

李必信 编著

程序切片技术及其应用

Program Slicing Technique and Its Applications

切片技术 及其应用

一个求最大或最小绝对值的程序	基于前置条件的切片
1 if(a>0 and b>0) then 2 if(a>0) then 3 if(a>b) then 4 alternate =a; 5 alternate =b; 6 else 7 alternate =a; 8 alternate =b; 9 fi 10 else 11 if(a>b) then 12 alternate =b; 13 alternate =a; 14 else 15 alternate =a; 16 alternate =b; 17 fi 18 fi 19 else 20 alternate =0; 21 alternate =0; 22 fi	1 if(a>0 and b>0) then 2 if(a>0) then 3 if(a>b) then 4 alternate =a; 6 else 7 alternate =b; 9 fi 10 else 11 if(a>b) then 12 alternate =b; 14 else 15 alternate =a; 17 fi 18 fi 19 else 20 alternate =0; 22 fi
基于前置条件的切片	基于规则的切片
3 if(a>b) then 4 alternate =a; 5 alternate =b; 6 else 7 alternate =b; 8 alternate =a; 9 fi	3 if(a>b) then 4 alternate =a; 6 else 7 alternate =b; 9 fi



科学出版社

www.sciencep.com

程序切片技术及其应用

李必信 编著

国家自然科学基金资助(项目编号:60473065)
计算机软件新技术国家重点实验室(南京大学)资助
东南大学优秀青年教师教学科研资助计划资助
东南大学科技出版基金资助

科学出版社

北 京

内 容 简 介

本书主要介绍程序切片的起源和发展,程序切片技术的图论基础,程序切片的各种变体(如静态切片、动态切片、有条件切片、并发切片、面向对象程序的切片、体系结构切片和规约切片等),计算程序切片的各种方法,以及程序切片技术在软件调试、波动分析、软件测试、度量、重用、程序理解、逆向工程和软件安全等方面的应用,并讨论了程序切片技术的发展趋向。

本书适合作为高等院校计算机软件专业学生学习“软件工程”、“软件分析与测试”等课程的参考书或工具书,也适合作为软件分析与测试研究人员的参考书。

图书在版编目(CIP)数据

程序切片技术及其应用/李必信编著. —北京:科学出版社,2006

ISBN 7-03-016944-1

I. 程… II. 李… III. ①软件—分析 ②软件—测试 IV. TP311.5

中国版本图书馆 CIP 数据核字(2006)第 013265 号

责任编辑:鞠丽娜 刘亚军/责任校对:都 岚

责任印制:吕春珉/封面设计:王 浩

科 学 出 版 社 出 版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

双 青 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2006年3月第 一 版 开本:B5(720×1000)

2006年3月第一次印刷 印张:19 1/2

印数:1—2 500 字数:390 000

定价:35.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

销售部电话 010-62136131 编辑部电话 010-62138978-8002(BI01)

前 言

计算机系统的发展给软件提出了更高的要求,软件质量提升这一课题的研究越来越得到从业人员的重视,而软件的可靠性、正确性、安全性等是保证软件具有高质量的关键因素。近年来,软件从业人员,特别是软件工程的从业人员,为保证开发中软件的质量,提出了多种技术与方法。其中类型推理、代码分析、软件测试、形式验证和 CMM 等就是典型代表。程序切片是由 M.Weiser 先生于 1979 年在他的博士论文中首先建立起来的一种程序分解技术。它通过寻找程序内部的相关性来分解程序,再通过对分解所得程序切片的分析达到对整个程序的分析 and 理解。

程序切片技术的研究有着极其重要的理论和实际意义。它极大地丰富了程序分析、程序理解、软件维护的理论基础,为软件工程的各个阶段提供理论和技术支持,保证软件开发过程沿着严谨、正确的道路走下去。例如,在需求规约层,我们可以通过对软件形式规约的切片,发现规约中存在的错误,分析产生错误的原因,进而纠正错误;在软件体系结构层,我们可以利用体系结构切片技术,逐步对体系结构规约进行精化(refinement),并验证体系结构的正确性,从而保证按此体系结构设计软件的正确性和可靠性;在设计阶段,我们同样可以利用模块间切片技术,尽可能地设计出高内聚、低耦合的模块,从而降低程序的复杂性,并为软件重用奠定良好的基础;在编码阶段,程序切片技术的应用更加广泛,它在程序分析、理解、测试、调试、度量以及维护过程中都有着极其广泛的用途。作为一种应用广泛的程序分析技术,程序切片正对软件工程领域产生越来越大的影响。

