中国计算机学会
学术著作丛书

计算几何
——算法分析与设计

周培德　著
卢开澄　审

清华大学出版社
广西科学技术出版社

中国计算机学会学术著作丛书

计　算　几　何

——算法分析与设计

周培德　著
卢开澄　审

清 华 大 学 出 版 社
广 西 科 学 技 术 出 版 社

(京)新登字 158 号

内 容 简 介

本书系统地介绍了计算几何中的基本概念、求解诸多问题的算法及复杂性分析，概括了求解几何问题所特有的许多思想方法、几何结构与数据结构。全书共分 11 章，包括：预备知识、几何查找、多边形、凸壳、Voronoi 图、交与并、矩形几何、几何体的排列、算法的运动规划、几何拓扑网络设计、随机几何算法与并行几何算法等。

本书可作为高等院校计算机专业研究生或本科高年级学生的教材，也可作为相关专业科技工作者的参考书。

图书在版编目(CIP)数据

计算几何——算法分析与设计/周培德著．—北京：清华大学出版社，1999
(中国计算机学会学术著作丛书)
ISBN 7-302-03801-5

Ⅰ．计…　Ⅱ．周…　Ⅲ．①计算几何-算法分析②计算几何-算法设计　Ⅳ．018

中国版本图书馆 CIP 数据核字(1999)第 63354 号

出版者：清华大学出版社(北京清华大学学研楼，邮编 100084)
http://www.tup.tsinghua.edu.cn
印刷者：清华大学印刷厂
发行者：新华书店总店北京发行所
开　本：787×1092 1/16　**印张**：19　**字数**：444 千字
版　次：2000 年 3 月第 1 版　2000 年 5 月第 2 次印刷
书　号：ISBN 7-302-03801-5/TP・2141
印　数：2001～4500
定　价：29.00 元

清华大学出版社　广西科学技术出版社
计算机学术著作出版基金

评审委员会

出 版 说 明

近年来，随着微电子和计算机技术渗透到各个技术领域，人类正在步入一个技术迅猛发展的新时期。这个新时期的主要标志是计算机和信息处理的广泛应用。计算机在改造传统产业，实现管理自动化，促进新兴产业的发展等方面都起着重要作用，它在现代化建设中的战略地位愈来愈明显。计算机科学与其它学科的交叉又产生了许多新学科，推动着科学技术向更广阔的领域发展，正在对人类社会产生深远的影响。

科学技术是第一生产力。计算机科学技术是我国高科技领域的一个重要方面。为了推动我国计算机科学及产业的发展，促进学术交流，使科研成果尽快转化为生产力，清华大学出版社与广西科学技术出版社联合设立了“计算机学术著作基金”，旨在支持和鼓励科技人员，撰写高水平的学术著作，以反映和推广我国在这一领域的最新成果。

计算机学术著作出版基金资助出版的著作范围包括：有重要理论价值或重要应用价值的学术专著；计算机学科前沿探索的论著；推动计算机技术及产业发展的专著；与计算机有关的交叉学科的论著，有较大应用价值的工具书；世界名著的优秀翻译作品。凡经作者本人申请，计算机学术著作出版基金评审委员会评审通过的著作，将由该基金资助出版，出版社将努力做好出版工作。

基金还支持两社列选的国家高科技重点图书和国家教委重点图书规划中计算机学科领域的学术著作的出版。为了做好选题工作，出版社特邀请“中国计算机学会”、“中国中文信息学会”帮助做好组织有关学术著作丛书的列选工作。

