

# 基于构造类别代数的变异分析

周晓煜,赵保华,屈玉贵  
(中国科学技术大学计算机系,安徽合肥 230027)

**摘要:** 在通信协议的一致性测试中,可以使用构造类别代数来对协议进行形式化的描述.本文将变异分析引入构造类别代数中,设计了应用于构造类别代数的几类变异算子,并且提出了使用变异分析对基于构造类别代数描述的协议进行一致性测试的基本步骤.这种方法为基于构造类别代数的协议测试提出了一个新的方向.

**关键词:** 构造类别代数;变异分析;变异算子

**中图分类号:** TP311 **文献标识码:** A **文章编号:** 0372-2112 (2002) 12A-2155-03

## Mutation Analysis Based on Constructed Type Algebra Specification

ZHOU Xiao-yu, ZHAO Bao-hua, QU Yu-gui  
(Department of Computer Science, University of Science and Technology of China, Hefei, Anhui 230027, China)

**Abstract:** The constructed type algebra specification has been used to specify communication protocols in protocol conformance testing. In this paper, mutation analysis is introduced into the constructed type algebra specification. Several kinds of mutant operators are designed and the method using mutation analysis based on constructed type algebra specification is proposed. This method provides a new probable direction for the protocol conformance testing based on the constructed type algebra specification.

**Key words:** constructed type algebra specification; mutation analysis; mutant operator

### 1 引言

通信协议的一致性测试,是指验证通信协议的待测实现的外部行为与其规范描述是否一致的过程.在文献[1]中,作者提出了使用构造类别代数来形式化协议,然后根据形式化描述的公理部分生成测试序列.但是,如何从形式化的公理生成测试序列,以及生成的测试序列是否能够对协议进行充分的测试,作者没有进行更进一步的讨论.

变异分析是一种基于故障的测试方法,通常用于软件的单元测试.这种方法通过在程序中引入错误,生成能够检测出这些错误的测试用例,从而检查待测程序中是否存在这些错误.

本文将变异分析引入构造类别代数,为基于构造类别代数描述的通信协议测试提出了一个新的方向.在基于构造类别代数的协议描述中,使用等式公理来描述协议的行为.针对协议的公理部分,本文中提出了几类变异算子,并且给出了应用这些算子生成测试用例的基本方法.

### 2 基于构造类别代数的协议描述

构造类别代数可以定义为一个四元组  $D = \langle S, C, E, Q \rangle$ , 其中,  $S$  是类别名的有限集合,  $C$  和  $E$  分别是构造函数和延拓

函数的有限集合,  $Q$  是  $C$  与  $E$  中定义的函数应满足的等式公理.一个通信协议可以用上述的构造类别代数来描述:使用函数来描述协议的行为;使用公理来规定协议行为应满足的要求.下面是使用构造类别代数对 RIP 协议中的路由算法的描述(详细说明参阅文[4]):

```

route_list =
S: route_list, address, Nat
C: Empty : → route_list (1)
Response : route_list, address, Nat, address → route_list (2)
E: Metric : route_list, address → Nat (3)
NextHop : route_list, address → address (4)
Request : route_list → route_list (5)
Q: rlist ∈ route_list; addr1, addr2, addr3 ∈ address; n ∈ Nat
Metric (Empty, addr1) = UNDEFINE (a)
Metric (Response (Empty, addr1, n, addr2), addr1) = n + 1 (b)
.....
Request(rlist) = rlist (i)

```

根据公理与外部环境的信息交互关系,可将所有的公理分为三类:

- 可控制公理** 对输入信息直接进行处理的公理.
- 可观察公理** 能够产生输出信息的公理.
- 内部公理** 不对输入信息直接处理,也不产生输出信息

收稿日期:2002-06-06;修回日期:2002-10-22

基金项目:自然科学基金重大研究计划项目(No. 90104010, No. 60241004);教育部博士点基金项目(No. 2000035802);安徽省自然科学基金项目(No. 01042208);国家 863 计划项目(No. 2001AA112062, No. 2001AA121016);中国科学院院长基金特别支持项目(院基计字 905 号)

的公理。

这样,我们能够给协议的待测实现一定的输入信息,激发可控制公理,间接控制内部公理,最后激发可观察公理来判断协议的待测实现是否正确。需要注意的是,可观察公理也可以是一个可控制公理,反之亦然。例如在上述的例子中,假定只能通过 Metric 函数了解路由信息,则公理(*a*)既可观察,又可控制的公理,而公理(*i*)只能够控制。

