

基于本体的极限与配合数据访问^①

黄 星¹, 钟艳如¹, 黄美发², 覃裕初¹

¹(桂林电子科技大学 计算机科学与工程学院, 桂林 541004)

²(桂林电子科技大学 机电工程学院, 桂林 541004)

摘 要: 为使极限与配合知识既能满足知识重用和语义互操作的需求, 又具有较高的数据访问效率, 以共享概念和数据存储两层结构组织极限与配合知识. 以极限与配合本体作为共享概念层, 以关系型数据库存储极限与配合数据, 以映射连接极限与配合本体和数据库, 从而可通过极限与配合本体直接访问极限与配合数据. 在满足极限与配合知识重用和语义互操作需求的同时, 保证了极限与配合数据的访问效率. 通过原型系统和基于该系统的工程实例, 验证了基于本体的极限与配合数据访问的可行性和有效性.

关键词: 极限与配合; 描述逻辑; OBDA; 本体; 数据库

Ontology-Based Limits and Fits Data Access

HUANG Xing¹, ZHONG Yan-Ru¹, HUANG Mei-Fa², QIN Yu-Chu¹

¹(School of Computer Science and Engineering, Guilin University of Electronic Technology, Guilin 541004, China)

²(School of Mechanical and Electrical Engineering, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: To meet both the needs of reusability and semantic interoperability of limits and fits knowledge and the needs of limits and fits data access performance, the tier structure of shared concepts and data storage is introduced to organize the limits and fits knowledge. Considering the limits and fits ontology as the shared concepts, we manage the limits and fits data with relational database management system and express the relationship between the data and the concepts by mappings. Then the limits and fits data can be accessed through the limits and fits ontology. In this way, the needs of limits and fits knowledge reusability and semantic interoperability are met, while ensuring the efficiency of limits and fits data access. The feasibility and effectiveness of ontology based limits and fits data access are demonstrated by a prototype system and a practical example.

Key words: limits and fits; DL; OBDA; ontology; database

当前, 主要的计算机辅助设计(Computer Aided Design, CAD)系统在对公差信息的处理上仍然不够理想, 在 CETOL 6 SigmaTM、eM-TolMateTM、VSA-GDTTM、VSA.3DTM等软件中, 公差类型和公差值均需要设计者手工指定. 为了改进 CAD 系统对公差信息的处理, 文献[1]和[2]分别研究了基于描述逻辑和基于本体的公差类型自动生成方法, 然而公差值的自动生成问题仍然没有得到解决. 本体能够实现真正意义上的知识共享、重用以及语义互操作^[3], 这使得基于本体的公差值自动生成成为可能. 为了实现基于本体

的公差值的自动生成, 首先要让各种涉及公差信息的本体能够快速高效地访问极限与配合^[4]数据, 然而目前尚无极限与配合数据本体化的文献可参考. 为此, 本文试图采用一种即满足本体语义需求又满足效率需求的方法, 实现极限与配合的数据访问.

极限与配合包含的标准公差、基本偏差、公差带及配合等数据结构庞杂且规模庞大, 难以直接构建包含这些数据的极限与配合本体的断言集(ABox). 为了在满足极限与配合的知识重用性和数据访问效率的同时, 免除直接构建极限与配合本体 ABox, 本文采用共

① 基金项目:国家自然科学基金(61163041);广西可信软件重点实验室基金(kx201120)

收稿时间:2014-02-24;收到修改稿时间:2014-03-18

享概念和数据存储两层结构组织极限与配合知识^[5]. DL-Lite_A 本体^[6]查询推理能力出色, 是极限与配合共享概念层最理想的选择. 关系型数据库作为目前数据存储的主流技术, 最适合极限与配合中数据存储层的维护. 本文通过使用 DL-Lite_A 本体和关系型数据库两层结构对极限与配合知识进行维护, 以映射连接极限与配合本体和数据库, 在满足极限与配合知识的重用和语义互操作需求的同时, 保证了极限与配合数据的访问效率.

全文框架如下: 首先, 介绍本文将涉及的基础概念; 其次, 分析极限与配合本体术语集(TBox)和极限与配合数据库关系模式; 再次, 讨论极限与配合数据库和极限与配合本体的连接; 最后, 开发原型系统并通过工程实例验证了基于本体的极限与配合数据访问的可行性和有效性.

1 相关概念

1.1 DL-Lite_A 本体

DL-Lite_A 包含原子概念(A)、基本概念(B)、一般概念(C)、全局概念(T_C)、原子值域(D)、基本值域(E)、一般值域(F)、全局值域(T_D)、原子角色(P)、基本角色(Q)、一般角色(R)、原子概念属性(U_C)、一般概念属性(V_C)等.

