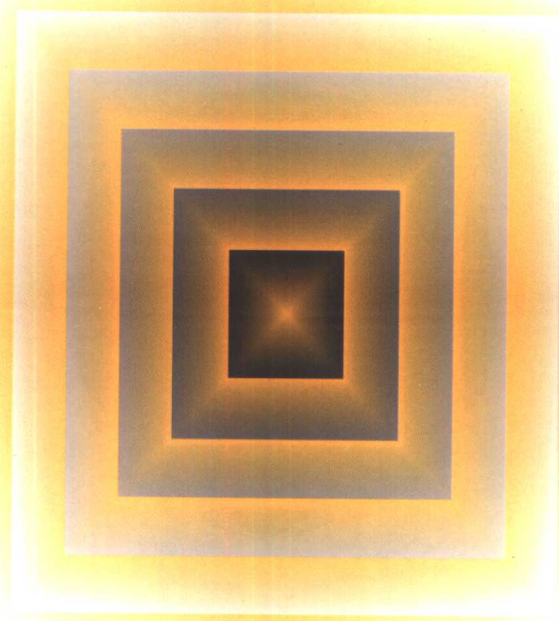


计算机数学

——计算复杂性理论与NPC、NP难问题的求解

西安交通大学数学研究生教学丛书

陈志平 徐宗本 编著



科学出版社

西安交通大学数学研究生教学丛书

计算机数学

——计算复杂性理论与 NPC、NP 难问题的求解

陈志平 徐宗本 编著

科学出版社

2001

内 容 简 介

本书全面、系统地介绍了计算复杂性理论的基本内容与各种 NPC 问题、NP 难问题等复杂问题的计算机求解方法. 前四章分别简要介绍了线性规划、多面体理论、网络规划与动态规划等预备知识. 第五至九章具体介绍了计算复杂性理论, 包括复杂性的定义与分类, 证明一个问题为 P 类或 NPC 类的基本方法, NPC 理论在分析、求解问题中的应用与近似算法的性能度量等. 第十至十六章则主要以整数规划为框架, 详细论述求解 NPC 及 NP 难问题各种不同形式的精确算法与近似算法.

本书可作为信息与计算科学、应用数学、计算机、管理科学等专业的研究生教材或本科生的选修课教材, 也可供有关的科研人员参考.

图书在版编目(CIP)数据

计算机数学——计算复杂性理论与 NPC、NP 难问题的求解/陈志平、徐宗本编著. —北京: 科学出版社, 2001. 8

(西安交通大学数学研究生教学丛书)

ISBN7-03-009151-5

I. 计… II. ①陈… ②徐… III. 电子计算机-数学-研究生-教材 IV. TP301.6

中国版本图书馆 CIP 数据核字(2001)第 03494 号

科学出版社 出版

北京朝阳城根北街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

*

2001年8月第 一 版 开本: B5(720×1000)

2001年8月第一次印刷 印张: 19

印数: 1—3 000 字数: 337 000

定价: 28.00 元

(如有印装质量问题, 我社负责调换〈环伟〉)

前 言

什么是计算机数学?依剑桥大学计算机系列丛书中《计算机数学》一书的作者 Cooke 和 Bez 的观点,早期国外所指计算机数学相当于我们现在所说的离散数学,它主要包括集合论、关系、代数结构、数据结构、基本的计算几何、形式语言和自动机等内容.因此,计算机数学是以离散量为研究对象的,它与计算机的软硬件结构密切相关,其讲授之目的在于帮助读者更深入地理解计算机的基本原理.然而,随着计算机科学与技术的蓬勃发展,特别是计算机在各个领域的广泛应用,计算机数学所研究的内容也在不断拓广与深入.人们发现由数理逻辑研究所产生的可计算性理论对于准确刻画计算机的计算能力非常有用,并由此导致了所谓的不可解性、难解性以及计算复杂性等理论.现在,谈到计算机数学较深层次的内容时,人们则主要是指形式语言与自动机、可计算性、不可解性与计算复杂性等理论.

