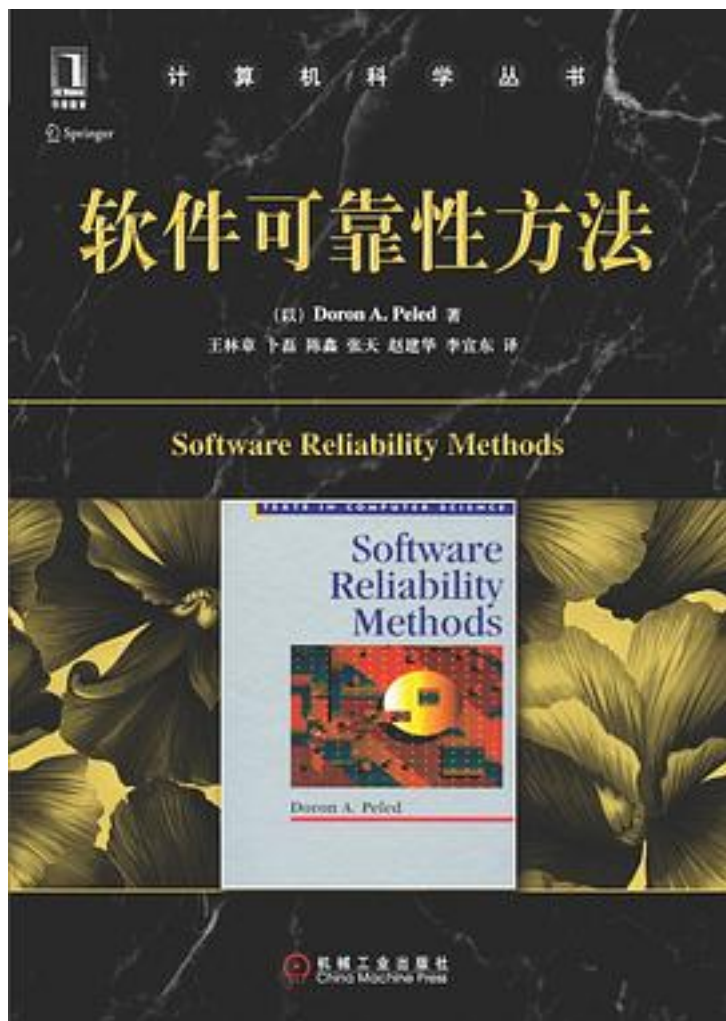


软件可靠性方法



[软件可靠性方法_下载链接1](#)

著者: (以色列) Doron A. Peled

出版者: 机械工业出版社

出版时间: 2012-3

装帧: 平装

isbn: 9787111365532

【名人推荐】

我第一次翻开这本书时，立刻被这本书的覆盖范围之广所深深打动，它覆盖了规约和建模、演绎验证、模型检验、进程代数、程序测试、状态与消息序列图。除了对每个方法进行了相当深入的介绍以外，本书还讨论了应当在何时选取何种方法以及在选择这些方法时所必须做出的权衡。书中结合当前工具，使用很多具有挑战性的实例来说明各种技术。我还没看见过其他任何覆盖同样内容的书籍能达到如此的深度。

同时，本书描述了应用形式化方法的过程：从建模和规约开始，然后选择一个合适的验证技术，最后测试程序。这些知识在实践中是十分必要的，但是却很少在软件工程的课本里面出现。我确信这本书将会取得巨大的成功。我向所有对软件可靠性问题感兴趣的读者强烈推荐这本书。

—— Edmund M. Clarke教授

图灵奖获得者，卡内基-梅隆大学

【内容简介】

用于创建可靠软件的形式化方法一直处于不断的开发和改进之中。最近，人们对于形式化方法工具的重要组成有了更深入的理解，从软件开发业界逐渐接受可靠性工具这一点就可以体现出来。

本书介绍了各种能解决软件可靠性问题的方法。理想情况下，形式化方法应该用起来直观，学起来简洁、快速，对开发过程的影响微乎其微。本书对各种方法进行了比较，揭示了它们各自的优点和缺点，同时紧扣自动机理论和逻辑这两个主题。在尽可能减少背景知识介绍的前提下，本书向非专家读者描述了多种技术，并且针对软件工程领域的研究人员和专业人士介绍了一些高级技术。