热诚希望得到广大计算机界同仁的支持和帮助。

清华大学出版社
广西科学技术出版社
计算机学术著作出版基金办公室

1992年4月

序　言

计算机是当代发展最为迅猛的科学技术，其应用几乎已深入到人类社会活动和生活的一切领域，大大提高了社会生产力，引起了经济结构、社会结构和生活方式的深刻变化和变革，是最为活跃的生产力之一。计算机本身在国际范围内已成为年产值达2500亿美元的巨大产业，国际竞争异常剧烈，预计到本世纪末将发展为世界第一大产业。计算机科技具有极大的综合性质与众多科学技术相交叉而反过来又渗入更多的科学技术，促进它们的发展。计算机科技内容十分丰富，学科分支生长尤为迅速，日新月异，层出不穷。因此在我国计算机科技尚比较落后的情况下，加强计算机科技的传播实为当务之急。

中国计算机学会一直把出版图书刊物作为学术活动的重要内容之一。我国计算机专家学者通过科学实践，做出了大量成果，积累了丰富经验与学识。他们有撰写著作的很大积极性，但相当时期以来计算机学术著作由于印数不多，出版往往遇到不少困难，专业性越强越有深度的著作，出版难度越大。最近清华大学出版社与广西科学技术出版社为促进我国计算机科学技术及产业的发展，推动计算机科技著作的出版工作，特设立“计算机学术著作出版基金”，以支持我国计算机科技工作者撰写高水平的学术著作，并将资助出版的著作列为中国计算机学会的学术著作丛书。我们十分重视这件事，并已把它列为学会本届理事会的工作要点之一。我们希望这一系列丛书能对传播学术成果、交流学术思想、促进科技转化为生产力起到良好作用，能对我国计算机科技发展具有有益的导向意义，也希望我国广大学会会员和计算机科技工作者，包括海外工作和学习的神州学人们能积极投稿，出好这一系列丛书。

中国计算机学会

1992年4月20日

前　言

1975 年，Shamos（沙莫斯）和 Hoey（霍伊）利用计算机有效地计算平面点集的 Voronoi 图，并发表了一篇著名论文，从此计算几何诞生了。自那时以来该研究领域取得了辉煌的成果，使得计算几何成为理论计算机科学领域中一个新的极有生命力的子领域，并且，计算几何中的研究成果已在计算机图形学、化学、统计分析、模式识别、地理数据库以及其他许多领域中得到了广泛的应用。

计算几何研究的典型问题由几何基元（geometric primitives）、查找、优化等问题类组成。首先，几何基元包括凸壳和 Voronoi 图、多边形的三角剖分、划分问题（partition problems）与相交问题。E^{d+1}中点集 S 的下凸壳在 E^d 中的投影恰好是点集 S 在 E^d 中投影点的 Delaunay 三角剖分，然后由 Delaunay 三角剖分可以容易地得到 Voronoi 图。换言之，Voronoi 图是凸壳的特例，因此构造 E^{d+1}中点集凸壳的算法也可以用于构造 E^d 中点集的 Voronoi 图。对多边形的三角剖分问题可以提出如下要求：设计复杂度低的算法构造多边形三角剖分以及设计三角形最小角最大化的三角剖分算法；分割线段长度之和最小的三角剖分算法。前者已有线性时间的算法。划分问题是多边形三角剖分的推广，它要求把几何体划分成若干好的部分。所谓好的部分通常是指下述两个目标之一：划分成尽量少的凸部分；各凸部分最小角最大化。另外，在几何体中可以加入 Steiner 点（新的顶点），然后再进行划分，使得划分线段长度之和最小化或者提出其他要求。二维中的典型相交问题是：给定平面上 n 条直线段，确定所有的相交线对。三维中的相交问题一般考虑两个凸多面体的交以及两个多面体的交。

其次，几何查找包括点定位、可视化、区域查找等问题。计算机图形学、数据库中的区域查找及地理图形中的点定位等都是几何查找中的典型例子。在平面细分（planar subdivision）中定位一个询问点或者在 $E^d(d\geqslant 3)$内由 n 个超平面构成的结构中定位询问点的问题是一个典型问题，现在不仅有解决这个问题的确定型算法，而且设计了动态随机增量算法。给定平面上 n 个顶点的简单多边形 P，由点 q 向任一方向引射线 l，确定 l 与 P 相交的第一条边，这个问题的解决为可视化问题的求解提供了前提。E^d 中给定点集 S 及区域集合 B，$b\in B$，要求在 b 中查找 S 中的点，这是区域查找问题。

