

面向对象的可视知识研究

何克清 马江 吕美玲 谢彪 金明源

(430072武汉大学软件工程国家重点实验室)

摘 要

As a research progress, this paper presents a new study on object-oriented visual knowledge. First, we propose some concepts and primary achievements, such as object-oriented visual knowledge, knowledge structure, inheritance, knowledge driven and inference, classification, etc. Then, we discuss the application prospect. Also, we put some problems in further research.

1. 前 言

人们对知识的认识和积累可以说是从可视的“形”开始的。在对图形、图象认识的基础上通过抽象获得相应的知识。当今,大量利用抽象知识的同时,我们不应该忽视可视知识的研究。而且,可视知识的利用十分直

型常元表示。第二,建立对象逻辑中项(公式)与第一步得到的类型的有效项之间的一一对应。第三,用“判断作为类型”的原则解释对象逻辑中的推导规则和证明。这样,ELF就为解释和实现各类逻辑系统提供了统一的框架和方法。

3. 概念作为类型

“概念作为类型”是李未在ALT类型理论中提出的原则。在软件工程和知识工程中有许多概念都是成对出现的,比如规范和程序,任务和解答,问题和算法,命题和证明等等,如果用类型理论中类型及其对象的定义来刻画这些概念,就可以利用类型理论中的方法来解决软件工程与知识工程中的各种问题。目前,ALT类型理论已用于描述变换式程序设计,为形式化描述软件开发过程打下

观、直接、自然、有效。

软件工程的发展,已形成了自己实用的图式方法与技术,如DFD, Petri网, PAD, FC, SC等知识。如何表示和利用这些可视知识,是软件工程中的重要课题之一。

知识工程所取得的方法与技术为可视知

了基础。

近来国外也出现了类似的工作,比如程序开发演算^[26]和软件开发的类型模型OROS^[27]。

六、结 论

类型理论和构造性数学都起源于对证明论和数学基础的研究,近来在计算机科学中引起了广泛注意^[20]。类型理论已成为函数式程序设计的逻辑基础^[25],与面向对象程序设计有密切的关系^[18];它不仅对证明检查、自动定理证明有重要意义,而且可作为人工智能中的知识表示语言,它与逻辑、范畴论、语义理论相结合,为软件开发的形式化与自动化开辟了诱人的前景。(参考文献共29篇略)。

识系统的研究奠定了基础, 计算机图形工作站等设备的出现提供了良好的环境和手段。软件开发中的可视知识系统的研究大有希望。

本文在已有研究^[1]的基础上, 提出了面向对象的可视知识的概念, 讨论了它的结构、继承性、推理, 并给出了一些应用分析。同时, 也提出了一些研究问题。

S.K.Chang关于可视语言^[1]的工作和J.W.Tai关于可视知识^[2]的观点将是我们讨论的基点。

2. 什么是面向对象的可视知识

文[2]中将可视性知识划分为两类, 即:

a. 为了表示可视性知识的知识, 称为第一类可视知识;

b. 被可视化的知识, 称为第二类可视知识。

其中, 第一类可视知识在表示与推理上与传统知识相同, 但本身不是可视的。其知识驱动的行为不仅包括推理中的一些行为, 还要包括将图形显示到终端上的行为, 从终端定义图形的行为, 将图形映射到计算机中的行为。第一类可视知识是可视知识的基本部分。

第二类知识本身是可视的。运用一些基本图形(S.K.Chang称为icon)来构造复杂的icon, 并利用icon之间的关系表示其结构、行为和活动。在当前的计算机系统中, 我们几乎不可能构造出一个完全由第二类可视知识组成的系统, 因为获取icon集合及构造复杂的icon的知识为第一类可视知识, 而且是必不可少的。