实际应用中,当我们想利用计算机做任何一件事情(包括传统的各种数值计算与越来越多的非数值计算)时,首先会考虑的是所给问题能不能用计算机解决?若能解决,则总是希望在内存空间占用量尽可能少的同时,用尽可能短的时间来完成其求解任务.这一要求引出一系列相互联系而又非常实际的问题,如:怎样设计出满足要求的求解算法;如何分析、区别算法的好坏;可否改进现有的算法使其更有效;求解所给问题最好可能的算法会是什么,等等.要从一般角度回答第一个问题,即涉及可计算性理论,而回答其他问题则涉及算法复杂性理论.可计算性理论从建立某个可恰当描述计算机运作原理的计算模型出发,精确定义什么是计算、什么是可计算,再进一步回答某个或某类问题可否用计算机求解的问题.若所给问题可以求解,则称该问题是可计算的或可解的,否则称为不可解.算法复杂性理论试图在一般框架下分析各种不同类型的问题,通过考察可能存在的求解某个问题不同算法的复杂性程度来衡量求解该问题的难易程度,如是否很难求解或较易求解等.据此理论,我们可把应用中的问题分类,并进而利用计算复杂性理论对各种算法按其有效性进行分类.计算复杂性理论研究的是可用于估计、定界任一可行算法求解某类问题所需和仅需的时间工作量、空间占用量等计算资源的技术或方法,并同时指出求解特定问题的复杂度下界.

显然,从借助计算机来解决各类具体问题的角度看,人们最感兴趣,或许也是最重要的,当属掌握和运用可计算性与不可解性理论、计算复杂性理论以

及与其密切相关的、有助于理解这些理论的形式语言与自动机理论。而作为计算机数学经典内容的离散数学,则主要是有助于人们理解计算机的基本构成与运作原理。关于可计算性与不可解性理论、形式语言与自动机,除一些外文专著外,国内已有张立昂教授、张鸣华教授、沈泓教授等翻译或编写的很好的教材与专著,而关于离散数学、数据结构的书就更多了。然而,国内现有涉及计算复杂性理论的书,要么是简单概述该理论的一些基本概念与部分结果,要么是从递归函数、可计算函数或符号计算的角度予以阐述,难于理解与直接应用。有鉴于此,并考虑到篇幅的限制,本书将重点放在讲述计算复杂性理论的基本内容及其应用上。特别地,为使读者易于理解,我们将图灵(Turing)机作为基本的计算模型进行论述,这是因为图灵机不仅具有直观性,而且与实际的数字计算机之间存在许多相似。

本书力求用通俗的语言来全面、系统地介绍包括复杂性的定义、问题复杂性的分类、证明问题为P类或NPC类的方法、NPC理论在分析与求解问题中的应用、近似算法的性能度量等在内的计算复杂性理论的基本内容。不同于已有著作,我们特别以整数规划作为描述NPC问题、NP难问题的一般框架,阐述如何利用整数规划的理论及算法来设计求解NPC、NP难问题等不同类型的精确型算法与近似算法。精确型算法将主要介绍割平面类算法、分解算法与分枝定界算法等,而近似算法则主要介绍贪婪型算法、局部搜索算法、原始-对偶方法等。在现有介绍整数规划解法的书中,绝大多数仅是通过具体实例来说明如何用不同的方法去求解,层次较低,不利于对方法本身的深入理解及供学术研究参考。为此,本书对求解整数规划问题的每一类常用方法,从其基本思想、理论分析、具体实施到应用等都将做全面、深入的论述。

以问题、复杂性的一般分析及各种复杂问题有效求解算法的设计为主线,是本书在内容结构上的独特之处。而在写作上,我们力求作到条理清楚、易于理解。本书的另一特点是其内容的相对独立性,由于在前四章中已介绍了本书所需的线性规划、多面体理论、网络规划和动态规划等基础内容,阅读本书其他章节则无需别的专门知识,只需具备计算机基本知识即可。

本书可作为信息与计算科学、应用数学、计算机、管理科学与信息工程等专业的研究生教材或高年级本科生的选修课教材,也可供从事上述相关领域研究的科研人员、经常使用计算机求解各种复杂问题的工程技术人员、高级管理人员等参考。

本书写作中,作者们得到了他们的同事、研究生及家属的广泛支持与帮助。特别是作者陈志平的妻子和同事赵彩娥女士,她对全书进行了校对工作。不仅如此,作为本书的第一位读者,她还对本书的讲述风格、某些章节的编排等提出了许多宝贵的意见。没有她的帮助,本书不可能按时完成。本书曾在西

安交通大学理学院 1999 级和 2000 级研究生中试用,听课的研究生也对书中部分内容提出了修改意见,作者对他们也一并表示感谢。

本书的出版得到了科学出版社多位同志,特别是林鹏和杨波同志的大力支持与帮助。西安交通大学理学院万萍女士也提供了许多帮助。另外,作者的研究生郜峰为本书绘制了所有的插图。作者对他们也表示感谢。