再次，几何优化包括参数查找和线性规划。参数查找技术是将一个优化问题的检验算法变成寻找解的算法，它必须满足某些条件（检验算法是可以并行的），并且具有广泛的应用性。例如，可用它来求解平面中 2 中心问题（选择斜率或距离，计算线段排列中第 k 个最左边的顶点），还可以用来完成三维空间中射线的安置。众所周知，有确定变元数目的线性规划问题已有线性时间算法求解，但对于广义线性规划是否存在多项式时间算法还有待进一步研究。

此外，计算几何中各种问题的下界的确定、推导下界的方法以及求解各种几何问题的算法的复杂性分析等，也是计算几何研究的重要内容。

计算几何中引入随机化之后，已经设计出非常有效的概率算法求解诸多几何问题。随机化给几何算法设计带来两种新的设计思想：基于随机抽样的分治方法；利用随机顺序插入产生随机增量结构。此外，随机几何算法的复杂性分析以及随机增量结构的非随机化也是重要的研究内容。

计算几何的新近发展包括几何抽样理论、计算实代数几何、计算拓扑、运动规划、并行计算几何、隐藏面的移动、结构和图形、网络生成以及计算机视觉中的几何问题等。计算机在各学科领域深层次的应用将为计算几何提出更多的研究问题，反之，计算几何的研究成果也将促进这些学科的进一步发展。

本书是以上述部分问题的前人研究成果与作者在本领域中所做的工作为基础撰写而成的。书中主要叙述解决某些几何问题的算法并兼顾复杂性分析。书中较详细地陈述了98个算法，其中39个算法是作者提出的，并编号为 Z_{1-1} 算法至 Z_{10-2} 算法。书中的某些Z算法是以基本形式出现的，对这些算法可以进行修改并降低复杂性的阶。

全书共分11章。第0章是预备知识。第1章介绍几何查找，包括点和点集的定位、范围查找与正交询问。第2章叙述多边形及多边形的划分。第3、4、7章分别介绍计算几何中的三个基本结构：凸壳、Voronoi图与几何体的排列。第5章阐述线段、多边形、半平面及凸多面体的交，多边形的并。第6章介绍矩形几何。第8章叙述算法的运动规划，所讨论的问题源于机器人学。第9章的中心论题是几何拓扑网络设计，其内容具有广泛的实用性。第10章介绍随机几何算法和并行几何算法。

描述算法有诸多形式。为了便于理解算法又不致于产生二义性，而且利于编制上机程序，本书描述算法就不拘泥于某一种形式，但以步骤、拟ALGOL语言、中文语句、数学表达式及逻辑运算符等相结合的描述形式为主。另外，本书以处于一般位置的几何体为讨论对象。

设计求解几何问题（或能转化为几何问题的问题）的算法应具备两个条件，一是分析并理解问题的几何特征，二是掌握计算几何中的几何结构、特殊的算法设计方法及相应的数据结构。计算几何和计算机科学中的算法设计与分析、数据结构等学科关系密切，它常常要用到这些学科的知识。但由于篇幅有限，本书将重点阐述计算几何学科的思想方法、几何结构、几何问题以及求解这些问题的算法和复杂性分析，而对算法分析、数据结构中的相关知识只简略地回忆。因此，阅读本书的读者应具备算法设计与分析、数据结构、程序设计等领域的知识，并能熟练地掌握某种高级语言，比如C语言，以便上机实现书中描述的算法。

本书可作为计算几何、计算理论、计算机图形学、计算机辅助设计、机器人学及相关领域科技工作者的参考书，也可作为计算机科学有关专业研究生或本科高年级学生的教材使用。

本书在撰写中引用了 Preparata F P 和 Shamos M I，O'Rourke J，Mulmuley K， MacGregor Smith J, Mehlhorn K, Chazelle B, Goodrich M T. 等诸多专家、学者的文献；特别是余荣老师提供了大量参考资料，为缩短本书的撰写周期付出了辛勤的劳动，作者在此表示衷心的感谢。此外，还要感谢清华大学出版社和广西科学技术出版社的“计算机学术著作出版基金”所给予的资助。

由于作者水平有限，书中定有缺点和错误，恳请广大读者批评指正。

周培德

1999.12

于北京理工大学计算机系

目　录

前言 ………………………………………………………………………… Ⅶ
第 0 章　预备知识 ……………………………………………………………… 1
0.1　算法与数据结构 …………………………………………………………… 1
0.1.1　算法 ………………………………………………………………… 1
0.1.2　数据结构 …………………………………………………………… 4
0.2　相关的几何知识 …………………………………………………………… 8
0.2.1　基本定义 …………………………………………………………… 8
0.2.2　线性变换群下的不变量 …………………………………………… 9
0.2.3　几何对偶性 ………………………………………………………… 10
0.3　计算模型 …………………………………………………………………… 11
第 1 章　几何查找(检索) ……………………………………………………… 14
1.1　点定位问题 ………………………………………………………………… 15
1.1.1　点 q 是否在多边形 P 内 …………………………………………… 15
1.1.2　确定点 q 在平面剖分中的位置 ………………………………… 20
1.2　范围查找问题 ……………………………………………………………… 27
1.2.1　多维二叉树(k-D 树)的方法 ……………………………………… 28
1.2.2　直接存取方法 ……………………………………………………… 30
1.2.3　范围树方法 ………………………………………………………… 31
1.3　判定点集是否在多边形内 ………………………………………………… 32
1.4　平面中线段集和空间中三角形集的正交询问 …………………………… 34
1.4.1　吊床询问及推广的吊床询问 ……………………………………… 35
1.4.2　正交限制 …………………………………………………………… 36
第 2 章　多边形 ………………………………………………………………… 38
2.1　凸多边形 …………………………………………………………………… 38
2.2　简单多边形 ………………………………………………………………… 42
2.3　多边形的三角剖分 ………………………………………………………… 47
2.4　多边形的凸划分 …………………………………………………………… 49
第 3 章　凸壳 …………………………………………………………………… 57
3.1　凸壳的基本概念 …………………………………………………………… 57
3.2　计算凸壳的算法(二维) …………………………………………………… 60
3.2.1　卷包裹法 …………………………………………………………… 60
3.2.2　格雷厄姆方法 ……………………………………………………… 61
3.2.3　分治算法 …………………………………………………………… 62

3.2.4 Z_{3-1}算法和 Z_{3-2}算法 …… 64
3.2.5 实时凸壳算法…… 66
3.2.6 增量算法…… 70
3.2.7 近似凸壳算法…… 71
3.3 计算凸壳的算法(三维)…… 71
3.3.1 基本概念…… 71
3.3.2 卷包裹法…… 73
3.3.3 分治算法…… 74
3.3.4 Z_{3-3}算法 …… 76
3.3.5 增量算法…… 77
3.4 凸壳的应用…… 78
3.4.1 确定任意多边形的凸、凹顶点 …… 78
3.4.2 利用凸壳求解货郎担问题…… 80
3.4.3 凸多边形直径…… 82
3.4.4 连接两个多边形成一条回路…… 84
第4章 Voronoi 图及其应用 …… 88
4.1 Voronoi 图的基本概念 …… 88
4.2 构造 Voronoi 图的算法…… 92
4.2.1 半平面的交…… 92
4.2.2 增量构造方法…… 93
4.2.3 分治法…… 95
4.2.4 减量算法…… 97
4.2.5 平面扫描算法…… 98
4.2.6 构造最远点意义下 Voronoi 图的算法 …… 100
4.3 平面点集的三角剖分 …… 101
4.3.1 平面点集三角剖分的贪心算法 …… 101
4.3.2 Delaunay 三角剖分与多边形内部点集的三角剖分 …… 103
4.3.3 平面点集三角剖分的算法 …… 105
4.4 Voronoi 图与三角剖分的应用 …… 110
4.4.1 最近邻近 …… 110
4.4.2 最大化最小角的三角剖分 …… 110
4.4.3 最大空圆 …… 111
4.4.4 最小生成树 …… 114
4.4.5 货郎担问题 …… 115
4.4.6 中轴 …… 116
4.4.7 Voronoi 图与凸壳的关系 …… 121
4.4.8 Voronoi 图的推广 …… 123
4.4.9 几何数据压缩 …… 130

第 5 章　交与并 …… 133
5.1　线段交的算法 …… 133
5.2　多边形的交 …… 139
5.2.1　凸多边形交的算法 …… 139
5.2.2　星形多边形交的算法 …… 143
5.2.3　任意简单多边形交的算法 …… 144
5.3　半平面的交及其应用 …… 146
5.3.1　半平面的交 …… 146
5.3.2　两个变量的线性规划 …… 147
5.4　多边形的并 …… 153
5.5　凸多面体的交 …… 157
第 6 章　矩形几何 …… 161
6.1　判定垂直、水平线段是否相交的算法 …… 161
6.2　矩形几何问题的特征及解决问题的途径 …… 162
6.3　矩形并的面积与周长 …… 163
6.4　矩形并的轮廓 …… 166
6.5　矩形并的闭包 …… 168
6.6　矩形并的非平凡轮廓和外轮廓 …… 171
6.7　矩形的交 …… 173
6.8　应用举例 …… 175
第 7 章　几何体的排列 …… 177
7.1　基本概念 …… 177
7.2　确定直线排列的算法 …… 180
7.3　对偶性 …… 181
7.4　Voronoi 图 …… 185
7.4.1　一维情况 …… 185
7.4.2　二维情况 …… 187
7.5　应用 …… 187
7.5.1　k-最近邻近 …… 187
7.5.2　删去隐藏面 …… 188
7.5.3　特征图 …… 189
7.5.4　点集的分割 …… 190
第 8 章　算法的运动规划 …… 192
8.1　最短路径 …… 192
8.1.1　可视图及其构造 …… 192
8.1.2　Dijkstra 算法 …… 193
8.2　移动圆盘 …… 196
8.3　平移凸多边形 …… 197

8.4 移动杆状机器人 …… 200
8.4.1 网格分解 …… 201
8.4.2 收缩方法 …… 203
8.5 机器人臂的运动 …… 204
8.5.1 可达性 …… 205
8.5.2 构造可达性 …… 206
8.6 可分离性 …… 209
8.6.1 多种可分离性 …… 209
8.6.2 借助于平移的可分离性 …… 209
8.6.3 分离问题是 NP-难的 …… 210
8.6.4 模拟河内塔问题 …… 211
第 9 章 几何拓扑网络设计 …… 212
9.1 $G(S)$问题 …… 212
9.1.1 最大间隙问题(MAX G) …… 213
9.1.2 最小覆盖问题(MIN C) …… 215
9.1.3 最近对问题(CPP) …… 218
9.1.4 所有最近邻近问题(ANNP) …… 219
9.1.5 邮局问题(POFP) …… 219
9.2 $G(E)$问题 …… 220
9.2.1 EMST 问题 …… 221
9.2.2 欧几里德 TSP …… 222
9.2.3 欧几里德最大生成树问题(EMXT) …… 223
9.3 $G(S,E)$问题 …… 224
9.3.1 欧几里德 Steiner 最小树问题(ESMT) …… 225
9.3.2 直线 Steiner 最小树问题(RSMT) …… 227
9.4 $G(\Omega)$问题 …… 228
9.4.1 有障碍物的最大空隙问题(MAX $G(\Omega)$) …… 229
9.4.2 具有障碍物的欧几里德最短路径问题(ESPO) …… 230
9.4.3 具有障碍物的 Steiner 最小树问题(ESMTO) …… 231
第 10 章 随机几何算法与并行几何算法 …… 236
10.1 分类和搜索线性表的随机算法 …… 236
10.1.1 随机二叉树 …… 237
10.1.2 跳越表 …… 240
10.2 增量算法 …… 241
10.2.1 四边形分解 …… 242
10.2.2 凸多胞形 …… 245
10.2.3 Voronoi 图 …… 248
10.2.4 构形空间 …… 250

