

图在软件测试中的应用

黎秀 潘虹

(电子科技大学成都学院计算机系 四川成都 610000)

摘要：图是拓扑学中的一个分支，同时在数学与计算机科学中，图的理论和图思想都是必不可少的部分。当然作为计算机科学中的一部分——软件测试，也必然离不开图。那么图的理论和图思想在软件测试中又有哪些方面的应用呢？本文将结合软件测的经典测试方法，即白盒测试和黑盒测试，讨论图理论和图思想在软件测试中的应用。

关键字：软件测试，图，程序流程图，控制流图，系统图，状态图

Abstract: Graph is a branch of topology. In mathematics and computer science, graph theory and graph thought are indispensable parts. Of course, as a part of computer science software testing is also inseparable from the graph. What are the applications of graph theory and graph thought in software testing? This paper will discuss the application of graph theory and graph idea in software testing based on the classical testing methods of software testing, namely white box testing and black box testing.

Keywords: Software testing, graph, program flow chart, control flow chart, system graph, state graph

1 图概述

图是拓扑学的一个分支，它的研究对象就是图，即通过直观的画图的方式来代替文字的表达。为什么要采用图的方式来代替文字表达呢？这跟图最大的特点有关，即直观性，俗话说“一图胜千言”，在采用文字描述过程中，难免会出现文字冗余或者啰嗦的情况，但是采用图来表示就大不一样，图中表示简单扼要，毫无冗余，令读者读起来清晰度高。在图的研究过程中，不难发现按照图中有无方向性，可以将图分为无向图和有向图，有向图指的是全部由有向边构成的图（如图1所示）；无向图则指的是全部由无向边构成的图（如图2所示）。在软件测试中，对于有向图和无向图均有使用。

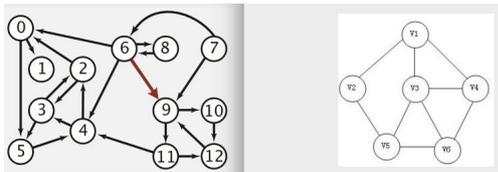


图1 无向图

图2 有向图

2 白盒测试中的图

白盒测试也称为结构测试或者逻辑驱动测试，是一种测试用例设计方法，它从程序的控制结构导出测试用例。白盒测试中最关键的部分是必须清楚了解程序内部逻辑结构，那么采用什么样的方式表示程序内部结构就至关重要了，经过长时间研究表明，采用程序流程图表示直观明了，同时也不会丢失某些逻辑路径。

程序流程图又被称之为框图即是使用各种框加上有向的边来表示程序的逻辑结构及其代码先后顺序。程序流程图是表示程序结构最简单明了的方法，也是人们最熟悉最喜欢的方式，如图3即为某程序片段的程序流程图表示。

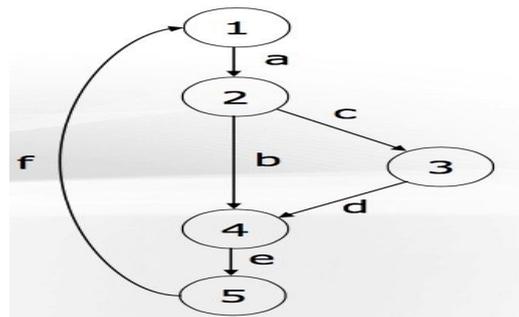


图3 某程序片段的程序流程图表示

由以上图可以看出，程序流程图中有矩形框、菱形框和圆形框之分，矩形框表示的是执行的处理，菱形框则表示判断，圆形框则表示的是程序片段起始和终止。然而，在白盒测试的逻辑测试过程中，测试人员并不关心菱形框与矩形框中的实际内容，更多的是关心逻辑走向。基于此，可将程序流程图简化，用圆圈代替矩形框和菱形框，即将所有的菱形框和矩形框一律用圆圈加序号代替，同时给每条边都指定唯一的名字，这就将程序流程图转化为标准的有向图，我将这种图取名为控制流图，如图4所示为某程序片段由程序流程图转换的控制流图。

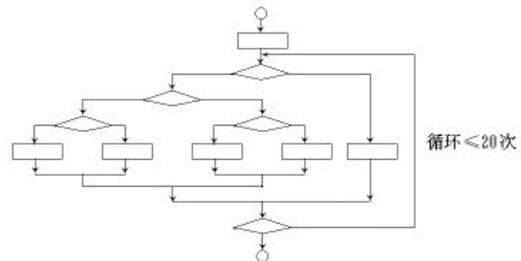


图4 某程序片段的控制流图

将程序流程图转换为控制流图之后，进一步将控制流图转换为控制流矩阵（即邻接矩阵），从而为程序片段的自动化测试提供了有利条件。

3 黑盒测试中的图

黑盒测试作为软件测试中的又一经典,必然也离不开图理论和图思想的支撑。黑盒测试又称功能测试、数据驱动测试或基于规格说明书的测试,是一种从用户观点出发的测试。黑盒测试有很多中不同的方法,其中因果图法和功能图法都离不开图的支撑。

首先看看图在因果图法中的应用。因果图顾名思义就是反映原因和结果之间关系的图,在因果图中,事件作为图的顶点,事件之间的关系作为图的边,再在传统图的概念上添加适当限制符号,表示原因与原因、结果与结果、原因与结果之间的制约关系。例如,在某段程序的需求说明中要求:第一行的字符只能是1或者2,第二行字符必须是一个字母,在这样的情况下,就给出正确的提示信息,但如果第一行字符不正确,就给出信息A,如果第二行字符不是字母就给出信息B。根据这样的需求要求,结合因果图的原理,将可以作出如图5所示的因果图。根据因果图将其直观的得出各个事件之间的关系,同时对设计正确且无遗漏的测试用例提供有利的保证。

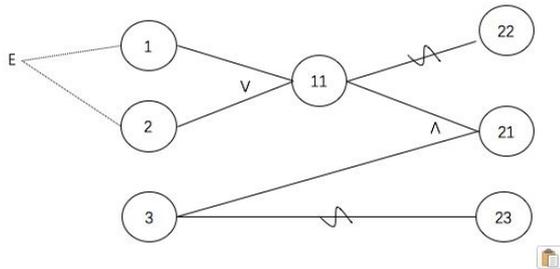


图5 因果图

以上是因果图中对图思想和图理论的应用,下面将再介绍一个关于图在黑盒测试中应用,即功能图法。在功能图法中,用功能图形式化的表示程序的功能说明,并机械的生成功能图的测试用例。为了更加形象的表示程序的功能说明,以及功能之间状态推移,巧妙的利用了状态推移图和逻辑功能模型辅助完成,这里主要介绍状态迁移图。

在软件设计过程中,许多需求均采用状态机的方式来描述,在对状态机的测试中,测试人员主要关注的是测试状态的转移的正确性上面。对于一个有限状态机而言,通过测试验证其在给定的条件内是否能够产生需要的状态变化,有没有不可达的状态和非法的状态,可能不可能产生非法的状态转移等。例如,现在的智能手机中都内嵌了音乐播放器的功

能,那么在测试音乐播放器的时候,如果某款手机的音乐播放器如下表1表示,则采用状态图如何表示其状态转移机制呢?

表1 音乐播放器的状态事件表

按键	Idle	例	播放	进	录音
R (倒)	倒	---	倒	倒	---
P (播放)	播放	播放	---	播放	---
F (进)	进	进	---	---	---
RC (录音)	录音	---	---	---	---
S (Idle)	---	Idle	Idle	Idle	Idle

根据状态图的使用原理和规则结合音乐播放器的状态事件表,可作出如下状态迁移图,即状态图,如图6所示,然后根据状态迁移图很容易得出状态事件树,从而根据状态事件树的一条分支设计一个测试用例,机械的完成测试用例的设计。

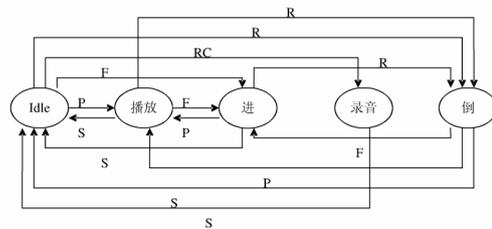


图6 音乐播放器的状态图

4 结束语

软件测试中最常用也是最有效的测试技术莫过于白盒测试技术和黑盒测试技术,而无论是在白盒测试中还是在黑盒测试中,最关键的部分都是软件测试用例的设计。每一种测试技术中的测试用设计方法都有很多,但是基于图的直观性和逻辑性,所以大部分测试用例设计方法都离不开图理论和图思想的支撑,本文是对图理论与图思想在两种关键软件测试技术中的应用总结。通过本文,将对软件测试用例设计过程中的图得到充分的理解与应用。

参考文献:

- [1] 《软件工程》 彭鑫, 赵文耘等, 机械工业出版社 2018 年
- [2] 《全程软件测试(第三版)》 朱少民等, 人民邮电出版社 2019 年
- [3] 《结构化并行程序设计: 高效计算模式》 迈克尔·麦库尔等, 机械工业出版社 2018 年