

## 时态数据库时间轴的动态逻辑模型\*

刘冬宁<sup>1,2</sup>, 汤庸<sup>2+</sup>

<sup>1</sup>(广东工业大学 计算机学院, 广东 广州 510006)

<sup>2</sup>(中山大学 计算机科学系, 广东 广州 510275)

### Dynamic Logic Model of Time Axes in Temporal Database

LIU Dong-Ning<sup>1,2</sup>, TANG Yong<sup>2+</sup>

<sup>1</sup>(Faculty of Computer Science, Guangdong University of Technology, Guangzhou 510006, China)

<sup>2</sup>(Department of Computer Science, SUN YAT-SEN University, Guangzhou 510275, China)

+ Corresponding author: E-mail: issty@mail.sysu.edu.cn, http://www.sysu.edu.cn

**Liu DN, Tang Y. Dynamic logic model of time axes in temporal database. 2010,21(4):694-701.**  
<http://www.jos.org.cn/1000-9825/3511.htm>

**Abstract:** Although axiomatic systems and proof method for temporal logic have found so far relatively few applications in the query language modeling of temporal database and that was proved by Gabbay, *et al* in 1994, the model of time axes still must be built axiomatically, which owns soundness and completeness and depicts the time axes in fine grain. Thus the essential characteristics of time and temporal attributes in database can be described exactly, and the result is used in the temporal querying. So the research starts from essential characteristics of time attributes. At first, this paper expounds the order relation and first order logic properties of time axes. Secondly, this paper axiomatically models that using Tense Logic and dynamic logic, which aims at reflecting the properties of axes in fine grain by logical analysis. In the part of dynamic logic modeling, respecting to the static of temporal logic system, this paper mostly deals with the dynamics in the new system. Based on Lin.Z system in Tense Logic, this paper makes out the dynamic Lin.Z system, which has some parameters and functions. The parameters of this system are based on the action, which helps the action exponential numerical and functional. The results of that embody the properties and representation method of the rule's lifecycle and point "Now" in temporal database, which positively helps the research in the field of temporal knowledge representation and temporal querying subsequently.

**Key words:** temporal database; time axes; Tense Logic; dynamic logic; action parameter

**摘要:** 尽管在 1994 年, Gabbay 等人论证了时态逻辑的公理化系统和证明论方法是不适合于时态数据库查询语言建模的, 但是仍需要通过对时间轴的公理化建模, 利用公理化系统的可靠和完全等性质对时间轴作“细精度”的语

\* Supported by the National Natural Science Foundation of China under Grant Nos.60673135, 60736020 (国家自然科学基金); the Program for New Century Excellent Talents in University of China under Grant No.NCET-04-0805 (新世纪优秀人才支持计划); the Natural Science Foundation of Guangdong Province of China under Grant Nos.7003721, 04105503 (广东省自然科学基金); the Science-Technology Plan of Guangdong Province of China under Grant Nos.2005B10101041, 2007B010200052 (广东省科技攻关计划)

Received 2008-04-05; Revised 2008-10-16; Accepted 2008-11-10; Published online 2009-06-09

义刻画,只有这样才能准确地数据库中反映时间和时间属性的本质特点,并应用于与时间轴模型直接相关的时态查询语言.因此,从分析时间轴的性质出发,对时间轴的序关系和谓词逻辑性质进行了研究和阐述.随后分别用经典的时态逻辑 Tense Logic 和动态逻辑对时态数据库的时间轴进行了公理化建模,这样刻画的目的是为了“细精度”地体现时间轴的特点,并借助逻辑的方法对其分析.在 TDB(temporal database)时间轴的动态逻辑建模部分,相对于原时态逻辑系统较为静态,着重处理了时间的动态性,并参照了 Tense Logic 中的 Lin.Z 系统,将其转化为动态 Lin.Z 系统,在其中添加了参数化处理.该参数化的处理是基于动作执行的,主要工作在动作指数的数值化和函数化两个方面,其结果体现了时态数据库中规则生存周期和“Now”节点的一些特点以及知识表达和解决方法,研究结果将对后续时态知识表达和时态数据库查询语言的研究起到积极的作用.