在 DL-Lite_A 中, 使用 δ 表示定义域, 使用 ρ 表示值域. Γ 表示常量集, 它包含对象常量 Γ_O 和值常量 Γ_V 两部分. 一个 DL-Lite_A 本体 $O = \langle T, A \rangle$, 其中 T 和 A 分别表示 TBox 和 ABox. TBox 是包涵断言 $B \subseteq C$ 、 $Q \subseteq R$ 、 $E \subseteq F$ 、 $UC \subseteq VC$ 、(funct Q)、(funct UC)等的集合. DL-Lite_A 本体的 ABox 是断言 $A(a)$ 、 $P(a,b)$ 、 $UC(a,b)$ 等的集合, a 、 b 和 c 是常量集 Γ 中的元素. DL-Lite_A 本体具有理想的查询推理效率^[7].

1.2 DL-Lite_A 本体的语义

DL-Lite_A 中的一个解释 $I = (\Delta^I, \cdot^I)$ 由非空解释域 Δ^I 和解释函数 \cdot^I 组成^[8]. 其中 Δ^I 由非空对象域 Δ_O^I 和非空值域 Δ_V^I 组成. Δ_V^I 是 $val(T_1), \dots, val(T_n)$ 的并, 这里 T_i 是值域, $val(T_i)$ 是 T_i 在 I 中对应的解释. \cdot^I 为 Γ 中的每个常量 c 分配一个 Δ^I 中的元素, 为每个概念和值域分配一个 Δ^I 的子集, 为每一个角色和概念属性分配一个 $\Delta^I \times \Delta^I$ 的子集. DL-Lite_A 使用唯一命名假设 (Unique Name Assumption), 即若有 Γ 中两个不同的常量 $c_1 \neq c_2$, 则有 $c_1^I \neq c_2^I$.

1.3 本体查询

一个本体 O 的连接查询(CQ) q 是形如 $q(x) \leftarrow \beta(x,y)$ 的表达式^[8]. 其中 x 是不同变量的元组, 叫做区别变量; y 是不出现在 x 中的变量的元组, 叫做无区别变量. $q(x)$ 叫做查询头. $\beta(x,y)$ 叫做查询体, 它在 DL-Lite_A 本体中是形如 $A(x)$ 、 $D(x)$ 、 $P(x,y)$ 、 $UC(x,y)$ 或 $x=y$ 的原子的连接, x 、 y 是 x 或 y 中的变量, 或者 Γ 中的常量. 一个联合连接查询(UCQ)形如 $q(x) \leftarrow \bigvee \beta(x,y_i)$, 它是查询头相同的 CQ 的一个集合. 对于某个 CQ 的查询体 $\beta(z)$, v 是与 z 参数数量相同的常量组, 则通过将 z 的每个变量用 v 中常量进行替换得到的 $\beta[z/v]$ 叫做基础实例.

1.4 数据库

本文中涉及的数据库(DB)为关系型数据库 $DB = \langle R, I \rangle$, 其中 R 表示关系模式, I 表示 R 的一个实例. 数据库 DB 的值常量集合 (Γ_{DB}) 表示所有出现在 I 中的数据的集合. 数据库 DB 的关系模式 R 的一个 SQL 查询(φ)表示由其实例 I 到一个数据元组的映射.

2 极限与配合本体的TBox

极限与配合由一系列 GPS 规范来保证, 它所涉及的国家标准主要包括 GB/T 1800-2009、GB/T 1801-2009、GB/T 5371-2004、GB/T 1803-2003 及 GB/T 1804-2000. 本文主要讨论公差精度中的标准公差等级、标准公差值及公称尺寸之间的关系, 孔的公差带中基本偏差代号、基本偏差值、标准公差等级及公称尺寸之间的关系, 轴的公差带中基本偏差代号、基本偏差值、标准公差等级及公称尺寸之间的关系. 另外本文还涉及孔公差带及其优先级、轴公差带及其优先级、孔轴配合的优先级及配合种类, 这些内容是配合选择的基础.

为了使极限与配合本体 OLF 具有较高的查询推理效率, 使用描述逻辑 DL-Lite^A 对所捕获到的极限与配合中的概念、概念结构、概念间属性及概念自身属性进行描述. 下文中, 所有概念名均使用大写首字母, 角色名使用 has 起始的斜体, 概念属性使用 has 起始的正体, 值域使用非 has 起始的小写正体. 于是极限与配合本体的 TBox 主要包含下述断言:

$$\begin{aligned} \text{NoIFS} &\subseteq \text{FitSystem}, \text{StdFS} \subseteq \text{FitSystem}, \text{Hole} \subseteq \text{Part}, \\ \text{Shaft} &\subseteq \text{Part}, \text{HoleTZ} \subseteq \text{TZone}, \text{ShaftTZ} \subseteq \text{TZone}, \\ \text{StdFS} &\subseteq \exists \text{ hasFit}, \text{StdFS} \subseteq \exists \text{ hasHole}, \\ \text{StdFS} &\subseteq \exists \text{ hasShaft}, \exists \text{ hasFit}^- \subseteq \text{InterFit}, \\ \exists \text{ hasHole}^- &\subseteq \text{Hole}, \exists \text{ hasShaft}^- \subseteq \text{Shaft}, \end{aligned}$$

