

17-19

有限模型的自动构造

TP3

张健

(中科院软件所, 北京 100080)

A

摘要 计算机科学的很多领域引入了形式化方法, 逻辑公式经常出现在各种技术中, 有时需要判定一个逻辑公式的协调性(可满足性), 并对可满足的公式给出一个具体的模型。本文介绍计算机科学和数学中的一些模型构造问题, 以及有限模型的自动构造技术。

一、引言

计算机科学的产生和发展始终受到数理逻辑的影响。最近十几年, 逻辑的手段(所谓的形式化方法)经常被用于刻画程序设计和智能行为的各个方面, 对此已经做了大量工作, 提出了各种逻辑系统。从技术上看, 在逻辑系统的机械化方面, 自动定理证明一直占主导地位。本文讨论与此密切相关的模型构造问题, 即给定一组逻辑公式(公理), 构造出它的一个(有限)模型。比如说, 给定抽象代数中群结构的几条公理, 以及正整数 n , 构造出一个 n 阶群的乘法表。

模型构造问题显然很重要, 同时又很困难。目前, 它正受到各国研究人员的重视, 国际自动推理学会主席 Larry Wos 在第 11 届自动演绎会议(CADE-11, 1992)的基调讲演中, 将“模型和反例的自动生成”列为一个重要的研究课题。J. Cunningham 在 Esprit 3125 计划的报告中写道, 推理有很多种, 其中演绎技术已经被广泛地研究; 但还有其它形式的推理值得注意, 包括模型生成, 它的应用之一是为给定的猜想构造反例。

在文[16]中, 我们阐述了自动构模的重要性及其应用, 并介绍了一个为等式系统构造有限模型的程序。概括地说, 构模有正反两方面的意义: 一是, 对于给定的一组合乎情理的条件, 它可以找出一些具体的满足

条件的例子; 二是, 对于错误的假设, 它能找出反例, 使人们加深对事物的认识。本文以下各节将详细讨论各种场合下的构模问题, 并简要介绍自动构模技术。

二、NP-难问题

NP-难问题中最著名的一个是命题逻辑中的可满足性问题(简称 SAT)。具体地说, 就是给定一个命题逻辑公式的合取范式, 判定该公式是否可满足。进一步, 如果它可满足, 则给出它的一个具体模型, 即给出每个变量的真假值, 使得整个公式的值为真。

从理论上讲, 一般认为不太可能对 NP-难问题给出有效的精确算法。然而, 可以设计出各种近似算法, 尽管它们不能对付所有的情形, 但在能对付的情形下效率却很高。最近的实验结果表明, 用局部搜索的方法能解决很多大规模的 SAT 问题^{[5], [10]}。同时, 三十年前提出的经典 SAT 算法 Davis-Putnam 过程^[1]又重新得到重视。这两类算法性能相比如何? 各适用于何种情形? SAT 问题的哪些实例可以有效地解决? 这些都有待于进一步研究。由于实际应用中的很多 NP-难问题都可在多项式时间内归约到 SAT, 寻找有效的 SAT 算法显然具有重要意义。

三、自动定理证明

模型构造与定理证明实际上是密切不可

分的,前者显示出逻辑公式的可满足性,而后者往往借助于逻辑公式的不可满足性。就命题逻辑而言,它是可判定的,其判定过程(如 Davis-Putnam 过程)既可用于构造模型,又可用于证明定理。

一阶逻辑中的自动定理证明常采用否定法,要证明 A 可推出 p , 只要从 " A and not p " 推导出矛盾。为此需要一些推导规则,如归结(resolution),重写(rewriting)等。在证明过程中,不断地对旧公式集施以推导规则,得到新公式,直至最终得出矛盾。影响否定法效率的一个关键因素是采用什么策略来使用推导规则^[13],比如说,从旧公式集中选哪两个公式进行二元归结。一种好的策略是,利用语义信息来帮助合理地使用规则^{[9],[12]}。在语义推导方法中,一般用到前提 A 的有限模型 M 。如果由 " A and not p " 推出的公式 q_1 和 q_2 在 M 上都成立,那就不用对其施以推导规则,因为我们的目的是要推出矛盾,而 " q_1 and q_2 " (或者, q_1 和 q_2 的归结式)显然在 M 上成立,不可能是"矛盾"。

除了经典的否定法外,还有基于模型生成的定理证明方法^[6],如日本第五代机计划中研制的定理证明器 MGTP (Model Generation Theorem Prover)^[3]。另外,在时序逻辑定理证明中,往往不用语法的推导规则,而是用模型检测(model checking)方法。上述这些方法都特别适合于变量值域有限的情形。

四、程序综合

程序综合(program synthesis)指的是,从形式化规格说明构造(或综合)出一个满足说明的可执行程序。这是自动程序设计的一个重要方面,它显然能提高编程效率,但是目前只能在某些领域自动地综合出较小的程序。