**关键词:** 时态数据库;时间轴;Tense Logic;动态逻辑;动作参数

**中图法分类号:** TP311

**文献标识码:** A

时间在信息系统中是无所不在的.针对时间的表达、推理与查询一直是时态数据库中最为关键的研究点.目前许多的推理查询语言已经使用于时态数据库,其理论基础也相对繁杂,但其中最为直接关联并得到广泛应用的就是基于时态逻辑的查询语言<sup>[1]</sup>.然而在 1994 年,Gabbay 等人论证了时态逻辑的公理化系统和证明论方法是不适合于时态数据库查询语言建模的<sup>[2]</sup>,相关时态逻辑的查询语言建模是非公理化的,这使得相关形式化研究工作陷入了一个困境.我们无法利用公理化系统的可靠性和完全性对时态逻辑所表达的一些时态数据库性质进行准确表达及语义刻画.例如,对于时态数据库的“Now”语义的刻画就一直是一个开问题,对它的刻画一直存在着语义过大或语义过小的问题<sup>[3]</sup>.

“Now”作为时间轴的一个元素,其语义是动态的,随着时间和时间单位的延伸与变化而动态地发生变化.另一方面,从时态数据库的全局来看,它也是一个随着时间轴向前推进的历史数据库序列  $H=(H_0, \dots, H_n, H_{n+1}, \dots)$ ,在知识表达处理过程中属于一个动态的概念.因此,要将时态数据库中这些动态的概念刻画清楚,一项重要的工作就是将时态数据库的时间元素进行准确刻画,即刻画反映时间元素的时间轴模型.

一般而言,基于结构的选择,时间轴模型可以初步划分为连续模型(continuous model)、步进模型(stepwise model)、离散模型(discrete model)和恒定模型(non temporal model)等.然而这样的划分是“粗精度”的<sup>[4]</sup>,我们仍需对模型内的一些重要的一阶甚至高阶性质加以探讨.另一方面,尽管时态逻辑的公理化系统和证明论方法是不适合于时态逻辑查询语言建模的,但是我们仍然需要通过对时间轴的公理化建模,利用公理化系统的可靠和完全等性质对时间轴作“细精度”的语义刻画.只有这样才能准确地数据库中反映时间和时间属性的本质特点,并应用于与时间轴模型直接相关的时态查询语言中.

事实上,关于时间特征和时态行为的公理化研究在国际上一直得到广泛研究<sup>[5,6]</sup>,这类研究较为集中于时态逻辑的本身扩展方面,如增加内定理和一阶模型、添加时态模态词与连接词并赋予新的语义等等.考虑到时态逻辑与时态知识推理的直观联系,这类研究较多集中在逻辑中属于静态的时态逻辑研究方面,或单纯地研究动态算法,忽略了动态的逻辑公理系统和语义变换,使得在动态性的公理化研究方面较为缺乏.2006 年,作为动态语义和更新语义的创始人,荷兰逻辑学家 Veltman 曾呼吁相关学者将研究焦点集中到这方面来<sup>[5]</sup>.本文正是这一角度出发,在文章的第 1 节初步分析并阐述了时态数据库时间轴的相关谓词逻辑性质,在第 2 节使用较为静态的 Tense Logic 对 TDB(temporal database)时间轴进行了刻画,而在第 3 节则利用动态逻辑对 Tense Logic 的 Lin.Z 系统进行了转化,并添加了相应的参数化处理,使得对时间轴的刻画更为“精细”、动态化和函数化,研究结果将对后续时态知识表达和时态数据库查询语言的研究起到积极的作用.

## 1 时间轴的性质分析

在对时间轴进行刻画之前,首先要对其性质进行分析,这方面的工作主要包括时间集合的序关系和时间轴的谓词逻辑性质.

## 1.1 时间集合的序关系

时间集合的序关系有两种表达方法,一种为偏序关系,另一种为非偏序关系.

作为偏序关系的表达方式,可令系统为  $\mathcal{T}=\langle T, \leq \rangle$ ,其中  $T=\{t_1, t_2, \dots, t_n\}$  为时间点的有限集合.“ $\leq$ ”表示  $T$  上的时序,  $t_i \leq t_{i+1}$  表示  $t_i$  不会出现在  $t_{i+1}$  之后,即  $t_i$  发生在  $t_{i+1}$  之前或与  $t_{i+1}$  同时发生.由此可以得到该系统是自反的、反对称的和传递的.

作为非偏序关系的表达方式,可令系统为  $\mathcal{T}'=\langle T, < \rangle$ ,其中  $T=\{t_1, t_2, \dots, t_n\}$  为时间点的有限集合,则“ $<$ ”同样表示  $T'$  上的时序,  $t_i < t_{i+1}$  表示  $t_i$  发生在  $t_{i+1}$  之前,由此也可以得到该系统是传递的,但不含自反性和反对称性等性质.

