

普适环境中上下文感知的带演算的 Bigraphs 描述

吴怀广¹ 姬厚灵² 毋国庆² 苗 玥³

(郑州轻工业学院计算机与通信工程学院 郑州 450002)¹ (武汉大学计算机学院 武汉 430079)²

(内蒙古包头医学院计算机科学与技术系 包头 014040)³

摘 要 针对普适环境中上下文感知系统的形式化建模问题,讨论了 Bigraphs 理论的扩展模型——柏拉图图形模型在此问题中的适用性及不足。在此基础上,利用带演算的 Bigraphs 反应系统对上下文感知系统进行描述,并给出了与柏拉图图形模型之间的比较。

关键词 Bigraphs 理论, Bigraphs 反应系统, 上下文感知, 形式化模型

中图分类号 TP311 **文献标识码** A

Calculational Bigraphical Model of Context-aware Systems in Ubiquitous Computing Environment

WU Huai-guang¹ JI Hou-ling² WU Guo-qing² MIAO Yue³

(College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China)¹

(School of Computer Science, Wuhan University, Wuhan 430079, China)²

(Department of Computer Science, Baotou Medical College, Baotou 014040, China)³

Abstract In ubiquitous computing environment, it is a important research to model the context-aware systems formally. Firstly, we discussed applicability and deficiency of a extended bigraphical model of context-aware systems called Polato Graphical Model. On the basis of analysis of Polato Graphical Model, we presented a calculational biographical model for description of context-aware systems and compared it with Polato Graphical Model through an example.

Keywords Bigraphs theory, Bigraphs reactive systems, Context aware, Formal models

1 引言

普适计算的思想是由 Xerox 的 Mark Weiser 于 1991 年首次提出的一种超越桌面计算的全新计算模式。它建立在分布式计算、通信网络、移动计算、嵌入式系统、传感器等技术飞速发展和日益成熟的基础上,体现了信息空间与物理空间的融合趋势,反映了人们对信息服务模式的更高需求——希望能随时、随地、自由地享用计算能力和信息服务,使人类生活的物理环境与计算机提供的信息环境之间的关系发生革命性改变^[1]。

计算的无所不在性、透明性以及自适应性是普适计算的关键特征。其中,无所不在性是指人们可以在工作和生活的任何场所获得服务;透明性是指获得这种服务的方式甚至可以在用户没有觉察的情况下由系统自主提供;自适应性实际上是强调如何有效感知物理空间、信息空间乃至设备和人的行为变化,使系统能动态调整自身的行为^[2]。计算能力的无所不在性,将通信和计算机构成的信息空间与人们生活和工作的物理空间融为一体,使其成为真正的智能空间^[3,4]。智能空间中,人们不断地与不同的计算设备进行隐形的交互,而在此过程中,计算系统需要根据与用户任务相关的上下文信息来确定向用户提供什么样的服务。因此,上下文感知是实

现普适计算环境中新型人机交互的基础,也是普适计算研究与实现中的支撑技术。

对于上下文感知的研究是多方面的,但其中多数是面向应用层级的。而关于如何建立一个形式化的模型,即模拟系统和环境之间的交互并对模型进行相关的推理的研究却相对较少。尽管研究者们已经意识到这个问题并且尝试对其进行定义、分类以及建模^[5-8],但是,其中多数定义是非形式化的,缺乏可靠的数学基础。因此,需要一个形式化的框架来理解和支持普适计算及上下文感知系统。

由 Robin Milner 及其合作者提出的 Bigraph 理论^[9-14]是一种基于图形的形式化理论工具,它同时强调了计算(物理或虚拟)的位置和连接,是比传统移动进程演算更一般的理论。该理论的主要目标是:(1)对普适计算系统进行建模;(2)为现有的移动与并发理论提供一个统一的元模型。Birkedal 等对利用 Bigraphs 反应系统(Bigraphical Reactive System, BRS)对上下文感知系统建模进行了深入的研究,提出了所谓的柏拉图图形模型(Plato-Graphical Model, PGM)。利用该模型,作者对一个简单的位置感知打印服务系统进行描述,并得出该模型适合对位置感知系统进行建模的结论^[15]。尽管 L. Birkedal 及 S. Debois 等人的技术报告^[16]中给出了在分类的 PGM 定义下的相关推出(Relative Push-out, RPO)及其衍

生迁移系统互模拟关系的证明讨论,但根据 Milner 关于同态赋类的定义,PGM 的分类规范却不是同态的^[11],并且存在反例使其不能满足文献[14]中给出的判定分类是否是安全的条件。

2009 年 10 月,在英国爱丁堡大学举行的 Bigraphs 模型发展研讨班上,Robin Milner 提出了采用在 Bigraphs 反应系统中嵌入演算系统的方式来描述上下文感知系统的想法。从形式上讲,与 PGM 一样,带演算的 Bigraphs 反应系统也是遵循整合 Bigraphs 反应系统的方法,将常规反应系统 $B_i = (\mathcal{K}_i, \mathcal{R}_i)$ 和 $B_c = (\mathcal{K}_c, \mathcal{R}_c)$ 演算反应系统组合在一起。但是,演算的 Bigraphs 反应系统较之前者更加严谨、成熟,与原有的 Bigraphs 理论有着更好的相容性,可以很好地处理 PGM 难以解决的诸如上面提到的分类的安全性以及同态赋类等问题。鉴于此,本章将通过在 Bigraphs 反应系统中嵌入执行演算功能的 Bigraphs 反应系统对上下文感知系统进行描述。随后,描述了文献[15]中给出的经典例子并通过与 PGM 的对比阐释了具有演算功能的 Bigraphs 反应系统描述上下文感知系统的表达能力。