本书主题和特点：

集中介绍目前常用的重要软件可靠性方法，并将它们互作比较，这些方法包括：演绎验证、自动验证、测试和进程代数

为具体项目的软件选择过程提供有用信息

提供了大量的练习、项目和连续性的实例，方便读者学习形式化方法并能够亲手使用这些工具

介绍了支持形式化方法的数学原理

对于该领域未来的研究方向，以及开发新方法和改进现有技术提出了有益的见解

作者介绍：

【作者简介】

Doron A. Peled 以色列巴依兰大学（Bar Ilan University）计算机科学系教授。主要研究领域为并发理论、形式化验证、形式化规约、编程语言语义、模型检验、有限自动机、软件测试、时序逻辑等。著有多部书籍和论文。

【译者简介】

王林章

南京大学计算机科学与技术系副教授、硕士生导师，南京大学计算机软件新技术国家重点实验室主任助理。主要研究方向为软件工程、新型软件测试方法、模型驱动软件测试与验证、自动化软件测试工具。目前面向本科生、研究生讲授软件工程、软件体系结构、软件测试等课程。

目录: 出版者的话

中文版序

译者序

英文版序

前言

第1章 引言1

1.1 形式化方法2

1.2 开发与学习形式化方法3

1.3 使用形式化方法5

1.4 应用形式化方法6

1.5 本书概要7

第2章 预备知识8

2.1 集合表示法8

2.2 字符串和语言9

2.3 图10

2.4 计算复杂度和可计算性12

2.5 扩展阅读16

第3章 逻辑和定理证明17

3.1 一阶逻辑17

3.2 项17

3.2.1 赋值和解释18

3.2.2 多个论域上的结构19

3.3 一阶公式19

3.4 命题逻辑23

3.5 证明一阶逻辑公式24

3.5.1 正向推理25

3.5.2 反向推理26

3.6 证明系统的属性26

3.6.1 正确性27

3.6.2 完备性27

3.6.3 可判定性27

3.6.4 结构完备性28

3.7 证明命题逻辑属性28

3.8 一个实用的证明系统29

3.9 证明示例31

3.10 机器辅助证明37

3.11 机械化定理证明器39

3.12 扩展阅读39

第4章 软件系统建模40

4.1 顺序系统、并发系统及反应式系统41

4.2 状态42

4.3 状态空间43

4.4 转换系统44

4.5 转换的粒度47

4.6 为程序建模的例子48

4.6.1 整数除法48

4.6.2 计算组合数49

4.6.3	Eratosthenes筛法	50
4.6.4	互斥	52
4.7	非确定性转换	53
4.8	将命题变量赋给状态	54
4.9	合并状态空间	55
4.10	线性视角	56
4.11	分支视角	57
4.12	公平性	58
4.13	偏序视角	61
4.13.1	一个银行系统的例子	61
4.13.2	线性化和全局状态	63
4.13.3	一个简单的例子	64
4.13.4	偏序模型的应用	65
4.14	形式化建模	65
4.15	一个项目的建模	67
4.16	扩展阅读	68
第5章	形式化规约	69
5.1	规约机制的属性	69
5.2	线性时序逻辑	70
5.3	公理化LTL	74
5.4	LTL规约示例	74
5.4.1	交通灯	74
5.4.2	顺序程序的属性	75
5.4.3	互斥	76
5.4.4	公平性条件	76
5.5	无限字上的自动机	77
5.6	使用Büchi自动机作为规约	79
5.7	确定性Büchi自动机	80
5.8	其他规约机制	81
5.9	复杂的规约	83
5.10	规约的完整性	83
5.11	扩展阅读	84
第6章	自动验证	85
6.1	状态空间搜索	86
6.2	状态表示方法	87
6.3	自动机结构体系	88
6.4	合并Büchi自动机	89
6.4.1	广义Büchi自动机	90
6.4.2	将广义Büchi自动机转换为简单Büchi自动机	91
6.5	Büchi自动机求补	92
6.6	检验空集	93
6.7	模型检验范例	94
6.8	将LTL转换为自动机	95
6.9	模型检验的复杂度	100
6.10	表示公平性	102
6.11	检验LTL规约	102
6.12	安全属性	103
6.13	状态空间爆炸问题	104
6.14	模型检验的优点	105
6.15	模型检验的缺点	105
6.16	选择自动验证工具	105
6.17	模型检验项目	105
6.18	模型检验工具	106
6.19	扩展阅读	106