由于作者水平有限,加之时间仓促,书中难免有不妥和错误之处,欢迎广大读者批评指正。

作 者

2000 年 10 月

于西安交通大学

目 录

第一章 线性规划	1
1.1 线性规划的基本概念	1
1.2 单纯形算法	4
1.3 字典序单纯形算法	7
1.4 对偶理论	10
1.5 内点算法	14
第二章 多面体理论	21
2.1 多面体的定义及其维数	21
2.2 用有效不等式与边界面来描述多面体	23
2.3 用极点和极射向表示多面体	26
第三章 图与网络规划	34
3.1 图的基本知识	34
3.1.1 图	34
3.1.2 有向图	36
3.1.3 图的表示	38
3.2 几类重要的图	41
3.3 最短路问题	42
3.4 最小权支撑树问题	44
3.5 最大流问题	45
第四章 动态规划方法	53
4.1 多阶段决策问题与动态规划的基本概念	53
4.2 动态规划方法的基本思想与最优性定理	55
4.3 最小权问题	59
4.4 背包问题	61
4.4.1 0-1 背包问题	61
4.4.2 整数背包问题	63
4.5 旅行商问题	65
第五章 算法复杂性概论	68
5.1 引言	68
5.2 基本概念	69

5.3	多项式时间算法与指数时间算法	71
第六章	问题复杂性的分类	75
6.1	判定问题与语言	75
6.2	算法的严格定义与 P 类问题	78
6.3	NP 类问题	80
6.4	多项式变换与 NP 完全问题	84
6.5	强 NP 完全问题	87
6.6	Co-NP 类问题	90
6.7	NP 困难问题	92
6.8	空间复杂性简介	95
第七章	证明问题为 NP 完全的或 P 的方法	97
7.1	证明问题为 NPC 的一般步骤	97
7.2	限制法(Restriction)	100
7.3	局部置换法(Local Replacement)	101
7.4	分量设计法(Component Design)	105
7.5	证明问题属于 P 类的方法	110
第八章	NP 完全理论在分析、求解新问题中的应用	114
8.1	分析新问题复杂性的双向研究方法	114
8.2	子问题分析法	116
8.3	求解 NPC 问题的算法类型	119
第九章	近似算法的性能度量与 NP 完全理论的应用	125
9.1	近似算法的性能度量	125
9.2	NP 完全理论在限定问题可近似程度中的应用	134
第十章	一般整数规划的基本性质	137
10.1	一般整数规划问题	137
10.2	整数规划与线性规划之间的关系	139
10.3	整数规划问题解的有界性	143
10.4	整数规划问题的计算复杂性	146
第十一章	割平面算法	150
11.1	分数割平面算法	150
11.2	整数割平面算法	154
11.3	导出有效不等式的方法	161
11.3.1	取整方法	161
11.3.2	同余方法	162
11.3.3	合并方法	162

11.3.4 超加函数法	163
11.4 混合整数规划问题的求解	167
11.5 覆盖问题的割平面算法	173
11.5.1 覆盖问题的描述	173
11.5.2 覆盖问题的割平面算法	175
第十二章 分解算法	179
12.1 拉格朗日松弛法	179
12.2 Benders 分解	185
12.3 一般分解方法	188
12.4 选址问题的分解算法	192
第十三章 分枝定界法	196
13.1 一般分枝定界法	196
13.2 使用线性规划松弛的分枝定界算法	199
13.2.1 剪枝准则	199
13.2.2 分枝方法	200
13.2.3 结节选取方法	202
13.2.4 分枝变量选择方法	203
13.3 0-1 背包问题的分枝定界算法	206
第十四章 匹配问题	210
14.1 匹配问题简介	210
14.2 最大匹配问题	212
14.2.1 二部图的匹配算法	214
14.2.2 非二部图的匹配算法	216
14.3 加权匹配问题	222
14.3.1 指派问题的求解	222
14.3.2 一般加权匹配问题	226
14.4 b 匹配问题与其他相关论题	232
14.4.1 b 匹配问题	232
14.4.2 匹配理论与算法的应用	236
第十五章 近似算法的设计与分类	240
15.1 近似算法概述	240
15.2 贪婪算法(Greedy Algorithms)	241
15.3 局部搜索法(Local Search Heuristics)	242
15.4 原始-对偶法	245
15.5 近似算法的其他设计方法	250