从表面上看,“ $<$ ”的表示方法使我们得到的序关系没有“ $\leq$ ”的表示方式好,但值得注意的是,在这样的“ $<$ ”表示方式下,对于  $\forall t_i$ ,正是由于没有了自反性和反对称性,反而更适于数据库的时间特点,后文中我们将对其作一阐述.

## 1.2 性质分析

在第 1.1 节中,我们简要地形式化描述了时间集合的序关系,其具有一般性.然而,时态数据库的时间表达具有更强的特殊性,它必须满足左序列性、右序列性、传递性和有穷间隔性以及反分支性等特点.

### 1.2.1 左序列性和右序列性

除非数据库系统刻意规定了数据库的起始事务时间或截止事务时间,否则,时态数据库的时间必须满足左序列性和右序列性.左序列性表示对于时间轴上的任意点,其先驱集为非空集,一阶形式化描述如下:

$$\forall x \exists y. Ryx.$$

右序列性表示对于时间轴上的任意点,其后继集为非空集,一阶形式化描述如下:

$$\forall x \exists y. Rxy.$$

满足了左序列性和右序列性,即满足了在数据库的任意表中,都可以在头元组或未元组中插入新的元组.

### 1.2.2 非分支性

在时态数据库的一般应用中,其时间轴主要呈现的是线性结构,而非一般用于模型检测的树形结构<sup>[7,8]</sup>,因此,时间轴必须具有非分支性.对于非分支性,它是一阶不可定义的,但模态可定义<sup>[9]</sup>,在本文第 2 节中,我们将给出其模态定义.

### 1.2.3 连续性与有穷间隔性

尽管时间给人的直观印象是连续的,然而在计算机科学中,由于要对时间作量化采样,因此连续性是不可取的.对于数据库的时间而言,其特点为在时间轴上的任意两点间,其中间可插入的点是无穷的,即有穷间隔性.它表明了在与时间相关的任意两元组之间可以有穷地插入相关元组.关于有穷间隔性,它是一阶不可定义的,但却模态可定义,在本文的第 2 节中,我们将对其进行模态定义.

### 1.2.4 其他性质

在时态数据库的时间性质方面,自反性和反对称性是有争论的两个性质.反对称性对时间提出了更强的序关系,如果在元组中没有出现时间循环的情况下,反对称性是可取的.但一旦出现了时间循环,则反对称性不可取.这类情况经常在时态数据库中发生,例如排班表等轮值问题.因此,反对称性对时态数据库的时间要求过强,除特殊应用以外,不应成为数据库中时间满足的必要条件.

所以,在第 1.1 节描述的两种表达时间序关系的方法中,  $\langle T, < \rangle$  显然优胜于  $\langle T, \leq \rangle$ .  $\langle T, < \rangle$  通过对时间自反性的消除,也消除了反对称性.同时,时间的自反性也一直是争论的,在不同的本体论立场上,对时间的自反性与反自反性有不同的论述<sup>[6,9]</sup>.作为时态数据库时间特点的一般性研究,我们认为自反性也是过强的条件.

## 2 TDB 时间轴的时态逻辑模型

基于上述对时态数据库时间轴的性质分析与论述,从本节开始,我们将对其进行逻辑的刻画与建模.显然,基于前文分析,  $\langle T, < \rangle$  是我们更适宜采用的代数模型.

另一方面,由于与时态数据库最直接关联的人工智能逻辑为时态逻辑,因此在本节中,我们首先使用最原始的 Tense Logic 来对时态数据库的时间进行逻辑刻画.首先使用 Tense Logic 的主要原因有两点,即分别基于 Tense Logic 的原始性和扩展性<sup>[1,6]</sup>.

## 2.1 Tense Logic

Tense Logic 是最原始、最基本的时态逻辑系统,它具有  $F, P, H, G$  两对共 4 个模态符,  $F$  和  $P$  分别为 Future 和 Past 的缩写,  $H$  和  $G$  分别为 Hitheto 和 Henceforth 的缩写.在这里,  $F$  和  $G$  的对偶性以及  $P$  和  $H$  的对偶性与经典模态逻辑中模态算子  $\square$  和  $\diamond$  类似,即:  $Fq = \neg G\neg q$ ;  $Pq = \neg H\neg q$ . 所不同的是,  $\square$  和  $\diamond$  在可能世界表示可达关系时是无向的,但  $F, P, H, G$  这 4 个模态算子都是有向的.