$InterFit \sqsubseteq \exists hasHoleTZ$, $InterFit \sqsubseteq \exists hasShaftTZ$,
 $Hole \sqsubseteq \exists hasHoleTZ$, $Shaft \sqsubseteq \exists hasShaftTZ$,
 $hasHoleTZ^- \sqsubseteq HoleTZ$, $\exists hasShaftTZ^- \sqsubseteq ShaftTZ$,
 $StdFS \sqsubseteq \delta(hasLNS)$, $StdFS \sqsubseteq \delta(hasMaxC)$,
 $StdFS \sqsubseteq \delta(hasMinC)$, $\rho(hasLNS) \sqsubseteq xsd:decimal$,
 $\rho(hasMaxC) \sqsubseteq xsd:decimal$, $\rho(hasMinC) \sqsubseteq xsd:decimal$,
 $Hole \sqsubseteq \delta(hasHFDI)$, $\rho(hasHFDI) \sqsubseteq xsd:string$,
 $Shaft \sqsubseteq \delta(hasSFDI)$, $\rho(hasSFDI) \sqsubseteq xsd:string$,
 $HoleTZ \sqsubseteq \delta(hasHFDI)$, $ShaftTZ \sqsubseteq \delta(hasSFDI)$,
 $InterFit \sqsubseteq \delta(hasPre)$, $InterFit \sqsubseteq \delta(hasBasis)$,
 $InterFit \sqsubseteq \delta(hasFitType)$, $\rho(hasPre) \sqsubseteq xsd:string$,

$\rho(hasBasis) \sqsubseteq xsd:string$, $\rho(hasFitType) \sqsubseteq xsd:string$,
 $Part \sqsubseteq \delta(hasFDV)$, $Part \sqsubseteq \delta(hasTV)$, $Part \sqsubseteq \delta(hasLNS)$,
 $Part \sqsubseteq \delta(hasTG)$, $Part \sqsubseteq \delta(hasSZ)$,
 $\rho(hasFDV) \sqsubseteq xsd:decimal$, $\rho(hasTV) \sqsubseteq xsd:decimal$,
 $\rho(hasTG) \sqsubseteq xsd:string$, $\rho(hasSZ) \sqsubseteq xsd:string$,
 $TZone \sqsubseteq \delta(hasSPre)$, $TZone \sqsubseteq \delta(hasMPre)$, $TZone \sqsubseteq \delta(hasBPre)$, $TZone \sqsubseteq \delta(hasTG)$,
 $\rho(hasSPre) \sqsubseteq xsd:string$, $\rho(hasMPre) \sqsubseteq xsd:string$,
 $\rho(hasBPre) \sqsubseteq xsd:string$, (funct *hasFits*), (funct *hasHole*)
 (funct *hasShaft*), (funct *hasHoleTZ*), (funct *hasShaftTZ*).