15.6 近似算法的分类	253
15.6.1 定常近似比算法	254
15.6.2 近似策略	255
15.6.3 最好可能近似比算法	258
15.6.4 比最好还要好的近似算法	260
15.6.5 与真正最优值仅一步之遥的近似算法	262
第十六章 对称旅行商问题	264
16.1 有效不等式的构造	264
16.2 松弛问题的构造	270
16.3 近似算法	273
16.3.1 最近邻法	274
16.3.2 最近插入法	274
16.3.3 贪婪可行法	275
16.3.4 k 边交换法	275
16.3.5 三角不等式与贪婪型算法的性能	275
16.3.6 支撑树加倍法	279
16.3.7 支撑树加完美匹配法	280
16.4 精确算法	282
16.4.1 指派问题加分枝定界算法	283
16.4.2 拉格朗日松弛加分枝定界算法	284
16.4.3 分数割平面加分枝定界算法	285
参考文献	289

第一章 线性规划

作为运筹学的一个重要分支,线性规划越来越广泛地被用于军事、工农业生产、各种商业活动以及政府管理部门等各个方面.今天,它已成为现代管理科学的重要基础与手段.关于线性规划理论、算法与应用的研究成果极为丰富,并不断有新的成果出现.本章仅对线性规划的几种典型求解算法与其对偶理论做一扼要介绍,目的是便于读者阅读和理解本书后面有关章节的内容.

1.1 线性规划的基本概念

线性规划是一种特殊的最优化问题.而最优化问题是研究某些具有数学结构的问题的最优解,通常表现为:对于给定的实际问题,研究如何从众多可行方案中选出某种意义下的最优方案.

一般地,可以用一二元集 (c, F) 来表示任一最优化问题,这里 c 表示目标函数,它是对不同解进行优劣比较的指标准则, $F \subseteq \mathbf{R}^n$ 表示可行解的集合,也常称为可行域或约束集.若用 $x \in \mathbf{R}^n$ 表示决策向量,则求解最优化问题等价于寻求某个 $x \in F$,使得 $c(x)$ 达到最小或最大.因此,最优化问题的一般形式可写成

$$\begin{aligned} \max \quad & c(x) \\ \text{s. t.} \quad & x \in F. \end{aligned}$$

若 $F = \mathbf{R}^n$,则称上述之问题为无约束最优化问题,并常简写为 $\max_{x \in \mathbf{R}^n} c(x)$.否则,就称其为约束最优化问题.约束最优化问题通常可表示为

$$\begin{aligned} \max \quad & c(x) \\ \text{s. t.} \quad & g_i(x) = 0, \quad i \in E, \\ & g_i(x) \leq 0, \quad i \in I, \\ & x \in X. \end{aligned}$$

这里 E 和 I 分别是等式约束、不等式约束的有限指标集, $g_i(x)$ 是约束函数. X 表示某种特殊的约束,通常是对变量的非负或上、下界要求.根据对决策变量、目标函数和约束函数的不同要求,可进一步将此一般的最优化问题划分为不同的类型.例如,当要求问题的全部或某些决策变量仅从某一离散集合,典型的情形是从整数的某一子集中取值时,就得到所谓的整数(混合整数)规划

问题. 其他常见的类型还有动态规划、网络规划、随机规划、多目标规划等. 而当目标函数和所有的约束函数均为 x 的线性函数时, 相应的最优化问题就是线性规划问题, 其一般形式为

$$\begin{aligned} \max \quad & z = c^T x \\ \text{s. t.} \quad & a_i^T x = (\leq, \geq) b_i, \quad i = 1, \dots, m, \\ & x_j \geq 0, \quad j \in J \subseteq N = \{1, 2, \dots, n\}. \end{aligned}$$

这里 m, n 为有限的正整数, $c, a_i (1 \leq i \leq m)$ 均为 n 维列向量, b_i 为常数. 因为 $\min z = -\max(-z)$, 对不等式约束 (\leq, \geq) 可以通过引进称为松弛变量的辅助变量来使其转化为等式约束. 若某个变量仅有有界约束, 如 $x_j \geq (\leq) l_j$, 则可令 $y_j = x_j - l_j (l_j - x_j)$, y_j 便成为非负变量; 而如果 x_j 为自由变量, 则可用 $x_j = u_j - v_j, u_j, v_j \geq 0$ 来代替 x_j . 所以, 对任一线性规划问题, 我们总可采用这些转换使之变为如下的标准形式:

$$\max \quad c^T x \quad (1.1)$$

$$\text{s. t.} \quad Ax = b, \quad (1.2)$$

$$x \geq 0, \quad (1.3)$$

其中 $c \in \mathbf{R}^n$ 称为价值系数向量, $A \in \mathbf{R}^{m \times n}$ 为约束系数矩阵, $m \leq n$ 为约束条件的个数, $b \in \mathbf{R}^m$ 称为约束右端向量. 记该线性规划问题的可行域为 $D = \{x \in \mathbf{R}^n \mid Ax = b, x \geq 0\}$. 若向量 $x \in D$, 则称 x 为一可行点或可行解.

对任一线性规划问题, 下面三种情况必占其一: 一是 $D = \emptyset$, 这时称该问题无解或者不可行; 二是 $D \neq \emptyset$ 但目标函数的值在 D 上无界, 这时称该问题无界; 三是 $D \neq \emptyset$ 且目标函数有有限值, 这时称该问题有最优解. 求解线性规划问题就是要判别它应属于哪一种情形, 而当问题有最优解时, 还要在 D 中找出使目标函数达到最大的点, 即问题的最优解, 以及相应的最优目标函数值. 为了刻画某一线性规划问题的具体性态, 进而求解它, 现在来引进一些用以描述问题可行解的概念. 以下不妨假设系数矩阵 A 是行满秩的, 即 $\text{rank}(A) = m$. 否则可通过对问题(1.1)~(1.3)作预处理以消去相关的行及对应的约束.

因假定 $\text{rank}(A) = m$, 故 A 中必有 m 个线性无关的列向量, 它们构成 $m \times m$ 的满秩方阵 B . 将 A 中其余各列组成的子阵记为 N . 于是可写 $A = (B, N)$. 把 $x = (x_1, \dots, x_n)^T$ 的分量也相应地划分为 x_B 和 x_N 两部分, 则 $Ax = b$ 可写作

$$Bx_B + Nx_N = b.$$

由于 B 是满秩方阵, B^{-1} 存在, 故由上述方程可求得

$$x_B = B^{-1}b - B^{-1}Nx_N.$$

现在把 x_N 看作一组自由变量, 给它任一组值 \bar{x}_N , 则可得到 x_B 相应的一组值

x_B , 于是 $x = \begin{pmatrix} \bar{x}_B \\ \bar{x}_N \end{pmatrix}$ 就是方程组 $Ax = b$ 的一个解. 若令 $\bar{x}_N = 0$, 则 $\bar{x}_B = B^{-1}b$.

我们把约束方程组这种特殊形式的解 $x = \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$ 称为基本解. 一般情形的具体定义如下.

设 B 是矩阵 A 中的一个 m 阶满秩方阵, 则称 B 为一个基; B 中 m 个线性无关的列向量称为基向量. 一般地, 通过选择不同的基向量可形成不同的基, 即 A 中存在不止一个基. x 中相应于基向量的分量称为基变量, 其他的变量称为非基变量. 令所有的非基变量取值为零, 所得解 $x = \begin{pmatrix} x_B \\ x_N \end{pmatrix} = \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$ 称为对应于 B 的基本解. 基本解满足 (1.2), 但不一定满足 (1.3), 所以未必是可行解. 但若基本解 $x \geq 0$, 则称 x 为基本可行解, 这时相应的基 B 称为可行基. 一个基本可行解 x , 如果它所有的基变量都取正值, 则说它是非退化的; 反之, 若有的基变量也取零值, 则说它是退化的. 一个可行基对应着一个基本可行解; 反之, 若一个基本可行解是非退化的, 那么它也对应着惟的一个可行基; 如果一个基本可行解是退化的, 则一般来说它可由不止一个的可行基得到. 一个线性规划问题, 如果它的所有基本可行解都是非退化的, 则称该问题是非退化的, 否则称其为退化的.

给定一个基本解, 我们很容易检验它是否为可行解, 从而是否为基本可行解. 那么, 给定一个可行解, 如何判别其是否为基本可行解呢? 其实, 由基本可行解的定义, 不难证明下列结论.

定理 1.1 可行解 x 是基本可行解当且仅当它的正分量所对应的列向量线性无关.

关于基本可行解的存在性及其重要性, 我们有下列关于线性规划的基本定理.

定理 1.2 若一个标准形式的线性规划问题有可行解, 则它必有基本可行解. 进而, 若该线性规划问题的目标函数有有限最优值, 则它必可在某个基本可行解处达到.

