定义可知 $i=0$ 时成立。令 Q_f 为 $\sum_{i+1}^{Q_f}$ 中的函数查询,$Q_f=M\circ(N_1,N_2,\cdots,N_n)$,其中 $M\in EX$,M 表示 $\bar{x}.\bar{T}.(\exists\bar{x}_1)\Psi(\bar{T})$,$N_j$ 在集合 Γ_i 中,N_j 表示为 $\bar{y}.\bar{R}.(\forall\bar{y}_{j,1})(\exists\bar{y}_{j,2})\cdots(\Theta\bar{y}_{j,i}).\Phi_j$,其中 $\bar{T}=(T_1,T_2,\cdots,T_n)$,$\Psi$ 和 Φ_j 是无量词的。对于每一个 j,用 $(\forall\bar{y}_{j,1})(\exists\bar{y}_{j,2})\cdots(\Theta\bar{y}_{j,i}).\Phi_j$ 替换 Ψ 中的 T_j,将否定提前使得表达式为前束范式形式,可以得到 Q_f 的表达式为 $\bar{x}.\bar{R}.(\exists\bar{x}_1)\Phi$,其中 Φ 在 $\exists_i$ 中。即 Q_f 可以表示为 $\bar{x}.\bar{R}.\Psi$,其中 Ψ 在 $\exists_{i+1}$ 中。

关系查询语言主要有 2 种类型,一种是逻辑语言,比如关系演算,由公式组成;另一种为代数语言,比如关系代数,由程序组成,其基本操作是代数(如连接和投影)[13]。文献[14]证明了关系演算与关系代数在语言表达能力上的等价性。因此,一阶查询层次结构同样可以对前述定义的集合进行投影来定义。如果用 P 表示如下形式的表达式的投影查询集合:

$$\bar{x}.R.(\exists\bar{y})R(\bar{x},\bar{y}), \tag{7}$$

那么,可以得出:

引理 5 $\sum_{i+1}^{Q_f}=P\circ\Gamma_i^{Q_f}$,并且当 $i\geqslant 1$ 时,$P\circ\sum_i^{Q_f}=\sum_i^{Q_f}$。

由引理 4 还可以得出结论:一阶层次结构可以精确描述一阶查询集合 F。

定理 2 $\cup_i\sum_i^{Q_f}=\cup_i\Gamma_i^{Q_f}=F$

组合可以看作是执行查询、保存查询结果中间值的一种方式,任何可以将查询结果存储在数据库中的查询语言,都可以计算一阶查询[10]。组合是文献[15]中计算所有一阶查询的方式,因此组合具有文献[2]中提到的完备性。

在复杂性理论[12]中,一阶查询层次结构与多项式时间层次结构$\{\sum_i^P,\Gamma_i^P\}$之间存在某种联系。

定理 3 对于任意的 i 以及扩充数据库 $B_f=(D,R_1,R_2,\cdots,R_k,S_1,S_2,\cdots,S_k)$,其中 B_f 中的关系 $R_i\neq\{\}$ 并且 $R_i\neq D^{a_i+1}$,集合 $\{\bar{d},N\mid N\in\sum_i$ 并且 $\bar{d}\in Q_{fN}(B_f)\}$ 在 $\sum_i^P$ 中是完备的。

证明 给定数据库,$\sum_i^P$ 中函数查询解答的复杂度与函数查询表达式的长度有关[10]。令 $G=\{\bar{d},N\mid N\in\sum_i$ 并且 $\bar{d}\in Q_{fN}(B_f)\}$

(1) 首先证明 $G\in\sum_i^P$。如引理 4(2)的证明,任意 $N\in\sum_i$,N 可以转化为等价的表达式 $\bar{y}.\bar{R}.(\forall\bar{y}_{1N})(\exists\bar{y}_{2N})\cdots(\Theta\bar{y}_{iN}).\Phi(\bar{y},\bar{y}_{1N},\cdots,\bar{y}_{iN})$,其中关系 $\bar{R}$ 具有扩充的属性的关系,Φ 是无量词的,并且转化过程中其符号数量没有增加。那么当且仅当(存在适当的多项式 $poly$):

$$\Theta s_i.\|s_i\|\leqslant poly(\|(\bar{d},N)\|). \tag{8}$$

成立时,$(\bar{d},N)\in G$ 成立。其中 s_j 是编码扩充数据库 B_f 中活动域 D[16] 上的向量 $\bar{d}$ 的位串。$\|s_i\|$、$\|(\bar{d},N)\|$ 分别表示 s_i、$(\bar{d},N)$ 的编码长度(以位为单位)。该表达式表示的函数查询在多项式时间内可计算,即 $G\in\sum_i^{Q_f}$。