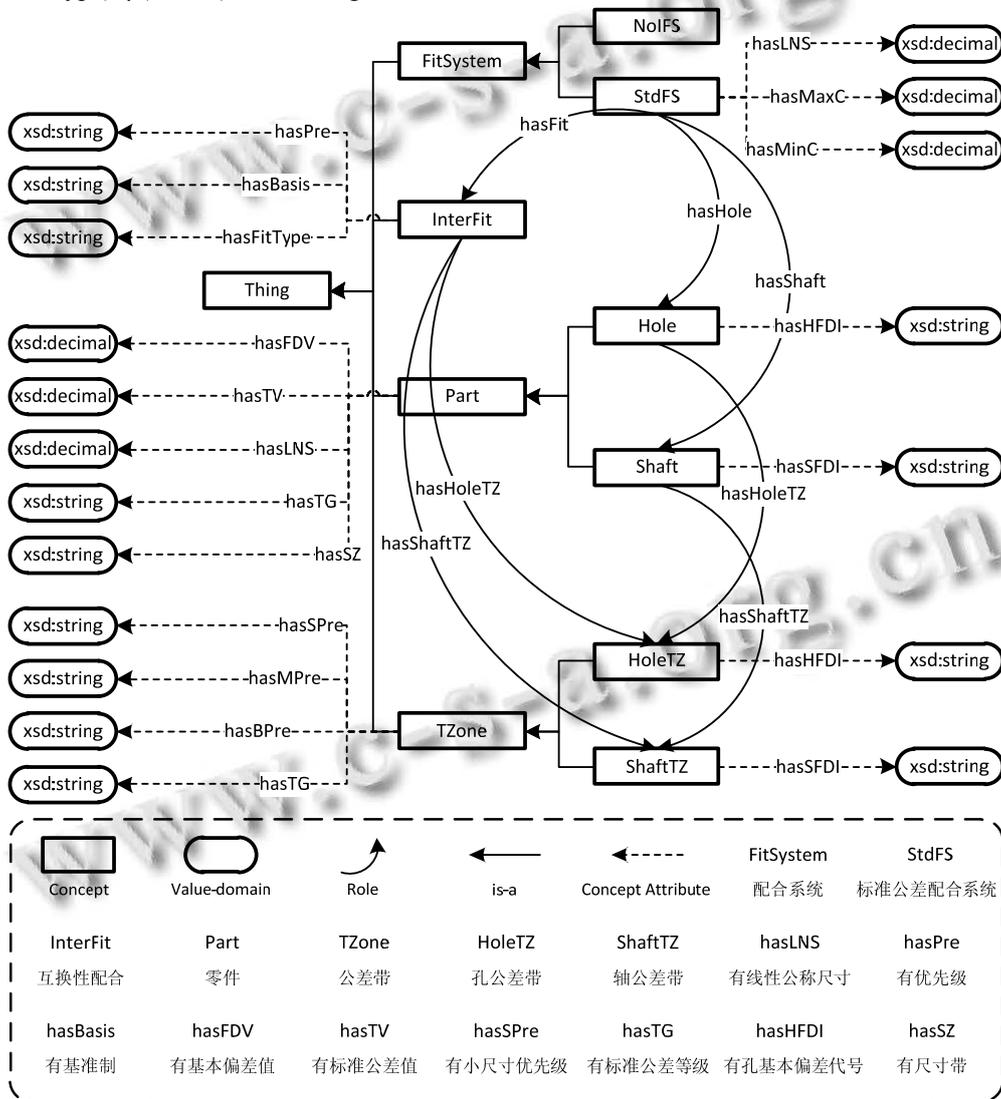


图 1 极限与配合本体 O_{LF} 的 TBox

这里值域 xsd:string 和 xsd:decimal 分别是可扩展标记语言模式数据类型(XML Schema Datatypes)中定

义的字符串和十进制数. 上述 DL-Lite_A 断言所描述的极限与配合本体 OLF 的 TBox 结构如图 1 所示.

3 极限与配合数据库的关系模式

极限与配合所涉及的数据主要包括标准 GBT1800 第一部分中规定的公称尺寸至 3150mm 的标准公差数值, 轴的基本偏差数值, 孔的基本偏差数值; GBT1800 第二部分规定的常用孔公差带及其优先级、常用轴公差带及其优先级、常用配合及其优先级。

极限与配合数据库 DB_{LF} 用于存储上述数据, 其关系模式 R 如图 2 所示。其中 fd 指代基本偏差 (fundamental deviation), t 指代公差 (tolerance), pre 指代优先级 (preference)。

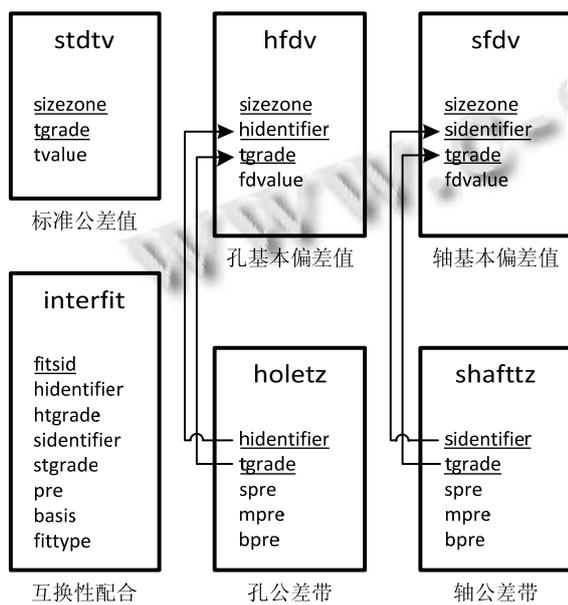


图 2 极限与配合数据库 DBLF 的关系模式 R

各表中, 值(如 tvalue)的数据类型均为十进制数 (numeric), 其它列的数据类型均为可变字符串 (varchar)。为简化 φ, 在表标准公差值表(stdtv)中尺寸段(sizezone)的值使用中间段落(GBT1800 中基本偏差表所用尺寸段)而非主段落(GBT1800 中标准公差值表所用尺寸段)。这样虽然会使 R 的实例 I 产生少量冗余, 但在连接本体时可以将每个段落视为一个个体标识, 与直接使用尺寸标识相比显著减少需要生成的个体数量。