Manna 和 Wolper^[7] 以及 Emerson 和 Clarke^[2] 独立地研究过并发程序的自动综合。并发程序的理解和编制很容易出错,通常要求并发程序满足一些性质,如安全性、活动

性等等,这些性质可用时序逻辑公式来描述。因此,并发程序综合可看成是时序逻辑中的模型构造问题。通过这种途径,已经能自动地综合出哲学家问题的并发程序。

五、数学问题求解

同机器定理证明一样,自动构模的一个主要应用领域在于数学研究。在引言中我们提过,构模有正反两面的意义:一是给反例,二是给正例。

给定一组公理 A 和猜想 c ,我们要研究 c 是否是 A 的定理,如果用定理证明器证出 c 是定理,问题就解决了。如果证不出,那么可以试图构造出 " A and not c " 的模型。如果这样的模型存在,就说明 c 不是 A 的定理。因此,我们不仅能证明定理,还能证明"非定理"(nontheorems)。它的一个应用是证明代数系统中某条公理独立于其它公理^[16](公理系统的无冗余性)。

另一方面,给定一个公理系统 A ,确定它的谱(spectrum)在组合学中是有意义的。(A 的谱是指集合 $\{n | n \text{ 是正整数,且存在 } A \text{ 的 } n \text{ 阶模型}\}$)。一种行之有效的方法是,对较小的 n ,用自动构模技术给出 n 阶模型或确定出这样的模型不存在;然后再基于这些小模型,用数学的办法得到更大的模型。笔者曾得出结论,Bennett 的拟群恒等式(quasigroup identities)没有 9 阶模型^[15]。此后, M. Fujita 等人利用他们的定理证明器 MGTP,在 ICOT 开发的 PIM (Parallel Inference Machine) 上,得到了一系列结果^[4]。

六、其它应用

形式化规格说明的一个重要性质是协调性(consistency),而证明协调性的一个常用办法是给出模型。对于软件系统来说,其规格说明的模型相当于一种实现,往往可由用户给出,但自动构模工具还是必要的。

模型构造技术在知识库更新和非单调推理中也起关键作用。如前所述,构模可以证

多态性

分类

特性

程序语言

⑤

19-22

多态性的分类及其特性

TP312

杜 诚 郭浩志

(国防科技大学计算机系 长沙 410073)

摘要 This paper presents a detailed discussion on the concept of the polymorphism and an analysis on the polymorphically typed system using type abstraction method.

一、前言

关于多态性概念的论述最早见于文[5]。在文[4]中, R. Milner 具体地提出了多态类型原则和自动类型检查算法 ω 。80 年人们在 ML 语言中实现了多态类型的思想, 其后相继出现了 Miranda, Russell, Poly 等具有多态类型系统的语言。随着类型在程序设计语言中的地位越来越受到人们的重视, 要求语言中有一个强有力的类型系统能够提供抽象机制以增强语言的表达能力, 并利于编写出正

确无误的程序^[2], 而多态类型系统就是能够满足上述要求的一种系统。由于在多态语言 (polymorphic language) 中, 值和变量, 函数的参数以及其返回值可有多种类型, 其类型可被定义成具有适于不同类型参数的操作的多态类型 (polymorphic type)。因此, 多态性思想允许同一实体在不同的环境中具有不同的行为解释, 因而发挥出不同的功能, 具有灵活有力的描述表达能力, 支持数据抽象, 利于方便自然地进行程序设计。多态性的具体表现形式多种多样, 而且也不只局限于具有

杜诚 硕士, 研究方向: 计算机网络、软件工程。郭浩志 教授, 研究方向: 软件工程、程序语言。

明非定理, 这些非定理带有旧知识库中缺少的信息, 能使知识库的能力增强。在开放逻辑系统^[6]中, R-重构和 N-重构都依赖于某种模型的存在。

七、自动构模技术

可满足性问题在一阶逻辑中是不可判定的, 在命题逻辑中也是 NP-难的, 因此自动构模很困难。对付这种困难有两类办法。一类是只考虑特殊的逻辑, 比如说, 命题 Horn 逻辑的可满足性问题有线性时间复杂度的算法。

另一类办法是只考虑有限模型。在域有限的情况下, 通过穷举, 总能判定可满足性。在实际系统中, 有限模型的构造通常可看作

约束满足 (constraint satisfaction) 问题。这需要将演绎和搜索技术结合起来 (通常采用回溯搜索)^{[4]、[14]、[16]}。目前, 针对一阶理论的通用自动构模程序还不多, 除了前面提到的日本的 MGTP 以外, 还有澳大利亚的 FINDER (Finite Domain Enumerator)^[11]。

八、结束语

模型的自动构造具有重要意义, 在很多场合有着广泛的应用, 同时也是个很难的问题, 在理论上有不少反面的结论。有限模型的构造技术目前还不成熟, 希望在不久的将来能有所突破, 并推动相关领域的发展。(参考文献共 16 篇略)