Tense Logic 的原始性使其较本质、较纯地抓住了时间的特性,而同时也正是由于其原始性,使得它的扩展性极强,对在后续研究中添加相应算子或各项交叉技术的使用提供了方便.根据 Taski 语义,施归纳于对任意公式  $\varphi$  的真值赋值,我们对 Tense Logic 中的模型  $\mathcal{M} = \langle T, <, \pi \rangle$  定义如下:

$$\begin{aligned} \mathcal{M}, t \models q & \quad \text{if } \pi(t)(q) = 1 \\ \mathcal{M}, t \models \neg \varphi & \quad \text{if not } \mathcal{M}, t \models \varphi \\ \mathcal{M}, t \models \varphi \wedge \psi & \quad \text{if } \mathcal{M}, t \models \varphi \text{ and } \mathcal{M}, t \models \psi \\ \mathcal{M}, t \models G\varphi & \quad \text{if } \forall s, t < s, \mathcal{M}, s \models \varphi \\ \mathcal{M}, t \models H\varphi & \quad \text{if } \forall s, s < t, \mathcal{M}, s \models \varphi \end{aligned}$$

其中,  $\pi$  为真值指派,  $\langle T, < \rangle$  为如前所述的代数模型.

下面为 Tense Logic 中最简单的模态系统  $K_t$  及其内定理:

- (CT) 所有命题逻辑的重言式  
 (US) if  $\varphi$  is a theorem, then so is  $\varphi[\psi/q]$   
 (MP) if  $\varphi$  and  $\varphi \rightarrow \psi$  are theorems, then so is  $\psi$   
 (TG) if  $\varphi$  is a theorem, then so are  $G\varphi$  and  $H\varphi$   
 (DB)  $G(q \rightarrow r) \rightarrow (Gq \rightarrow Gr)$   
        $H(q \rightarrow r) \rightarrow (Hq \rightarrow Hr)$   
 (CV)  $q \rightarrow GPq$   
        $q \rightarrow HFq$   
 (4)  $Gq \rightarrow GGq$   
        $Hq \rightarrow HHq$

其中, (CT) 规则包含了所有命题逻辑的重言式,即经典命题逻辑系统中的所有内定理均为模态系统  $K_t$  的内定理; (CV) 规则描述了  $F$  和  $G$ 、 $P$  和  $H$  的对应关系,对一阶性质不起任何对应作用; (DB) 规则类似经典模态逻辑系统中的 (K) 规则,它表明模态算子  $G$  和  $H$  对于逻辑连接词“ $\rightarrow$ ”具有分配性; (4) 规则描述了一阶性质中的传递关系; (US) 规则为替换规则,它和 (MP) 规则都与经典模态逻辑系统一致; 规则 (TG) 为模态产生规则,与经典模态逻辑系统中的 (RN) 规则类似.

## 2.2 刻画 TDB 时间轴的逻辑系统 Lin.Z

若在 Tense Logic 上刻画 TDB 的时间,其主要工作为将上述讨论的一阶和非一阶性质对应为时态逻辑公式,并添加入系统中.对应定理是各人工智能逻辑的三大组成部分之一.在此,我们将左右序列性、非分支性和有穷间隔性等性质对应为模态公式,分别如下:

- 左序列性(LS):  $P\top$
- 右序列性(FS):  $F\perp$
- 非分支性(NB):  $PFq \rightarrow (Pq \vee q \vee Fq)$   
 $FPq \rightarrow (Pq \vee q \vee Fq)$

- 有穷间隔性(FI):  $(G(Gq \rightarrow q) \rightarrow (FGq \rightarrow Gq)) \wedge (H(Hq \rightarrow q) \rightarrow (PHq \rightarrow Hq))$

由此我们得到刻画 TDB 时间的逻辑系统,其内定理为  $K$ , 的内定理及上述定理,这恰好为刻画整数轴的时态逻辑系统  $Lin.Z$ .

$Lin.Z$  系统具有较好的形式化性质,它具有可靠性、完全性和可判定性,文献[9]提供了相应的证明方法,限于篇幅,这里略过.据此,我们可以对时态知识进行表达.

例如,我们希望在表 1 中查询以下两个事实:

例 1:找出仅在一个公司工作过的人员.

例 2:找出曾经失业过的人员.