第7章 演绎式软件验证	107
7.1 流程图程序的验证	107
7.2 含数组变量的验证	111
7.2.1 含数组变量赋值的问题	112
7.2.2 修改证明系统	112
7.3 完全正确性	114
7.4 公理式程序验证	117
7.4.1 赋值公理	117
7.4.2 空语句公理	117
7.4.3 左强化规则	117
7.4.4 右弱化规则	118
7.4.5 顺序组合规则	118
7.4.6 if-then-else规则	118
7.4.7 while规则	118
7.4.8 begin-end规则	119
7.4.9 示例：整数除法	119
7.5 并发程序的验证	121
7.6 演绎验证的优点	124
7.7 演绎验证的缺点	125
7.8 证明系统的正确性和完备性	126
7.9 组合性	127
7.10 演绎验证工具	128
7.11 扩展阅读	128
第8章 进程代数与等价关系	129
8.1 进程代数	130
8.2 通信系统的演算	131
8.2.1 动作前缀	131
8.2.2 选择	132
8.2.3 并发组合	132
8.2.4 限制符	133
8.2.5 重标记	133
8.2.6 等式定义	133
8.2.7 agent 0	135
8.2.8 传值agent	135
8.3 示例：Dekker算法	135
8.4 建模问题	137
8.5 agent之间的等价性	138
8.5.1 迹等价	139
8.5.2 失败等价	139
8.5.3 模拟等价	140
8.5.4 互模拟和弱互模拟等价	142
8.6 等价关系的层级	142
8.7 用进程代数研究并发	143
8.8 计算互模拟等价	145
8.9 LOTOS	147
8.10 进程代数工具	148
8.11 扩展阅读	148
第9章 软件测试	150
9.1 审查和走查	151
9.2 控制流覆盖准则	152
9.2.1 语句覆盖	153
9.2.2 边覆盖	153
9.2.3 条件覆盖	153
9.2.4 边/条件覆盖	154

9.2.5 条件组合覆盖	154
9.2.6 路径覆盖	154
9.2.7 不同覆盖准则的比较	155
9.2.8 循环覆盖	155
9.3 数据流覆盖准则	155
9.4 传播路径条件	157
9.4.1 示例：GCD程序	159
9.4.2 含有输入语句的路径	160
9.5 等价类划分	160
9.6 待测代码预处理	160
9.7 检查测试套件	161
9.8 组合性	162
9.9 黑盒测试	163
9.10 概率测试	164
9.11 测试的优点	165
9.12 测试的缺点	166
9.13 测试工具	166
9.14 扩展阅读	166
第10章 组合形式化方法	167
10.1 抽象	167
10.2 组合测试与模型检验	171
10.2.1 直接检验	171
10.2.2 黑盒系统	172
10.2.3 组合锁自动机	172
10.2.4 黑盒死锁检测	172
10.2.5 一致性测试	173
10.2.6 检验重置的可靠性	175
10.2.7 黑盒检验	176
10.3 净室方法	177
10.3.1 验证	177
10.3.2 证明审查	177
10.3.3 测试	177
10.4 扩展阅读	178
第11章 可视化	179
11.1 在形式化方法中运用可视化	179
11.2 消息序列图	180
11.3 可视化流程图和状态机	182
11.4 层次状态图	184
11.4.1 层次化状态	184
11.4.2 统一的出口和入口	185
11.4.3 并发	185
11.4.4 输入和输出	185
11.5 程序文本的可视化	186
11.6 Petri网	186
11.7 可视化工具	188
11.8 扩展阅读	188
结束语	189
参考文献	191
• • • • •	(收起)

标签

形式化方法

软件工程

可靠性

软件可靠性

计算机科学

计算机

软件

验证

评论

一本浓缩了精华的大survey，写得很好，翻译从国内整体水平来讲，尚属不错，对那些有insights的部分没有离谱的感觉，中规中矩，当然内容上有不少小bug。

总要把不喜欢的事情尽快做完，才能去做喜欢的事情，正如科研之于我。把不喜欢、不擅长的事情做完，才有成就感。把喜欢的、擅长的事情做好，才有幸福感。P.S. 书的确是好书，不可否认，不能以主观的喜好来代替客观的评判。

软件可靠性方法是一本非常经典的讲解软件形式化方法的书，书的作者Doron A. Peled不仅介绍了形式化方法所用到的逻辑、自动机理论等基础的数学知识，还深入的讨论了软件形式化常用到的推理验证、软件测试、模型检验等重要的软件形式化方法。

介绍了提高软件可靠性的方法，model checking,theorem prove,software testing

[软件可靠性方法 下载链接1](#)

书评

[软件可靠性方法 下载链接1](#)