编著本书的目的在于向那些已学过大学“软件工程”、“程序设计语言”等计算机软件基础课程而希望进一步深入学习程序分析方法的读者,提供一部全面介绍程序切片技术的基础读物。本书内容包括程序切片的基本原理和方法,基本工具和技巧,以及程序切片在软件工程各个方面的运用。本书力求以浅显易懂的形式讲述基本理论和方法,并且紧密结合实践。

本书共分上、中、下三篇和附录。

上篇主要阐述程序切片的发展历史和基本原理,它由 9 章构成,分别阐述程序切片基本概况(第 1 章),图论基础(第 2 章),静态程序切片(第 3 章),动态程序切片(第 4 章),条件程序切片(第 5 章),并发程序切片(第 6 章),面向对象程序切片(第 7 章),规约切片(第 8 章),程序切片变体(第 9 章)。

中篇主要阐述程序切片的基本应用,它由 8 章构成,分别论述程序切片在软件调试(第 10 章),波动分析(第 11 章),软件测试(第 12 章),软件维护(第

13 章), 软件度量 (第 14 章), 软件安全 (第 15 章), 软件重用 (第 16 章), 应用扩展 (第 17 章) 等方面的应用现状及热点研究课题。

下篇主要介绍我们在层次切片模型方面的工作, 并针对一些新型程序切片课题进行相关讨论, 旨在开拓读者的视野。本部分由层次切片模型及其实现 (第 18 章)、层次切片模型的应用 (第 19 章) 和结束语 (第 20 章) 3 章构成。

附录给出了汉英名词对照和缩略语英汉对照。

作者多年来一直从事程序切片技术研究, 并担任东南大学“软件分析与测试”, “软件开发方法与技术”等课程的教学工作。本书主要内容来源于作者的博士论文和相关的研究论文, 以及国内外一些著名的相关文献。

作者衷心感谢恩师南京大学的郑国梁教授, 师兄南京大学的李宣东教授、陈家骏教授、赵建华教授, 以及东南大学的罗军舟教授、徐宝文教授和瞿裕忠教授, 是他们的鼓励和无私的帮助激发了我完成本书写作的决心; 同时还要感谢我在南京大学学习期间一起做研究的 SEG 同仁, 他们是郑滔、樊晓聪、崔萌、梁佳、刘小东、张勇祥、钱雨、沈驿梅、朱平、谭毅、刘芳和陈雨亭等; 另外还要感谢东南大学江苏省软件质量研究所的聂长海老师、张迎周老师、史亮同学和章晓芳同学, 以及东南大学计算机系 ERICS 组的老师和同学, 他们是周颖、张鹏程、王艳臣、莫军辉、苏志勇、杨利利和周宇等。

这里还要特别感谢上海交通大学的赵建军教授, 多年来, 很多灵感得益于赵教授的论文和与赵教授的多次交流。北京大学的梅宏教授对本书稿提出了许多具有建设性的意见, 在此表示衷心的感谢。作者在南京大学学习期间以及在东南大学工作期间均得到了计算机软件新技术国家重点实验室 (南京大学) 的大力支持。另外, 作者在研究程序切片技术的过程中得到了国家 863 基金和国家自然科学基金资助, 在此一并表示感谢。

最后, 感谢科学出版社的鞠丽娜老师和刘亚军老师, 正是她们的关心和支持, 才使得本书的写作、出版、发行得以顺利完成。

程序切片技术发展到今天, 已经形成多个分支, 并积累了丰富的理论研究和经验研究成果。由于时间问题, 本书不可能包含所有有关程序切片的内容, 感兴趣的读者还需一些辅助资料才能对它有更加深入地理解。由于作者水平有限, 错误和遗漏在所难免, 恳请读者批评指正。

作 者

2005 年 11 月于东南大学

目 录

上篇 程序切片技术基本原理

第 1 章 概论	3
1.1 程序切片技术的起源和发展	3
1.1.1 从数据流方程到程序依赖图	4
1.1.2 从可执行的程序切片到不可执行的程序切片	5
1.1.3 静态切片、动态切片和有条件切片	6
1.1.4 后向切片和前向切片	7
1.1.5 从源程序代码切片到软件规约切片	7
1.1.6 从单一切片到多种切片	8
1.2 程序切片技术的应用概述	9
1.2.1 程序调试	9
1.2.2 软件维护	9
1.2.3 回归测试	10
1.2.4 软件度量	11
1.2.5 软件重用	11
1.2.6 软件安全	11
1.3 程序切片工具简介	12
1.3.1 支持 C 语言的 PST	12
1.3.2 支持 Ada 语言的 PST	13
1.3.3 支持 Oberon-2 语言的 PST	13
1.3.4 支持 Java 语言的 PST	13
1.3.5 其他 PST	14
小结	14
思考题	14
参考文献	15
第 2 章 图论基础	17
2.1 控制流图	17
2.1.1 基本模块	17
2.1.2 控制流图定义	18
2.1.3 基本属性	19
2.2 控制流分析	20