它们可以表达为

- 1)  $PF(\exists c. Works(r,c) \wedge \neg \exists c'. (FP Works(r,c') \wedge c' \neq c));$
- 2)  $P\exists c. Works(r,c) \wedge \neg \exists c. Works(r,c) \wedge F\exists c. Works(r,c).$

对上述知识表达应分两部分来理解:第 1 部分为对时间的表达,第 2 部分为结合时间的数据库知识的表达.首先,在这里,对时间的表达利用的是  $Lin.Z$  系统的表示方法,其所依赖的时间轴是由  $Lin.Z$  系统刻画的;其次,注意到一方面时态数据库知识表达及查询是构筑在既定的时间轴上的,另一方面其仍需保持谓词集的特点,因此,这里必须使用模态词结合量词的方法作整体的知识表达.尽管这样一来,表示方法便成为非公理化的方法<sup>[2]</sup>,但其表示时间和知识的基础(即时间轴)已被公理化刻画<sup>[1]</sup>,因此保证了知识刻画的精度与规范性,这正是我们结合时间轴研究时态知识表达与时态查询的主要目标和方法.类似的方法在后文的动态逻辑知识表示中依然有效.

**Table 1** Personnel employment record

**表 1** 人员工作履历表

Valid	Works	
	Name	Company
[2001~2003]	John	DEC
[2005~forever]	John	HP
[2002,2006]	Marry	HP
[2006~forever]	Marry	Microsoft
[2008~forever]	Tom	HP
...	...	...

### 3 TDB 时间轴的动态逻辑模型

Tense Logic 和  $Lin.Z$  系统是最早的时态逻辑,具有很强的原始性和扩展性.但无论是 Tense Logic 本身还是其后扩充的系统,对于时间的表达和推理主要是静态的.相对于动态的、参数化的时态数据库与时间轴,这样的静态在一定程度上影响了刻画的精度.由此,为了增加系统的动态刻画,一个可行的思路是将  $Lin.Z$  系统扩展为动态逻辑(dynamic logic)的系统.使用动态逻辑进行扩展的原因在于,动态逻辑是基于程序语义的<sup>[10]</sup>,这是其最突出的特点,尤其适合于计算机科学理论和应用.基于此,我们可以定义各种程序动作、参数、函数,而不再使用静态的模态连接词来表示时态数据库的知识.

#### 3.1 动态 $Lin.Z$ 逻辑系统

在设计动态  $Lin.Z$  系统之前,需要先定义正规动作(regular action)和正规动作集.

**定义 1(正规动作集).** 令集合  $B$  表示原子动作集合,  $Act(B)$  为一最小集,使得

- (i)  $B \subseteq Act(B)$ ;
- (ii) 如果  $\alpha, \beta \in Act(B)$ , 则  $\alpha; \beta \in Act(B)$ ;
- (iii) 如果  $\alpha, \beta \in Act(B)$ , 则  $\alpha \cup \beta \in Act(B)$ ;
- (iv) 如果  $\alpha \in Act(B)$ , 则  $\alpha^0, \alpha^{-1}, \alpha^*, \alpha^+ \in Act(B)$ ;
- (v) 除此之外,没有其他动作.

其中,动作  $\alpha; \beta$  表示“先做  $\alpha$ , 后做  $\beta$ ”;动作  $\alpha \cup \beta$  表示“或者做  $\alpha$ , 或者做  $\beta$ ”;动作  $\alpha^0$  是一个 skip 动作,即空操作;动

作 $\alpha^{-1}$ 表示如果动作 $\alpha$ 是有向的,那么它的逆向动作为 $\alpha^{-1}$ ,特殊的 $\alpha, \alpha^{-1}$ 等于 $\alpha^0, \alpha^*$ 和 $\alpha^+$ 分别为动作 $\alpha$ 的无穷星闭包(含 $\alpha^0$ )和无穷+闭包(不含 $\alpha^0$ ).

此外,我们可以定义一些缩写,令 $\alpha^n$ 表示连续 $n$ 个 $\alpha$ 动作,则 $\alpha^{n+1}$ 表示 $\alpha^n, \alpha$ .

根据正规动作集,我们已可定义线性时间轴上的基于程序语义的全部基本动作,如先后顺序的动作、选择动作、停顿动作和持续动作等\*\*.

基于对动作的语义解释,我们可以进一步定义相关模态词.