(2)通过将量化的布尔公式规约到 G 来证明在 $\sum_i^P$ 中是完备的。

令 T 是 B_f 中的关系,$T\neq\{\}$ 且 $T\neq D^{a_i+1}$。给定如下形式的量化布尔公式 Ψ:

$$(\exists P_{1,1},P_{1,2},\cdots,P_{1,k1})(\forall P_{2,1},P_{2,2},\cdots,P_{2,k2})\cdots(\Theta P_{i,1},P_{i,2},\cdots,P_{i,ki})\Phi(P_{1,1},P_{2,2},\cdots,P_{i,ki}),$$

其中,$P_{i,ki}$ 是命题符号。当且仅当 Ψ 为真时,$((),N)\in G$ 成立。其中 N 为表达式:$().\bar{R}.(\exists\bar{x}_{1,1},$

$\bar{x}_{1,2}$，…，$\bar{x}_{1,k1}$）（$\forall \bar{x}_{2,1}$，$\bar{x}_{2,2}$，…，$\bar{x}_{2,k2}$）…（$\Theta \bar{x}_{i,1}$，$\bar{x}_{i,2}$，…，$\bar{x}_{i,ki}$）Φ（$T(\bar{x}_{1,1})$，$T(\bar{x}_{2,2})$，…，$T(\bar{x}_{i,ki})$，$N \in \sum_i$。由量化布尔公式的完备性[12]可以得出 G 在 $\sum_i^P$ 中的完备性[10]。

当扩充数据库 B_f 中的关系 $R_i = \{\}$ 或者 $R_i = D^{a_i+1}$ 时，对于任意 i，集合$\{\bar{d}, N \mid N \in \sum_i$ 并且 $\bar{d} \in Q_{fN}(B_f)\}$ 在多项式时间内是可计算的。

定理 4　对于任意 i，$\sum_i^{Q_f} \subset \neq \sum_{i+1}^{Q_f}$。

类似证明见文献[10,17]。

定理 5　对于任意 $i \geqslant 1$，$\sum_i^{Q_f} \neq \Gamma_i^{Q_f}$。

证明　对任意 $i \geqslant 1$，假设 $\sum_i^{Q_f} = \Gamma_i^{Q_f}$，$\sum_{i+1}^{Q_f} = P \circ \Gamma_i^{Q_f} = P \circ \sum_i^{Q_f} = \sum_i^{Q_f}$。与定理 4 矛盾，假设不成立。

文献[18]从类型角度给出相似证明，其证明 2 个长度相同的量化前缀在有限数据库上没有相同的表达能力。例如公式∀∃∀∀∀∃逻辑上不等于任何公式∀∃∃∀∀∃，反之亦然。

定理 6　对于任意 i，$\sum_i^{Q_f} \cup \Gamma_i^{Q_f} \subset \neq \sum_{i+1}^{Q_f} \cap \Gamma_{i+1}^{Q_f}$。

证明　结合定理 1(3)，这里主要证明 $\sum_i^{Q_f} \cup \Gamma_i^{Q_f} \neq \sum_{i+1}^{Q_f} \cap \Gamma_{i+1}^{Q_f}$。由定理 5 可知，已知有在 $\sum_i^{Q_f}$ 中而不在 $\Gamma_i^{Q_f}$ 中的函数查询 Q_f，Q_f 表示为 $\bar{x}.\bar{R}.\Phi$。令 T 是新的 0 元谓词符号，函数查询 Q_f' 表示为 $\bar{x}.(\bar{R}, T).(T \wedge \Psi) \vee (\neg T \wedge \neg \Psi)$。由引理 4 可知，$Q_f'$ 在 $\Gamma_{i+1}^{Q_f}$ 中，但不在 $\sum_i^{Q_f} \cup \Gamma_i^{Q_f}$ 中。因为如果 Q_f' 在 $\sum_i^{Q_f}$ 中，且表示为 $\bar{x}.(\bar{R},T).\Phi(T)$，那么 $\neg Q_f$ 可以表示为 $\bar{x}.\bar{R}.\Phi$，推出 $\neg Q_f \in \sum_i^{Q_f}$，与已知矛盾。

3　结束语

大数据环境下函数查询解答的复杂度问题是制约大数据查询的瓶颈，解决该问题的关键首先是了解函数查询的层次结构特征。本文的研究成果为下一步研究基于函数查询结构特征的查询解答复杂度分析奠定理论基础。

参考文献

[1] KUHNS J L. Answering questions by computer: a logical study: RM-5428-PR[R]. USA: Rand Corporation, 1967.