4 极限与配合本体和数据库的连接

4.1 连接的原理

为通过极限与配合本体 O_{LF} 访问极限与配合数据库 DB_{LF} 中的数据 Γ_{DB}, 首先需要连接 O_{LF} 和 DB_{LF}。参

照数据集成的经验^[9], 本文通过将 DB_{LF} 中 Γ_{DB} 映射为 O_{LF} 的实例即 ABox 的方式, 来实现这种连接。注意到 DB_{LF} 中 Γ_{DB} 均为数据, 为了区分这些数据与 O_{LF} 实例中的对象 Γ_O, 需要为 Γ_O 中的元素构造特殊的标识符。借鉴文献[8]的方法, 本文使用 Γ_{DB} 中的元素来构造 Γ_O 中的元素。于是引入函数 f 的集合 Λ, 其中 f 的参数数量 n 大于 0, t 是 Γ_{DB} 中元素构成的 n 的元组, f(t) ∈ Γ_O 叫做可变对象项, 它与 Γ_O 中的元素相对应。此时所要构造的极限与配合本体 O_{LF} 的 ABox 的对象和数据均来自 DB_{LF}, 即 Γ_V 是 Γ_{DB} 的子集, Γ_O 从 Γ_{DB} 构造而来。

为了连接 DB_{LF} 和 O_{LF}, 可以使用 R 的 SQL 查询 φ 来获取 Γ_{DB} 中元素, 使用 CQ 来获取 O_{LF} 中的概念、角色、值域、概念属性等元素。于是一个映射断言形如 φ(x) → ψ(y), 其中 φ(x) 是一个 R 上的一个 SQL 查询, ψ(y) 是 O_{LF} 上的一个连接查询, x 是结果变量, y 是包含以 x 中元素为参数的可变对象项, x 中元素在內的变量组。映射集 M 由若干映射断言组成。在查询 O_{LF} 本体时, 通过使用分解 M 中各 ψ 的原子所得到的原子与 φ 的对应关系, 将对 O_{LF} 的查询转化为对 DB_{LF} 的查询, 从而得到 Γ_{DB} 中的数据。

此连接的语义, 即一个如前所述的解释 I=(Δ^I, ·^I), 且 ·^I 为每个可变对象项, 均分配 Δ^I 中一个不同的元素。I 对于一个映射断言 φ(x) → ψ(y), 和在 DB_{LF} = <R, I> 中的每个 SQL 查询结果 v ∈ φ(I), 以及由此得到的每个基础实例 φ[x/v] → ψ[x/v], 若 SQL 查询结果 ans(φ[x/v], DB_{LF}) 为真, 则 ψ[x/v]^I 也为真(对于 ψ 的原子类似)。

4.2 连接的建立

根据上述讨论, 首先构建函数集 Λ。由于 R 中 stdtv 关于零件(即孔和轴), hfdv 关于孔, sfdv 关于轴, interfit 关于配合, holetz 关于孔公差带, shafttz 关于轴公差带, 故 Λ 中至少应包含五个函数如表 1 所示, 即 Λ = {h, s, f, ht, st}。interfit 中 fitsid 分量存储的常量属于 Γ_O 而非 Γ_V, 可直接作为函数 f 的参数。

表 1 用于标识 O_{LF} 中对象的函数

函数名	在 O _{LF} 中对应对象的概念 (Concept)	参数数量	参数 (所属分量名)
h	Hole	3	sizezone, hidentifier, tgrade
s	Shaft	3	sizezone, sidentifier, tgrade
f	InterFit	1	fitsid
ht	HoleTZ	2	hidentifier, tgrade
st	ShaftTZ	2	sidentifier, tgrade

尽管由 stdtv 可得到一元包涵依赖^[10] $Hole \subseteq Part$ 和 $Shaft \subseteq Part$, 但因零件基本偏差值已经按孔和轴分为 hfdv 和 sfdv, 故这里不对极限与配合本体 O_{LF} 的 TBox 进行依赖优化。

由于 R 中 Γ_{DB} 属于 stdtv 的子集并不包含分量 hidentifier 和 sidentifier, 故需分别根据 28 个孔基本偏差代号(A, B, C, CD, D, E, EF, F, FG, G, H, JS, J, K, M, N, P, R, S, T, U, V, X, Y, Z, ZA, ZB, ZC)和函数 h 来得到 OLF 中 Hole 相应的对象, 分别根据 28 个轴基本偏差代号(a, b, c, cd, d, e, ef, f, fg, g, h, js, j, k, m, n, p, r, s, t, u, v, x, y, z, za, zb, zc)和函数 s 来得到 OLF 中 Shaft 相应的对象。用 hi 表示 28 个孔基本偏差代号的通配符, 用 si 表示 28 个轴基本偏差代号的通配符, 则用于连接 OLF 与 DBLF 的映射集 M 如下:

a. $SELECT * FROM stdtv \rightarrow Hole(h(sizezone, hi, tgrade)) \wedge hasSZ(h(sizezone, hi, tgrade), sizezone) \wedge hasTG(h(sizezone, hi, tgrade), tgrade) \wedge hasTV(h(sizezone, hi, tgrade), tvalue).$