**定义 2(模型与赋值 1).** 令模型  $\mathcal{M}=\langle W, R, \pi \rangle$ ,其中  $W$  为可能世界集, $R$  为  $W$  上的一个二元关系, $\pi$  为其上的赋值定义,则对于  $\forall w, u \in W$ :

$$\begin{aligned} \mathcal{M}, w \models q & \quad \text{if } \pi(w)(q)=1 \\ \mathcal{M}, w \models [\alpha]\varphi & \quad \text{if } \forall u, wRu, u \models \varphi \\ \mathcal{M}, w \models \langle \alpha \rangle \varphi & \quad \text{if } \exists u, wRu, u \models \varphi \end{aligned}$$

假设我们将动作 $\alpha$ 定义为一个时间轴上的前向搜索动作,则 $[\alpha]\varphi$ 表示“将来,必然 $\varphi$ ”, $\langle \alpha \rangle \varphi$ 表示“将来,可能 $\varphi$ ”(即在将来存在一点使得 $\varphi$ 成立),显然 $[\alpha]\varphi \equiv \neg \langle \alpha \rangle \neg \varphi$ ,相对地, $[\alpha^{-1}]\varphi$ 表示“过去,必然 $\varphi$ ”, $\langle \alpha^{-1} \rangle \varphi$ 表示“过去,可能 $\varphi$ ”(即在过去存在一点使得 $\varphi$ 成立),显然 $[\alpha^{-1}]\varphi \equiv \neg \langle \alpha^{-1} \rangle \neg \varphi$ .由此,其在时间轴模型的语义定义如下:

**定义 3(模型与赋值 2).** 令时间轴模型  $\mathcal{M}=\langle T, <, \pi \rangle$ ,其中  $T$  为时间点的可能世界集, $<$  为  $T$  上的一个二元关系, $\pi$  为其上的赋值定义,则对于  $\forall t, s, u \in W$ :

$$\begin{aligned} \mathcal{M}, t \models q & \quad \text{if } \pi(t)(q)=1 \\ \mathcal{M}, t \models \neg \varphi & \quad \text{if not } \mathcal{M}, t \models \varphi \\ \mathcal{M}, t \models \varphi \wedge \psi & \quad \text{if } \mathcal{M}, t \models \varphi \text{ and } \mathcal{M}, t \models \psi \\ \mathcal{M}, t \models [\alpha]\varphi & \quad \text{if } \forall s, t < s, \mathcal{M}, s \models \varphi \\ \mathcal{M}, t \models [\alpha^{-1}]\varphi & \quad \text{if } \forall s, s < t, \mathcal{M}, s \models \varphi \\ \mathcal{M}, t \models \langle \alpha \rangle \varphi & \quad \text{if } \exists s, t < s, \mathcal{M}, s \models \varphi \\ \mathcal{M}, t \models \langle \alpha^{-1} \rangle \varphi & \quad \text{if } \exists s, s < t, \mathcal{M}, s \models \varphi \end{aligned}$$

由此,我们可以将原 Tense Logic 的 Lin.Z 系统转化为动态 Lin.Z 系统,其内定理如下:

(CT) 所有命题逻辑的重言式

(US) if  $\varphi$  is a theorem, then so is  $\varphi[\psi/q]$

(MP) if  $\varphi$  and  $\varphi \rightarrow \psi$  are theorems, then so is  $\psi$

(TG) if  $\varphi$  is a theorem, then so are  $[\alpha]\varphi$  and  $[\alpha^{-1}]\varphi$

(DB)  $[\alpha](q \rightarrow r) \rightarrow ([\alpha]q \rightarrow [\alpha]r)$   
 $[\alpha^{-1}](q \rightarrow r) \rightarrow ([\alpha^{-1}]q \rightarrow [\alpha^{-1}]r)$

(4)  $[\alpha]q \rightarrow [\alpha][\alpha]q$   
 $[\alpha^{-1}]q \rightarrow [\alpha^{-1}][\alpha^{-1}]q$

(LS)  $\langle \alpha^{-1} \rangle \top$

(RS)  $\langle \alpha \rangle \perp$

(NB)  $\langle \alpha^{-1} \rangle \langle \alpha \rangle q \rightarrow (\langle \alpha^{-1} \rangle q \vee \langle \alpha \rangle q)$   
 $\langle \alpha \rangle \langle \alpha^{-1} \rangle q \rightarrow (\langle \alpha^{-1} \rangle q \vee \langle \alpha \rangle q)$

(FI)  $([\alpha]([\alpha]q \rightarrow q) \rightarrow (\langle \alpha \rangle [\alpha]q \rightarrow [\alpha]q)) \wedge ([\alpha^{-1}]( [\alpha^{-1}]q \rightarrow q) \rightarrow (\langle \alpha^{-1} \rangle [\alpha^{-1}]q \rightarrow [\alpha^{-1}]q))$

注意到,在转换后的动态 Lin.Z 系统中,不再包含内定理(CV),这是因为原来时态模态词的对应性现在已改成动作的对应性.

\*\*注意到在正规动作集中,我们并没有定义测试动作 $\alpha?$ ,这样做的原因主要有两点:首先,动作 $\alpha?$ 是一个非线性序的分支动作,体现了动作的跳转;其次,含有 $\alpha?$ 的逻辑系统只具有模型完全性,不具有更高的框架完全性,影响了逻辑系统的公理化性质,故不采用.

由于动态 Lin.Z 系统是原 Lin.Z 系统的简单动态逻辑转换,因此用同样的方法可以证明其具有可靠性、完全性和可判定性,详见文献[10].

针对动态 Lin.Z 系统,在例 1 和例 2 中,我们可以将知识表示为

- 1)  $\langle \alpha^{-1} \rangle \langle \alpha \rangle F(\exists c. Works(r,c) \wedge \neg \exists c'. (\langle \alpha \rangle \langle \alpha^{-1} \rangle Works(r,c') \wedge c' \neq c));$
- 2)  $\langle \alpha^{-1} \rangle \exists c. Works(r,c) \wedge \neg \exists c. Works(r,c) \wedge \langle \alpha \rangle \exists c. Works(r,c).$

在这里,我们不再使用静态的模式连接词对知识进行表达,而是使用了可具有丰富内涵的表示动作的模式连接词进行表达.例如,这里可将基本动作  $\alpha$  定义为时间轴上的前向搜索动作,后文中我们将  $\alpha$  定义为直接后继搜索动作,有兴趣的读者也可以尝试着将其定义为其他更具语义特色的动作.

### 3.2 动作参数化处理

由于引入了时间轴上的搜索动作,因此动态 Lin.Z 系统较静态的原 Lin.Z 系统更为精准地刻画了时间轴.然而为了进一步刻画时间轴并扩展逻辑系统,针对时态数据库应用,我们仍可进一步在其上作参数化处理<sup>[5,6]</sup>,例如在语义上将  $\alpha$  定义为直接后继.

**定义 4(模型与赋值 3).** 令时间轴模型  $\mathcal{M} = \langle T, <, \pi \rangle$ , 其中  $T$  为时间点的可能世界集,  $<$  为  $T$  上的一个二元关系,  $\pi$  为其上的赋值定义, 则对于  $\forall t, s, u \in W$ :

- $$\begin{aligned} \mathcal{M}, t \models [\alpha] \varphi & \quad \text{if } \forall s. t < s \wedge \mathcal{M}, s \models \varphi \wedge \neg \exists u. t < u \wedge u < s \\ \mathcal{M}, t \models [\alpha^+] \varphi & \quad \text{if } \forall s. t < s, \mathcal{M}, s \models \varphi \\ \mathcal{M}, t \models \langle \alpha \rangle \varphi & \quad \text{if } \exists s. t < s \wedge \mathcal{M}, s \models \varphi \wedge \neg \exists u. t < u \wedge u < s \\ \mathcal{M}, t \models \langle \alpha^+ \rangle \varphi & \quad \text{if } \exists s. t < s, \mathcal{M}, s \models \varphi \end{aligned}$$

为节省篇幅,在定义 4 中我们省略了其他连接词和过去时连接词的定义,只用将来时连接词作突出定义.这里表明了  $\alpha$  是一个直接后继动作,发生  $\alpha$  动作的两个状态点之间不存在其他状态点作为过渡.由此根据传递性,原  $G\varphi$  的语义也改为  $[\alpha^+] \varphi$ . 注意到,  $\alpha^*$  在这里是不可以使用的,因为  $\alpha^*$  包含了动作  $\alpha^0$ , 它是一个空操作.

同时我们注意到,此时的  $\alpha$  动作在算子;和  $[\ ]$  下是可以叠加的,例如  $[d^m][d^n] \varphi = [d^m; d^n] \varphi = [d^{m+n}] \varphi$ , 特殊地,  $[\alpha][\alpha^{-1}] \varphi = [\alpha; \alpha^{-1}] \varphi = [\alpha^0] \varphi = \varphi$ .

在这里,我们实际上等于在原系统上扩充了两个一元时态连接词,  $[\alpha]$  相当于经典时态逻辑中表示将来直接后继的  $\circ$ ,  $[\alpha^{-1}]$  相当于表示过去直接后继的  $\bullet$ . 据此,我们可以表示更复杂的知识,如:

例 3: 在表 1 中找出从来没有在一间公司工作时间超过连续两年的员工. 其知识表示为

$$[(\alpha^{-1})^+] [\alpha^+] (\neg \exists c. Works(r,c) \wedge [\alpha] Works(r,c) \wedge [\alpha^2] Works(r,c)).$$

上例中,  $[\alpha^2]$  实际上是  $[\alpha; \alpha]$  (或  $[\alpha][\alpha]$ ) 的叠加,是连续两个正规动作的迭代,也是将  $\alpha$  数值化处理后的结果.更进一步地,我们还可将其函数化处理.

动态逻辑的一个最突出的特点就是其语义是模仿程序语义的<sup>[10]</sup>, 尤其适合于计算机科学理论和应用. 在对于  $\alpha$  的处理中,可进一步模仿程序对其处理,例如增加函数  $Til(t)$ , 它表示截止到时间  $t$  为止.

直观上,  $\alpha^{Til(t)}$  表示一直执行动作  $\alpha$  直到时间  $t$  为止,其与经典的时态逻辑连接词 until 类似,但又有所不同. 主要区别在于 until 是连接词,作用在命题上;  $Til(t)$  是函数,作用在动作上.

$Til(t)$  函数的使用是有效的,例如我们若在系统中想表示规则  $\varphi$  的有效期截止 2006 年,则可以将上述规则表示为  $[\alpha^{Til(2006)}] \varphi$ ; 类似地,我们还可以增加函数  $Snc(t)$ , 它与连接词 since 类似,但同样只作用在连接词上. 例如,表 1 中 Marry 在 HP 公司的工作时间为 2002 年~2006 年(即规则的生存周期),可将其表示为

$$[\alpha^{Snc(2002) \wedge Til(2006)}] Works(Marry, HP).$$

更进一步地,尽管在时态数据库查询中 Now 存在着语义过大和语义过小的问题,而且它是一个动态的概念. 但是作为时间轴上的点,在时间轴的动态搜索过程中, Now 是可以定义和采样的. 因此,如果在系统中想表示规则  $\varphi$  的生存周期是从 2007 年至今,则可表示为  $[\alpha^{Snc(2007) \wedge Til(Now)}] \varphi$ . 见表 1, Tom 在 HP 公司的工作时间则可据此表示为  $[\alpha^{Snc(2007) \wedge Til(Now)}] Works(Tom, HP)$ . 这样的表示由于是建立在已刻画的时间轴上,因此,其语义是不模糊的,动态的.

## 4 结束语

通过上述逻辑系统的建模,我们从公理系统角度上较为“精细”地刻画了时态数据库时间轴的性质.尤其是在动态逻辑刻画部分,不仅刻画了时间轴的一般性质,而且刻画了时间轴的动态性,同时还体现了时态数据库的一些特点,例如规则的生存周期和“Now”节点等.其中,借助于动态逻辑程序语义的特点,动作参数化的知识表达和解决方法也为时态数据库一些开问题的解决带来一定的启发,对后续时态知识表达和时态数据库查询的研究起到了积极的作用.

### References:

- [1] Chomicki J, Saake G. Logics for Databases and Information Systems: Temporal Logic Information Systems. Boston/Dordrecht/London: Kluwer Academic Publishers, 1998.
- [2] Gabbay DM, Hodkinson I, Reynolds M. Temporal Logic: Mathematical Foundations and Computational Aspects. Oxford University Press, 1994.
- [3] Jensen CS, ed. Temporal database management. 2000. <http://www.cs.auc.dk/~csj/Thesis/>
- [4] Fox C, Lappin S. Foundations of Intensional Semantics. Malden, Oxford, Carlton: Blackwell Publishing, 2005.
- [5] Veltman F. Imperatives in update semantics. In: Proc. of the 5th Int'l Conf. on Logic and Cognition. Guangzhou, 2006. 128–134.
- [6] Hasle PV. Temporal Logic from Ancient Ideas to Artificial Intelligence. Netherlands: Kluwer Academic Publishers, 1995.
- [7] Clarke EM, Grumberg JO, Peled DA. Model Checking. Cambridge: The MIT Press, 1999.
- [8] Wooldridge M, Huget MP, Fisher M, Parsons S. Model checking for multiagent systems: The mable language and its applications. Int'l Journal on Artificial Intelligence Tools (IJAIT), 2006,15(2):195–225.
- [9] Goble L. The Blackwell Guide to Philosophical Logic. Malden, Oxford: Blackwell Publishers Inc., 2001.
- [10] Harel D, Kozen D, Tluy J. Foundations of Computing Series: Dynamic Logic. Cambridge: The MIT Press, 2000.



刘冬宁(1979—),男,江西永新人,博士,CCF会员,主要研究领域为时态数据库,人工智能逻辑.



汤庸(1964—),男,博士,教授,博士生导师,CCF高级会员,主要研究领域为数据库,协同软件.