[2] CODD E F. Relational completeness of data base sublanguages [M]//RUSTIN R. Data Base Systems. Englewood Cliffs, NJ: Prentice-Hall, 1972:65-98.

[3] GALLAIRE H, MINKER J. Logic and data bases[M]. New York, Plenum Press, 1978.

[4] CHANDRA A K, MERLIN P M. Optimal Implementation of conjunctive queries in relational data bases[C]// Proceedings of 9th ACM Symposium on Theory of Computing. Boulder, Colorado:ACM, 1977: 77-90.

[5] AHO A V, SAGIV Y, ULLMAN J D. Equivalences among relational expressions [J]. Society for Industrial and Applied Mathematics Journal on Scientific and Statistical Computing, 1979, 8(2): 218-246.

[6] BUNEMANP, FRANKEL R E. Fql- a functional query language: a preliminary report[C]//ACM SIGMOD Conference. Boston, Massachusetts:ACM, 1979: 52-58.

[7] FAN Wenfei, GEERTS F, NEVEN F. Making queries tractable on big data with preprocessing [C]//The 39th International Conference on Very Large Data Bases. Italy: Curran Associates, Inc, 2013, 6(9):685-696.

[8] FAN Wenfei, HUAI Jinpeng. Querying big data: bridging theory and practice[J]. Journal of Computer Science and Technology, 2014, 29(5): 849-869.

[9] NICOLAS S, SUGIBUCHI T. Hifun-a high level functional query language for big data analytics [J]. Journal of Intelligent Information Systems, 2018,51(3): 529-555.

[10] CHANDRA A K, HAREL D. Structure and complexity of relational queries[J]. Journal of Computer and System Science, 1982, 25(1):99-128.

[11] ZHAO Wenfeng, LIU Guohua, CHEN Zhao, et al. Querying big data from a database perspective[C]//The 2017 4th International Conference on Systems and Informatics(ICSAI 2017). Hangzhou, China:IEEE, 2017: 1433-1437.

[12] STOCKMEYER L J. The polynomial - time hierarchy [J]. Theoretical Computer Science, 1977, 3(1): 1-22.

[13] VARDI M. The complexity of relational query languages[C]// Proceedings of 14th ACM Symposium on Theory of Computing. San Francisco, California, USA :ACM, 1982: 137-146.

[14] KLUG A. Equivalence of relational algebra and relational calculus query languages having aggregate functions[J]. Journal of the ACM, 1982,29(3): 699-717.

[15] ZLOOF M M. Query by example: Operations on the transitive Closure: RC5526[R]. USA: IBM Research Yorktown Heights, 1976.

[16] ABITEBOUL S, HULL R, VIANU V. Foundations of databases [M]. The United Kingdom:Addison-Wesley, 1995.

[17] ROGERS H J. Theory of recursive functions and effective computability[M]. New York: McGraw-Hill, 1967.

[18] KEISLER H J, WALKOE W J. The diversity of quantifier prefixes[J]. Symbolic Logic, 1973, 38(1) :79-85.

[19] SPYRATOS N. A functional model for data analysis[C]// Proceedings of the 7th International Conference on Flexible Query Answering Systems. Milan, Italy:Springer, 2006, :51-64.

[20] CHANDRA A K. Theory of database queries[C]//7th ACM Symposium on Principles of Database Systems. New York:ACM, 1988: 1-9.

[21] CHANDRA A K, HAREL D. Computable queries for relational data bases[J]. Journal of Computer and System Sciences, 1980, 21(2): 156-178.

[22] AHO A V, ULLMAN J D. Universality of data retrieval languages [C]//Proceedings of 6th ACM Symposium on Principles of Programming Languages. San Antonio, Texas:ACM, 1979: 110-117.