b. $SELECT * FROM stdtv \rightarrow Shaft(s(sizezone, si, tgrade)) \wedge hasSZ(s(sizezone, si, tgrade), sizezone) \wedge hasTG(s(sizezone, si, tgrade), tgrade) \wedge hasTV(s(sizezone, si, tgrade), tvalue).$

c. $SELECT * FROM hfdv \rightarrow Hole(h(sizezone, hidentifier, tgrade)) \wedge hasSZ(h(sizezone, hidentifier, tgrade), sizezone) \wedge hasHFDI(h(sizezone, hidentifier, tgrade), hidentifier) \wedge hasTG(h(sizezone, hidentifier, tgrade), tgrade) \wedge hasFDV(h(sizezone, hidentifier, tgrade), fdvalue).$

d. $SELECT * FROM sfdv \rightarrow Shaft(s(sizezone, sidentifier, tgrade)) \wedge hasSZ(s(sizezone, sidentifier, tgrade), sizezone) \wedge hasSFDI(s(sizezone, sidentifier, tgrade), sidentifier) \wedge hasTG(s(sizezone, sidentifier, tgrade), tgrade) \wedge hasFDV(s(sizezone, sidentifier, tgrade), fdvalue).$

e. $SELECT * FROM interfit \rightarrow Interfit(f(fitsid)) \wedge hasHoleTZ(f(fitsid), ht(hidentifier, htgrade)) \wedge hasShaft(f(fitsid), st(sidentifier, stgrade)) \wedge hasPre(f(fitsid), pre) \wedge hasBasis(f(fitsid), basis) \wedge hasFitType(f(fitsid), fitype).$

f. $SELECT * FROM holetz \rightarrow HoleTZ(ht(hidentifier, tgrade)) \wedge hasHFDI(ht(hidentifier, tgrade), hidentifier) \wedge hasTG(ht(hidentifier, tgrade), tgrade) \wedge hasSPre(ht(hidentifier, tgrade), spre) \wedge$

$hasMPre(ht(hidentifier, tgrade), mpre) \wedge hasBPre(ht(hidentifier, tgrade), bpre).$

g. $SELECT * FROM shafttz \rightarrow ShaftTZ(st(sidentifier, tgrade)) \wedge hasSFDI(st(sidentifier, tgrade), sidentifier) \wedge hasTG(st(sidentifier, tgrade), tgrade) \wedge hasSPre(st(sidentifier, tgrade), spre) \wedge hasMPre(st(sidentifier, tgrade), mpre) \wedge hasBPre(st(sidentifier, tgrade), bpre).$

包含通配符的 a 和 b 分别对应 28 条映射公理, 即 M 中共包含 61 条公理, 且 M 可根据需要进行扩展。于是得到了一个极限与配合的 DL-LiteA 本体 $O_{LF} = \langle T, A \rangle$, 其中 $A = \langle M, DB_{LF} \rangle$ 是由所构造的映射集 M 映射和极限与配合数据库 DB_{LF} 所形成的虚拟 ABox。此时通过对 O_{LF} 的查询, 可以访问 Γ_{DB} 中的任意数据。

5 基于本体的极限与配合数据访问系统

5.1 系统结构

本文所要构建的基于本体的极限与配合数据访问系统, 作为一个基础模块, 是我们所开发的软件“尺寸极限与配合辅助系统”的一部分, 另外也为公差自动生成、自动装配等等研究提供极限与配合知识的访问接口。如图 3 所示, 它包含上面所构建的极限与配合本体 O_{LF} 、极限与配合数据库 DB_{LF} 、映射集 M、一个推理机 Quest^[11]以及用于操作推理机和处理推理结果的“查询表达和结果转化”部分。

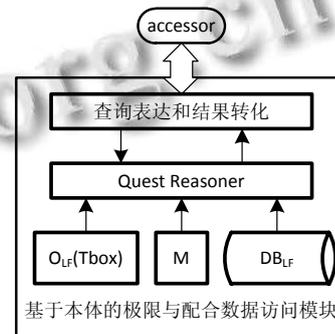


图 3 基于本体的极限与配合数据访问模块

5.2 系统实现

使用 web 本体语言(Web Ontology Language, OWL)的基于 DL-Lite 的子集 OWL 2 QL^[12]编码 O_{LF} 的 TBox, 使用数据库 PostgreSQL 实现 DB_{LF} , 并以类 Turtle^[13]的语法编写 Quest 可识别的对应于 M 的映射集如表 2 所示, 其中映射公理 interfit0、interfit1、interfit2 对应于 M 中的映射公理 e。按照图 3 的结构,

使用 Quest 推理机连接 Q_{LF} 、 M 、 DB_{LF} 对 O_{LF} 进行查询时, 通过 Quest 推理机展开(转化)本体查询, 查询返回 DBLF 中的对应数据, 完成极限与配合数据的访问。