本文第 2 节给出 Bigraphs 的基本理论以及 PGM 的相关介绍;第 3 节讨论了带演算的 Bigraphs 反应系统;第 4 节给出一个经典案例分析;最后总结全文并探讨未来的工作。

2 Bigraphs 基础及柏拉图图形模型介绍

2.1 Bigraphs 基本介绍

单个 Bigraphs 可以分解为两个子图:位置图(place graph)用以表示各个计算节点的所在位置,节点之间允许嵌套;连接图(link graph)则只描述节点间的连接关系,而忽略节点间的位置关系。位置图和连接图是从两个不同的角度对同一个 Bigraphs 观察所得到的结果。在纯 Bigraph 中,位置图和连接图是相互独立的,此处参考文献[9]对 Bigraphs 进行介绍。

如图 1 所示,在 Bigraph $F: \in \rightarrow \langle 3, \{xy\} \rangle$ 中,虚线框表示一个区域(region),每个 Bigraph 中的区域个数用一个自然数 n 来标识, n 是一个有限的序数,即 $n = \{0, 1, \dots, n-1\}$ 。 v_1, v_3, v_4, v_5 为 Bigraph 中节点的标识, v_1, v_3 分别隶属于区域 0, 1, v_4 嵌套了 v_5 隶属于区域 2。图 1 的左下方位置图 F^P 就表示了以区域为树的根节点的森林。 e_1, e_2, x, y 表示 F 中 v_1, v_3, v_4, v_5 的连接关系。其中, e_1, e_2 是分别连接两个节点的边,称之为封闭的连接。 x, y 只有一端分别连接在 v_1, v_4 上,而另外一端与其它 Bigraphs 进行连接。在 Bigraphs 中规定在 Bigraphs 图上方伸出的连接为外部连接,反之,如图 2 中的 x, y 在图的下方伸出则称之为内部连接,内部与外部连接分别作为连接图的内部名和外部名。

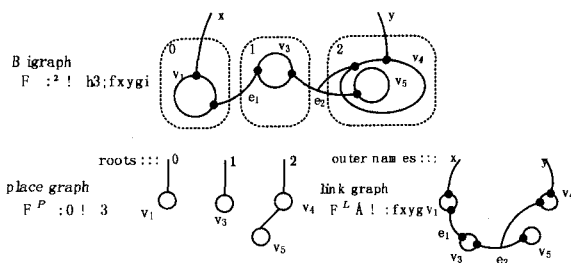


图 1 Bigraph F 之结构图

在图 2 中, Bigraph H 的两个区域包含了用灰色矩形表示的 3 个地点(site)。地点的表示方式和区域的表示方式一样,地点和区域构成了位置图的内部界面和外部界面。因为每个 Bigraph 都是由位置图和连接图组合而成的合成图,所以, Bigraphs 的内部界面形式为 $\langle m, X \rangle$, 其中 m 为地点总数, X 为连接图中内部名集合。同理, Bigraph 的外部界面的形式为 $\langle n, Y \rangle$, 其中 n 为区域总数, Y 为连接图中外部名集合。对于 Bigraph F , 其内部界面形式为 $\langle 0, \phi \rangle$, 记为 $\in$, 称之为源;其外部界面形式为 $\langle 3, \{xy\} \rangle$, 即有 0, 1, 2 3 个区域, x, y 两个外部名。对于 Bigraph H , 其内部界面形式为 $\langle 3, \{xy\} \rangle$, 即有 0, 1, 2 3 个地点, x, y 两个内部名;其外部界面形式为 $\langle 2, \phi \rangle$, 也就是说, 有两个区域但没有连接的外部名。

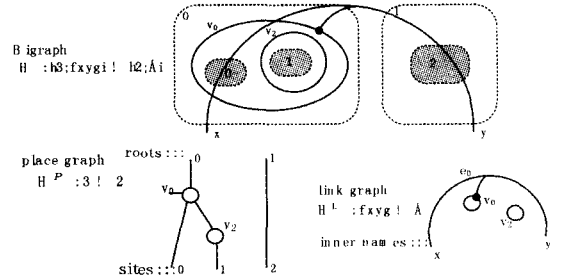


图 2 Bigraph H 之结构图

从范畴论的观点,将位置图视为一个态射,其对象是序数集,则位置图 $G^P: m \rightarrow n$ 表示 G^P 具有 m 个地点和 n 个区域。同样地,连接图则是对象为名字集的态射,连接图 $G^L: X \rightarrow Y$ 表示 G^L 具有内部名 X 和外部名 Y 。Bigraphs 的对象则是序数集和名字集构成的二元组,即有 $G: \langle m, X \rangle \rightarrow \langle n, Y \rangle$, 一个 Bigraph G 可以写成 $\langle G^P, G^L \rangle$ 的形式。Bigraph F 和 Bigraph H 的 3 种态射都在相应的图中给出了相应的表达式。

Bigraphs 间的组合有一种基本的组合形式:复合。复合规则要求参与复合的两个 Bigraphs 的节点和边是互不相同的,并且还要满足范畴论中两个态射之间复合的条件。例如,在 Bigraph F 和 Bigraph H 中,其节点 $\{v_1, v_3, v_4, v_5\} \cap \{v_0, v_2\} = \phi$;其边 $\{e_1, e_2\} \cap \{e_0\} = \phi$; $F: \in \rightarrow \langle 3, \{xy\} \rangle$ 的陪域与 $H: \langle 3, \{xy\} \rangle \rightarrow \langle 2, \phi \rangle$ 的定义域相同。所以,设 F 和 H 进行复合操作生成的 Bigraph 为 G , 则 $G = H \circ F = \in \rightarrow \langle 2, \phi \rangle$ 。生成的图 G , 如图 3 所示。

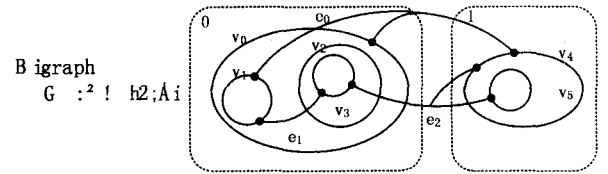


图 3 Bigraph F 和 H 复合操作

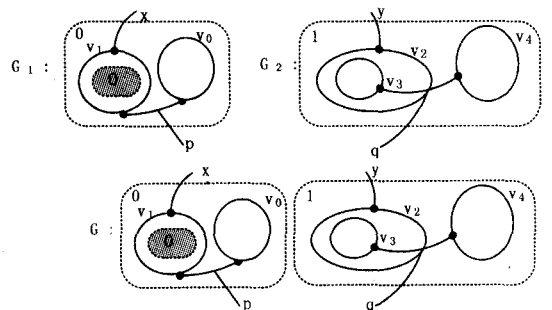


图 4 Bigraph 扩展张量积

前面的这种复合操作有时称为垂直合成,它是将 F 的区域驻留到 H 的地点中,然后将 F 的外部名和 H 的内部名连接到一起。还有一种组合操作称之为水平合成一张量乘积即除了要求参与合成的 Bigraphs 的节点和边互不相交以外, Bigraphs 的内部界面和外部界面也互不相同。该操作只是将两个 Bigraphs 按照“从左到右”的顺序并列在一起。例如, $G_1: h1; fpgi! h1; fxgi, g_2: hA; fggi! h1; fgyi$, 则 $G = G_1 \otimes G_2 = G: \langle 1, \{pq\} \rangle \rightarrow \langle 2, \{xy\} \rangle$, 其操作如图 4 所示。

Bigraphs 的动态变化是通过 Bigraph 反应系统来进行表述的。反应关系中的一个反应规则 (R, R') 表示一个 Bigraph 对自身进行重新配置后变化成为另一个 Bigraph R' , 其中 R 称为反应物, 而 R' 称之为生成物。一条反应规则的反应物和生成物中可以带有限的任意多的参数, 并且反应物和生成物的参数具有映射关系, 这也使得反应物的参数可能被复制或丢弃。反应规则可以根据具体的应用进行定义, 如果 Bigraphs 中有和反应物相匹配的部分, 则该 Bigraphs 可以根据反应规则将其对应部分代替为生成物。例如, 图 5 表述的反应规则表示一个人在家中通过 PDA 将办公室电脑上的文件信息拷贝到 PDA 内部。一个 Bigraphs 反应系统则包含了 Bigraphs 以及一个反应规则集, 其利用范畴论中的重要数学性质 (RPO) 可以推导并证明在 Bigraphs 反应系统中得到的互模拟关系实际上是一个同余。

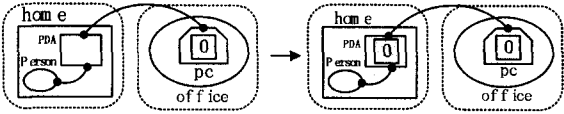


图 5 Bigraph 反应规则

2.2 柏拉图图形模型 (PGM)

为了解决 Bigraphs 反应系统模拟上下文感知系统非常不灵活且难以操作的不足, Birkedal 等给出的柏拉图图形模型是将反应规则不再作为直接执行的对象, 而是将反应规则用作描述发生在现实世界中的变迁以及编译执行语言的操作语义, 如此就可以执行关于现实世界描述上的查询了。他们将上下文感知计算的 Bigraphs 反应系统模型分成 3 个 Bigraphs 反应系统, 其具体结构是: 上下文 C ——模拟应用环境; 上下文的代替物 P ——模拟应用环境有关的信息; 计算代理 A ——查询上下文的代替物的应用。反应规则要么用来表述上下文 BRS 中的动态变迁, 要么用来模拟上下文替代物 BRS 中的查询。文献 [15] 给出的柏拉图图形模型定义可以描述为: 一个柏拉图图形模型是 Bigraphs 反应系统的三元组 (C, P, A) , 并且 $\mathcal{M} = \text{CUPUA}$ 本身也是一个 Bigraphs 反应系统, 其中 $C \perp A$ 。模型的每个状态是一个 Bigraph $\vec{x}. (C \parallel P \parallel A)$, 这里的 $C \in C, P \in P, A \in A, \vec{x}$ 是名字向量组。在此定义中, $C \perp A$ 表示上下文 C 的 $(\mathcal{K}_C, \mathcal{R}_C)$ 和代理 A 的 $(\mathcal{K}_A, \mathcal{R}_A)$ 的独立性关系, 即 $C \perp A$ 成立当且仅当 $\mathcal{K}_C$ 和 $\mathcal{K}_A$ 是不相交的。CUPUA 表示反应系统间的合成, 即, 当 $\mathcal{K}_C, \mathcal{K}_P, \mathcal{K}_A$ 都满足存在于 $\mathcal{K}_C \cap \mathcal{K}_P \cap \mathcal{K}_A$ 元素时, $\text{CUPUA} = (\mathcal{K}_C \cup \mathcal{K}_P \cup \mathcal{K}_A, \mathcal{R}_C \cup \mathcal{R}_P \cup \mathcal{R}_A)$ 。在此模型中, C 中每个元素模拟一个可能的上下文, 而 P 中的每个元素则是模拟一个可能上下文的模型。由于上下文与代理之间的独立性, 代理只能观察或操作上下文的代替物 P , 而不是上下文本身。若代理要查询或变更上下文只有通过使用 P 作为传感器或执行器。

利用 $\mathcal{M} = \text{CUPUA}$ 作为描述上下文感知系统的基本形式体系可以在建模的过程中直接写出相关的反应规则, 进而使得建模变得非常直观易懂。另一方面可以相应地得到一个互模拟关系并且这个互模拟关系是一个同余关系。因此, 代理间的互模拟关系使得上下文及其替代物不能对其进行区分, 也就是说, 如果 $\sim$ 是 $\mathcal{M}$ 中的互模拟关系, 并且对于两个代理 $A, A' \in A$ 之间有 $A \sim A'$, 那么对于任意的 $C \in C, P \in P$ 和名字向量组 $\vec{x}$, 都有 $\vec{x}. (C \parallel P \parallel A) \sim \vec{x}. (C \parallel P \parallel A')$ 。

为了保证在 $\mathcal{M}$ 的状态组合时 C, P, A 各自的控制类是分离的, Birkedal 等给出了位置分类的分类柏拉图图形模型。其定义形式如下 [16]。

定义 1 假设 $\mathcal{M} = \text{CUPUA}$ 是一个柏拉图图形模型, 其中, $C = (\mathcal{K}_C, \mathcal{R}_C), P = (\mathcal{K}_P, \mathcal{R}_P)$ 及 $A = (\mathcal{K}_A, \mathcal{R}_A)$ 。通过 $\Theta = \{\mathcal{K}_C, \mathcal{K}_P, \mathcal{K}_A\}$ 定义 $\mathcal{M}$ 上的分类规范, 并且, 对于初始 Bigraphs 分类的条件为 $\Phi(f; S_{\leq n} \rightarrow S) = \text{ctrl}(f) \subseteq S \wedge S_i = S (\forall i \leq n)$ 。分类的 Bigraph f' 则都可以通过分解成得到 $f' = f_0 \cdots f_{n-1}$, 而且若 $f_i (i \leq n-1)$ 都是良分类的, 那么 f' 也是良分类的。设 ϕ 是一个对反应规则 $\mathcal{R}_C, \mathcal{R}_P$ 和 $\mathcal{R}_A$ 的 Θ 指派, 使得在该指派下每个反应规则都是良分类的。将 $\mathcal{M}$ 定义成是 $\mathcal{M}$ 通过 (Θ, Φ, ϕ) 分类后得到的形式, 即 $\mathcal{M}$ 为所谓的分类的柏拉图图形模型, 对于 $\mathcal{M}$ 的状态则是具有外部名类型为 $\mathcal{R}_C, \mathcal{R}_P$ 和 $\mathcal{R}_A$ 的良分类 Bigraphs。

3 带演算的 Bigraphs 反应系统

若在一个智能建筑中有表示人 (控制类型为 A)、房子 (控制类型为 R) 以及计算机 (控制类型为 C) 所描述的 Bigraphs 系统, 其反应系统有如下规则:

Rule 1: $A_{xy} | R. (-_0 | -_1) \longrightarrow \triangleright R. (A_{xy} | -_1)$

表述一个人进入房间的行为动作。那么如何进一步约束此规则使得 Rule 1 可以表述只有满足一定条件的人才能进入房间呢? 假设只有持规定磁卡的用户才能进入房间, 可以要求每个控制中包含一个 Admit 控制。控制 Admit 中含有的控制类型是身份验证器 Tag_m , 其中 m 表示它有一个端口。另外, 给控制 A 增加一个端口 m 用来和 Tag_m 进行匹配链接。若 A 通过了身份验证器的验证, 则允许 A 进入相应的 R , 其规则精化如下:

Rule 1': $A_{xym} | R. (\text{Admit}. (-_0 | Tag_m) | -_1 | -_2) \longrightarrow \triangleright /mR. (A_{xym} | \text{Admit}. (-_0 | Tag_m) | -_2)$

如果允许进入房间的人可以随意使用房间内的计算机, 其规则如下:

Rule 2: $C_{xy} | A_{xym} \longrightarrow \triangleright /y(C_{xy} | A_{xym})$

更进一步地, 假设只允许符合规定年龄的人才可以对计算机进行操作, 那么在 Bigraphs 的描述中就需要表达出人的年龄与规定年龄进行比较的演算, 继而根据演算结果来判定是否可以使用计算机。这样, 在智能建筑的 Bigraphs 反应系统中需要一个子 Bigraphs 反应系统 BRS_{ml} 来完成演算的功能。 BRS_{ml} 的反应关系由以下 $L1; L2; L3$ 3 条反应规则组成:

L1: $Leq. (-_0 | Zero) \longrightarrow \triangleright \text{True}$

L2: $Leq. (\text{Succ}. (-_0) | Zero) \longrightarrow \triangleright \text{False}$

L3: $Leq. (\text{Succ}. (-_0) | \text{Succ}. (-_1)) \longrightarrow \triangleright Leq. (-_0 | -_1)$

至此, 可以将提供演算能力的 BRS_{ml} 嵌入到智能建筑的 BRS 反应系统中, 上面的规则 Rule 2 扩展后就可以表达使用

计算机的条件。首先,在反应规则中引入演算:

$$\text{Rule 2.1: } A_{ym} | \text{Age. } -_0 | C_{ye}. (\text{AgeAdmit. } -_1 | -_2) \longrightarrow / y(A'_{ym} | \text{Age. } -_0 | C'_{ye}. (\text{AgeAdmit. } -_1 | -_2 | \text{Pass.} \\ (\text{Leq. } (-_0 | -_1)))$$

该规则中的 A'_{ym} 、 C'_{ye} 表示将 A_{ym} 、 C_{ye} 委托到交互的反应中去,如果演算的结果允许发生反应,那么委托成功,否则委托被撤销。生成物中的 $/y$ 表示此时的链接是封闭的,也就是说,当前的演算对于 A'_{ym} 和 C'_{ye} 都是唯一的。

接下来就是演算是否允许请求用户来操作计算机,其演算反应规则如下:

$$\text{Rule 2.2: } /y(A'_{ym} | C'_{ye} (- | \text{Pass. True})) \longrightarrow /y(A_{ym} | C_{ye})$$

$$\text{Rule 2.3: } /y(A'_{ym} | C'_{ye} (- | \text{Pass. False})) \longrightarrow (A_{ym} | C_{ye})$$

实际上,规则 Rule 2.1、Rule 2.2 以及 Rule 2.3 执行了一个判定条件。尽管该条件比较简单,但是从概念上讲,它说明了可以将演算应用到任意复杂的情况中。因此,嵌入的演算增加了十分重要的表述能力。另外,由上面的反应规则可以看出,原智能系统的控制类也因增加演算功能的缘故而有所增加。

在形式上,若原智能建筑 Bigraphs 模型系统记为 $B_G(\mathcal{K}, \mathcal{R})$,在嵌入一个演算 $B_G(\mathcal{K}', \mathcal{R}')$ 后其构造为 $B_G(\mathcal{K} \cup \mathcal{K}', \mathcal{R} \cup \mathcal{R}')$ 。嵌入 BRS_{int} 的和原有的 BRS 是如何进行相互作用的以及在进演算时会不会破坏原有反应系统中的生成物都是自然要考虑的问题。基于这些问题,文献[17]对包含 BRS_{int} 的 Bigraphs 反应系统进行了相关的探讨研究,并且给出了一些重要的结果。本节就是在此基础上描述上下文感知的 Bigraphs 演算系统。

3.1 演算 Bigraphs 反应系统

基于文献[1]中关于生长理论的研究,Robin 及其合作者将种子发展成为带有参数的非原子的形式,用模式匹配的方法来扩展相应的规则,继而得到具有表达演算能力的反应系统。从上面的例子可以看到,该演算反应系统除了独立作为一个反应系统以外,它还可以嵌入到一般的 Bigraphs 反应系统中。在文献[17]中,将演算的这个子反应系统称为展开(unfolding),其反应记为 $\leftrightarrow$;而将该演算子系统嵌入的宿主反应系统反应称之为常规反应(liberal),其反应记为 $\longrightarrow$ 。本文也按照这样的称谓和记法。

参数的展开利用了类似于等编程语言中的参数模式的概念,对于不同参数的模式其种子的展开也是不相同的。因此,每一个带参数的种子 K 具有模式的集合 Δ_K ,其中每种模式是一个不包含种子的离散的 Bigraphs $P; hm; Ai; h1; k; Xi$ 。值得注意的是; Δ_K 由控制用法标签来确定, $h1; k; Xi$ 中 k 表示控制的类型。在上面的例子中,控制是一个带参数的种子,其模式的集合为 $\Delta_{leq} = \{- \otimes \text{Zero}, \text{Zero} \otimes \text{Succ. } -, \text{Succ. } -_0 \otimes \text{Succ. } -_1\}$ 。一个带参数的展开规则是一个形式为 $(K; P, U, \eta)$ 的反应规则。在此形式中, K 是一个带参数的种子, $P \in \Delta_K$ 是 K 的一个模式。对于一个 Bigraphs 反应系统 $B_G(\mathcal{K}, \mathcal{R})$,如果 $\mathcal{R}$ 中的每个反应规则都是带参数的反应规则,那么该反应系统被称为是一个展开的 BRS。

展开关系 $\leftrightarrow$ 称为是交汇的是指:若有 $g \leftrightarrow^* g_0$ 与 $g \leftrightarrow^*$

g_1 ,则一定存在 g' 使得 $g_0 \leftrightarrow^* g'$ 和 $g_1 \leftrightarrow^* g'$ 成立。如果模式集合 Δ 中的模式之间两两之间互不相同,则称该模式集是正交的。也就是说,对于任意的参数 d 和 $e, P, Q \in \Delta$ 且 P 与 Q 是两种不同的模式,则一定有 $P. d \neq Q. e$ 。在一个展开的 BRS $B_G(\mathcal{K}, \mathcal{R})$ 中,若对于每一个带参数的种子 K ,其模式集 Δ_K 都是正交的,则称该展开的 BRS $B_G(\mathcal{K}, \mathcal{R})$ 是正交的。在此基础上,域为 $\in$ 的种子为一个分子形式 $K; P. d$,此时,若一个生长反应规则的形式为:

$$K; P \leftrightarrow G$$

式中, P 和 Q 的内部宽度分别为 m, m^0 ,则其实例化映射为 $' : m^0 i! m$ 来决定种子的 m 个参数中的实例化情况,若 $d = d_0 \otimes \dots \otimes d_{m-1}$,则 d_0 的定义为:

$$d' = d_{\eta(0)} \parallel \dots \parallel d_{\eta(m-1)}$$

此时,生长规则 $K; P \leftrightarrow G$ 的展开关系为:

$$K; P. d \leftrightarrow G. d$$

同样地,给出与文献[12]中交汇的展开定理相同的定理。

定理 1 在一个正交的展开 Bigraphs 反应系统 $B_G(\mathcal{K}, \mathcal{R})$ 中,展开关系是交汇的。

证明:略

在此定理下就可以给出演算 Bigraphs 反应系统的概念。

定义 2 若一个 Bigraphs 反应系统的反应规则集是展开规则的正交集,则称该反应系统为演算 BRS。

由展开的交汇特性定理及定义 2 可以得到如下的推论。

推论 1 演算 BRS 中的展开是交汇的。

由以上的定义、定理及命题可知,引言部分给出的 $B_G(\mathcal{K}, \mathcal{R})$ 是一个演算 BRS。一般而言,演算 BRS 除了种子之外还有一些控制被称为构造器,更具体地,若构造器为原子控制则称为常量(Constants)。例如 $B_G(\mathcal{K}, \mathcal{R})$ 中的 Zero 和 Succ 都是构造器且 Zero 是一个常量。

3.2 常规 BRS 与演算 BRS

通过前面的例子可以知道,演算 BRS 可以嵌入到一个称之为宿主 BRS 的常规 BRS 中。然而,如何才能保证演算中 BRS 的展开规则不会破坏常规 BRS 中的反应呢? 做如下的限定:

- 演算 BRS 中的反应物与常规 BRS 的反应物不重叠。
- 常规 BRS 的反应物不允许出现在演算 BRS 的参数中。也就是说,常规 BRS 可以在其反应中复制或丢弃演算 BRS 的反应物,反之则不能。

同进程演算中受困的概念一样,这里也给出 Bigraphs 中关于受困的概念。

定义 3 在 Bigraphs 中,如果一个节点或位置出现在另一个节点中,则称该节点或位置是受困的。

方便起见,我们将带演算 BRS 的 Bigraphs 反应系统简称为带演算的 BRS,其详细的定义如下:

定义 4 若给定演算 BRS $C = B_G(\mathcal{K}_C, \mathcal{R}_C)$ 是一个 Bigraphs 反应系统 $A = (\mathcal{K}_A, \mathcal{R}_A)$ 的子反应系统,设 A 中的常规反应记为 $L = B_G(\mathcal{K}_L, \mathcal{R}_L)$,并且满足如下条件:

- A 中的反应物不包含 C 中的种子。
- 在 A 的常规反应规则 (R, R', η) 中,如果生成物 R' 中的位置受困于一个隶属演算控制类型的节点,那么该地点在反应物 ∂ 中的像 $\eta(i)$ 也是如此。

• A 的常规反应的每个反应物若是受固的,则必受固于 $\mathcal{K}_A \setminus \mathcal{K}_C$ 中的控制类型。

此时,称 $A=(\mathcal{K}_A, \mathcal{R}_A)$ 是一个带演算的 BRS。

在上述定义的这三条约束中,第一条禁止了 L 和 C 中反应物的重叠,第二、三条保证了 $\leftrightarrow$ 反应规则的交汇特征且并不妨碍 $\rightarrow$ 的反应。由此,有如下定理。

定理 2 在一个带演算的 BRS 中,其演算 BRS 的展开 $\leftrightarrow$ 是交汇的,并且遵守常规反应 $\rightarrow$ 。也就是说,对于基 Bigraphs g ,如果有 $f \rightarrow g, f \leftrightarrow^* f'$,那么就有 $f' \rightarrow g', g \leftrightarrow^* g'$ 。

证明:略

4 案例研究

为了便于和柏拉图图形模型描述的上下文感知系统进行比较,本节将利用带演算的 Bigraphs 反应系统对文献[15]中给出的简单上下文感知打印系统进行建模。这个简单的上下文感知打印系统可以描述如下。

在一个办公大楼中有两种类型的打印机:一类是新式的 PCLi5e 兼容打印机,另外一类则是老式的 raw 打印机。并且,大楼的 IT 设备管理人员可以移动或者更换任意一种类型的打印机,但同时也要保证有打印机存在。系统规定任何一种类型的打印机只能同时处理一个打印任务,办公大楼中的所有打印任务由一个中央打印服务器进行统一排队。而且,只有当前环境中没有类型为 PCLi5e 兼容打印机的情况下,打印服务器才会给老式的打印机发送打印任务。倘若环境中存在 PCLi53 兼容打印机并且都在执行打印任务,那么新的打印任务只好排队等待。该系统需要感知当前环境中打印机的类型及其存在的数量来决定如何分配打印任务。

在简单上下文感知打印系统的柏拉图图形模型 B 中,其上下文 $C=(\mathcal{K}_C, \mathcal{R}_C)$ 的反应规则如下:

$$loc(-_0) \rightarrow loc(-_0 | /x. prt_x(raw)) \quad (1)$$

$$loc(-_0) \rightarrow loc(-_0 | /x. prt_x(pcl)) \quad (2)$$

$$loc(-_0 | prt_x(-_1)) \rightarrow (-_0) | x/ \quad (3)$$

$$prt_x(dat_x | (-_0)) \rightarrow loc(-_0 | /z \quad (4)$$

依照上下文的表述方式,其上下文代替物 $P=(\mathcal{K}_P, \mathcal{R}_P)$ 中的反应规则如下:

$$jobs(doc_x | -_0) \parallel prt_{sy}(pcl) \parallel prt_y(pcl) \rightarrow jobs(-_0) \parallel prt_{sy}(pcy) \parallel prt_y(pcl | dat_x) \quad (5)$$

$$jobs(doc_x | -_0) \parallel x. prt_{sy}(pcl) \parallel prt_{sy}(raw) \parallel prt_y(raw) \rightarrow jobs(-_0) \parallel /x. prt_{sy}(pcl) \parallel prt_{sy}(raw) \parallel prt_y(raw | doc_x) \quad (6)$$

$$/x. prt_x(pcl) \parallel prt_t(pcl) \rightarrow prt_y(pcl) \parallel prt_{sy}(pcl) \quad (7)$$

$$/x. prt_x(raw) \parallel prt_{sy}(raw) \rightarrow prt_y(raw) \parallel prt_{sy}(raw) \quad (8)$$