### 3 变异分析

变异分析是一种常用于单元测试的测试技术,它的基本思想是:将一组变异算子作用于被测程序,在程序代码中引入微小的语法变化,从而产生一个语法正确,但语义与被测程序不同的新程序,称为变异体。然后,针对变异体进行分析,生成测试用例集合,使得对于任意一个变异体,满足:

(*a*)存在一个测试用例,使得该变异体与被测程序产生不同的输出结果,或者(*b*)该变异体与原程序等价,即对于任意的输入,该变异体都与被测程序产生相同的输出结果。

针对变异算子的设计,已经有一些研究人员进行了研究,例如,Moithra 工具<sup>[2]</sup>,是针对 Fortune77 设计的变异分析工具,使用了 22 个变异算子。针对标准 C 语言,也设计了相应的变异算子<sup>[3]</sup>。

针对每个与被测程序不等价的变异体,需要生成测试用例来将该变异体与被测程序区分开。这样的测试用例必须满足以下三个条件:

**可达性** 必须能够执行到被变异的语句。

**必要性** 必须能够使被变异的语句产生与正确语句不同的结果。

**充分性** 该语句产生的不同结果必须能够最终影响到程序的输出,否则该错误无法被观察到。

根据以上分析,针对被测程序的变异体集合,可以设计满足上述三个条件的测试用例集合,完成对被测程序的测试。

### 4 基于构造类别代数的变异分析方法

构造类别代数使用公理来描述协议的行为,公理的定义类似于程序设计语言中的函数。因此,对公理的分析,可以借鉴对函数的分析方法。另一方面,构造类别代数中具有一个特殊的类型——构造类别,这个类别中所有元素都是用构造函数来表示的,针对这种类型,需要设计特殊的变异算子。

这一部分中将介绍基于构造类别代数的变异算子的设计,以及测试用例的生成方法。

#### 4.1 变异算子设计

根据构造类别代数的特点,可以将变异算子分为以下三类:

##### (1) 基本变异算子

这类算子主要是针对公理的变异而设计的,可以参考程序设计语言<sup>[2,3]</sup>中变异算子的设计。例如在第 2 部分的例子中,假定 UNDEFINE 是一个特殊的常量,则可以使用常量替换算子对公理(*b*)进行变异,得到变异体:

$$\text{Metric}(\text{Response}(\text{Empty}, \text{addr1}, n, \text{addr2}), \text{addr1}) = 0$$

##### (2) 构造类型变异算子

这一类算子主要是针对构造类型而定义的。对于构造类型的变量,可以看作一般类型变量,定义如变量替换等算子。针对构造类型的特殊性——构造定义,需要增加一类特殊的算子:构造类型变量的构造函数替换。例如,在第 2 部分的例子中,我们可以对公理(*i*)进行变异,得到如下的变异公理:

$$\text{Request}(rlist) = \text{Empty}$$

$$\text{Request}(rlist) = \text{Response}(rlist, \text{addr1}, n, \text{addr2})$$

##### (3) 公理变异算子

这一类算子是将公理部分作为整体来考虑设计的,此时,每条公理就是被变异的元素,可以定义两个变异算子:删除公理算子和增加公理算子,分别用于将某条公理删除,或者增加新的公理来产生变异体。

#### 4.2 测试用例生成

将变异算子作用于协议描述,得到变异体集合后,需要设计测试用例,来区分这些变异体。测试用例生成的基本过程如下:

(1) 选择变异算子,生成一个与协议描述不等价的变异体集合;

(2) 任意选择集合中的一个变异体进行分析,生成测试用例;

(3) 对集合中所有的变异体执行该测试用例,删除被区分的变异体,得到新的变异体集合;

(4) 如果变异体集合为空,则结束;否则,转(2)。

在通信协议测试中,一个测试用例对应于一个输入/输出序列。根据可达性、必要性和充分性要求,可将一个测试用例分解为三部分:

**切换序列** 从初始状态到达应用变异公理所需要经过的序列;

**变异公理应用** 变异公理被应用时,执行变异公理中被变异部分,并且产生错误结果所需要的序列;

**变异结果输出序列** 应用变异公理后,使得产生的错误结果影响到可观察公理的序列。

下面对上述三个部分,分别进行分析。

##### (1) 切换序列的生成

首先,从变异公理出发,分析应用变异公理所需要满足的条件。然后,递归分析每个条件中对其他公理的应用,直到条件集合中只包含可控制公理,选择符合条件集合要求的输入/输出即可。