表 2 映射集 M 的编码

映射ID	目标查询	源查询
stdtv_h_hi	<code>{sizezone}hi{tgrade} a :Hole; :hasSZ {sizezone}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasTV {tvalue}^^xsd:decimal .</code>	<code>SELECT sizezone, tgrade, tvalue FROM stdtv</code>
stdtv_s_si	<code>{sizezone}si{tgrade} a :Shaft; :hasSZ {sizezone}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasTV {tvalue}^^xsd:decimal .</code>	<code>SELECT sizezone, tgrade, tvalue FROM stdtv</code>
hfdv	<code>{sizezone}{hidentifier}{tgrade} a :Hole; :hasSZ {sizezone}^^xsd:string; :hasHFDI {hidentifier}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasFDV {fdvalue}^^xsd:decimal .</code>	<code>SELECT sizezone, hidentifier, tgrade, fdvalue FROM hfdv</code>
sfdv	<code>{sizezone}{sidentifier}{tgrade} a :Shaft; :hasSZ {sizezone}^^xsd:string; :hasSFDI {sidentifier}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasFDV {fdvalue}^^xsd:decimal .</code>	<code>SELECT sizezone, sidentifier, tgrade, fdvalue FROM sfdv</code>
interfit0	<code>{fitsid} a :InterFit; :hasHoleTZ :{hidentifier}{htgrade} .</code>	<code>SELECT fitsid, hidentifier, htgrade FROM interfit</code>
interfit1	<code>{fitsid} a :InterFit; :hasShaftTZ :{sidentifier}{stgrade} .</code>	<code>SELECT fitsid, sidentifier, stgrade FROM interfit</code>
interfit2	<code>{fitsid} a :InterFit; :hasPre {pre}^^xsd:string; :hasBasis {basis}^^xsd:string; :hasFitType {fittype}^^xsd:string .</code>	<code>SELECT fitsid, pre, basis, fittype FROM interfit</code>
holetz	<code>{hidentifier}{tgrade} a :HoleTZ; :hasHFDI {hidentifier}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasSPre {spre}^^xsd:string; :hasMPre {mpre}^^xsd:string; :hasBPre {bpre}^^xsd:string .</code>	<code>SELECT hidentifier, tgrade, spre, mpre, bpre FROM holetz</code>
shafttz	<code>{sidentifier}{tgrade} a :ShaftTZ; :hasSFDI {sidentifier}^^xsd:string; :hasTG {tgrade}^^xsd:string; :hasSPre {spre}^^xsd:string; :hasMPre {mpre}^^xsd:string; :hasBPre {bpre}^^xsd:string .</code>	<code>SELECT sidentifier, tgrade, spre, mpre, bpre FROM shafttz</code>

6 极限与配合数据访问实例

简单协议资源描述框架查询语言^[14](Simple Protocol and RDF Query Language, SPARQL)是 W3C 推荐的 RDF 查询语言, 由于 OWL 基于 RDF, 而 O_{LF} 使用 OWL 实现, 故可使用 SPARQL 对 O_{LF} 进行查询. 本文中 O_{LF} 的 IRI 为 `<http://www.semanticweb.org/adm/onto/2013/1/slaf>`, 故在 SPARQL 查询中使用前缀 PREFIX : `<http://www.semanticweb.org/adm/onto/2013/1/slaf#>` 和 PREFIX xsd: `<http://www.w3.org/2001/XMLSchema#>`.

6.1 基本数据访问

标准公差和基本偏差是极限与配合中最基本的数据, 它们分别确定了公差带的大小和公差带距离零线的位置. 为简化公差带和配合的种类, 国家标准对孔、轴公差带及其所组合成的配合进行了限制, 这就使得

在选择公差带和配合时, 需要查询其优先级等信息. 下面验证是否可通过查询 O_{LF} 访问上述数据.

查询某个公称尺寸介于 6mm 到 10mm 之间的零件在标准公差等级为 IT6 时所具有的标准公差值. 为此构造 SPARQL 查询语句如表 3(a)所示, 返回结果 9.0, 即所求标准公差值为 9.0 μ m. 以孔为例, 查询某个公称尺寸介于 560mm 到 630mm 的孔在基本偏差代号为 F, 标准公差等级为 IT8 时的基本偏差值. 为此构造 SPARQL 查询语句如表 3(b)所示, 返回结果为 76.0, 即所求基本偏差值为 76 μ m. 以孔公差带为例, 查询孔公差带 J8 分别对应于公称尺寸小于 18mm、公称尺寸 18mm 至 500mm 的优先级, 为此构造 SPARQL 查询和返回结果如表 3(c)所示. 查询配合 H7/n6 的配合种类和优先级, 所构造 SPARQL 查询和返回结果如表 3(d)所示. 数据访问成功.