那末, 什么是面向对象的可视知识呢? 可视知识是直接面向对象的, 它有如下二个方面的意义。

a. 面向对象的可视知识属第二类可视知识;

b. 使用面向对象模型的知识表示和利用。

3. 面向对象可视知识的结构

可视知识的结构包括两部分, 即“形”和“意”。S.K.Chang在1985年提出了icon理论, 他认为一个可视语言(Visual Language)可表示为一个树文法, 节点为icon。节点的结构可以用一个双重表达式说明, 即 (X_i, X_j) 。其中 X_i 表示icon的物理部分, 即the image, 也是可视知识的“形”部分; X_j 表示与这个物理部分有关的意义部分, 即logical part, icon之间的关系往往用一组规则表示。如图1所示。图中, ip是icon的image部分, 即“形”; lp是icon的逻辑部分, 即“意”, R是icon之间的关系。

由图1组成的结构, 同样也可以作为可视知识的结构。

其中, ip的表示方法有图文法, 属性文法^[3]等; lp的表示方法有概念图(conceptual graph)^[4], 属性文法等。

下面, 我们利用面向对象的知识表示和利用模型来加以描述, 得到面向对象的可视知识的结构。

如图2所示, 知识单元icon对应着面向对象模型中的知识单元对象(Object)。icon的ip和lp均由属性和操作两部分组成, 它们的

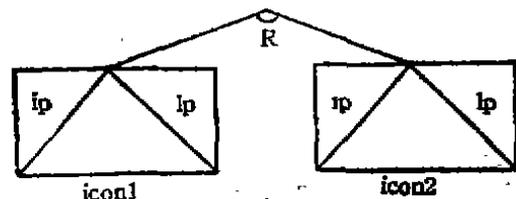


图1

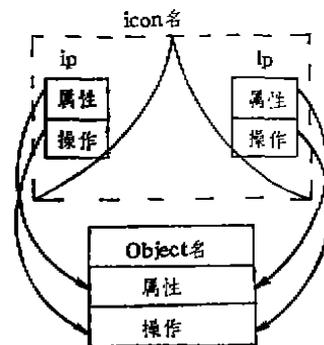


图2

属性之和对应着对象的属性,操作之和对应着对象的操作。

对象的属性包括icon的操作数据、状态、性质、约束、以及与其它icon之间的静态关系,如继承关系、父子关系、逆关系等。

对象的操作包括icon内ip和lp的操作,操作由方法(method)集组成。

在面向对象的可视知识系统中,我们称上述知识单元为Object-icon,简称为Oicon。Oicon同样也有class和instance之分,其含义与传统意义相同。

图3给出了一个“银行”的例子,其ip和lp中的属性、操作组成了Object中的属性与操作。所以Oicon的结构比对象的结构更精细,因为在传统的对象结构中,一般不区分“形”的属性与“意”的属性以及“形”的操作与“意”的操作。

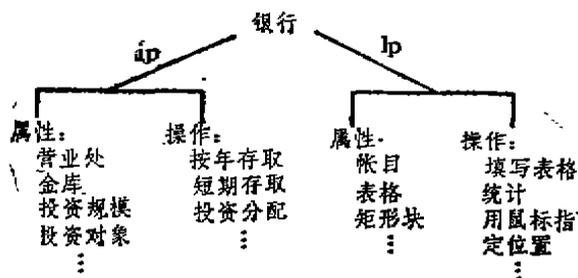


图3

4. 关于继承性

在面向对象的可视知识系统中,Oicon之间同样具有有如[5]所述的继承性:

APO(A-Part-Of)继承:一个Oicon和它的组成部分(Oicon)之间有抽象继承关系,一个Oicon继承它的组成Oicon的属性和操作。

AMO(A-Member-Of)继承:一个集合Oicon可以继承它的元素Oicon的属性和操作。

AKO(A-Kind-Of)继承:一个类型Oicon可以继承它的Super类型的属性和操作。

AIO(A-Instances-Of)继承:一个例示

Oicon继承它所在类型Oicon的属性和操作。

这里,继承的含义与面向对象计算模型中的含意完全相同。值得说明的是,关于Oicon的图形结构的继承内含在ip的属性和操作的继承之中。

当然,也存在单重继承与多重继承的区别。

5. 消息与消息传递

在面向对象的可视知识系统中,Oicon之间的动态关联是通过消息及消息传递来实现的,从而进行知识驱动或推理。

面向对象的可视知识与面向对象的知识在表示方法与利用方法上是相同的。不同的是前者强调知识的图形表示与构造,是可视的;后者不是可视的。

除传统意义下的对象间的消息及消息传递外,这里,应注意可视知识的结构表达及结构推理的研究。这是一个深专家系统(deep expert system)的研究课题。关于Oicon的获取问题还涉及到类比学习(analogy)的研究。这也是下一代专家系统中研究的课题之一^[2]。相应地,除传统的消息模式(一元消息,二元消息,键字消息)外,如何根据Oicon的特点扩充消息模式,也是一个重要的研究课题。

6. 面向对象的可视知识的分类

在以上讨论的基础上,我们可以对面向对象的可视知识系统进行分类,以利于我们探讨和发展这类知识系统的策略。

文[2]中,对于可视知识结构的三要素,使用两个分类原则进行了分类。根据Oicon特点,本文扩充了第二个分类原则,因而增加了知识系统的类型。

原则1^[2] Oicon中的三要素的任一个是采用1-D(第一类可视知识)或2-D(第二类可视知识)表示。

原则2 Oicon三要素中的任一个是系统事先提供或用户可定义^[2],或者系统动态产生。

原则2中“系统动态产生”的含义是:在

面向对象的知识模型中，Oicon之间存在着继承关系，系统可以动态产生新的Oicon。例如，

ins(a): 接受消息的类Oicon生成对应的例示(instance)Oicon，取名为a。

根据第一个原则，系统分类列表如表1所示。

表1 第一种分类^[2]

类型	ip	lp	R
D·0	1-D	1-D	1-D
D·1	2-D	1-D	1-D
D·2	2-D	2-D	1-D
D·3	2-D	2-D	2-D

D·0类面向对象的可视知识系统全部由第一类知识组成，因而，它与传统的知识系统没有什么区别。

D·3类面向对象的可视知识系统全部由第二类知识组成，也就是说，其所有知识表示和利用可以全部用图形来描述。这是针对特殊领域的知识类型，如动画系统的知识领域等。

很显然，D·0是基础，如果没有D·0支持，D·1、D·2、D·3都不可能成为非常有效的系统。

根据扩充了的原则2，系统分类如下：

表2 第一种分类

类型	ip	lp	R
S·0	S	S	S
S·1	U	S	S
S·2	U	U	S
S·3	U	U	U
S·4	S	U	S
S·5	S	S	U
S·6	U	S	U
S·7	S	U	U
S·8	T	S	S
S·9	T	T	S
S·10	T	T	T
S·11	S	T	S
S·12	S	S	T

S·13	T	S	T
S·14	S	T	T
S·15	T	U	U
S·16	T	T	U
S·17	U	T	U
S·18	U	U	T
S·19	T	U	T
S·20	U	T	T
S·21	S	U	T
S·22	S	T	U
S·23	U	S	T
S·24	U	T	S
S·25	T	S	U
S·26	T	U	S

其中，S表示系统事先提供的，U表示为用户可定义；T表示系统动态产生的。

S·0类型表示所有Oicon的要素都是由系统事先提供，用户只能使用，不能定义。这对于任何一种高级语言都是自然的，但对可视知识系统而言是很不方便的。

S·3类型表示所有Oicon的要素都是由用户可定义的。这样的系统的效率是非常有限的。

S·10类型表示所有Oicon的要素都是由系统动态产生的，这只有理论上的意义，实际上不可能存在。

S·1~S·26中的任何类型都应该基于S·0，也就是说S·0类型是一个面向对象的可视知识系统所需要的最基本功能。

7. 应用

可以说，面向对象的可视知识在人-机交互式环境、动画、软件开发、信息系统的开发中应用广泛，下面分析应用前景。

①Hypersystem中的应用

Hypertext 是一种管理信息的技术，它将数据或信息存储在许多节点中，而节点中的数据可以是正文、源程序、图形、图象、动画、声音及其组合。利用这种方式而构成的系统称为Hypermediasystem，简称为Hyper-system。

Hypercard 是美国苹果公司1987年开发的以文字、图表、图片、数字化照片为基本信息单元(卡片)的 Hypersystem。一张卡片相当于一个节点,卡片内容可以通过工具构造。工具包括icon及icon库、编辑器等。

我们可用Oicon及Oicon库构造更为复杂的Oicon,形成知识 Card,利用知识Card构造更为复杂的知识,……。在Oicon之间,或在Card之间通过消息传递实现知识驱动和推理,从而开发知识处理类型的 Hypercard 系统。

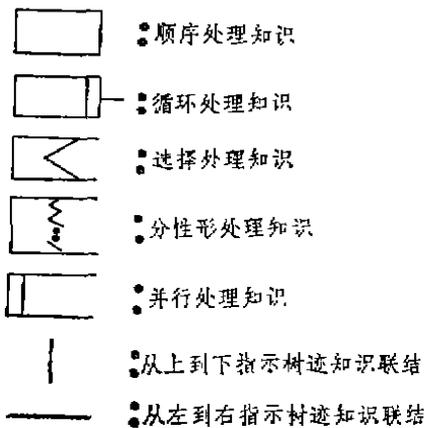
② 软件开发中的应用

软件开发中的实用方法与技术都与图形方法有关。如DFD方法, Petri网方法, PAD方法, FC方法等。下面我们以可视程序设计知识系统为例说明。

“可视”在本例中指PAD^[1]图。一个程序设计与调试包括以下三个方面:

- a. 程序的静态结构。在本系统中,静态结构(包括数据结构)是用PAD表示的。
- b. 程序的动态行为。本系统是由Oicon之间消息(控制与数据)传递表示的。
- c. 界面。本系统使用PAD表示输入、输出数据结构。

可视知识的基本图形Oicon可定义如下:



以上的系统的图式元素ip, 均可作为类Oicon。而lp和R都可利用传统规则和方法表示。如图4所示。

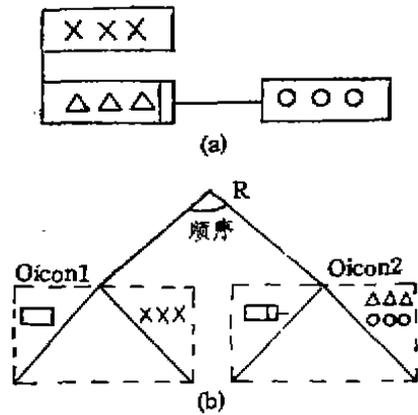


图4

图(a)所示程序设计知识表示为图(b)所示Oicon构造。其中ip, lp部分的描述可以采用传统程序设计语言或PDL, R部分根据算法知识可以利用规则描述。由于PAD方法中, 7种PAD图形的类(class) Oicon可以由系统给出, 所以本系统为S.7类型。

8. 结论

本文作为一个研究讨论, 并没有讨论面向对象可视知识的表示、利用的各种详细方法与技术。但作为开端, 提出了一些概念与结构供大家讨论。我们将继续进行各个问题的详细研究, 陆续发表。

参 考 文 献

- [1] Shi-Kuo Chang, Visual Languages: A Tutorial and Survey, IEEE Software Vol.4 No.1 Jan. 1987
- [2] 戴汝为等, 关于可视知识的讨论, 自动化所人工智能部研究报告1990
- [3] 戴汝为等, A Syntactic-Semantic Approach for Pattern Recognition and Knowledge Representation
- [4] 何克清, 计算机软件工程学, 武汉大学出版社 1983
- [6] 何克清, 软件开发知识的表示与利用模型SOKM, 《面向对象的软件工程研究》, 武汉大学出版社 1989.10