证明可参见张建中(1997).

由于基本可行解是由矩阵 A 中选择 m 个线性无关的列形成基矩阵后才得以确定, 而从 n 个列向量中选择 m 个列的可能性总共有 C_n^m 个, 因而基本可行解的数目是有限的, 至多有 C_n^m 个. 定理 1.2 后半部分的结论告诉我们: 要求解标准形式的线性规划问题, 只需在基本可行解的集合中进行搜索即可. 这样一来, 为求解线性规划问题, 我们似乎只要求出并比较所有有限多个的基

本可行解就行了. 其实这是不切实际的, 因为尽管基本可行解数目有限, 但当 n 较大时, 其个数也将很大. 所以, 如何按一定的规则在基本可行解的一个子集中进行搜索对于有效求解线性规划问题非常重要. 下一节介绍的单纯形算法即可达到这一目的.

1.2 单纯形算法

解标准形式线性规划问题的单纯形算法是由 G. B. Dantzig 在 1947 年提出的, 它是在所论线性规划问题的基本可行解中确定最优解的一种迭代法. 它产生一个使目标函数值单调上升的基本可行解序列. 因为可行域 D 有界的线性规划问题有有限多个基本可行解, 故若不出现退化解, 则单纯形算法必经有限步迭代后终止于问题的最优解.

单纯形算法的主要思想是: 先找一个基本可行解, 判别它是否为最优解, 如不是, 则找出一个更好的基本可行解, 再进行检验. 如此进行迭代, 直至找到最优解, 或者判定原问题无界. 因此, 这里主要有两个问题: 一是寻找初始的基本可行解; 二是如何判别当前基本可行解的最优性或怎样对其进行修正以建立迭代格式. 先考虑后一问题. 为此, 假设 $\text{rank}(A) = m < n$, 且假定已找到了一个非退化的基本可行解.

我们说找到了一个基本可行解, 其实是指找到了一个基 B , 并且已把方程组 (1.2) 式变换成等价的方程组

$$x_B + B^{-1}Nx_N = B^{-1}b. \quad (1.4)$$

为叙述方便, 不妨设 B 由 A 的前 m 列构成, 即 $B = (a_1, \dots, a_m)$, 相应地 $x_B = (x_1, \dots, x_m)$. 引进记号

$$\begin{aligned} \bar{a}_j &= B^{-1}a_j = (\bar{a}_{1j}, \dots, \bar{a}_{mj})^T, & j &= 1, \dots, n, \\ \bar{b} &= B^{-1}b = (\bar{b}_1, \dots, \bar{b}_m)^T, & \bar{N} &= B^{-1}N. \end{aligned}$$

则 (1.4) 式可以写成

$$x_B + \sum_{j=m+1}^n x_j \bar{a}_j = \bar{b},$$

或等价地写成

$$x_B + \bar{N}x_N = \bar{b}. \quad (1.5)$$

显然, 如果所取的基不同, 则对应的方程表达式也不同. 若 $\bar{b} \geq 0$, 则 (1.5) 就对应着基本可行解 $\bar{x} = \begin{pmatrix} \bar{b} \\ 0 \end{pmatrix}$. 而当 $\bar{b} > 0$ 时, 该解为非退化的基本可行解. 对 (1.2) 式进行了上述变换以后, 目标函数也要作相应的变换, 即将它用非基变量来表示. 相应于 x 的划分对 c 亦进行对应的划分后, 则有

$$\begin{aligned}
z &= c^T x = c_B^T x_B + c_N^T x_N \\
&= c_B^T (\bar{b} - \bar{N} x_N) + c_N^T x_N \\
&= c_B^T \bar{b} + \sum_{j=m+1}^n (c_j - c_B^T \bar{a}_j) x_j.
\end{aligned} \tag{1.6}$$

若用 z_0 表示基本可行解 \bar{x} 所对应的目标函数值, 则 $z_0 = c_B^T \bar{b}$, 即(1.6)中的常数项. 因 $\bar{a}_1, \dots, \bar{a}_m$ 均为单位向量, 故当 $j=1, \dots, m$ 时, $c_j - c_B^T \bar{a}_j = 0$. 若引入下面的记号

$$\sigma_j = c_j - c_B^T \bar{a}_j, \quad j = 1, \dots, n,$$

或用向量表示为

$$\sigma = (\sigma_1, \dots, \sigma_n) = c^T - c_B^T B^{-1} A = (\sigma_B, \sigma_N) = (0, c_N^T - c_B^T B^{-1} N),$$