在预热(剔除首次查询)的情况下,分别对上述各 SPARQL 查询及其功能相同的 SQL 查询计时.取十次计时的平均值,结果见表 3.从表中数据可以看到,在访问极限与配合数据时,尽管通过 Q_{LF} 访问的耗时略大于直接查询 DB_{LF} 的耗时,但其效率仍然较高.考虑到 O_{LF} 对知识共享和重用以及语义互操作的支持,这种性能上的微小损失是可接受的.

6.2 复杂数据访问

在实际极限与配合选用中,常用方法有计算法、类比法、实验法等.其中计算法是根据使用要求,通过理论公式计算来确定极限与配合.为了验证基于本体的极限与配合数据访问的可行性,现以计算法为例:

假设有公称尺寸为 $\Phi 3000$ 的孔和轴,要求配合最大间隙为 0.45mm ,最小间隙为 0.14mm ,在基孔制、孔轴同级配合的情况下,如何设计在生产时选用配合?

上述问题对应的 RDF 图如图 4(a)所示,htg、htv、stg、stv、sfdi、sfdv 分别表示孔标准公差等级、孔标准公差值、轴标准公差等级、轴标准公差值、轴基本偏差代号、轴基本偏差值,其中 $htv=stv \leq 155\mu\text{m}$.

为求得 O_{LF} 中与上述问题匹配的个体,首先使用图 4(b)中 SPARQL 语句 1 查询 O_{LF} ,得到孔和轴匹配的标准公差代号为 IT6;然后使用 SPARQL 语句 2 查询 O_{LF} ,得到满足要求的孔和轴;最后匹配结果如图 4(c)所示.在本文所开发的软件“尺寸极限与配合辅助系统”中,以公称尺寸 3000mm 、最大间隙 $450\mu\text{m}$ 、最小间隙 $155\mu\text{m}$ 为输入,系统运行结果如图 5 所示,即所应选用的配合为 $\Phi 3000 \text{ H6/f6}$.



图 5 软件系统运行结果

7 总结

本文构建了极限与配合的 DL-LiteA 本体和极限与配合数据库,分别以两者作为极限与配合知识的概念层和数据层,然后使用可变对象函数和本体与数据

库间的映射集完成了两者的连接.通过工程实例验证,采用这种结构,可以在保证极限与配合知识重用和支持语义互操作特性的同时,高效率地访问极限与配合数据.对本体和数据库间映射的优化,添加规则支持,以及以此为基础的 CAD 系统公差值自动生成,是进一步需要开展的研究工作.

参考文献

- 覃裕初,钟艳如,常亮,等.基于几何公差描述逻辑的公差类型的自动生成.计算机集成制造系统,2013,19(7):1490-1499.
- Zhong YR, Qin YC, Huang MF, et al. Automatically generating assembly tolerance types with an ontology-based approach. Computer-Aided Design, 2013, 45(11): 1253-1275.
- Gruber TR. A translation approach to portable ontology specifications. Knowledge Acquisition, 1993, 5(2): 199-220.
- 蒋向前.现代产品几何量技术规范(GPS)国际标准体系.机械工程学报,2004,40(12):133-138.
- Calvanese D, Giacomo GD, Lembo D, et al. Ontology-based database access. The 15th Italian SympOLFium on Advanced Database Systems (SEBD 2007). Torre Canne di Fasano (BR). Proc. of SEBD 2007, 2007. 324-331.
- Calvanese D, Giacomo GD, Lembo D, et al. Linking data to ontologies: The description logic DL-LiteA. The 2006 International Workshop on OWL: Experiences and directions (OWLED 2006). Athens, OWLED 2006, 2006.
- Calvanese D, Giacomo GD, Lembo D. Data complexity of query answering in description logics. Artificial Intelligence, 2013, 195: 335-360.
- Poggi A, Lembo D, Calvanese D, et al. Linking data to ontologies. Journal on Data Semantics, 2008, X: 133-173.
- Lenzerini M. Data integration: A theoretical perspective. The 21st ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. Madison, ACM, 2002. 233-246.
- Rodriguez-muro M, Calvanese D. Dependencies: Making ontology based data access work in practice. BARCELOP. The 5th Alberto Mendelzon Int. Workshop on Foundations of Data Management. Santiago, 2011.
- Rodriguez-muro M, Calvanese D. Quest, an OWL 2 QL reasoner for ontology-based data access. KLINOV P. Proc. of OWL: Experiences and Directions Workshop 2012. Heraklion, Crete, 2012.
- Calvanese D, Carroll J, Giacomo G D. OWL 2 Web Ontology Language Profiles (2nd Edition). http://www.w3.org/TR/2012/REC-owl2-profiles-20121211/. [2012-12-11].
- Beckett D, Berners-lee T. Turtle-Terse RDF Triple Language. http://www.w3.org/TR/2013/CR-turtle-20130219 [2013-02-19].
- Prudhommeaux E, Seaborne A, Harris S. SPARQL 1.1 Query Language. http://www.w3.org/TR/2012/PR-sparql11-query-20121108/. [2012-11-08].