在简单上下文感知打印系统的柏拉图图形模型中,代理 $A=(\mathcal{K}_A, \mathcal{R}_A)$ 是十分简单的 Bigraphs 反应系统,其反应规则如下:

$$jobs(-_0) \rightarrow jobs(-_0 | /x. doc_x) \quad (9)$$

由感知打印系统的柏拉图图形模型可以看出:为了满足模型中 $C \perp A$ 的条件,打印的任务文档在 C 和 A 中分别用控

制 dat 和 doc 来表示 g 。因此,可以在带演算的 BRS 的系统模型描述中将打印任务文档统一用 doc 来表示。对于柏拉图图形模型中的反应规则,在带演算的模型描述过程中进行说明。

假定将系统带演算的模型记为 $A=(\mathcal{K}_A, \mathcal{R}_A)$,其常规反应规则应该包括允许系统中打印机的添加或删除即反应规则 (1-3);由于文档任务控制的变化,表述打印机执行打印任务的反应规则 4 在 A 中应写成:

$$prt_x(dat_x | -_0) \rightarrow loc(-_0) | /z \quad (4')$$

中央服务器的打印任务队列可以由反应规则 9 来描述。对于柏拉图模型中的增加打印机到打印机列表的反应规则 7、8,在模型中其表述方式如下:

$$/x. prt_x(pcl) prt_{py}(-_0) \rightarrow prt_{py}(prt_y(pcl) | -_0) \quad (7')$$

$$/x. prt_x(raw) prt_{py}(-_0) \rightarrow prt_{py}(prt_y(raw) | -_0) \quad (8')$$

而对于如何选定使用何种类型的打印机进行打印,其反应规则为:

$$jobs(doc_x | -_0) | prt_p(-_1) | prtr(-_2) \rightarrow /m. jobs_m'(doc_x | -_0) | prt_{pm}'(-_1) | prtr_m'(-_2) | Admit(Jud(-_1 | -_2)) \quad (10)$$

$$/m. jobs_m'(doc_x | -_0) | prt_{pm}'(-_1) | prtr_m'(-_2) | Admit.P \rightarrow jobs(-_0) | prt_x(pcl | doc_x) \quad (11)$$

$$| Admit.R \rightarrow jobs(-_0) | prt_x(raw | doc_x) \quad (12)$$

该演算系统的反应规则为:

$$Jud(P|R) \leftrightarrow P \quad (13)$$

$$Jud(P|zero) \leftrightarrow P \quad (14)$$

$$Jud(Zero|R) \leftrightarrow R \quad (15)$$

由演 BRS 算的定义 2 可知 $A_{cal}=(\mathcal{K}_{cal}, \mathcal{R}_{cal})$ 是一个演算 BRS。更进一步地,由带演算的 BRS 定义及其定理可以得到 $A=(\mathcal{K}_i \cup \mathcal{K}_{cal}, \mathcal{R}_i \cup \mathcal{R}_{cal})$ 是一个带演算的 Bigraphs 反应系统。至此,我们就给出了简单的上下文感知打印系统的一个带演算 BRS 模型描述。在模型 A 中,虽然看似增加了反应规则的数量,但其表述能力得到了明显的增强。同时,由于带演算的 BRS 理论具有良好的理论构建使得 PGM 中的分类安全性及同态赋类等问题都可以很好地解决。

结束语 上下文感知是普适计算中的关键技术,而 Bigraphs 理论作为普适计算抽象机能否严格而有效地对普适计算中的核心问题即上下文感知进行形式化描述是其应用于实践的关键环节。在介绍普适计算及上下文感知等应用背景的基础上,本文首先给出了 Bigraphs 理论基本内容的介绍,讨论了已有的上下文感知系统的 Bigraphs 模型——柏拉图图形模型的适用性及其存在的缺陷,给出了利用带演算的 Bigraphs 反应系统进行上下感知系统建模的相关讨论比较。并且利用一个经典的感知系统案例来展示了带演算的 Bigraphs 反应系统在上下文感知系统建模方面的表达能力。

参考文献

- [1] 吴朝晖,潘刚. 普适计算(中国计算机学会论文集) [M]. 北京:清华大学出版社,2009
- [2] 王海涛,宋丽华. 普适计算——新一代计算模式和理念 [J]. 电信科学,2008,24(2),66-71
- [3] 沈理. 普适计算 [J]. 计算机工程与科学,2005,27(7),77-82
- [4] 徐光祐,史元春. 普适计算[J]. 计算机学报,2003,26(9),76-81