2.2.1 控制流	20
2.2.2 控制流的表示方法	20
2.2.3 支配节点和后必经节点	21
2.2.4 循环识别	23
2.3 数据流分析	24
2.3.1 可到达定义	24
2.3.2 数据流方程	26
2.3.3 活性分析	28
2.4 数据依赖和控制依赖	30
2.4.1 控制依赖	30
2.4.2 数据依赖	31
2.5 程序依赖图	31
2.5.1 过程内依赖图	32
2.5.2 过程间依赖图	32
小结	32
思考题	33
参考文献	33
第3章 静态程序切片	35
3.1 引言	35
3.2 Mark Weiser 程序切片	35
3.2.1 初步理解	36
3.2.2 基本术语	37
3.2.3 Mark Weiser 的数据流算法	42
3.3 过程内切片	43
3.3.1 构造程序依赖图	43
3.3.2 图可达性算法	44
3.3.3 例子分析	44
3.4 过程间切片	46
3.4.1 构造系统依赖图	46
3.4.2 构造特征子图的算法	48
3.4.3 过程间切片的算法	51
小结	54
思考题	54
参考文献	54

第4章 动态程序切片	56
4.1 引言	56
4.2 基本术语	57
4.2.1 程序依赖图和静态切片	57
4.2.2 执行历史和动态切片准则	58
4.3 Agrawal 和 Horgan 的动态切片	59
4.3.1 动态切片方法 1	59
4.3.2 动态切片方法 2	60
4.3.3 动态切片方法 3	62
4.3.4 动态切片方法 4	63
4.4 Korel 和 Laski 的动态切片	65
小结	65
思考题	66
参考文献	66
第5章 有条件程序切片	68
5.1 引言	68
5.2 有条件切片	69
5.2.1 准静态切片	69
5.2.2 同时动态切片	70
5.2.3 一般的有条件切片模型	70
5.2.4 有条件切片计算	72
5.3 切片模型关系分析	76
5.4 分割分析	77
小结	78
思考题	78
参考文献	79
第6章 面向对象程序切片	80
6.1 引言	80
6.2 传统系统依赖图的缺陷分析	81
6.3 面向对象系统依赖图	83
6.3.1 OOSDG 的基本组成模型	83
6.3.2 OOSDG 对 SDG 的扩充	84
6.3.3 类依赖图	86
6.3.4 虚函数调用图的构造	88
6.3.5 OOSDG 的构造算法	95

6.3.6 基于 OOSDG 的程序切片算法	97
小结	97
思考题	97
参考文献	98
第 7 章 并发程序切片	100
7.1 引言	100
7.2 Cheng 的并发程序切片思想	101
7.3 Krinke 多线程程序静态切片方法	102
7.3.1 线程控制流图	102
7.3.2 线程程序依赖图	104
7.3.3 基于 tPDG 的切片	105
7.4 Nanda 和 Ramesh 的并发程序切片方法	109
7.5 并发程序的动态切片	112
7.5.1 进程图和静态程序依赖图	112
7.5.2 进程图到并发图	113
7.5.3 构建 DPDG	113
7.6 面向对象并发程序的切片方法	114
7.6.1 Zhao 的早期方法	114
7.6.2 Java 并发程序的切片方法	115
小结	116
思考题	117
参考文献	117
第 8 章 规约切片	118
8.1 形式规约切片	118
8.1.1 静态形式规约切片	118
8.1.2 动态形式规约切片	121
8.1.3 其他形式规约切片	121
8.2 基于规约的程序切片	122
8.2.1 前置和后置条件	122
8.2.2 基于规约的切片	123
8.3 体系结构规约切片	126
8.3.1 体系结构规约	126
8.3.2 体系结构切片定义	128
8.3.3 体系结构信息流图和体系结构切片的计算	129
8.4 动态软件体系结构切片	130