例如,在第 2 部分的例子中,对公理(*i*)的变异结果

$$\text{Request}(rlist) = \text{Empty}$$

进行分析可知,要想应用这个公理,必须对路由表进行初始化,即 Empty/-。

##### (2) 变异公理应用

在应用变异公理时,需要保证能够执行到变异公理中被变异的部分,产生错误结果。我们可以从被变异的部分出发,逆向推出要到达这一部分,并且产生错误结果所需要满足的条件。继续使用上面的例子,为了产生错误的结果,必须满足

路由表原本不为空,因此可得  $\text{Response}(rlist, addr1, n, addr2) / - , \text{Request}(rlist) / -$ . (其中  $rlist$  表示上一个公理应用后得到的结果路由表)

### (3) 变异结果的输出

应用变异公理产生错误结果后,需要通过可观察公理表现出来.从变异公理出发,以广度优先策略扩展生成一棵树,使得父节点能够影响子节点的执行结果,直到有分支到达可观察公理为止.这样,从变异公理到可观察公理之间的分支,就构成了变异结果输出序列应该满足的条件.

在上面的例子中,使用  $\text{Metric}$  和  $\text{NextHop}$  函数进行观察,可得  $\text{Metric}(rlist, addr1) / n, \text{NextHop}(rlist, addr1) / addr2$ .

## 5 例子

为了验证上述变异分析方法的有效性,我们仍然使用了第 2 部分中提到的路由算法的例子.

首先,针对 RIP 路由算法的构造类别代数描述,使用上述几类变异算子.共生成了 84 个变异体,其中等价变异体有 9 个.针对不等价的 75 个变异体,使用上述的方法,生成了 18 条测试用例(篇幅限制,在此不一一列出).

(1)  $\text{Empty}() / - , \text{Metric}(rlist, addr1) / \text{UNDEFINE}$

(2)  $\text{Empty}() / - , \text{Response}(rlist, addr1, 2, addr2) / - , \text{Metric}(rlist, addr1) / 3$

(3)  $\text{Empty}() / - , \text{Response}(rlist, addr2, 0, addr2) / - , \text{Response}(rlist, addr1, 2, addr2) / - , \text{Response}(rlist, addr3, 5, addr2) / - , \text{Metric}(rlist, addr1) / 3$

(4)……

然后,针对使用 C 语言编写的 RIP 路由算法的实现部分,使用 C 语言的变异算子在该程序中随机引入单个错误,得到 44 个不等价的变异体.将生成的 18 个测试用例作用于这些变异体上,能够全部区分出这 44 个变异体.

这个例子说明,基于构造类别代数的描述进行变异分析产生的测试用例,在发现待测实现中可能出现的错误是有效的.

## 6 结论

本文将变异分析引入基于构造类别代数的协议测试中,根据构造类别代数的特点,将变异算子分为三类:基本变异,构造类型变异和公理变异,并且给出了使用变异分析方法生成测试用例的基本步骤.这种方法为基于构造类别代数的协议测试提出了一个新的方向.

基于构造类别代数的变异分析仍在进一步的研究之中,很多方面需要继续扩充和完善.例如,变异算子的更为详细的设计,以及变异算子集合是否充分,测试用例生成方法的形式化等,还需要进一步的研究.

### 参考文献:

- [1] 孙宇霖,屈玉贵,赵保华.一种通信协议测试序列生成的新方法[J].通信学报,2001,22(6):122-127.
- [2] Weichen Eric Wong. On Mutation and Data Flow[D]. Indiana USA: Purdue University, Software Engineering Research Center, 1993.
- [3] Hiralal Agrawal, Richard A DeMillo, et al. Design of mutant operators for the C programming language[R]. Indiana USA: Purdue University, Software Engineering Research Center, 1989.
- [4] 孙宇霖.基于构造类别代数协议测试理论的研究[D].安徽合肥:中国科学技术大学计算机科学技术系,2002.

### 作者简介:



周晓煜 女,1977年6月生于内蒙古,博士研究生,研究方向:软件工程,通信软件和协议工程.

赵保华 男,1947年8月生于安徽,教授,博士生导师,研究方向:软件工程,通信软件和协议工程.

屈玉贵 女,1945年5月生于山东,教授,博士生导师,研究方向:软件工程,通信软件和协议工程.