- [5] Julien C, Payton J, Roman G C. Reasoning About Context-Awareness in the Presence of Mobility [A]// Proc. of the 2nd International Workshop on Foundations of Coordination Languages and Software Architectures (FOCLASA 2003) [C]. Marseille, France, 2003; 259-276
- [6] Roman G C, Julien C, Payton J. A Formal Treatment of Context Awareness [C] // Wermelinger M, Margaria-Steffen T, eds. FASE 2004. Vol. 2984 of LNCS. Springer Heidelberg, 2004; 12-36
- [7] Yan L, Sere K. A Formalism for Context-Aware Mobile Computing [A]// Proceedings of the IEEE 3rd International Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Networks [C]. Cork, Ireland, 2004; 14-21
- [8] Cafezeiro I, Haeusler E H. Semantic interoperability via category theory [C] // John Grundy, Sven Hartmann, Alberto H. F. Laender, Leszek Maciaszek, and John F. Roddick, eds. Tutorials, posters, panels and industrial contributions at the 26th international conference on Conceptual modeling. Vol. 83 (ER '07). Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 2007; 197-202
- [9] Milner R. The Space and Motion of Communicating Agents [M]. Cambridge: Cambridge University Press, 2009
- [10] Krivine J, Milner R, Troina A. Stochastic bigraphs [A] // Bauer A, Mislove M, eds. Proceedings of the 24th Conference on the Mathematical Foundations of Programming Semantics (MFPS 2008) [C]. volume 218 of Electronic Notes in Theoretical Computer Science, Elsevier, 2008; 73-96
- [11] Milner R. Pure bigraphs: Structure and dynamics [J]. Information and Computation, 2006, 204 (1): 60-122
- [12] Leifer J J, Milner R. Transition systems, link graphs and Petri nets [J]. Mathematical Structures in Computer Science, 2006, 16 (6): 989-1047
- [13] Milner R. Axioms for bigraphical structure [J]. Mathematical Structures in Computer Science, 2005, 15(6): 1005-1032
- [14] Jensen O H. Mobile processes in bigraphs [D]. Dep. of Computer Science, Aalborg University, 2006
- [15] Birkedal L, Debois S, Elsborg E, et al. Bigraphical models of context-aware systems [A] // Aceto L, Ingólfssdóttir A, eds. Proceedings of the 9th International Conference on Foundations of Software Science and Computation Structure (FoSSaCS'06) [C]. Lecture Notes in Computer Science, vol. 3921, Springer-Verlag, 2006; 187-201
- [16] Birkedal L, Debois S, Elsborg E, et al. Bigraphical Models of Context-aware Systems [R]. 74. IT Univ. of Copenhagen, 2005
- [17] Milner R. Seminar Notes on developments in Bigraphs. Universities of Cambridge and Edinburgh [EB/OL] <http://www.cl.cam.ac.uk/archive/rm135/uam-theme.html>, 2009

(上接第 103 页)

$$s \stackrel{\text{sources}(a,u)}{\approx} t \sqsupset \text{step}(s,a) \stackrel{\text{sources}(a,u)}{\approx} \text{step}(t,a) \text{ dom}(a) \notin \text{sources}(a,u) \\ (a \circ a, u) \sqsupset s \stackrel{\text{sources}(a,u)}{\approx} \text{step}(s,a)$$

具体证略。

定理 6 将一个多进程行为与一个机器 M 对应, 如果 M 满足定义 6 的 RMA, 且满足条件:

$$n \in \text{alter}(u) \wedge n \in \text{observe}(u) \sqsupset u \rightsquigarrow v$$

则该多进程动态行为是可信的。

证明: 利用定理 5 来证明定理 6。

首先证明满足输出一致性。输出一致性由定义 6 的 RMA 假设 1 即可得。

其次证明满足弱单步一致性。弱单步一致性根据定义 6 的 RMA 假设 1, 可以改写为:

$$s \stackrel{\text{dom}(a)}{\sim} t \wedge \forall n \in \text{observe}(u). \text{contents}(s,n) = \text{contents}(t,n) \\ \sqsupset \forall n \in \text{observe}(u). \text{contents}(\text{step}(s,a),n) = \text{contents}(\text{step}(t,a),n) \quad (7)$$

分情况讨论上述式(7)成立即可。在证明的过程中, 要依次用到 RMA 假设 2 以及式(7)的前条件。

最后证明局部干扰性成立。证明逆否命题 $s \not\rightsquigarrow \text{step}(s,a) \sqsupset \text{dom}(a) \rightsquigarrow u$ 即可。利用式(5)的前条件, 以及 RMA 的假设 3 立即可得。

具体证略。

结束语 软件动态行为度量是可信计算必须要解决的关键问题之一, 也是目前国内外的研究热点。但是, 由于软件形式的多样性和攻击方式的复杂性, 使得目前这方面的研究还

处于初步阶段, 还有很多问题需要深入研究^[1]。

软件(或系统)在运行的过程中必然要涉及到信息的流动, 因此, 无干扰模型作为一种重要的信息流模型, 在可信计算相关的研究中, 特别是可信链的研究当中, 获得了广泛的应用。本文也尝试以无干扰模型为基础, 对建模后的软件动态行为进行分析, 提出了一种基于信息流的软件动态行为可信性分析方法, 并从单进程行为和多进程行为的角度分别给出了软件动态行为可信性的判定定理。本文的结论对于建模完成后的软件动态行为可信性推演具有参考意义, 也能够与相应的软件动态行为建模方法相结合从而构成一个完整的软件动态行为度量框架。

参 考 文 献

- [1] 沈昌祥, 张焕国, 王怀民, 等. 可信计算的研究与发展[J]. 中国科学: 信息科学, 2010, 40(2): 139-166
- [2] Zhang F, Xu M D, You L. A Behavior Modeling Method based on System Call and Algebra Process CCS[J]. Proceedings of the 5th China Trusted Computing and Information Security(CTCIS'11), Journal of Wuhan University (Science Edition)
- [3] Milner R. Communication and Concurrency[M]. London: Prentice-Hall, 1989
- [4] Focardi R, Gorrieri R. Classification of Security Properties: Part I Information Flow[J]. Foundations of Security Analysis and Design, LNCS, 2001, 2171: 331-396
- [5] Rushby J. Noninterference, Transitivity, and Channel-Control Security Policies[R]. Computer Science Laboratory, SRI International, 2005