则(1.6)式可记为

$$z = c_B^T \bar{b} + \sigma x = z_0 + \sigma x.$$

若 $\sigma \leq 0$, 则对原问题的任一可行解 x , 由 $x \geq 0$ 知 $\sigma x \leq 0$, 从而 $c^T x = z_0 + \sigma x \leq z_0$, 这表明 \bar{x} 必为原问题的最优解. 相反, 若向量 σ 有某个分量 $\sigma_p > 0$, 显然 $m+1 \leq p \leq n$, 则由上式或等价地(1.6)式知, 通过增大相应非基变量 x_p 的值, 保持其余非基变量不变, 并保证 $x \geq 0$, 则可以使目标函数值在可行域内进一步增大. 这意味着只要 σ 存在正的分量, 则当前的基本可行解 \bar{x} 就不是最优解.

当对应于基 B 的基本可行解为非最优解时, 单纯形法通过交换 B 与 N 中的一对列向量来形成一个新的基, 记为 B^+ , 使得相应于 B^+ 的基本解为基本可行解, 且使目标函数值要优于上一次迭代的函数值. 具体做法如下:

由于增大相应于正分量 σ_p 的非基变量 x_p 可使目标函数值增大, 且使该变量从非基变量变为基变量. 因此, 为使目标函数上升较快, 在 σ 有不止一个正分量时, 应取相应于

$$\sigma_p = \max\{\sigma_j \mid j = 1, \dots, n\}$$

的非基变量 x_p 作为入基变量, 而取相应的列向量 a_p 进入基 B . 为找出应从 B 中移出的离基向量 a_q , 考察 x_B 受非基变量 x_p 增值的影响, 因为

$$x_B = B^{-1} b - B^{-1} N x_N = \bar{b} - B^{-1} a_p x_p = \bar{b} - x_p \bar{a}_p.$$

如果 $\bar{a}_p \leq 0$, 则任意增加 x_p 的值都可保证所形成的向量 x 可行, 并同时使目标函数值不断增大, 这表明可行域 D 是无界的, 从而原问题无界. 否则, 如果 \bar{a}_p 中存在正的分量, 则增加 x_p 的值将使 x_B 中相应分量的值减小, 为确保所得基本解的可行性, x_p 的值不应超过

$$x_p = \frac{\bar{b}_q}{\bar{a}_{qp}} = \min_{\bar{a}_{ip} > 0} \frac{\bar{b}_i}{\bar{a}_{ip}}. \tag{1.7}$$

这时 x_B 中的第 q 个分量 $x_q = \bar{b}_q - x_p \bar{a}_{qp} = 0$. 其余分量仍保持大于 0, 即 x_q 从基变量变为非基变量, 相应的列向量 a_q 为离基向量. 换句话说, 用 N 中对应于 x_p 的列向量 a_p 交换 B 中对应于 x_q 的列向量 a_q 来得到 B^+ , 这一交换过程可表示为

$$B^+ = B + (a_p - a_q)e_q^T, \quad (1.8)$$

其中 e_q 为 \mathbf{R}^m 中的第 q 个单位坐标向量. 因为 $1 + e_q^T B^{-1}(a_p - a_q) = \bar{a}_{qp} > 0$, 所以由 Sherman-Morrison 公式知 B^+ 非奇异, 从而构成 A 的一个新基. 进而, 相应于该基的基本解为

$$\begin{aligned} x_i &= \bar{b}_i - \frac{\bar{b}_q \bar{a}_{ip}}{\bar{a}_{qp}}, \quad 1 \leq i \leq m, \quad x_p = \frac{\bar{b}_q}{\bar{a}_{qp}}, \\ x_i &= 0, \quad m+1 \leq i \leq n, \quad i \neq p. \end{aligned} \quad (1.9)$$

该解为一可行解, 即已找到对应于新基 B^+ 的基本可行解.

有了以上的论述, 则可以给出单纯形算法的迭代步骤. 鉴于 σ 在上述过程中所起的作用, 常称 σ 为检验向量, 其分量称为检验数.

· 单纯形算法

步 1 找到一初始的基本可行解.

步 2 计算 \bar{N}, \bar{b} 及检验向量 σ 等的值. 令 $\sigma_p = \max\{\sigma_j | j=1, \dots, n\}$.