10.3 动态算法…… 253
10.4 随机抽样…… 257
10.4.1 具有限界的构形空间…… 257
10.4.2 顶-向下的抽样 …… 258
10.4.3 底-向上的抽样 …… 260
10.4.4 动态抽样…… 261
10.5 并行几何算法…… 264
10.5.1 凸壳问题…… 266
10.5.2 排列与分解…… 268
10.5.3 邻近…… 269
10.5.4 几何搜索…… 270
10.5.5 可视性和最优化…… 270
算法索引…… 272
参考文献…… 275

第0章 预备知识

本书叙述的内容不属于欧几里得的几何证明公理化范畴，而是属于欧几里得的几何构造，即由算法和复杂性分析所组成。欧几里得的几何构造满足算法的所有要求：无二义性、有穷性、确定性、能行性、输入、输出、正确性等。在欧几里得的几何构造中，限定了可允许使用的工具（直尺和圆规）及原始运算（圆规的一个脚置于一个给定点或一条直线上；作一个圆；直尺的边通过一个给定点；作一条直线）。但欧几里得原始运算并不能胜任所有的几何计算（比如角的三等分），这一点直到 19 世纪，阿贝尔、伽罗华等数学家才给出了证明。

在一个几何构造过程中，执行原始运算的总次数称为该过程的复杂性度量，这个概念对应于算法的时间复杂度。同样，还有对应于算法的空间复杂度的概念。这是欧几里得几何构造过程复杂性的定量测度。

称为计算几何的学科大致有下述几种：Forrest 等人依据样条函数处理曲线和曲面（实际上更接近于数值分析）；Minsky 和 Papert 写的一本名为《感知机》的书（副标题为“计算几何”），该书陈述用简单回路构成的网络实现模式识别的可能性（应属于人工神经网络）；计算机图形学是研究用计算机进行图形信息处理（包括表示、输入、输出、存储、显示、检索与变换等）和图形运算（如图的并、交运算）的一门学科，而不是算法分析；几何定理的机器证明，主要研究定理证明的探索方法及证明过程的推断，而不是几何本身。本书讨论的内容与上述计算几何学科所研究的内容不同，而是属于 Shamos 的文章（1975a）中命名的“计算几何”。为了不与上述“计算几何”的命名混淆，并突出 Shamos 的计算几何是研究几何问题的算法及复杂性的，故本书中将 Shamos 的计算几何称为 S 计算几何，而书名仍称为计算几何。

欧几里得货郎担问题、最小生成树问题、隐藏线（面）问题和线性规划问题等许多问题是S 计算几何研究的基本问题。在 19 世纪的文献中，已经出现了对这些问题的算法研究，但对几何问题进行几何算法的系统研究还是近 20 多年的事情。

本书将通过对几何问题的研究，在以下各章节中给出 S 计算几何的观点、研究方法与几种重要的几何结构。S 计算几何的一个基本观点是，经典的几何对象的表征常常不适合于有效算法的设计。因此有必要建立一些概念及相关的性质，以适应于有效算法的设计。

本章将介绍有关算法与数据结构的一些知识、几何知识及计算模型。这些内容是以后章节所需要的。

0.1 算法与数据结构

0.1.1 算法

众所周知，算法是求解一个问题类的无二义性的有穷过程。这里的过程是指求解问题

执行的一步一步动作的集合，每一步动作只需要有限的存储单元和有限的操作时间。另外，如果详细说明一台典型的计算机以及与这种计算机通信的语言，那么凡用这种语言编写的，可以在给定的计算机上执行的过程便称为算法。随机存取机器(RAM)、图灵机等可以作为典型的计算机，拟 ALGOL 语言作为描述算法而非执行的语言。应该指出，算法不等于程序，因此描述算法的方式将是多种形式的，比如在拟 ALGOL 语言的描述中可以使用数学记号和自然语言。为了把算法转换成上机程序，还需要进行编程工作。

算法的复杂性包括算法的时间复杂性和算法的空间复杂性。为了说明复杂性的概念，先介绍问题规模的概念。用一个与问题相关的整数量来衡量问题的大小，该整数量表示输入数据量的尺度，称为问题的规模。比如，行列式的规模可以用其阶数 n 作为它的规模；图问题的规模可以用其边数或顶点数作为它的规模等等。

利用某算法处理一个问题规模为 n 的输入所需要的时间，称为该算法的时间复杂性。它显然是 n 的函数，记为 $T(n)$。

类似地可以定义算法的空间复杂性 $S(n)$。

以下主要讨论算法的时间复杂性。由于一般不需要知道精确的时间耗费，只要知道时间耗费的增长率大体在什么范围内即可，因此我们引入算法复杂性阶的概念。

如果对某一正常数 c，一个算法在时间 cn^2 内能处理规模为 n 的输入，则称此算法的时间复杂性是 $O(n^2)$，读作“n^2 阶”。一般定义如下：

如果存在正常数 c 和 n_0，使得当 $n \geqslant n_0$ 时有 $T(n) \leqslant cf(n)$，则称 $T(n)$ 是 $O(f(n))$，记作 $T(n)=O(f(n))$。此时，$f(n)$ 是 $T(n)$ 的增长率的一个上界。

如果 $T(n)=O(f(n))$ 和 $f(n)=O(T(n))$ 同时成立，则称 $T(n)$ 是 $\theta(f(n))$，记作 $T(n)=\theta(f(n))$。此时，$T(n)$ 和 $f(n)$ 的增长率是同阶的。

如果存在正常数 c，使得对无穷多个 n 关系式 $T(n) \geqslant cf(n)$ 成立，则称 $T(n)$ 是 $\Omega(f(n))$，记作 $T(n)=\Omega(f(n))$。此时，$f(n)$ 是 $T(n)$ 的增长率的一个下界。

如果 $T(n)=O(f(n))$ 和 $T(n)=\Omega(f(n))$ 同时成立，则有 $T(n)=\theta(f(n))$。

根据上述定义，两个函数如果同阶，那么它们可以相差一个常数因子，还可以相差比阶低的项，即函数中的低阶项并不影响它的阶数。这时，如要进一步分析 $T(n)$，就应考虑常数因子和低阶项对 $T(n)$ 的影响。

一般情况下，都是取一个简单形式的函数作为阶数的规范表示，如 n^2, n^3, n^6 等。实用中也是这样处理的。

一个算法的时间复杂性如果是 $O(2^n)$，则称此算法需要指数时间。而时间复杂性如果是 $O(n^k)$(k 为有理数)，则称此算法需要多项式时间。当 n 很大时，指数时间和多项式时间存在很大的差别。以多项式时间为限界的算法称为有效算法。有效算法的一个本质特征是可以按常规的方法在较短的时间内用一个确定的计算装置进行机械的计算。该计算装置的典型模型是图灵机和 RAM(random access machine)机。

算法的时间复杂性分为最坏情况的时间复杂性和平均情况的时间复杂性。对于给定规模为 n 的各种输入，某算法执行每条指令所需要的时间之和的最大值，称为该算法的最坏情况的时间复杂性；对于给定规模为 n 的各种输入，执行每条指令所需要的时间之和的平均值，称为平均情况的时间复杂性(或期望复杂性或平均特性)。由于规模为 n 的“输

入”出现的概率不同，所以有时要考虑加权平均特性。

为了求平均特性，必须对输入量的分布作某种假设。然而切合实际情况的假设，在数学上往往不易处理。因此，确定平均特性比确定最坏情况的时间复杂性难。

在最坏情况的时间复杂性和平均特性的定义中，都提到“指令”。由于不同的机器执行某一指令(如加法指令)所需要的时间可能不同，因此同一算法在不同机器上执行时所需的时间也可能不同。为了消除这一差距，人们引入一个理想的计算模型，用它代替具体的机器，以建立时空复杂性概念。这个理想的计算模型称为随机存取机器(RAM)，它只有一个累加器，而且不允许自行修改指令。

RAM 机器由一条只读输入带，一条只写输出带，一个程序存储器，一个存储器和指令计数器组成，如图 0-1 所示。

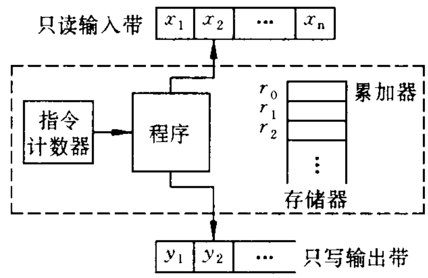

图 0-1　RAM 机器

输入带被划分为一系列方格，每个方格可以放一个整数。每当从输入带读出一个符号时，带头向右移动一格。输出带也被划分为一系列方格。执行一条写指令时，在输出带当前处于带头下的方格里打印一个整数，且带头右移一格，带头不能改写已印出的整数。

存储器由一系列寄存器 $r_0, r_1, r_a, \cdots$ 组成，每个寄存器能放下一个任意大小的整数。寄存器 r_i 的个数不受限定。

RAM 的程序不存放在存储器里，于是可假设程序不能修改自身。程序是带(也可不带)标号的指令序列，指令同实际的计算机一样，有算术指令、输入输出指令、间接寻址和转移指令等。用这些指令编写的程序称为 RAM 程序。RAM 的所有计算都在第一个寄存器 r_0(称为累加器)中进行。r_0 中也可放一个任意大小的整数。

RAM 程序的最坏情况时间复杂性和平均特性的定义与上面相同。

下面介绍最佳算法的概念。

在解决某一问题 P 的一类算法 $\mathscr{A}$ 中，需要操作次数最少的算法称为求解问题 P 算法类 $\mathscr{A}$ 的最佳算法。这里是用指定基本操作来确定算法类的。而算法 A 的基本操作是指 A 的关键操作，并且 A 的操作总数与基本操作次数成正比(当 n 增加时)。

形式上，设算法类 $\mathscr{A}$ 求解问题 P。对任一 $A \in \mathscr{A}$，其时间复杂性为 $T_A(n)$，定义

$$C_p(n) = \min_{A \in \mathscr{A}} \{T_A(n)\}$$

称为问题 P 关于 $\mathscr{A}$ 的计算复杂性。问题 P 的计算复杂性是 P 所固有的。由于 A 是 $\mathscr{A}$ 中任一算法，所以 $C_p(n)$ 难以估计。为此可以对问题 P 先求出在算法类 $\mathscr{A}$ 下 $C_p(n)$ 的下界(称为下界问题)，然后努力寻找达到这一计算复杂度的算法。

若能构造函数 $g(n)$，并可以证明，对任一 $A \in \mathscr{A}$，有 $T_A(n) \geqslant g(n)$ 成立，则 $g(n)$ 是 $C_p(n)$ 的一个下界。

设已找到求解 P 的算法 A，$A \in \mathscr{A}$，并分析 A 的复杂性 $T_A(n)$，有

$$C_p(n) \leqslant T_A(n) = f(n)$$

则 $f(n)$ 是 $C_p(n)$ 的一个上界。