8.5 JVM 规约切片	131
小结	132
思考题	132
参考文献	132
第 9 章 新型切片变体	135
9.1 无定型切片	135
9.1.1 程序投影	135
9.1.2 无定型简单性度量	136
9.1.3 无定型静态切片	137
9.1.4 无定型有条件切片	138
9.1.5 无定型切片的实现以及相关问题	139
9.2 削片	139
9.2.1 静态削片	139
9.2.2 动态削片	140
9.2.3 削片的性质和构造策略	141
9.3 砍片	143
小结	143
思考题	144
参考文献	144

中篇 程序切片技术的基本应用

第 10 章 程序调试	147
10.1 引言	147
10.1.1 什么是程序调试	147
10.1.2 为什么用切片进行调试	148
10.2 如何用切片辅助程序调试	150
10.2.1 调试中的错误分析及切片选取	150
10.2.2 面向对象程序切片与调试	151
10.3 基于切片的调试工具	151
10.3.1 C-Debug	151
10.3.2 SPYDER	152
小结	153
思考题	154
参考文献	154

第 11 章 波动分析	155
11.1 引言	155
11.2 程序切片与 REA 过程	157
11.3 后向切片存在的必要性	157
11.3.1 定义修改	157
11.3.2 使用修改	158
11.3.3 控制修改	158
11.4 程序切片运算	159
11.5 直接波动和诱导波动	161
11.5.1 通用程序切片	161
11.5.2 REA 例子	162
小结	163
思考题	164
参考文献	164
第 12 章 软件测试	165
12.1 引言	165
12.2 基于程序切片的软件测试	165
12.2.1 例子简介	166
12.2.2 依赖图模型	167
12.2.3 基本性质	169
12.3 回归测试	170
12.3.1 受影响的定义-使用关系类型	171
12.3.2 BackwardWalk 算法	172
12.3.3 ForwardWalk 算法	173
12.3.4 进一步讨论	176
12.3.5 回归测试的一般步骤	181
12.4 基于切片技术的软件测试工具模型	182
小结	184
思考题	184
参考文献	184
第 13 章 软件维护	186
13.1 引言	186
13.2 软件维护模型	187
13.3 分解切片与软件维护	187
13.3.1 分解切片	188

13.3.2 使用分解切片的几条规则	189
13.4 联合切片与软件维护	191
13.4.1 联合切片	191
13.4.2 利用联合切片维护软件	192
13.5 基于切片的软件维护模型	193
小结	193
思考题	193
参考文献	194
第 14 章 复杂性度量	195
14.1 引言	195
14.2 早期基于切片的度量	197
14.3 内聚度量	198
14.3.1 数据切片	198
14.3.2 胶水、强力胶水和粘性	200
14.3.3 内聚度量	200
14.3.4 类内切片和类内聚	202
14.4 耦合度量	205
14.4.1 Java 源代码中存在的耦合问题分析	206
14.4.2 基于切片的 Java 耦合度量框架	210
小结	214
思考题	214
参考文献	214
第 15 章 软件安全	215
15.1 引言	215
15.2 软件安全分析的几种常用方法	216
15.2.1 失效模式效应分析法	216
15.2.2 软件故障树分析法	217
15.2.3 Petri 网分析法	217
15.3 临界安全组件与软件故障树分析	219
15.3.1 共同模式失效问题	219
15.3.2 临界安全组件	219
15.4 基于程序切片共同模式失效分析方法	220
15.4.1 基本原理	221
15.4.2 例子分析	222
小结	225

思考题	225
参考文献	225
第 16 章 软件重用	227
16.1 引言	227
16.2 转换切片与重用	228
16.3 有条件切片与重用	228
16.4 规约驱动切片与重用	230
16.5 软件体系结构切片与重用	233
小结	234
思考题	234
参考文献	234
第 17 章 应用扩展	235
17.1 程序分析理解	235
17.1.1 Lucia 等人的初步思想	236
17.1.2 Korel 的大型程序理解手段	237
17.1.3 Kumar 的 CONCEPT 技术	238
17.2 逆向工程和再工程	239
17.2.1 传统切片	239
17.2.2 接口切片	240
17.3 Tip 的类型错误定位方法	241
17.3.1 静态语义规约和类型检查	241
17.3.2 项重写和依赖追踪	242
17.3.3 切片精确度问题	243
17.4 程序验证	244
17.5 其他应用	245
小结	245
思考题	246
参考文献	246

下篇 程序切片技术展望

第 18 章 层次切片模型及其实现	251
18.1 面向对象程序的层次结构模型	251
18.2 层次切片模型	252
18.3 SSA 算法的基本思想	253
18.4 HSM 和 SSA 的实现	254

18.4.1 代码信息树	256
18.4.2 依赖图的生成算法和切片算法	262
18.5 Jato——Java 程序切片工具	273
18.5.1 依赖图生成层	274
18.5.2 切片生成层	275
18.6 层次切片复杂度	277
小结	278
思考题	278
参考文献	278
第 19 章 层次切片模型的应用	280
19.1 静态信息流分析	280
19.2 耦合度量	282
19.2.1 方法 m_1 和 m_2 之间的耦合度量	283
19.2.2 类 c 的耦合度量	283
19.3 内聚度量	284
19.3.1 子功能内聚	284
19.3.2 功能内聚	284
19.4 复杂度度量	285
小结	286
思考题	286
参考文献	286
第 20 章 结束语	288
20.1 基本原理总结	288
20.2 基本应用总结	288
20.3 未来研究课题	289
20.3.1 程序切片的形式语义	289
20.3.2 无定型程序切片	289
20.3.3 规约切片	290
20.3.4 基于规约程序切片	290
20.3.5 软件体系结构切片	290
20.3.6 程序切片应用	290
参考文献	291

附 录

附录 A 汉英名词对照	293
附录 B 缩略语英汉对照	296

上篇 程序切片技术基本原理

20 世纪 70 年代初期,随着计算机软件技术的发展,特别是计算机软件规模的扩大,相当一部分人正致力于研究较大规模计算机程序的调试和维护问题。在软件调试过程中有人发现:当一个大的计算机程序被分解成一个个较小的程序片段以后,很容易被构造、理解和维护。于是一些用于程序分解的技术和方法被研究人员提出来。例如,信息隐藏、数据抽象和 HIPO 等技术就是其典型代表。这些技术并非互斥,而是互相补充地完成程序分解和抽象的任务。通过对这些技术的反复运用,一些新型技术又不断涌现出来,使得程序分解技术得到进一步的发展和充实,其中程序切片起到了其他技术无可替代的作用,它的出现和发展注定了要在程序分解领域引起一场革命。

程序切片是一种用于分解程序的程序分析技术,它的原理和方法是由 M.Weiser 于 1979 年在他的博士论文中首次建立,后来分别于 1981 年和 1984 年在他的两篇论文中又得到了进一步的完善和推广。M.Weiser 博士不仅建立了程序切片的概念,并提出了基于控制流图的计算程序切片的算法,以及基于程序切片的软件度量基本框架等,为后来人们扩展程序切片及其应用奠定了坚实的基础。程序切片一经问世,就引起了软件从业人员极大的兴趣,很多研究者们开始进行这一课题的研究,经过 20 多年的努力,程序切片技术理论日趋完善,已经形成了多个分支,应用范围几乎遍及软件工程学科的各个方面。纵观全局,程序切片经历了以下几个发展阶段。

(1) 基于数据流方程的程序切片阶段

这一阶段主要是指由 M.Weiser 提出的程序切片概念,以及运用基于 CFG 的数据流方程来计算程序切片的算法。

(2) 基于依赖图的程序切片阶段

这一阶段主要包括:K.J.Ottenstein 等人分别于 1984 年和 1987 年引入了基于程序依赖图的图可达性算法,它能够计算当时流行的过程内后向切片;S.Horwitz 等人分别于 1988 年、1990 年和 1992 年引入了前向切片的概念及算法,过程间切片概念以及基于系统依赖图的两步遍历图可达性算法;B.Korel 和 J.Laski 于 1988 年,H.Agrawal 和 J.R.Horgan 于 1990 年分别引入了动态切片的概念和算法。

(3) 面向对象程序切片阶段

随着面向对象程序设计语言和设计方法的兴起,人们自 1996 年开始了面向

对象的程序切片技术的研究。其中, M.J.Horrald 等人利用类依赖图来扩充传统的系统依赖图使其能够表示面向对象的 C++ 程序, 然后利用改进的两步遍历图可达性算法来计算 C++ 程序的程序切片; C.Steindl 通过对各种各样的控制流和数据流的计算, 并结合他们所使用的一个项目语言来讨论面向对象程序的切片计算问题; 另外, J.Zhao 教授, D.Liang 博士和 Z.Chen 博士等人分别在面向对象动态程序切片、使用对象依赖图切片面向对象程序, 以及面向对象 Java 程序切片等方面各自提出了自己的观点。

(4) 程序切片发展“百花齐放”阶段

这一阶段的主要标志就是出现了各种各样的程序切片变体, 如切片、砍片, 还有数据切片、层次切片和无定型切片等。同时, 程序切片技术的应用也在此阶段得到了极大的丰富和发展。除了传统的基于切片的软件调试、软件测试和程序理解以外, 程序切片还被广泛运用于程序重组、逆向工程、软件维护、程序验证以及软件可靠性分析和建模等方面。

以上四个阶段的划分没有严格的时间限制, 甚至前后阶段之间存在着较大的时间重叠, 笔者只是为了问题叙述的方便, 仅供读者参考。另外, 除了这里提到的作者以外, 还有一大批从事程序切片研究并做出突出贡献的同仁, 他们的工作也是极具理论和应用价值的。例如, 有关并发程序切片, 有关有条件切片等, 我们将用独立的章节分析和介绍这些成果。

本篇主要是对程序切片技术发展的一个回顾, 所包含内容尽量避免深奥的理论赘述, 相信初学者可以比较轻松地学习每一个章节, 同时又能够较快地把握整个技术发展的轮廓。

本篇的主要内容组织如下。

第 1 章介绍程序切片技术的发展、基本原理和应用, 以帮助读者快速入门。

第 2 章介绍控制流图、程序依赖图、控制流分析和数据流分析的基本原理。

第 3 章主要介绍 M.Weiser 程序切片的基本思想, 过程内切片和过程间切片等, 目的是让读者了解程序切片发展早期阶段的有关情况。

第 4 章详细论述动态程序切片的基本原理, 讨论计算动态切片的基本方法。

第 5 章简述有条件切片的基本思想及其应用。

第 6 章讨论并发程序的切片问题。

第 7 章从面向对象程序的特点出发, 论述计算面向对象程序切片时遇到的问题和挑战, 以及目前采用的计算面向对象程序切片的方法等。

第 8 章重点阐述针对规约语言的程序切片思想和相应的算法等。

第 9 章讨论的是一些新型的程序切片问题, 对其中的许多问题进行初步的论述和探讨。

第 1 章 概 论

本章比较全面地概述了程序切片技术的起源和发展现状,把程序切片技术的发展概括为:从基于数据流方程的算法到基于依赖图的算法,从可执行的程序切片到不可执行的程序切片,从静态切片到动态切片再到有条件切片,从后向切片到前向切片,从代码切片到规约切片,从单一切片到多种切片的过程。内容还包括:切片准则的变化和发展,程序切片技术的应用现状,切片工具的研制和使用等。旨在让读者对这一热门技术有一个概括的了解,为阅读后续章节提供线索。

1.1 程序切片技术的起源和发展

程序切片的基本思想由 M.Weiser 于 1979 年在他的博士论文中首次建立。1981 年他又在国际软件工程大会 (ICSE'81) 上报告了这一新发现,立即引起了当时计算机界不小的震动,他提交给大会的论文《程序切片》(Program Slicing) 获得当届“今后十年最佳论文奖”。当时的情况是, M.Weiser 等人正在进行自动程序抽象的理论和实践研究,他们通过对源程序代码的分析以及对控制流图 (control flow graph, CFG) 的研究发现:程序的某一个输出只与源程序中部分语句和控制谓词 (control predicate) 有关,删除其他的语句和谓词并不影响该输出的结果。这表明:对该输出来说,源程序与删除不相关语句和谓词以后所得的可执行程序在语义上是一致的。M.Weiser 等人把这种只与某个输出有关的语句和谓词构成的程序称溪源程序的一种静态切片 (static slice),并提出了基于 CFG 的计算程序切片的算法。1984 年, M.Weiser 博士在国际权威杂志 IEEE Transactions on Software Engineering 上发表了同样题为《程序切片》的论文,进一步形式地阐述了程序切片技术的基本思想,以及如何基于数据流方程计算程序切片的方法。从此,越来越多的专家学者开始了这一课题的理论和应用研究。如今,程序切片技术的研究和发展已经经历了 20 多年的漫长旅程,在理论和应用两个方面都取得了丰硕的成果,这些研究成果在计算机学科的很多方面发挥了应有的作用。而且从各方面来看,程序切片技术还会作为一种主要手段在软件开发的各个阶段、程序设计语言、硬件描述语言、其他规约语言和形式化模型等的分析和测试方面继续发挥不可低估的作用。笔者根据多年从事程序切片技术的研究体会,从程序切片的算法、切片准则的演化和切片变体的形成、切片技术的应用以及切片工具等方面简要介绍程序切片技术的发展历程。