步 3 若 $\sigma_p \leq 0$, 停止, 已找到线性规划问题的最优解 $x = \begin{pmatrix} \bar{b} \\ 0 \end{pmatrix}$ 及最优值 $c_B^T \bar{b}$.

步 4 若 $\bar{a}_p \leq 0$, 停止, 原问题无界.

步 5 由 (1.7) 式确定 q 的值, 交换列向量 a_p 与 a_q , 并分别由 (1.8) 式和 (1.9) 式确定新的基矩阵及对应的基本可行解. 转步 2.

要应用上述的单纯形算法, 还需要有一个确定初始基本可行解的方法. 已有多种方法可用于确定初始解, 这里只介绍一种称为两阶段法的典型方法. 因必要时可通过在约束方程的两边同乘以 -1 来使约束右端项变为正值, 故不失一般性, 可设 $b \geq 0$.

对标准的线性规划问题 (1.1) ~ (1.3) 引入 m 个人工变量, 并记其为 $r = (r_1, r_2, \dots, r_m)^T$, 考虑如下的辅助问题:

$$\begin{aligned} \max g &= - \sum_{i=1}^m r_i \\ \text{s. t. } Ax + r &= b, \\ x &\geq 0, r \geq 0. \end{aligned} \quad (1.10)$$

若记辅助问题的可行区域为 D' , 则显然 $x \in D$ 和 $\begin{pmatrix} x \\ 0 \end{pmatrix} \in D'$ 等价, 而存在某个

$\begin{pmatrix} x \\ 0 \end{pmatrix} \in D'$ 的充要条件是辅助问题的最优值为 0. 因此可通过解此辅助问题来获得原问题的初始解.

对于辅助问题(1.10), 由人工变量对应的 m 列构成了可行的单位基. 因 $b \geq 0$, 它对应于一基本可行解 $\begin{pmatrix} x \\ r \end{pmatrix} = \begin{pmatrix} 0 \\ b \end{pmatrix}$, 于是由此则可应用上述的单纯形算法进行迭代. 由于对人工变量的非负要求, 问题(1.10)的目标函数以零为上界, 从而必有最优解. 求得的最优解不外乎有两种可能:

一是 $\max g < 0$. 这表明不存在 x , 使得 $\begin{pmatrix} x \\ 0 \end{pmatrix} \in D'$, 亦即不存在 $x \in D$. 此时, 原问题(1.1)~(1.3)没有可行解, 计算可终止.

二是 $\max g = 0$. 这时自然有 $r = 0$, 即所有的人工变量均变为非基变量, 把它们除去后即可得到原问题(1.1)~(1.3)的基本可行解, 这一段的工作称为第一阶段. 基于所得基本可行解及相对应的可行基, 就可开始第二阶段了, 即采用上述的单纯形算法来正式开始求解原问题(1.1)~(1.3).

在前面的讨论中, 我们假定基本可行解是非退化的, 即 $\bar{b} > 0$. 若标准型线性规划问题的每一个基本可行解都是非退化的, 则用单纯形法计算时, 每一次迭代都使目标函数值严格下降, 因此出现过的解不可能再次出现. 由于基本可行解的个数有限, 单纯形算法的迭代过程一定在有限步内结束. 但是, 当出现退化解时, 单纯形算法的迭代有可能出现循环, 从而永远得不到最优解.

为了防止循环, 人们对上述单纯形算法中选取入基变量的方法进行了改进, 提出了一些新的选取规则, 如简便的 Bland 规则等. 另一个防止循环的典型方法是由 Dantzig 等人所提出的字典序方法, 并由此产生了所谓的字典序单纯形法. 因我们在后面介绍求解整数规划问题的割平面算法时要用到这种方法, 下面就对该法的做一较详细的介绍.

1.3 字典序单纯形算法

若引进一新的变量 x_0 来表示目标函数(1.1), 则标准型的线性规划问题(1.1)~(1.3)可简单地写为

$$\max \{x_0 \mid x_0 = c^T x, Ax = b, x \geq 0\}.$$

假设 A 的秩为 m , 则线性方程组 $Ax = b$ 的解集是一个 $n - m$ 维的线性流形, 其中有 $n - m$ 个变量可任意取值. 对 A 中的任意基 B , 记它的非基变量为参变量 $t_1, \dots, t_d, d = n - m$. 以这些非基变量为参数, 用高斯消去法从 $Ax = b$ 中解出基变量, 则可将标准型的线性规划问题化成如下的参数形式: