

《最优化基础——模型与方法》系列教材序言

最优化是人们在工程技术、科学研究和经济管理的诸多领域中经常遇到的问题。结构设计要在满足强度要求等条件下使所用材料的总重量最轻；资源分配要使各用户利用有限资源产生的总效益最大；安排运输方案要在满足物资需求和装载条件下使运输总费用最低；编制生产计划要按照产品工艺流程和顾客需求，尽量降低人力、设备、原材料等成本使总利润最高。可以预料，随着科学技术尤其是计算机技术的不断发展，以及数学理论与方法向各门学科和各个应用领域的更广泛、更深入的渗透，在即将到来的 21 世纪信息时代，最优化理论和技术必将在社会的诸多方面起着越来越大的作用。

解决实际生活中优化问题的手段大致有以下几种：一是靠经验的积累，凭主观作判断；二是做试验选方案，比优劣定决策；三是建立数学模型，求解最优策略。虽然由于建模时要作适当简化，可能使结果不一定非常完善，但是它基于客观数据，求解问题简便、灵活、经济，而且规模可以很大（将来会越来越大）。人们还可以吸收从经验得到的规则，用实验来不断校正建立的模型。随着数学方法和计算机技术的进步，用建模和数值模拟解决优化问题这一手段，将会越来越显示出它的效能和威力。显然，在决策定量化、科学化的呼声日益高涨的今天，数学建模方法的推广应用是符合时代潮流和形势发展需要的。

最优化理论、模型与方法所包含的内容很多，国内已出版了不少教材和专著介绍其各个分支。但是一方面，近年来发展起来的、有着广泛应用背景的规划模型（如随机规划、模糊规划等），以及一些已经为许多人采用、受到广泛关注的优化算法（如模拟

退火、遗传算法等), 还缺乏详细和系统的介绍; 另一方面, 一些偏重优化理论和方法的教材, 其要求难以与工科学生的数学知识衔接, 也缺少对于应用来说十分重要的建模过程和软件介绍, 而一些比较通俗的运筹学教材, 则在加强理论基础, 适应学生将来从事科研工作需要上考虑较少。我们这套教材试图弥补以上两方面的缺陷, 力求体现下述特点:

1. 内容既包含传统的线性规划与非线性规划等部分, 又纳入有广泛应用前景的随机规划和模糊规划; 在传统内容中, 既注重典型的数学思想和方法的系统叙述, 又引入丰富的建模实例。

2. 数学基础既与工科学生所学知识衔接, 又考虑到研究生阅读文献、从事科研工作的需要, 适当提高理论基础的起点。

3. 对一般教材介绍的诸多算法进行精选, 配合介绍一些应用软件, 并引入近年来迅速发展的若干新算法。

本系列教材将陆续出版, 首批四册为: 《线性与非线性规划》、《网络优化》、《现代优化计算方法》、《随机规划与模糊规划》。

由于水平所限, 书中难免有缺陷和错说, 诚恳希望读者予以批评指正。

《最优化基础 —— 模型与方法》系列教材编委会

1998 年 5 月

系列教材编委会成员名单

(姓氏笔划为序)

主 编: 姜启源 谭泽光

编 委: 刘宝碇 邢文训 陈宝林 林翠琴 胡冠章

黄红选 谢金星

序 言

在现实世界上，人们制定决策时经常会碰到两类不确定性现象：一类是随机现象，一类是模糊现象。描述、刻画随机现象的量称为随机变量，而描述、刻画模糊现象的量称为模糊集。为了方便，我们不妨把二者分别称为随机参数和模糊参数。含有随机和模糊参数的数学规划分别称为随机规划和模糊规划。既然随机性和模糊性都是用来处理不确定性的，我们将随机规划和模糊规划统称为不确定规划。本书将为随机规划和模糊规划提供统一的原理，并为一般不确定环境下的优化理论打下基础。

在很多实际问题中，如管理、工程、经济、工业以及生态等领域，系统是一个广泛使用的概念，而一个复杂的决策系统通常具有多维性、多样性、多功能性和多准则性，并带有随机或模糊参数。对于随机规划问题中所出现的随机变量，出于不同的管理目的和技术要求，采用的方法自然也不同。第一类处理随机规划中随机变量的方法是所谓的期望值模型，即一种在期望值约束下，使目标函数的概率期望达到最优的模型。第二类方法是由 Charnes 和 Cooper 提出的机会约束规划，主要针对约束条件中含有随机变量，且必须在观测到随机变量的实现之前作出决策的情况。考虑到所作决策在不利情况发生时可能不满足约束条件，而采用一种原则：即允许所作决策在一定程度上不满足约束条件，但该决策应使约束条件成立的概率不小于某一置信水平 α 。第三类随机规划是相关机会规划，它是使事件的机会函数在不确定环境下达到最优的方法。在确定性规划以及期望值模型和机会约束规划中，

当对实际问题建模以后,可行集本质上是确定的,这就可能导致所给出的最优解在实际中无法执行,而相关机会规划并不假定可行集是确定的。实际上相关机会规划的可行集被描述为所谓的不确定环境。虽然相关机会规划也给出一个确定的解,但这个解只是要求在实际问题中尽可能地执行。显然,相关机会规划的这一特点与确定性规划、期望值模型和机会约束规划是截然不同的。

沿用随机环境中机会约束规划的思想,在模糊环境中,假定模糊约束成立的可能性不小于置信水平 α , 这样就可以建立模糊机会约束规划、机会约束多目标规划和机会约束目标规划。类似地,沿用随机环境下相关机会规划的思想,亦有模糊相关机会规划、相关机会多目标规划和相关机会目标规划理论。

随着计算机的飞速发展和革新算法的不断涌现,许多复杂的优化问题已可以通过计算机求解。虽然目前计算机的能力还只能处理小规模的不确定规划模型,但是,我们坚信计算机的能力将会大幅度提高。这就为求解更加复杂的优化问题提供了一个契机,它不仅表现在已有的复杂模型可以通过计算机求解,而且表现在我们可以提出更丰富的建模思想。基于这一事实,本书采用全新的观点处理随机规划和模糊规划,并且允许不确定规划中的目标函数和约束函数是非线性的,随机参数的密度函数或模糊参数的隶属函数可以有更一般的形式,模型的结构可以更加复杂等等。

本书为求解传统方法所不能解决的随机规划和模糊规划模型,设计了一系列基于随机模拟或模糊模拟的遗传算法。虽然遗传算法有耗时多、速度慢等缺点,但对传统方法无法处理的问题,遗传算法是一种非常有效的方法,而且随着计算机速度的提高,实际问题将可以在合理的计算时间内得到解决。

本书共分 12 章。第 1 章主要介绍数学规划的基本概念,如线性规划、非线性规划、多目标规划、目标规划以及整数规划,同时也勾画出了随机规划和模糊规划的理论框架,第 2 章为求解优

化问题,如单目标规划、多目标规划和目标规划,提供了一个遗传算法,并通过一些数值例子解释了遗传算法的有效性。第3章列举了生成随机数的方法,并介绍模糊集合的一些基础理论,以及随机模拟和模糊模拟的技术。第4章给出了期望值模型的一些基本性质。第5章讨论了带有随机参数的机会约束规划。第6章给出一些机会约束规划模型的应用。第7章讨论了随机环境下的相关机会规划模型。第8章通过相关机会规划模型对随机决策系统进行了建模。第9章把随机机会约束规划推广到模糊机会约束规划。而第10章把随机相关机会规划推广到模糊相关机会约束规划。传统的数学规划模型提供的是使一些目标函数达到最优的清晰决策,然而,对实际问题,有时应该提供的是模糊决策而不是清晰决策,所以第11章建立了带有模糊决策的模糊规划的理论构架。在第9章和第11章所讨论的模糊系统中的机会约束规划模型本质上是一种 Maximax 模型(乐观模型),即极大化可能达到的最大收益。与 Maximax 模型的思想不同,第12章介绍了 Minimax 机会约束规划模型,其思想是极大化可能达到的最小收益。

本书可作为高等院校有关专业的高年级大学生和研究生的教材,也可作为运筹学、管理科学、计算机科学、系统科学、信息科学与工程等方面的学者和技术人员的参考书。

目 录

序言	ix
第 1 章 数学规划简介	1
1.1 线性规划	1
1.2 非线性规划	3
1.3 多目标规划	6
1.4 目标规划	8
1.5 整数规划	10
1.6 不确定规划	12
第 2 章 遗传算法	15
2.1 优化问题	16
2.2 表示结构	18
2.3 处理约束条件	18
2.4 初始化过程	19
2.5 评价函数	20
2.6 选择过程	22
2.7 交叉操作	23
2.8 变异操作	24
2.9 遗传算法程序	24
2.10 遗传算法与上升法	25
2.11 数值例子	26

第 3 章 随机模拟和模糊模拟	37
3.1 随机数的产生	38
3.2 随机模拟	47
3.3 模糊集合理论	50
3.4 模糊模拟	57
第 4 章 期望值模型	64
4.1 期望值算子	65
4.2 期望值模型	66
4.3 凸性	68
4.4 补偿模型	70
4.5 基于随机模拟的遗传算法	70
4.6 注	73
第 5 章 机会约束规划	74
5.1 机会约束规划模型	75
5.2 确定性等价类	79
5.3 一些性质	83
5.4 随机模拟	88
5.5 基于随机模拟的遗传算法	90
5.6 注	94
第 6 章 机会约束规划的应用	95
6.1 生产过程	95
6.2 饲料混合问题	97
6.3 随机资源分配	98
6.4 开放存储网络	101
6.5 资金预算	112

第 7 章 相关机会规划	117
7.1 背景: 供给 - 分配系统	117
7.2 随机集合	121
7.3 不确定环境	124
7.4 事件和机会函数	125
7.5 相关机会规划	128
7.6 相关机会多目标规划	130
7.7 相关机会目标规划	133
7.8 执行最优解	136
7.9 机会函数的随机模拟	137
7.10 基于随机模拟的遗传算法	138
7.11 注	143
第 8 章 随机决策系统建模	144
8.1 水资源供给 - 分配问题	144
8.2 生产过程	152
8.3 开放存储网络	154
8.4 资金预算	159
第 9 章 模糊机会约束规划	164
9.1 机会约束规划模型	165
9.2 清晰等价类	168
9.3 模糊模拟	173
9.4 基于模糊模拟的遗传算法	175
9.5 资金预算	179
9.6 注	183
第 10 章 模糊环境下的相关机会规划	184
10.1 相关机会规划	184

10.2	相关机会多目标规划	186
10.3	相关机会目标规划	188
10.4	机会函数的模糊模拟	191
10.5	基于模糊模拟的遗传算法	192
10.6	注	197
第 11 章	带有模糊决策的模糊规划	198
11.1	模糊决策	198
11.2	机会约束规划模型	200
11.3	相关机会规划模型	202
11.4	模糊模拟	208
11.5	基于模糊模拟的遗传算法	212
11.6	数值例子	216
11.7	注	222
第 12 章	Minimax 机会约束规划模型	223
12.1	Maximax 模型	223
12.2	Minimax 模型	227
12.3	Minimax 与 Maximax	229
12.4	模糊模拟	232
12.5	数值例子	234
12.6	注	238
参考文献		240
一些常用的符号		251
索引		252

第 1 章

数学规划简介

数学规划是运筹学的一个重要分支，并已被广泛地应用到很多领域。数学规划可以描述为在一些数学关系诸如等式或不等式表示的约束条件下，求一个（或一组）函数的极值问题的方法。常见的数学规划有线性规划、非线性规划、多目标规划、目标规划、整数规划、多层规划、动态规划以及本书重点讨论的随机规划和模糊规划等等。

本章里，介绍一些数学规划的基本概念和处理技术，为引入随机规划和模糊规划打下基础。

1.1 线 性 规 划

作为优化领域最重要的工具之一，线性规划是用来处理在线性等式及不等式组的约束条件下，求线性函数的极值问题的方法。线性规划的标准形式可以写为

$$\left\{ \begin{array}{l} \max \quad c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{s.t.} \\ \quad a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ \quad a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \quad \quad \quad \cdots \\ \quad a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n = b_m \\ \quad x_i \geq 0, \quad i = 1, 2, \cdots, n. \end{array} \right. \quad (1.1)$$

也可以表示成矩阵的形式

$$\begin{cases} \max & \mathbf{C}^T \mathbf{x} \\ \text{s.t.} & \\ & \mathbf{Ax} = \mathbf{B} \\ & \mathbf{x} \geq 0, \end{cases} \quad (1.2)$$

其中 $\mathbf{C} = (c_1, c_2, \dots, c_n)^T$, $\mathbf{A} = (a_{ij})_{m \times n}$ 和 $\mathbf{B} = (b_1, b_2, \dots, b_m)^T$ 为已知系数, 而 $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$ 为决策向量.

在线性规划的标准形式 (1.2) 中, 所有决策变量 $x_i, i = 1, 2, \dots, n$ 均假定为非负的. 在实际问题中这一假定通常是成立的. 如若不然, 我们亦可将其化成符合这一假定的等价的线性规划. 例如, 若变量 x_i 没有非负性假设, 我们可以用 $x'_i - x''_i$ 取代它, 其中 x'_i 和 x''_i 是两个新的变量, 且 $x'_i \geq 0$ 和 $x''_i \geq 0$. 这样, 线性规划问题被转化为具有非负变量的等价的线性规划.

在许多实际问题中, 一些约束含有不等式符号 \leq 或 \geq . 对于含有 \leq 的约束, 可在约束左端加上一个非负变量使其成为等式约束. 同样地, 对含有 \geq 的约束, 可在约束左端减去一个非负变量使其成为等式约束. 在约束中新增加的变量称为松弛变量, 而原来的变量称为结构变量.

一个解 \mathbf{x} 称为线性规划 (1.2) 的可行解, 如果 \mathbf{x} 满足 $\mathbf{Ax} = \mathbf{B}$, 且 $\mathbf{x} \geq 0$. 由所有可行解构成的集合称为可行集. 一个可行解 \mathbf{x}^* 称为线性规划 (1.2) 的最优解, 如果对所有的可行解 \mathbf{x} , 有 $\mathbf{C}^T \mathbf{x} \leq \mathbf{C}^T \mathbf{x}^*$ 成立.

欧式空间 \mathbb{R}^n 的子集 S 称为凸的, 如果 S 中任意两点的连线也在 S 中, 即 S 是凸的当且仅当对任意的 $\mathbf{x}_1, \mathbf{x}_2 \in S$ 和 $\lambda \in [0, 1]$, 有 $\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2 \in S$. 设 \mathbf{x}_1 和 \mathbf{x}_2 是 (1.2) 的任意两个可行解,

及 $0 \leq \lambda \leq 1$, 那么

$$\begin{cases} \mathbf{A}(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2) = \lambda \mathbf{A} \mathbf{x}_1 + (1 - \lambda) \mathbf{A} \mathbf{x}_2 = \mathbf{B} \\ \lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2 \geq \lambda \mathbf{0} + (1 - \lambda) \mathbf{0} = \mathbf{0}, \end{cases}$$

这意味着 $\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2$ 对于线性规划 (1.2) 是可行的. 因此, 线性规划的可行集永远是凸的.

一个点 \mathbf{x} 称为凸集 S 的极点, 如果 $\mathbf{x} \in S$, 且 \mathbf{x} 不能表示成 S 中其它任何两点的凸组合. 已经证明, 在可行集 S 有界的情况下, 线性规划 (1.2) 的最优解一定是可行集中的一个极点. 这一事实奠定了 Dantzig^[28] 所给出的一种求解线性规划非常有效且广泛使用的方法——单纯形法的理论基础. 简单地说, 单纯形法仅检查可行集的极点, 而不是所有的可行解. 首先, 单纯形法选择一个极点作为初始点, 然后, 再选择另一个极点, 以改善目标函数值. 重复以上过程, 直到目标函数值不能改进为止. 最后的一个极点就是最优解. 为了解决大规模的或特殊结构的线性规划问题, 一些学者相继提出了一些先进的技术, 如修正单纯形法、对偶单纯形法、原始对偶单纯形法、Wolfe-Dantzig 分解法以及 Karmarkar 内点算法. 详细内容, 读者可以查阅有关线性规划的书籍和论文.

1.2 非线性规划

非线性规划是用来处理在非线性等式及不等式组的约束条件下, 求非线性函数的极值问题的方法. 很多实际问题可以归结为非线性规划问题. 非线性规划的一般形式可以写为

$$\begin{cases} \min f(\mathbf{x}) \\ \text{s.t.} \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (1.3)$$

即在一组约束下, 极小化 (有时是极大化) 一个函数. 如果在非线性规划 (1.3) 中没有约束, 则称为无约束非线性规划. 如果函数 $f(\mathbf{x})$ 和 $g_j(\mathbf{x})$ ($j = 1, 2, \dots, p$) 均为凸函数, 则非线性规划 (1.3) 称为凸规划. 如果 $f(\mathbf{x})$ 可以表示为 $f(\mathbf{x}) = f_1(x_1) + f_2(x_2) + \dots + f_n(x_n)$, 则非线性规划 (1.3) 称为可分离规划. 如果函数 $f(\mathbf{x})$ 是二次的, 并且所有函数 $g_j(\mathbf{x}), j = 1, 2, \dots, p$ 都是线性的, 则非线性规划 (1.3) 称为二次规划. 如果函数 $f(\mathbf{x})$ 和 $g_j(\mathbf{x}) (j = 1, 2, \dots, p)$ 的形式是 $\sum_j a_j \prod_{i=1}^n x_i^{b_{ij}}$, 且对所有的序号 j , $a_j > 0$, 则称非线性规划 (1.3) 为几何规划.

在非线性规划 (1.3) 中, 向量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 称为决策向量, 其中 n 个变量 x_1, x_2, \dots, x_n 称为决策分量. 决策向量 \mathbf{x} 的函数 f 称为目标函数. 集合

$$S = \{\mathbf{x} \in \mathbb{R}^n \mid g_j(\mathbf{x}) \leq 0, j = 1, 2, \dots, p\} \quad (1.4)$$

称为可行集. 满足条件 $\mathbf{x} \in S$ 的解称为可行解. 非线性规划问题的目的在于找到一个解 $\mathbf{x}^* \in S$ 使得

$$f(\mathbf{x}^*) \leq f(\mathbf{x}), \quad \forall \mathbf{x} \in S, \quad (1.5)$$

如此的解 \mathbf{x}^* 称为最优解. 这种情况下的最优解也称为极小解. 最优解 \mathbf{x}^* 所对应的目标值 $f(\mathbf{x}^*)$ 称为最优值.

对于一个极大化问题

$$\begin{cases} \max f(\mathbf{x}) \\ \text{s.t.} \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (1.6)$$

可以通过把目标函数乘以 -1 , 变成在同样约束下的极小化问题, 而两个数学规划具有同样的最优解. 有时, 约束集合中不仅含有

不等式, 而且含有等式, 例如,

$$\begin{cases} g_j(\mathbf{x}) \leq 0, & j = 1, 2, \dots, p \\ h_k(\mathbf{x}) = 0, & k = 1, 2, \dots, q. \end{cases}$$

显然, q 个等式意味着我们可以删除非线性规划中的 q 个变量, 即可以用余下的变量表示这 q 个变量. 可以通过解方程组 $h_k(\mathbf{x}) = 0 (k = 1, 2, \dots, q)$ 做到这一点. 所以, 在本书中, 使用 (1.3) 作为单目标非线性规划的标准形式.

对一些结构特殊的非线性规划模型, 通过数学理论分析问题的结构, 已经建立了大量的经典方法. 在理论方面, Kuhn-Tucker 条件占有一席之地. 下面对这一理论做一简单介绍. 首先, 我们给出一些概念. 不等式约束 $g_j(\mathbf{x}) \leq 0$ 称为在点 \mathbf{x}^* 是起作用的约束, 如果 $g_j(\mathbf{x}^*) = 0$. 一个满足 $g_j(\mathbf{x}^*) \leq 0$ 的点 \mathbf{x}^* 称为规则的, 如果所有起作用的约束的梯度向量 $\nabla g_j(\mathbf{x})$ 是线性独立的. 假设 \mathbf{x}^* 是非线性规划 (1.3) 的规则点, 并假设所有函数 $f(\mathbf{x})$ 和 $g_j(\mathbf{x}) (j = 1, 2, \dots, p)$ 可导. 如果 \mathbf{x}^* 是局部最优解, 则存在 Lagrange 算子 $\lambda_j (j = 1, 2, \dots, p)$ 使下面的 Kuhn-Tucker 条件成立,

$$\begin{cases} \nabla f(\mathbf{x}^*) + \sum_{j=1}^p \lambda_j \nabla g_j(\mathbf{x}^*) = 0 \\ \lambda_j g_j(\mathbf{x}^*) = 0, & j = 1, 2, \dots, p \\ \lambda_j \geq 0, & j = 1, 2, \dots, p. \end{cases} \quad (1.7)$$

如果所有函数 $f(\mathbf{x})$ 和 $g_j(\mathbf{x}), j = 1, 2, \dots, p$ 均为凸的, 并且可导, \mathbf{x}^* 满足 Kuhn-Tucker 条件 (1.7), 则可以证明点 \mathbf{x}^* 是问题 (1.3) 的全局最小解.

现在考虑一个无约束问题, 即在区域 \mathbb{R}^n 上极小化一个实值函数. 在实际问题中, 当函数的一阶导数或二阶导数计算比较困难或根本无法计算时, 通常无法使用经典方法. 无论导数存在与

否,通常的方法是首先选择 \mathcal{R}^n 中的一个点,尽可能地将该点选择在最小值存在的地方. 如果没有任何这方面的信息,则可以随机地选取. 然后,由该点出发,通过分析该点处函数的性质,沿着使目标函数改善的方向,产生下一个点,接着重新分析函数的性质,这个过程一直延续到终止条件满足为止. 基于这种思路的求解方法统称为上升法. 根据目标函数 f 的性质方面的信息,可分为直接法、梯度法和 Hessian 法. 直接法仅要求函数在各点有意义. 梯度法要求函数 f 的一阶导数存在. Hessian 法则要求二阶导数存在. 众所周知,直接法是一种模式搜索方法,其目的是沿着有利的方向进行加速搜索, Rosenbrock 法、Powell 法、Brent 法、Stewart 法等均为此类搜索. 共轭方向法和共轭梯度法属于梯度法这一类. Hessian 方法包括牛顿法、Raphson 法和变尺度法. 这些方法都有自己的使用范围,并不是对所有的问题都有效,有效与否一般依赖于目标函数的性质.

对解决约束最优化问题,一般使用的方法有可行方向法、梯度投影法、罚函数法和线性近似法. 这类方法通常会收敛到局部最优解,而非全局最优解.

对优化问题,也涌现了一批革新算法,如神经网络、模拟退火法以及遗传算法等,而且这些方法受到了越来越多的重视. 在本书的第 2 章中,我们将详细介绍遗传算法,并贯穿本书始终.

1.3 多目标规划

非线性规划是在一组约束条件下,极大化(有时是极小化)一个实值目标函数. 然而,很多实际决策问题中,通常包含多个不相容的目标,而且要求同时考虑. 作为单目标规划的推广,多目标规划定义为在一组约束条件下,极大化(或极小化)多个不同的

目标函数. 其一般形式为

$$\begin{cases} \max [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (1.8)$$

其中 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 是一个 n 维决策向量, $f_i(\mathbf{x})$ 是目标函数, $i = 1, 2, \dots, m$, $g_j(\mathbf{x}) \leq 0$ 是系统约束, $j = 1, 2, \dots, p$.

当目标函数处于冲突状态时, 不存在最优解使所有目标函数同时最优化. 对这种情况, 我们使用有效解这一概念, 表示在不牺牲其他目标函数的前提下, 不可能再改进任何一个目标函数. 具体地说, 一个解 \mathbf{x}^* 称为有效解, 如果不存在 $\mathbf{x} \in S$ 使

$$f_i(\mathbf{x}) \geq f_i(\mathbf{x}^*), \quad i = 1, 2, \dots, m$$

且不等号至少对一个序号 j 成立. 众所周知, 在连续的情况下, 所有有效解构成的集合实际上是个有效前沿面. 一个有效解也称为非支配解、非劣解或 Pareto 解. 具体详细的解释, 感兴趣的读者可以查阅有关的书籍如文献 [128].

如果决策者把 m 个目标函数集成在一起, 构成一个实值偏好函数, 我们可以在相同的约束条件下, 极大化偏好函数. 这个模型称为妥协模型, 而其解称为妥协解. 第一个常见的妥协模型是通过对目标函数进行加权而建立起来的. 第二种方法是极小化 $(f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x}))$ 到一个理想的向量 $(f_1^*, f_2^*, \dots, f_m^*)$ 的距离函数, 其中 f_i^* 分别是第 i 个目标在不考虑其它目标时的最优值, $i = 1, 2, \dots, m$. 第三种是利用人机交互法去寻找妥协解. 目前已得到各种各样的人机对话法, 如可行区域削减法、加权向量空间削减法、准则锥削减法和线性搜索法.

1.4 目标规划

目标规划首先由 Charnes 和 Cooper^[21] 提出, 并由许多学者进行过研究. 目标规划可以看成是多目标优化问题的一个特殊的妥协模型, 目前已广泛应用到实际问题当中. 在多目标决策问题中, 假设决策者对每一个目标设计了一个目标值, 其思想是极小化各目标函数与目标值的偏差 (正偏差、负偏差或正负偏差). 在实际问题中, 一个目标通常只有在牺牲另一些目标的情况下, 才能实现, 而这些目标一般是不相容的. 因此, 在这些不相容的目标之间, 根据其重要性, 建立一个优先级是非常必要的, 并按照这个优先级为所有目标排序; 尽可能地实现更多的目标. 为了平衡多个冲突的目标, 根据决策者的目标值和优先结构, 一些实际的管理问题可以建模为目标规划模型. 目标规划的一般形式可以表示如下,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij}d_i^+ + v_{ij}d_i^-) \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m \end{array} \right. \quad (1.9)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差, 定义为

$$d_i^+ = \begin{cases} f_i(\mathbf{x}) - b_i, & f_i(\mathbf{x}) > b_i \\ 0, & f_i(\mathbf{x}) \leq b_i, \end{cases} \quad (1.10)$$

d_i^- = 目标 i 偏离目标值的负偏差, 定义为

$$d_i^- = \begin{cases} 0, & f_i(\mathbf{x}) \geq b_i \\ b_i - f_i(\mathbf{x}), & f_i(\mathbf{x}) < b_i, \end{cases} \quad (1.11)$$

\mathbf{x} = n 维决策向量,

f_i = 目标约束中的函数,

g_j = 系统约束中的函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

p = 系统约束个数.

有时, 目标规划模型 (1.9) 可以写成如下形式,

$$\left\{ \begin{array}{l} \text{lexmin} \left\{ \sum_{i=1}^m (u_{i1}d_i^+ + v_{i1}d_i^-), \dots, \sum_{i=1}^m (u_{il}d_i^+ + v_{il}d_i^-) \right\} \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{array} \right. \quad (1.12)$$

其中 lexmin 表示按字典序极小化目标向量.

解决线性目标规划的一个成功的方法是单纯形法. Saber 和 Ravindran^[117] 总结了求解非线性目标规划的方法, 这些方法的有效性各不相同. 可将它们分类如下:

- (a) 基于单纯形的方法, 如可分离规划技术、渐进规划方法^[59]、二次目标规划技术. 这些方法的主要思想在于把非线性目标规划转化为一组近似的线性目标规划, 以便使用目标规划单纯形法.
- (b) 直接搜索法, 如修正模式搜索算法^[56, 59]和梯度搜索算法^[26]. 在这种方法中, 把给定的非线性目标规划问题转化为一组单目标非线性规划问题, 然后, 使用解决单目标非线性规划时已经讨论过的直接搜索方法加以解决.
- (c) 基于梯度的方法^[81, 117], 该方法的思想是利用约束的梯度, 确认一个可行的方向, 然后, 以可行方向法为基础求解目标规划.
- (d) 人机交互法^[135, 103, 36], 在多次重复的人机交互过程中, 因为决策者参与了求解的过程, 因此可以得到满意解.
- (e) 遗传算法^[49, 60], 这种方法可以处理结构复杂的非线性目标规划模型, 但是所花费的 CPU 时间比较多.

1.5 整数规划

整数规划是一种特殊的数学规划, 其所含变量均假定取整数值. 如果数学规划中既含整数变量又含普通变量, 我们称这种数学规划为混合整数规划. 这样, 整数规划可分为如下几类:

- (a) (混合) 线性整数规划;
- (b) (混合) 非线性整数规划;
- (c) (混合) 多目标整数规划;
- (d) (混合) 目标整数规划

等等. 如果所有变量假定取值为 0 或 1, 则该数学规划称为 0-1 规划.

理论上, 解决整数规划可以使用枚举法, 但真正使用这种方

法求解实际问题却是不现实的。目前,求解整数线性规划比较有效的方法是由 Balas^[3] 和 Dakin^[27] 提出的分枝定界法。分枝定界法首先求解相应的线性规划,如果该问题证明无可行解,则原来的问题也无可行解,说明原来的问题是不合理的。如果找到整数解,则它一定是原来问题的最优解。如果所得的解是一个分数,则将其标记为节点 1(一个等待节点)。在等待节点,一定存在一个分量,记为 x , 取值分数。这个变量称为分枝变量。那么我们可以在目前的线性规划问题中加上 $x \leq [x]$ 或 $x \geq [x] + 1$, 其中 $[x]$ 表示 x 的整数部分,这样有两个分枝方向,两种选择可能性,从而产生两个新的问题。我们可以分别使用以上过程解决两个新的问题。当不存在更多的等待节点时,终止整个过程。在所有找到的整数解中,最好的作为最优整数解。如果没有找到整数解,说明原来问题无解。解决线性整数规划的另一种技术是割平面法,由 Gomory^[51] 提出。割平面法适用于一般的混合线性整数规划。使用这种方法求解线性整数规划的时候,先放宽可行域条件,把它当作一个线性规划来解。如果对应的线性规划的解是一个整数,则它也是原来线性整数规划的最优解。否则,在原问题中添加一些额外约束(即割平面),得到一个新的问题。对新的约束问题求解,得到一个新的解。重复以上过程直到找到整数解或证明问题不可行为止。

对于一个含有上十万个变量的线性规划,数字计算机可以在合理的时间内求得其最优解。但对整数规划,没有那么幸运。实际上,我们并没有一个很好的求解整数规划的方法。不同的算法通常适用不同类型的问题,尤其是一些特殊结构的问题。

最初的分枝定界枚举法不适于求解非线性整数规划,主要是分枝规则只有在线性假定的条件下才有效。为此, Dakin^[27] 提出了一个修改意见,使分枝规则不再依赖线性条件。另外, Taha^[129] 建议使用智能枚举法解决非线性整数规划问题。

1.6 不确定规划

在现实世界中,人们制定决策时经常会碰到不确定性现象。这种不确定性现象包括我们所熟悉的两类:一类是随机现象,一类是模糊现象。描述、刻画随机现象的量称为随机变量,描述、刻画模糊现象的量称为模糊集。为了方便,我们不妨把二者分别称为随机参数和模糊参数。含有随机(模糊)参数的数学规划称为随机(模糊)规划,二者统称为不确定规划。随机(模糊)规划为解决带有随机(模糊)参数的优化问题提供了有力的工具。有兴趣的读者可参阅文献 [121, 132, 64, 75, 31, 38, 65, 151, 153, 100]。

一个复杂的决策系统通常具有多维性、多样性、多功能性和多准则性,并带有随机参数或模糊参数。对于随机规划问题中所出现的随机变量,出于不同的管理目的和技术要求,采用的方法也不尽相同。首先,似乎最自然的方法是:取这些随机变量所对应的函数的概率平均值(数学期望),从而把随机规划转化为一个确定的数学规划(只是多了几个多重积分)。这种在期望值约束下,使目标函数的概率期望达到最优的模型通常称为期望值模型。

第二种方法是机会约束规划,由 Charnes 和 Cooper^[20] 提出,主要针对约束条件中含有随机变量,且必须在观测到随机变量的实现之前作出决策的问题。考虑到所作决策在不利的情况发生时可能不满足约束条件,而采用一种原则:即允许所作决策在一定程度上不满足约束条件,但该决策应使约束条件成立的概率不小于某一置信水平 α 。对一些特殊情况,机会约束规划问题可以转化为等价的确定性数学规划问题,但对较复杂的机会约束规划问题,通常很难做到这一点。文献 [60] 和 [147] 建议利用基于随机模拟的遗传算法来求解一般机会约束规划问题以及机会约束多目标规划和机会约束目标规划问题。

有时,一个复杂的决策系统要涉及到多项任务,称之为事件,

决策者往往希望这些事件实现的概率(即机会函数)尽可能的大. 为了解决这类问题, Liu^[85]提出了第三种随机规划, 即相关机会规划并推广到相关机会多目标规划和相关机会目标规划. 简单地说, 相关机会规划是使事件的机会函数在不确定环境下达到最大值的优化问题. 在确定性规划以及期望值模型和机会约束规划中, 当对实际问题建模以后, 可行集则实质上已经确定, 这就可能导致所给定的最优解在实际执行时根本无法实现, 相反, 相关机会规划并不假定可行集是确定的. 实际上相关机会规划的可行集描述为所谓的不确定环境. 虽然相关机会规划也给出一个确定的解, 但这个解只是要求在实际问题中尽可能地实现. 显然, 相关机会规划的这一特点是前面所提到的两种随机规划不具有的. 然而, 这类问题在现实生活中确实存在, 有关相关机会规划的实例和可能的应用可见文献 [82, 83, 86].

另外, 对于处理含有模糊参数的最优化问题, 模糊数学规划提供了一套有力的方法. 过去的一些方法可在文献 [153] 找到, Luhandjula^[98] 也对模糊规划作了详细的总结. 沿用随机规划中 Charnes 和 Cooper^[20] 所提出的机会约束规划的思想, 在模糊环境中, 假定模糊约束成立的可能性不小于置信水平 α , 这样可以得到模糊机会约束规划. Luhandjula^[96, 97] 和 Yazenin^[142] 等人考虑了模糊线性规划以及模糊多目标线性规划问题, 并根据 Zadeh^[145] 提出的可能性理论, 提出了一系列把机会约束转化为清晰等价类的思想. 但是, 这仍然只能解决一些特殊的模糊规划. 然而, 随着计算机的飞速发展和革新算法的不断涌现, 许多复杂的优化问题完全可以通过计算机得以解决, 出于此目的, Liu 和 Iwamura^[87, 88] 讨论了一般模糊机会约束规划以及机会约束多目标规划和机会约束目标规划, 并给出了基于模糊模拟技术的遗传算法. Liu^[90] 证明了上面提到的模糊机会约束规划实际上是一种 maximax 模型, 即极大化可能的最大目标函数, 为了模型思想的完整性, 还讨论

了模糊决策系统中的 minimax 机会约束规划模型. 沿用随机规划中相关机会约束规划的思想, Liu^[89] 给出了模糊相关机会规划、相关机会多目标规划和相关机会目标规划.

传统的数学规划模型提供清晰向量作为最优解使一些目标达到最优值. 但是, 如果从实际的需要出发, 有时应该提供模糊决策而不是清晰决策. Bouchon-Meunier 等人在文献 [15] 中总结了各种各样的极大化建立在模糊集上的数值函数的方法. Buckley 和 Hayashi^[16] 通过选择一个最优的模糊集去极大化一个实值函数, 为此设计了模糊遗传算法, 并应用到模糊优化问题、模糊极大流问题、模糊回归以及模糊控制等领域. 更一般地, Liu 和 Iwamura^[93] 提供了带有模糊决策的机会约束规划、机会约束多目标规划以及机会约束目标规划的理论提架. 另外, Liu^[94] 建立了带有模糊决策的相关机会规划、相关机会多目标规划以及相关机会目标规划等一系列模型, 并为求解这些模糊模型设计了基于模糊模拟的遗传算法.

在本书中, 从规划论的角度出发, 我们将重载随机与模糊的概念, 为随机规划和模糊规划提供统一的理论, 并称这一理论为不确定规划.

第 2 章

遗 传 算 法

遗传算法 (Genetic Algorithm) 是一种通过模拟自然进化过程搜索最优解的方法。过去 30 年中, 在解决复杂的全局优化问题方面, 遗传算法已取得了成功的应用, 并受到了人们广泛的关注。在优化问题中, 如果目标函数是多峰的, 或者搜索空间不规则, 就要求所使用的算法必须具有高度的鲁棒性, 以避免在局部最优解附近徘徊。遗传算法的优点恰好是擅长全局搜索。另外, 遗传算法本身并不要求对优化问题的性质作一些深入的数学分析, 从而对那些不太熟悉数学理论和算法的使用者来说, 无疑是方便的。

大家知道, 生物遗传物质的主要载体是染色体, 在遗传算法中, 染色体通常是一串儿数据 (或数组), 用来作为优化问题的解的代码, 其本身不一定是解。遗传算法一般经过这样几个过程: 首先, 随机产生一定数目的初始染色体, 这些随机产生的染色体组成一个种群。种群中染色体的数目称为种群的大小或种群规模。然后, 用评价函数来评价每一个染色体的优劣, 即染色体对环境的适应程度 (称为适应度), 用来作为以后遗传操作的依据。接着, 进行选择过程, 选择过程的目的是为了从当前种群中选出优良的染色体, 使它们成为新一代的染色体。判断染色体优良与否的准则就是各自的适应度, 即染色体的适应度越高, 其被选择的机会就越多。通过选择过程, 产生一个新的种群。对这个新的种群进行交叉操作, 交叉操作是遗传算法中主要的遗传操作之一。接着进行变异操作, 变异操作的目的是挖掘种群中个体的多样性, 克服有可能陷入局部解的弊病。这样, 经过上述运算产生的染色体

称为后代. 然后, 对新的种群 (即后代) 重复进行选择、交叉和变异操作, 经过给定次数的迭代处理以后, 把最好的染色体作为优化问题的最优解.

有关遗传算法 (包括进化规划, 进化算法和进化策略) 的更进一步知识, 读者可参看文献 [50] 和 [104]. 在解决全局优化问题方面, 遗传算法显示了非常广泛的应用前景, 并已应用到最优控制、运输问题、旅行商问题、调度、生产计划、资源分配、统计及模式识别等.

遗传算法已取得了理论上和应用上的巨大成功. 在本章里, 将介绍遗传算法的基本知识, 为学习本书的后续部分打好基础.

2.1 优 化 问 题

首先, 回顾一下单目标规划、多目标规划以及目标规划模型的基本形式.

单目标数学规划的一般形式可以写为

$$\begin{cases} \max f(\mathbf{x}) \\ \text{s.t.} \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (2.1)$$

即在一组约束条件 $g_j(\mathbf{x}) \leq 0$, $j = 1, 2, \dots, p$ 下, 使目标函数 $f(\mathbf{x})$ 达到极大 (有时也可能是极小). 其中 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 是一个 n 维决策向量, $f(\mathbf{x})$ 和 $g_j(\mathbf{x}) (j = 1, 2, \dots, p)$ 是普通的实值函数.

在同样的一组约束条件下, 使 m 个不同的目标函数达到最优的问题可用多目标规划描述, 其一般形式为

$$\begin{cases} \max [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ g_j(\mathbf{x}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (2.2)$$

其中 $f_i(\mathbf{x})$ ($i = 1, 2, \dots, m$) 和 $g_j(\mathbf{x})$ ($j = 1, 2, \dots, p$) 是决策向量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 的普通的实值函数.

根据决策者给定的目标值和优先结构, 非线性目标规划的一般形式可以写为

$$\left\{ \begin{array}{ll} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij}d_i^+ + v_{ij}d_i^-) & \\ \text{s.t.} & \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, & i = 1, 2, \dots, m \\ g_j(\mathbf{x}) \leq 0, & j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, & i = 1, 2, \dots, m, \end{array} \right. \quad (2.3)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

$\mathbf{x} = n$ 维决策向量,

f_i = 目标约束中的函数,

g_j = 系统约束中的函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

p = 系统约束个数.

在上面提到的三种数学规划模型中, 目标函数不一定是单峰的, 可行集也不一定是凸的. 另外, 值得一提的是, 在上述模型中, 认为不存在等式约束, 正如在后面讨论中提到的那样, 等式

约束是可以消除的.

2.2 表示结构

最优化问题的解有两种表示方法: 二进制向量或浮点向量. 使用二进制向量作为一个染色体来表示决策变量的真实值, 向量的长度依赖于要求的精度, 但使用二进制代码的必要性已经受到了一些批评. 在求解复杂优化问题时, 二进制向量表示结构有时不太方便.

另一种表示方法是用浮点向量, 每一个染色体由一个浮点向量表示, 其长度与解向量相同. 这里用向量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 表示最优化问题的解, 其中 n 是维数. 则相应的染色体也是 $\mathbf{V} = (x_1, x_2, \dots, x_n)$.

2.3 处理约束条件

处理约束条件的关键在于 (a) 删除约束条件中的所有等式, (b) 设计恰当的遗传操作以保证所有新产生的染色体在可行集中.

在数学规划模型中, 如果存在一些等式约束, 例如, $h_k(\mathbf{x}) = 0$, $k = 1, 2, \dots, q$, 可以通过解这些约束构成的方程组, 用其它的变量替换其中的 q 个变量, 以消除 q 个等式约束. 例如, 不失一般性, 其前 q 个变量通过解方程组得到 $x_k = \hat{h}_k(x_{q+1}, x_{q+2}, \dots, x_n)$, $k = 1, 2, \dots, q$, 则可以将最优化模型中的变量 x_1, x_2, \dots, x_q 由其它变量表示, 于是原问题化成一个 $(n - q)$ 维的只有不等式约束的新问题.

为了保证染色体是可行的, 必须对遗传操作过程中得到的每一个染色体进行检查, 具体的方法将在后面的章节作详细的介绍. 对每个最优化问题, 最好设计一个 C 语言子函数, 其输出值 1 表示染色体是可行的, 0 表示不可行. 例如, 对约束条件 $g_j(\mathbf{x}) \leq 0$,

$j = 1, 2, \dots, p$, 可以作如下的一个子函数:

从 $j = 1$ 开始循环

若 $(g_j(x) > 0)$, 返回 0 ;

直到 $j = p$ 结束

返回 1.

最后, 值得注意的是, 在设计程序时, 应当注意到一些隐含约束, 即有些点虽然是可行解, 但不可能是最优解. 例如, 数学规划模型

$$\max_{x \in \mathbb{R}} e^{-x^2}$$

的可行集是全体实数 \mathbb{R} . 但是, 通过数学分析, 可以确认, 最优解一定在区间 $[-5, +5]$ 上. 所以, 为减少程序的搜索空间, 应当增加约束

$$-5 \leq x \leq 5.$$

这一类隐含约束一般不难发现, 尤其对实际的管理问题. 因此, 尽可能地增加隐含条件以减少搜索空间, 可以大大加快问题的进化过程.

2.4 初始化过程

定义整数 *pop_size* 作为染色体的个数, 并且随机产生 *pop_size* 个初始染色体. 一般情况下, 由于优化问题的复杂性, 解析地产生可行的染色体是困难的. 此时, 可以采用下述两种方法之一作为初始化过程. 具体实施时依赖于决策者所提供的信息.

第一种方法, 设决策者能够给出可行集中的一个内点, 记为 V_0 . 定义一个足够大的数 M , 以保证遗传操作遍及整个可行集. 此大数 M 不仅在初始化过程中使用而且在变异操作中使用. 按照下面的方法产生 *pop_size* 个染色体. 在 \mathbb{R}^n 中, 随机选择一个方向 d , 如果 $V_0 + M \cdot d$ 满足不等式约束, 则将 $V = V_0 + M \cdot d$

作为一个染色体, 否则, 置 M 为 0 和 M 之间的一个随机数, 直到 $V_0 + M \cdot d$ 可行为止. 由于 V_0 是内点, 所以在有限步内, 可以找到满足不等式约束的可行解. 重复以上过程 pop_size 次, 从而产生 pop_size 个初始染色体 $V_1, V_2, \dots, V_{pop_size}$.

如果决策者不能给出这样的内点, 但可以确定一个包含最优解 (不一定是整个可行集) 的区域. 显然, 任何情况下, 只要不介意所给区域过大, 决策者总能提供这样的区域. 一般情况下, 把该区域设计成一个易于计算的形状, 如可以设计成一个 n 维超立方体. 从这个超立方体中产生一个随机点, 并检验其可行性. 如果可行, 则作为一个染色体. 否则, 从超立方体中重新产生随机点, 直到得到可行解为止. 重复以上过程 pop_size 次, 则得到 pop_size 个初始可行的染色体 $V_1, V_2, \dots, V_{pop_size}$.

2.5 评 价 函 数

评价函数 (用 $eval(V)$ 表示) 用来对种群中的每个染色体 V 设定一个概率, 以使该染色体被选择的可能性与其种群中其它染色体的适应性成比例, 即通过轮盘赌, 适应性强的染色体被选择产生后代的机会要大.

第一种方法, 设目前该代中的染色体为 $V_1, V_2, \dots, V_{pop_size}$, 可以根据染色体的序进行再生分配, 而不是根据其实际的目标值. 无论何种数学规划 (单目标、多目标或目标规划) 都可以作一合理假设, 即在染色体 $V_1, V_2, \dots, V_{pop_size}$ 中, 决策者可以给出一个序的关系, 使染色体由好到坏进行重排, 就是说, 一个染色体越好, 其序号越小. 后面将通过一些例子来说明这一点. 设参数 $\alpha \in (0, 1)$ 给定, 定义基于序的评价函数为

$$eval(V_i) = \alpha(1 - \alpha)^{i-1}, \quad i = 1, 2, \dots, pop_size. \quad (2.4)$$

$i = 1$ 意味着染色体是最好的, $i = pop_size$ 说明是最差的.

第二种方法是通过对适应度的适当缩放调整 (称为适应度定标) 来设计评价函数. 用 $f_1, f_2, \dots, f_{pop_size}$ (即染色体 $V_1, V_2, \dots, V_{pop_size}$ 各自的目标值) 表示原来的适应度. Goldberg^[50] 提出一种线性适应度定标方案,

$$f'_i = af_i + b, \quad i = 1, 2, \dots, pop_size, \quad (2.5)$$

其中 $f'_i, i = 1, 2, \dots, pop_size$ 为新的适应度, a 和 b 为参数. 这种方法实际上假定使用者了解目标函数的性质, 从而才能设计合理的参数 a 和 b . 这种情况下, 评价函数定义为

$$eval(V_i) = f'_i / \sum_{j=1}^{pop_size} f'_j, \quad i = 1, 2, \dots, pop_size. \quad (2.6)$$

Gen, Liu 和 Ida^[47] 构造了另外一种基于指数适应度的评价函数, 其介于基于序的评价函数和线性适应度定标方案之间. 首先, 定义三个优先参数 p_1, p_0 和 p_2 ($0 < p_1 < p_0 < p_2 < 1$), 以确定三个临界数 u_1, u_0 和 u_2 (取自于 pop_size 个染色体目标值构成的集合), 使在目前的 pop_size 染色体构成的集合中分别有 $(p_1 \cdot pop_size), (p_0 \cdot pop_size)$ 和 $(p_2 \cdot pop_size)$ 个染色体的目标值分别小于 u_1, u_0 和 u_2 .

对极大化问题, 原来的适应度 u_1 映射到 $e^{-1} \approx 0.37$, u_0 映射到 1, u_2 映射到 $2 - e^{-1} \approx 1.63$. 则原来的适应度 u 和指数适应度 u' 之间的关系为

$$u' = \begin{cases} \exp \left[-\frac{u - u_0}{u_1 - u_0} \right], & u < u_0 \\ 2 - \exp \left[-\frac{u - u_0}{u_2 - u_0} \right], & u \geq u_0. \end{cases} \quad (2.7)$$

此函数表示在图 2.1 中.

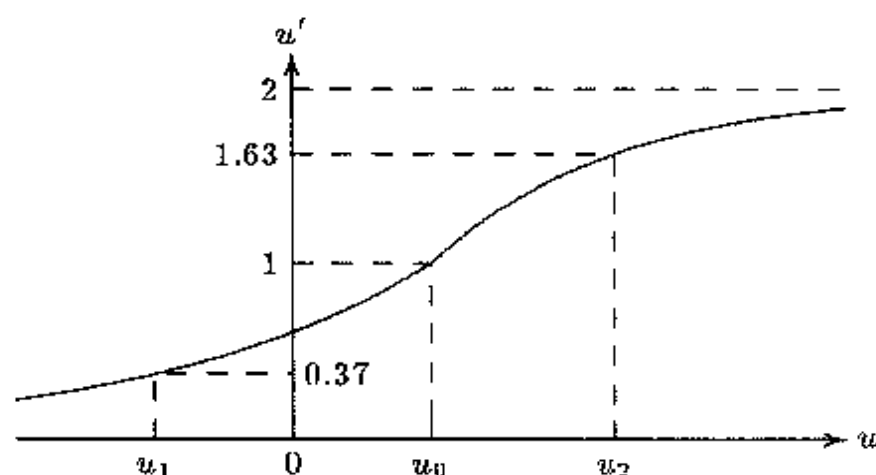


图 2.1 极大化问题的指数适应度

此时, 定义如下的评价函数,

$$eval(\mathbf{V}_i) = u'_i / \sum_{j=1}^{pop_size} u'_j, \quad i = 1, 2, \dots, pop_size, \quad (2.8)$$

其中 u'_i 分别是染色体 \mathbf{V}_i , $i = 1, 2, \dots, pop_size$ 各自的指数适应度.

2.6 选 择 过 程

选择过程是以旋转赌轮 pop_size 次为基础的. 每次旋转都为新的种群选择一个染色体. 赌轮是按每个染色体的适应度进行选择染色体的. 无论使用哪一种评价函数, 选择过程总可以写成如下形式

步骤 1 对每个染色体 \mathbf{V}_i , 计算累积概率 q_i

$$\begin{cases} q_0 = 0, \\ q_i = \sum_{j=1}^i eval(\mathbf{V}_j). \quad i = 1, 2, \dots, pop_size; \end{cases} \quad (2.9)$$

步骤 2 从区间 $(0, q_{pop_size}]$ 中产生一个随机数 r ;

步骤 3 若 $q_{i-1} < r \leq q_i$, 则选择第 i 个染色体 V_i ($1 \leq i \leq pop_size$);

步骤 4 重复步骤 2 和步骤 3 共 pop_size 次, 这样可以得到 pop_size 个复制的染色体.

在上述过程中, 并没有要求满足条件 $q_{pop_size} = 1$. 实际上, 可以用 q_{pop_size} 除以所有的 q_i , $i = 1, 2, \dots, pop_size$, 使 $q_{pop_size} = 1$, 新得到概率同样与适应度成比例. 只要我们不介意概率方面解释上的困难, 这一点并没有在遗传过程中产生任何影响.

2.7 交叉操作

首先定义参数 P_c 作为交叉操作的概率, 这个概率说明种群中有期望值为 $P_c \cdot pop_size$ 个染色体来进行交叉操作.

为确定交叉操作的父代, 从 $i = 1$ 到 pop_size 重复以下过程: 从 $[0, 1]$ 中产生随机数 r , 如果 $r < P_c$, 则选择 V_i 作为一个父代.

用 V'_1, V'_2, V'_3, \dots 表示上面选择的父代, 并把它们随机分成下面的对

$$(V'_1, V'_2), (V'_3, V'_4), (V'_5, V'_6), \dots$$

我们以 (V'_1, V'_2) 为例解释怎样对上面所有的对进行交叉操作. 首先, 从开区间 $(0, 1)$ 中产生一个随机数 c , 然后, 按下列形式在 V'_1 和 V'_2 之间进行交叉操作, 并产生两个后代 X 和 Y

$$X = c \cdot V'_1 + (1 - c) \cdot V'_2, \quad Y = (1 - c) \cdot V'_1 + c \cdot V'_2. \quad (2.10)$$

如果可行集是凸的, 这种凸组合交叉运算在两个父代可行的情况下, 能够保证两个后代也是可行的. 但是, 在许多情况下, 可行集

不一定是凸的, 或很难验证其凸性, 此时必须检验每一后代的可行性. 如果两个后代均可行, 则用它们代替其父代, 否则, 保留其中可行的 (如果存在的话), 然后, 产生新的随机数 c , 重新进行交叉操作, 直到得到两个可行的后代或循环给定次数为止, 无论如何, 仅用可行的后代取代其父代.

2.8 变异操作

定义参数 P_m 作为遗传系统中的变异概率, 这个概率表明, 总体中有期望值为 $P_m \cdot \text{pop_size}$ 个染色体用来进行变异操作.

类似于交叉操作中选择父代的过程, 由 $i = 1$ 到 pop_size , 重复下列过程: 从区间 $[0, 1]$ 中产生随机数 r , 如果 $r < P_m$, 则选择染色体 V_i 作为变异的父代. 对每一个选择的父代, 用 $V = (x_1, x_2, \dots, x_n)$ 表示, 按下列方法进行变异. 在 \mathbb{R}^n 中随机选择变异方向 d , 如果 $V + M \cdot d$ 是不可行的, 那么, 置 M 为 0 和 M 之间的随机数, 直到其可行为止. 其中 M 是初始化过程定义的一个足够大的数. 如果在预先给定的迭代次数之内没有找到可行解, 则置 $M = 0$. 无论 M 为何值, 总用

$$X = V + M \cdot d \quad (2.11)$$

代替 V .

2.9 遗传算法程序

经过选择、交叉和变异操作, 我们得到一个新的种群, 准备进行下一代进化. 对上述步骤经过给定的循环次数之后, 遗传算法终止. 对一般优化问题, 遗传算法可以归纳如下:

遗传算法程序

输入参数 $\text{pop_size}, P_c, P_m$;

通过初始化过程产生 *pop_size* 个染色体;

重复

对染色体进行交叉和变异操作;

计算所有染色体的评价函数;

根据某种抽样机制选择染色体;

直到满足终止条件.

我们知道, 最好的染色体不一定出现在最后一代中, 所以在进化开始, 必须把最好的染色体保留下来, 记为 V_0 , 如果在新的种群中又发现了更好的染色体, 则用它代替原来染色体 V_0 . 在进化完成之后, 这个染色体就可以看做是优化问题的解.

2.10 遗传算法与上升法

上升法是直接法、梯度法和 Hessian 法的通称. 上升法首先在最优解可能存在的地点选择一个初始点, 然后通过分析目标函数的特性, 由初始点移到一个新的点, 然后再继续这个过程. 为了更好地理解遗传算法, 现在把上升法和遗传算法作一比较.

上升法的搜索过程是确定的, 通过产生一系列的点收敛到最优解 (有时是局部最优解), 而遗传算法的搜索过程是随机的, 它产生一系列随机种群序列. 二者的主要差异可以归纳如下两点:

- (a) 上升法的初始点仅有一个, 由决策者给出; 遗传算法的初始点有多个, 随机产生.
- (b) 通过分析目标函数的特性, 上升法由上一点产生一个新的点; 遗传算法通过遗传操作, 在当前的种群中经过交叉、变异和选择产生下一代种群.

对同一优化问题, 遗传算法所使用的机时比上升法所花费的机时要多, 但遗传算法可以处理一些上升法不能解决的复杂的优化问题.

2.11 数值例子

遗传算法已编写成 C 语言程序. 在后面给出的数值例子中, 所使用的参数为: 种群规模为 30, 交叉概率为 0.2, 变异概率为 0.5, 而评价函数中的参数 α 为 0.05. 遗传算法对于这些参数的设置是非常鲁棒的, 改变这些参数对所得的结果不会有太大的影响.

例 1 单目标规划. 考虑非凸集合上的优化问题

$$\left\{ \begin{array}{l} \max f(\mathbf{x}) = \frac{x_1^2 x_2 x_3^2}{2x_1^3 x_3^2 + 3x_1^2 x_2^2 + 2x_2^2 x_3^3 + x_1^3 x_2^2 x_3^2} \\ \text{s.t.} \\ x_1^2 + x_2^2 + x_3^2 \geq 1 \\ x_1^2 + x_2^2 + x_3^2 \leq 4 \\ x_1, x_2, x_3 > 0. \end{array} \right. \quad (2.12)$$

已经知道目标函数的最大值为 $f(\mathbf{x}) = 0.1537$, 图 2.2 中的阴影部分表示可行集在 $x_3 = 0$ 处的横截面. 从图中可以看到, 可行集是非凸的.

下面用遗传算法求解此问题. 用染色体 $\mathbf{V} = (x_1, x_2, x_3)$ 来作为解的代码. 染色体 $\mathbf{V} = (x_1, x_2, x_3)$ 的可行性由下面的检验函数检验

如果 $(x_1 \leq 0 \parallel x_2 \leq 0 \parallel x_3 \leq 0)$, 返回 0;

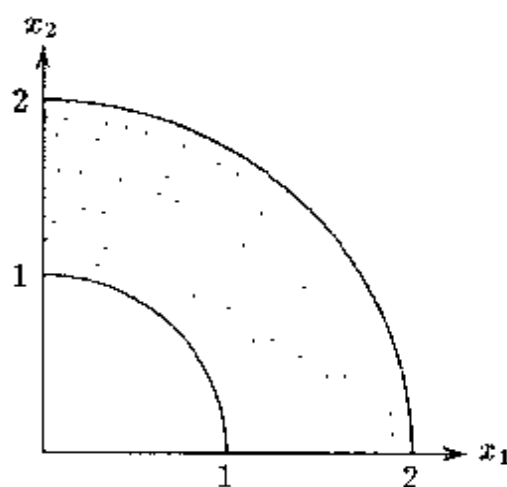
如果 $(x_1^2 + x_2^2 + x_3^2 < 1)$, 返回 0;

如果 $(x_1^2 + x_2^2 + x_3^2 > 4)$, 返回 0;

返回 1.

其中检验函数值 0 表示不可行, 1 表示可行. 容易知道可行集包含于下列超几何体中

$$\Omega = \{(x_1, x_2, x_3) \mid 0 \leq x_1 \leq 2, 0 \leq x_2 \leq 2, 0 \leq x_3 \leq 2\}.$$

图 2.2 在 $x_3 = 0$ 处可行集的横截面

我们可以很容易从这样的超几何体中抽样, 例如, 取

$$x_1 = \mathcal{U}(0, 2), \quad x_2 = \mathcal{U}(0, 2), \quad x_3 = \mathcal{U}(0, 2), \quad (2.13)$$

其中函数 $\mathcal{U}(a, b)$ 用来产生区间 $[a, b]$ 上的均匀分布的随机数, 具体细节将在第 3 章予以讨论. 如果这个染色体不可行, 则拒绝接受, 由式 (2.13) 重新产生一个新的染色体, 如果产生的染色体可行, 则接受它作为种群的一名成员. 经过有限次抽样以后, 得到 30 个可行的染色体

$$\begin{aligned} V_1 &= (0.3903, 0.6723, 1.2507) & V_2 &= (0.9167, 0.2930, 0.3297) \\ V_3 &= (0.2373, 0.1267, 1.7370) & V_4 &= (0.8523, 0.9683, 1.4477) \\ V_5 &= (0.1280, 0.8337, 1.1807) & V_6 &= (0.3283, 0.6830, 1.8263) \\ V_7 &= (1.1223, 0.6363, 1.2303) & V_8 &= (0.5020, 0.8447, 1.0840) \\ V_9 &= (0.0490, 1.7077, 0.2813) & V_{10} &= (0.5643, 0.5450, 1.6913) \\ V_{11} &= (1.1430, 0.6000, 0.3623) & V_{12} &= (1.6243, 1.0153, 0.5573) \\ V_{13} &= (0.7953, 1.3563, 1.1223) & V_{14} &= (0.1240, 1.7903, 0.5593) \\ V_{15} &= (1.2320, 0.0733, 0.9930) & V_{16} &= (1.4473, 1.3397, 0.2947) \\ V_{17} &= (0.3960, 0.6173, 1.2623) & V_{18} &= (0.5420, 0.4000, 1.6593) \end{aligned}$$

$$\begin{aligned}
V_{19} &= (0.1517, 1.0047, 0.5590) & V_{20} &= (1.2550, 1.2957, 0.6413) \\
V_{21} &= (0.1313, 0.8217, 1.4523) & V_{22} &= (0.2383, 1.2930, 0.3637) \\
V_{23} &= (1.3047, 0.4163, 0.4673) & V_{24} &= (1.7893, 0.5220, 0.4343) \\
V_{25} &= (1.1910, 0.1460, 0.5890) & V_{26} &= (0.6023, 1.3187, 0.3897) \\
V_{27} &= (0.9907, 0.8447, 0.9030) & V_{28} &= (0.7467, 1.2017, 1.0873) \\
V_{29} &= (0.9263, 1.5153, 0.0503) & V_{30} &= (0.0823, 0.1867, 1.1217)
\end{aligned}$$

直接计算, 得到如下的初始适应度的值, 即目标值

$$\begin{aligned}
f(V_1) &= 0.0727 & f(V_2) &= 0.0674 & f(V_3) &= 0.0854 \\
f(V_4) &= 0.1277 & f(V_5) &= 0.0082 & f(V_6) &= 0.0401 \\
f(V_7) &= 0.1482 & f(V_8) &= 0.0906 & f(V_9) &= 0.0022 \\
f(V_{10}) &= 0.1144 & f(V_{11}) &= 0.0539 & f(V_{12}) &= 0.0663 \\
f(V_{13}) &= 0.0971 & f(V_{14}) &= 0.0068 & f(V_{15}) &= 0.0294 \\
f(V_{16}) &= 0.0197 & f(V_{17}) &= 0.0792 & f(V_{18}) &= 0.1269 \\
f(V_{19}) &= 0.0170 & f(V_{20}) &= 0.0711 & f(V_{21}) &= 0.0071 \\
f(V_{22}) &= 0.0215 & f(V_{23}) &= 0.0784 & f(V_{24}) &= 0.0616 \\
f(V_{25}) &= 0.0560 & f(V_{26}) &= 0.0327 & f(V_{27}) &= 0.1275 \\
f(V_{28}) &= 0.1013 & f(V_{29}) &= 0.0006 & f(V_{30}) &= 0.0158
\end{aligned}$$

从中可以发现, 染色体 V_7 是其中最好的染色体, 而染色体 V_{29} 是其中最差的. 在此次进化中, 保留染色体 V_7 , 记为 V_0 . 如果在以后的进化过程中, 发现比 V_0 更好的染色体, 则用它取代 V_0 . 根据染色体的目标值, 由好到坏重排染色体如下

$$\begin{aligned}
V'_1 &= (1.1223, 0.6363, 1.2303)(V_7) \\
V'_2 &= (0.8523, 0.9683, 1.4477)(V_4) \\
V'_3 &= (0.9907, 0.8447, 0.9030)(V_{27}) \\
V'_4 &= (0.5420, 0.4000, 1.6593)(V_{18})
\end{aligned}$$

$$\begin{aligned}V'_5 &= (0.5643, 0.5450, 1.6913)(V_{10}) \\V'_6 &= (0.7467, 1.2017, 1.0873)(V_{28}) \\V'_7 &= (0.7953, 1.3563, 1.1223)(V_{13}) \\V'_8 &= (0.5020, 0.8447, 1.0840)(V_8) \\V'_9 &= (0.2373, 0.1267, 1.7370)(V_3) \\V'_{10} &= (0.3960, 0.6173, 1.2623)(V_{17}) \\V'_{11} &= (1.3047, 0.4163, 0.4673)(V_{23}) \\V'_{12} &= (0.3903, 0.6723, 1.2507)(V_1) \\V'_{13} &= (1.2550, 1.2957, 0.6413)(V_{20}) \\V'_{14} &= (0.9167, 0.2930, 0.3297)(V_2) \\V'_{15} &= (1.6243, 1.0153, 0.5573)(V_{12}) \\V'_{16} &= (1.7893, 0.5220, 0.4343)(V_{24}) \\V'_{17} &= (1.1910, 0.1460, 0.5890)(V_{25}) \\V'_{18} &= (1.1430, 0.6000, 0.3623)(V_{11}) \\V'_{19} &= (0.3283, 0.6830, 1.8263)(V_6) \\V'_{20} &= (0.6023, 1.3187, 0.3897)(V_{26}) \\V'_{21} &= (1.2320, 0.0733, 0.9930)(V_{15}) \\V'_{22} &= (0.2383, 1.2930, 0.3637)(V_{22}) \\V'_{23} &= (1.4473, 1.3397, 0.2947)(V_{16}) \\V'_{24} &= (0.1517, 1.0047, 0.5590)(V_{19}) \\V'_{25} &= (0.0823, 0.1867, 1.1217)(V_{30}) \\V'_{26} &= (0.1280, 0.8337, 1.1807)(V_5) \\V'_{27} &= (0.1313, 0.8217, 1.4523)(V_{21}) \\V'_{28} &= (0.1240, 1.7903, 0.5593)(V_{14}) \\V'_{29} &= (0.0490, 1.7077, 0.2813)(V_9) \\V'_{30} &= (0.9263, 1.5153, 0.0503)(V_{29})\end{aligned}$$

根据评价函数 (2.4), 及 $\alpha = 0.05$, 有

$$\text{eval}(\mathbf{V}_i') = 0.05 \cdot (1 - 0.05)^{i-1}, \quad i = 1, 2, \dots, \text{pop_size}.$$

于是得到

$$\begin{aligned} q_1 &= 0.0500 & q_2 &= 0.0975 & q_3 &= 0.1426 & q_4 &= 0.1855 \\ q_5 &= 0.2262 & q_6 &= 0.2649 & q_7 &= 0.3017 & q_8 &= 0.3366 \\ q_9 &= 0.3698 & q_{10} &= 0.4013 & q_{11} &= 0.4312 & q_{12} &= 0.4596 \\ q_{13} &= 0.4867 & q_{14} &= 0.5123 & q_{15} &= 0.5367 & q_{16} &= 0.5599 \\ q_{17} &= 0.5819 & q_{18} &= 0.6028 & q_{19} &= 0.6226 & q_{20} &= 0.6415 \\ q_{21} &= 0.6594 & q_{22} &= 0.6765 & q_{23} &= 0.6926 & q_{24} &= 0.7080 \\ q_{25} &= 0.7226 & q_{26} &= 0.7365 & q_{27} &= 0.7497 & q_{28} &= 0.7622 \\ q_{29} &= 0.7741 & q_{30} &= 0.7854 \end{aligned}$$

现在准备旋转赌轮 30 次. 首先由计算机在区间 $(0, q_{30}] = (0, 0.7854]$ 上产生随机数, 得到 0.0328, 其大于 $q_0 = 0$, 而小于 $q_1 = 0.0500$, 所以选择染色体 $\mathbf{V}_1'(\mathbf{V}_7)$ 作为新种群的一名成员. 第二次产生的随机数为 0.1284, 大于 $q_2 = 0.0975$, 而小于 $q_3 = 0.1426$, 所以 $\mathbf{V}_3'(\mathbf{V}_{27})$ 也被选中. 经过 30 次选择之后, 得到一个新的种群

$$\begin{aligned} \mathbf{V}_1'' &= (1.1223, 0.6363, 1.2303) & \mathbf{V}_2'' &= (0.9907, 0.8447, 0.9030) \\ \mathbf{V}_3'' &= (0.7467, 1.2017, 1.0873) & \mathbf{V}_4'' &= (1.1223, 0.6363, 1.2303) \\ \mathbf{V}_5'' &= (0.9167, 0.2930, 0.3297) & \mathbf{V}_6'' &= (0.5420, 0.4000, 1.6593) \\ \mathbf{V}_7'' &= (0.9167, 0.2930, 0.3297) & \mathbf{V}_8'' &= (0.8523, 0.9683, 1.4477) \\ \mathbf{V}_9'' &= (1.3047, 0.4163, 0.4673) & \mathbf{V}_{10}'' &= (0.9263, 1.5153, 0.0503) \\ \mathbf{V}_{11}'' &= (0.0823, 0.1867, 1.1217) & \mathbf{V}_{12}'' &= (0.5020, 0.8447, 1.0840) \\ \mathbf{V}_{13}'' &= (0.9263, 1.5153, 0.0503) & \mathbf{V}_{14}'' &= (0.9263, 1.5153, 0.0503) \\ \mathbf{V}_{15}'' &= (0.3960, 0.6173, 1.2623) & \mathbf{V}_{16}'' &= (1.7893, 0.5220, 0.4343) \\ \mathbf{V}_{17}'' &= (0.8523, 0.9683, 1.4477) & \mathbf{V}_{18}'' &= (0.9263, 1.5153, 0.0503) \end{aligned}$$

$$\begin{aligned}
V''_{19} &= (0.6023, 1.3187, 0.3897) & V''_{20} &= (0.1313, 0.8217, 1.4523) \\
V''_{21} &= (0.9263, 1.5153, 0.0503) & V''_{22} &= (1.3047, 0.4163, 0.4673) \\
V''_{23} &= (1.1223, 0.6363, 1.2303) & V''_{24} &= (1.6243, 1.0153, 0.5573) \\
V''_{25} &= (1.1910, 0.1460, 0.5890) & V''_{26} &= (0.9907, 0.8447, 0.9030) \\
V''_{27} &= (0.1517, 1.0047, 0.5590) & V''_{28} &= (1.1223, 0.6363, 1.2303) \\
V''_{29} &= (0.0490, 1.7077, 0.2813) & V''_{30} &= (1.6243, 1.0153, 0.5573)
\end{aligned}$$

接着, 对新的种群进行遗传操作, 即交叉和变异操作. 交叉概率 $P_c = 20\%$, 说明平均有 6 个染色体进行交叉操作. 从区间 $[0, 1]$ 上产生随机数, 得 0.6437, 大于 $P_c = 0.20$, 第一个染色体 V''_1 没有选中. 第二次产生的随机数为 0.1256, 小于 $P_c = 0.20$, 第二个染色体 V''_2 被选中用来作为交叉操作的一个父代. 这样, 经过 30 次, 选中 10 个染色体

$$V''_2, V''_3, V''_4, V''_6, V''_{12}, V''_{13}, V''_{15}, V''_{16}, V''_{19}, V''_{28},$$

将它们随机分成五组

$$(V''_{16}, V''_{19}), (V''_3, V''_6), (V''_{15}, V''_2), (V''_4, V''_{12}), (V''_{13}, V''_{28}).$$

由于被选中的染色体是偶数个, 所以容易配对, 若是奇数个, 只需去掉一个即可. 由式 (2.10) 实现交叉操作, 得到如下的一个种群

$$\begin{aligned}
V'''_1 &= (1.1223, 0.6363, 1.2303) & V'''_2 &= (0.5631, 0.6812, 1.1614) \\
V'''_3 &= (0.6068, 0.6540, 1.4781) & V'''_4 &= (1.0155, 0.6722, 1.2051) \\
V'''_5 &= (0.9167, 0.2930, 0.3297) & V'''_6 &= (0.6818, 0.9477, 1.2685) \\
V'''_7 &= (0.9167, 0.2930, 0.3297) & V'''_8 &= (0.8523, 0.9683, 1.4477) \\
V'''_9 &= (1.3047, 0.4163, 0.4673) & V'''_{10} &= (0.9263, 1.5153, 0.0503) \\
V'''_{11} &= (0.0823, 0.1867, 1.1217) & V'''_{12} &= (0.6088, 0.8088, 1.1092) \\
V'''_{13} &= (0.9810, 1.2702, 0.3794) & V'''_{14} &= (0.9263, 1.5153, 0.0503) \\
V'''_{15} &= (0.8236, 0.7808, 1.0040) & V'''_{16} &= (1.6400, 0.6222, 0.4287)
\end{aligned}$$

$$\begin{aligned}
V_{17}''' &= (0.8523, 0.9683, 1.4477) & V_{18}''' &= (0.9263, 1.5153, 0.0503) \\
V_{19}''' &= (0.7517, 1.2184, 0.3953) & V_{20}''' &= (0.1313, 0.8217, 1.4523) \\
V_{21}''' &= (0.9263, 1.5153, 0.0503) & V_{22}''' &= (1.3047, 0.4163, 0.4673) \\
V_{23}''' &= (1.1223, 0.6363, 1.2303) & V_{24}''' &= (1.6243, 1.0153, 0.5573) \\
V_{25}''' &= (1.1910, 0.1460, 0.5890) & V_{26}''' &= (0.9907, 0.8447, 0.9030) \\
V_{27}''' &= (0.1517, 1.0047, 0.5590) & V_{28}''' &= (1.0677, 0.8814, 0.9013) \\
V_{29}''' &= (0.0490, 1.7077, 0.2813) & V_{30}''' &= (1.6243, 1.0153, 0.5573)
\end{aligned}$$

类似于交叉过程, 经过计算机运行, 有以下染色体被选择作为父代用来进行变异操作

$$\begin{aligned}
&V_1''', V_3''', V_4''', V_7''', V_9''', V_{10}''', V_{13}''', V_{14}''', V_{15}''', \\
&V_{16}''', V_{17}''', V_{18}''', V_{19}''', V_{21}''', V_{22}''', V_{23}''', V_{24}'''.
\end{aligned}$$

在 \mathbb{R}^3 中, 随机产生一个方向 $d = (0.1130, -0.8072, 0.1279)$, 在这个方向上, 对染色体 V_1''' 进行变异. 不妨取 M 为 10, 检验

$$V_1''' + M \cdot d = (2.2523, -7.4357, 2.5093),$$

知其不可行. 从 0 到 M 之间产生的一个随机数为 0.2345, 置 $M = 0.2345$. 检验函数显示

$$V_1''' + M \cdot d = (1.1488, 0.4470, 1.2333)$$

是可行的. 所以用 $(1.1488, 0.4470, 1.2333)$ 代替

$$V_1''' = (1.1223, 0.6363, 1.2303).$$

经过变异操作, 得到种群

$$\begin{aligned}
V_1'''' &= (1.1488, 0.4470, 1.2333) & V_2'''' &= (0.5631, 0.6812, 1.1614) \\
V_3'''' &= (0.1701, 0.6540, 1.4550) & V_4'''' &= (0.9978, 0.6722, 1.2077) \\
V_5'''' &= (0.9167, 0.2930, 0.3297) & V_6'''' &= (0.6818, 0.9477, 1.2685) \\
V_7'''' &= (0.9811, 0.2930, 0.1346) & V_8'''' &= (0.8523, 0.9683, 1.4477)
\end{aligned}$$

$$\begin{aligned}
V_9''' &= (1.3047, 0.4163, 0.4285) & V_{10}''' &= (0.8192, 1.5153, 0.0503) \\
V_{11}''' &= (0.0823, 0.1867, 1.1217) & V_{12}''' &= (0.6088, 0.8088, 1.1092) \\
V_{13}''' &= (0.9810, 1.2702, 0.6591) & V_{14}''' &= (0.1878, 1.5153, 0.0503) \\
V_{15}''' &= (0.8236, 0.7808, 0.7472) & V_{16}''' &= (1.7911, 0.6222, 0.3176) \\
V_{17}''' &= (0.6160, 0.9683, 1.4477) & V_{18}''' &= (1.0359, 1.5153, 0.0503) \\
V_{19}''' &= (0.7517, 1.2268, 0.3953) & V_{20}''' &= (0.1313, 0.8217, 1.4523) \\
V_{21}''' &= (0.9263, 1.7268, 0.3645) & V_{22}''' &= (1.2524, 0.4163, 0.4673) \\
V_{23}''' &= (1.1223, 0.6363, 1.2303) & V_{24}''' &= (1.6243, 0.8745, 0.5573) \\
V_{25}''' &= (1.1910, 0.1460, 0.5890) & V_{26}''' &= (0.9907, 0.8447, 0.9030) \\
V_{27}''' &= (0.1517, 1.0047, 0.5590) & V_{28}''' &= (1.0677, 0.8814, 0.9013) \\
V_{29}''' &= (0.0490, 1.7077, 0.2813) & V_{30}''' &= (1.6243, 1.0153, 0.5573)
\end{aligned}$$

到此, 已经得到新一代的种群.

经过 150 代之后, 得到最好的解为

$$x^* = (0.8597, 0.5273, 1.3245),$$

其目标值为 $f(x^*) = 0.1537$, 已接近已知的最优解, 所花费的 CPU 时间为 4.1 秒. 进化过程由图 2.3 给出.

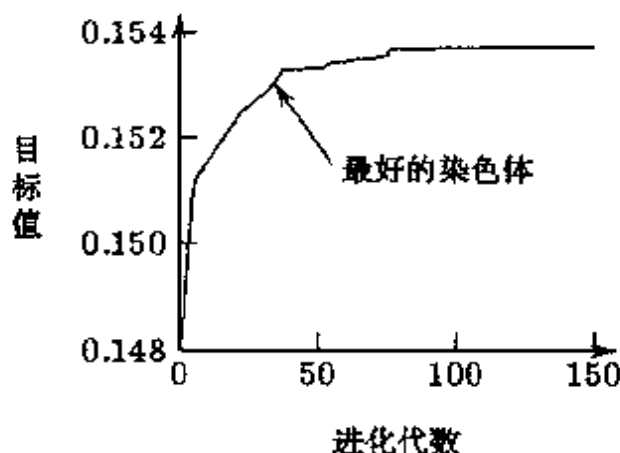


图 2.3 例 1 中的进化过程

例 2 目标规划. 考虑非线性目标规划模型

$$\left\{ \begin{array}{l} \text{lexmin} \quad \{d_1^-, d_2^-, d_3^+, d_4^- + d_4^+\} \\ \text{s.t.} \\ \sum_{i=1}^5 x_i \sin(i \cdot x_i) + d_1^- - d_1^+ = 18 \\ \sum_{i=1}^4 x_i \sin(i \cdot x_i) + d_2^- - d_2^+ = 15 \\ \sum_{i=1}^3 x_i \sin(i \cdot x_i) + d_3^- - d_3^+ = 10 \\ \sum_{i=1}^2 x_i \sin(i \cdot x_i) + d_4^- - d_4^+ = 0 \\ x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \leq 100. \end{array} \right.$$

此模型是很复杂的, 目标函数不但是多峰的, 而且是高度非线性的. 传统方法无法处理这类非线性目标规划, 但遗传算法却是有效的.

目标规划的目标函数的一般形式为

$$\sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-),$$

其中 P_j 为优先因子, 表示各个目标的相对重要性, 对所有的 j , 有 $P_j \gg P_{j+1}$, 但它并不适合作为目标规划的评价函数, 因为对优先因子, 仅有信息 $P_j \gg P_{j+1}$. 事实上, 对染色体, 有如下的序的关系: 对任意两个染色体, 如果在较高的优先级中目标值相等, 则在当前优先级中, 较小的目标值对应的那个比较好. 这种关系是可行集上的一个序, 从而可以按这个序重排这些染色体. 如果两个染色体具有相同的目标值, 则可以随机重排这两个染色体.

除了在每个种群中按上述序的关系重排染色体这一点外, 目标规划的进化过程与单目标规划的进化过程没有什么区别.

经过 6000 次进化后, 遗传算法给出的解为

$$\mathbf{x} = (-1.860, 0.745, 6.823, 6.685, 1.995),$$

它满足前三个目标, 但最后一个目标值为 2.525. 虽然不知道这个问题的精确最优解, 但通过大量的实验, 没有发现比这更好的解, 所以我们对这个结论是满意的. 进化 6000 代, 所花费的 CPU 时间是 68.8 秒, 进化过程如图 2.4.

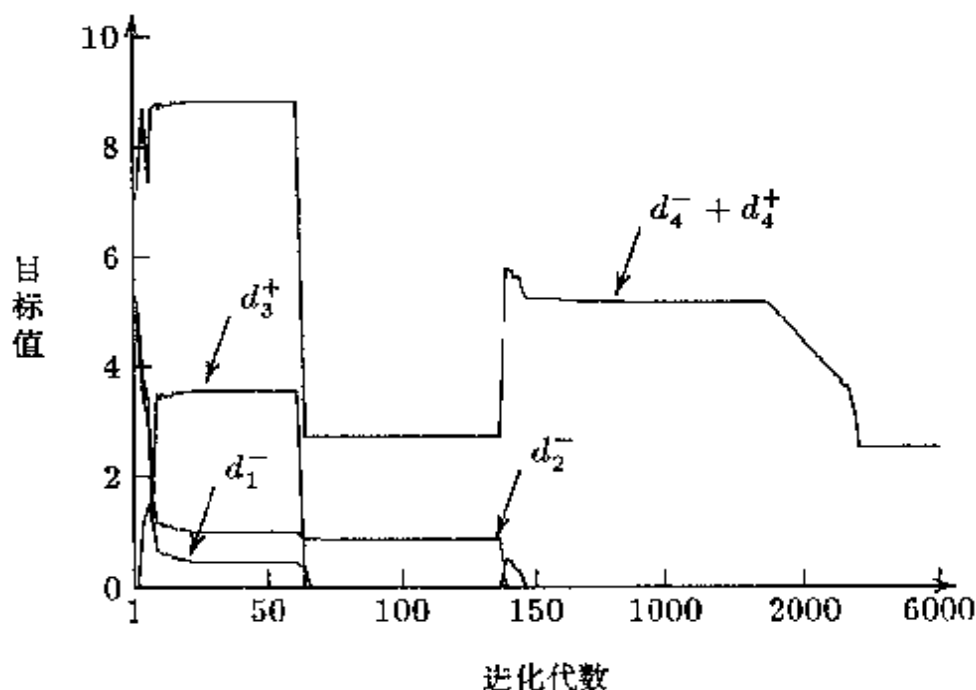


图 2.4 例 2 的进化过程

例 3 多目标规划. 虽然我们可能对多目标规划的所有有效解感兴趣, 但不一定非要找出所有的有效解. 解决多目标优化问题的目的在于实际过程中执行这个解, 而同一时刻不可能执行所有的解, 因此, 一个满意的解对某些实际问题已经足够了.

无论一个问题有多少个目标, 合理的假设是决策者能够对任

意两个可能解进行比较, 即指出它们的好坏. 可以采用下面的方法之一评估可能的解. 例如, 可采用加权和方法, 对 m 个目标函数 $f_i, i = 1, 2, \dots, m$ 进行合并

$$f(\mathbf{x}) = \lambda_1 f_1(\mathbf{x}) + \lambda_2 f_2(\mathbf{x}) + \dots + \lambda_m f_m(\mathbf{x}),$$

其中 $\lambda_1, \lambda_2, \dots, \lambda_m$ 是权重因子, $0 \leq \lambda_i \leq 1, i = 1, 2, \dots, m$, 且 $\lambda_1 + \lambda_2 + \dots + \lambda_m = 1$. 又如, 可采用距离函数法, 根据理想目标向量 $(f_1^0, f_2^0, \dots, f_m^0)$ 把 m 个目标函数合并成

$$f(\mathbf{x}) = \left(\sum_{i=1}^m |f_i(\mathbf{x}) - f_i^0|^\lambda \right)^{\frac{1}{\lambda}},$$

其中 λ 是正的参数. 采用上述方法可以对多目标规划的任意一组可行解由好到坏进行重排.

现考虑非凸集合上的多目标规划

$$\left\{ \begin{array}{l} \max f_1(\mathbf{x}) = x_1^2 + x_2^2 \\ \max f_2(\mathbf{x}) = x_3 / (1 + x_1 + x_2) \\ \text{s.t.} \\ x_1^2 - x_2^2 + x_3^2 \geq 1 \\ x_1^2 + x_2^2 + x_3^2 \leq 4 \\ x_1, x_2, x_3 > 0. \end{array} \right. \quad (2.14)$$

如果使用加权和方法重排染色体, 例如, 对目标函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 分别给予权重 0.4 和 0.6. 经过 2000 代以后, 遗传算法给出的最优解为 $\mathbf{x}^* = (0.0000, 1.9837, 0.2545)$, 其对应的目标值是

$$f_1(\mathbf{x}^*) = 3.9352, \quad f_2(\mathbf{x}^*) = 0.0853.$$

所花费的 CPU 时间为 32.4 秒.

第 3 章

随机模拟和模糊模拟

在研究一个过程或系统时，由于问题的复杂性，往往不能直接利用解析方法去研究。这时，可以利用计算机模拟来处理这类大型复杂的问题。

模拟通过反映系统本质的数学模型，运用数字计算机对过程或系统的运行进行模仿，从而定量地获得系统的性状指标，为决策服务。虽然模拟是一种不很精确的技术，其提供的仅仅是一个统计估计而不是一个精确解，耗时较多，费用较高，但对那些无法用解析方法处理的模型，模拟也许是唯一能够获得问题答案的方法。

根据环境的不同，模拟可分为随机模拟和模糊模拟。随机模拟是一种实现随机（或确定）系统抽样实验的技术，其基础是从给定的概率分布中抽取随机变量。随机模拟也称为 Monte Carlo 模拟。

而模糊模拟是一种实现从模糊系统模型中做抽样实验的技术。大量数值例子表明，在检验模糊约束以及估计模糊系统（或子系统）的可能性方面，模糊模拟取得了成功的应用。

在本章后面几节，主要介绍随机数的产生，模糊集基本理论以及随机模拟技术和模糊模拟技术，目的在于为后面的不确定规划的计算作准备。

3.1 随机数的产生

随机数的产生技术是随机模拟的基础. 设 x 为随机变量, 概率分布为 $F(\cdot)$. 由于 $F(\cdot)$ 是非减函数, 且值域为 $[0, 1]$, 其逆函数 $F^{-1}(\cdot)$ 在 $[0, 1]$ 上有定义. 设 u 是 $[0, 1]$ 上均匀分布的随机变量, 则有

$$\Pr \{F^{-1}(u) \leq y\} = \Pr \{u \leq F(y)\} = F(y), \quad (3.1)$$

那么变量

$$x = F^{-1}(u) \quad (3.2)$$

的概率分布函数必为 $F(\cdot)$. 由此, 要产生服从分布为 $F(\cdot)$ 的随机数, 就只要产生 $[0, 1]$ 上的均匀分布的随机数 u , 然后由 $F^{-1}(u)$ 求得. 这种产生随机数的方法称为逆变换法.

例如, 设 F 为参数为 1 的指数分布, 即

$$F(x) = 1 - e^{-x}, \quad x \geq 0.$$

首先求出 F 的反函数 $F^{-1}(y) = -\ln(1-y)$. 若 u 是 $[0, 1]$ 上均匀随机变量, 则 $F^{-1}(u) = -\ln(1-u)$ 服从参数为 1 的指数分布.

从原理上讲, 逆变换法可以求得服从任何连续分布的随机数, 然而在有些情况下反函数无解析表达式以至于计算很复杂, 此时需要采用其他方法. 下面介绍一些服从常见概率分布的随机数的产生方法.

均匀分布: $U(a, b)$ 设随机变量 x 服从 $[a, b]$ 上的均匀分布, 其概率密度函数为:

$$f(x) = \begin{cases} \frac{1}{b-a}, & a \leq x \leq b \\ 0, & \text{其它,} \end{cases} \quad (3.3)$$

记为 $U(a, b)$. 均匀分布的随机数是构造其他分布的随机数的基础. 均匀分布的随机数可以由计算机附加的硬件来生成, 但是它

不具有再生性. 现在通用的方法是由计算机按一定的算法生成从统计上看满足独立性及均匀性的一串数字. 这样的数称为伪随机数.

用算法生成一串随机数, 至少应满足: 该算法容易执行, 而生成的速度快; 产生的数字序列能通过独立性及均匀性的统计检验.

下面介绍一种目前常见的产生随机数的方法, 称为同余法.

令

$$x_{i+1} = ax_i + c \pmod{m}, \quad i = 1, 2, \dots, n-1, \quad (3.4)$$

其中 a 为正整数, 称为乘子, c 为非负整数, 称为增量, $0 \leq x_1 < m$, x_1 称为种子, m 称为模数, 同时也是伪随机数序列的长度. 这样, 对任何一个初始值 x_1 , 可由 (3.4) 产生一个序列 $\{x_1, x_2, \dots, x_n\}$. 通过下式, 就可以产生区间 $[0, 1]$ 上的均匀分布的随机数

$$u_i = \frac{x_i}{m-1}, \quad i = 1, 2, \dots, n. \quad (3.5)$$

等价地, $[a, b]$ 上的均匀分布的随机数可以由

$$u_i = a + \frac{x_i}{m-1}(b-a), \quad i = 1, 2, \dots, n \quad (3.6)$$

产生. 由 (3.4) 可以看出, 模 m 越大, 随机数的均匀性越好. 从计算的角度来讲, 取 $m = 2^l$ 最方便, 其中 l 为正整数, 例如取 $a = 2^7 + 1, c = 1, m = 2^{35}$.

在 C 语言的库函数中, 有两个产生伪随机数的子函数. 一个是

```
#include <stdlib.h>
```

```
int rand(void)
```

其产生的随机数是 0 和 R_MAX 之间的伪随机整数, 在 `stdlib.h` 中, $R_MAX = 2^{15} - 1$. 另一个子函数是


```
#include <stdlib.h>
int random(int num)
```

其产生的随机数是 0 和 $num - 1$ 之间的伪随机整数. 这样, 区间 $[a, b]$ 上的随机数可以按下列程序产生

```
u = random(num);    // num. 一个整数, 如10000
u = u/(num - 1);
返回 a + u(b - a).
```

指数分布: $\mathcal{E}\mathcal{X}\mathcal{P}(\beta)$ 设随机变量 x 服从指数分布, 其概率密度函数为

$$f(x) = \begin{cases} \frac{1}{\beta} e^{-x/\beta}, & 0 \leq x < \infty, \beta > 0 \\ 0, & \text{其它,} \end{cases} \quad (3.7)$$

记为 $\mathcal{E}\mathcal{X}\mathcal{P}(\beta)$. 可由下列方法产生服从指数分布的随机数.

```
由  $\mathcal{U}(0, 1)$  生成  $u$ ;
 $x = -\beta \ln(u)$ ;
返回  $x$ .
```

Γ 分布: $\mathcal{G}(\alpha, \beta)$ 设随机变量 x 服从 Γ 分布, 其概率密度函数为

$$f(x) = \begin{cases} \frac{x^{\alpha-1} e^{-x/\beta}}{\beta^\alpha \Gamma(\alpha)}, & 0 \leq x < \infty, \alpha > 0, \beta > 0 \\ 0, & \text{其它,} \end{cases} \quad (3.8)$$

记为 $\mathcal{G}(\alpha, \beta)$. 注意到 $\mathcal{G}(1, \beta)$ 分布与 $\mathcal{E}\mathcal{X}\mathcal{P}(\beta)$ 分布是等价的. 当 α 是整数时, 按下列方法产生 Γ 分布的随机数.

```
 $x = 0$ ;
重复
  由  $\mathcal{E}\mathcal{X}\mathcal{P}(1)$  生成  $v$ ;
 $x = x + v$ ;
```

$$\alpha = \alpha - 1;$$

直到($\alpha = 1$)

返回(βx).

β 分布: $B(\alpha, \beta)$ 设随机变量 x 服从 β 分布, 其概率密度函数为

$$f(x) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha)\Gamma(\beta)} x^{\alpha-1} (1-x)^{\beta-1}, \quad \alpha > 0, \beta > 0, 0 \leq x \leq 1, \quad (3.9)$$

记为 $B(\alpha, \beta)$. 当 α 和 β 是整数时, 产生随机数的过程如下:

由 $G(\alpha, 1)$ 生成 y_1 ;

由 $G(\beta, 1)$ 生成 y_2 ;

$$x = y_1 / (y_1 + y_2);$$

返回 x .

正态分布: $N(\mu, \sigma^2)$ 设随机变量 x 服从正态分布, 其概率密度函数为

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp \left[-\frac{(x - \mu)^2}{2\sigma^2} \right], \quad -\infty < x < +\infty, \quad (3.10)$$

记为 $N(\mu, \sigma^2)$, 其中 μ 为均值, σ^2 为方差. 产生随机数的过程如下:

由 $U(0, 1)$ 生成 μ_1 ;

由 $U(0, 1)$ 生成 μ_2 ;

$$z = [-2 \ln(\mu_1)]^{\frac{1}{2}} \sin(2\pi\mu_2);$$

$$x = \mu + \sigma z;$$

返回 x .

对数正态分布: $LOGN(\mu, \sigma^2)$ 设 x 服从正态分布 $N(\mu, \sigma^2)$, 称 $y = e^x$ 服从对数正态分布, 其概率密度函数为:

$$f(y) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma y} \exp \left[-\frac{(\ln(y) - \mu)^2}{2\sigma^2} \right], & 0 \leq y < \infty \\ 0, & \text{其它,} \end{cases} \quad (3.11)$$

记为 $LOGN(\mu, \sigma^2)$. 产生随机数的过程如下:

由 $N(0, 1)$ 生成 z ;

$x = \mu + \sigma z$;

$y = e^x$;

返回 y .

柯西分布: $C(\alpha, \beta)$ 设随机变量 x 服从柯西分布, 其概率密度函数为

$$f(x) = \frac{\beta}{\pi [\beta^2 + (x - \alpha)^2]}, \quad \alpha > 0, \beta > 0, -\infty < x < \infty, \quad (3.12)$$

记为 $C(\alpha, \beta)$. 随机数产生过程为:

由 $N(0, 1)$ 生成 u ;

$x = \alpha - \beta / \tan(\pi u)$;

返回 x .

韦布分布: $W(\alpha, \beta)$ 设随机变量 x 服从韦布分布, 其概率密度函数为

$$f(x) = \begin{cases} \frac{\alpha}{\beta^\alpha} x^{\alpha-1} e^{-(x/\beta)^\alpha}, & 0 \leq x < \infty, \alpha > 0, \beta > 0 \\ 0, & \text{其它,} \end{cases} \quad (3.13)$$

记为 $W(\alpha, \beta)$. 韦布分布的随机数产生过程为:

由 $EXP(1)$ 生成 v ;

$x = \beta v^{1/\alpha}$;

返回 x .

卡方分布: $\chi^2(k)$ 设随机变量 z_1, z_2, \dots, z_k 服从标准正态分布 $N(0, 1)$, 称随机变量

$$y = \sum_{i=1}^k z_i^2 \quad (3.14)$$

服从自由度为 k 的卡方分布, 记为 $\chi^2(k)$. 卡方分布的随机数生成过程为

由 $\mathcal{N}(0, 1)$ 生成 $z_i, i = 1, 2, \dots, k$;

$$y = \sum_{i=1}^k z_i^2;$$

返回 y .

t 分布: $\mathcal{S}(k)$ 设随机变量 z 服从标准正态分布 $\mathcal{N}(0, 1)$, y 服从卡方分布 $\chi^2(k)$, 则称随机变量

$$x = \frac{z}{\sqrt{y/k}} \quad (3.15)$$

服从自由度为 k 的 t 分布, 记为 $\mathcal{S}(k)$. t 分布的随机数生成过程为

由 $\mathcal{N}(0, 1)$ 生成 z ;

由 $\chi^2(k)$ 生成 y ;

$$x = z / \sqrt{y/k};$$

返回 x .

F 分布: $\mathcal{F}(k_1, k_2)$ 设随机变量 y_1 服从卡方分布 $\chi^2(k_1)$, y_2 服从卡方分布 $\chi^2(k_2)$, 则称随机变量

$$x = \frac{y_1/k_1}{y_2/k_2} \quad (3.16)$$

服从 F 分布, k_1 和 k_2 为自由度, 记为 $\mathcal{F}(k_1, k_2)$. F 分布的随机数生成过程为

由 $\chi^2(k_1)$ 生成 y_1 ;

由 $\chi^2(k_2)$ 生成 y_2 ;

$$x = \frac{y_1/k_1}{y_2/k_2};$$

返回 x .

三角分布: $T(a, b, m)$ 一个随机变量 x 服从三角分布, 如果其

概率密度函数为

$$f(x) = \begin{cases} \frac{2(x-a)}{(b-a)(m-a)}, & a < x \leq m \\ \frac{2(b-x)}{(b-a)(b-m)}, & m < x \leq b \\ 0, & \text{其它,} \end{cases} \quad (3.17)$$

记为 $T(a, b, m)$, 其中 $a < m < b$. 三角分布随机数生成过程为

$c = (m-a)/(b-a)$;

由 $U(0, 1)$ 生成 u ;

若 $u < c$, 则 $y = \sqrt{cu}$;

否则 $y = 1 - \sqrt{(1-c)(1-u)}$;

返回 $a + (b-a)y$.

二项分布: $BN(n, p)$ 一个随机变量 x 服从二项分布, 如果其概率密度函数为

$$f(x) = \frac{n!}{(n-x)!x!} p^x (1-p)^{n-x}, \quad x = 0, 1, 2, \dots, n, \quad (3.18)$$

记为 $BN(n, p)$, 其中 n 为正整数, p 为 0 和 1 之间的数. 二项分布随机数生成过程为

$x = 0$;

重复下列循环 n 次

由 $U(0, 1)$ 生成 r ;

若 $r \leq p$, 则 $x = x + 1$;

返回 x .

泊松分布: $P(\lambda)$ 一个随机变量 x 服从均值为 $\lambda (\lambda > 0)$ 的泊

松分布, 如果其概率密度函数为

$$f(x) = \begin{cases} \frac{\lambda^x e^{-\lambda}}{x!}, & x = 0, 1, 2, \dots \\ 0, & \text{其它,} \end{cases} \quad (3.19)$$

记为 $\mathcal{P}(\lambda)$. 泊松分布随机数生成过程为

$x = 0$;

$b = 1$;

mark:

由 $\mathcal{U}(0, 1)$ 生成 u ;

$b = bu$;

若 $b \geq e^{-\lambda}$, 则 $x = x + 1$, 并转移到 mark 处;

返回 x .

几何分布: $\mathcal{GE}(p)$ 一个随机变量 x 服从具有参数 p 的几何分布 ($0 < p < 1$), 如果其概率密度函数为

$$f(x) = \begin{cases} p(1-p)^x, & x = 0, 1, 2, \dots \\ 0, & \text{其它,} \end{cases} \quad (3.20)$$

记为 $\mathcal{GE}(p)$. 几何分布随机数生成过程为:

由 $\mathcal{U}(0, 1)$ 生成 r ;

返回 $\ln r / \ln(1-p)$ 的整数部分.

负二项分布: $\mathcal{NB}(r, p)$ 一个随机变量 x 服从参数为 r 和 p 的负二项分布 ($r = 1, 2, \dots$ 和 $0 < p < 1$), 如果它是 r 个独立的具有参数为 p 的几何分布变量之和, 记为 $\mathcal{NB}(r, p)$. 负二项分布随机数生成过程为:

由 $\mathcal{GE}(p)$ 生成 $y_i, i = 1, 2, \dots, r$;

返回 $y_1 + y_2 + \dots + y_r$.

多维正态分布: $\mathcal{N}(\mu, \Sigma)$ 一个 n 维随机向量 \mathbf{x} 服从多维正态

分布, 如果其概率密度函数为

$$f(x_1, x_2, \dots, x_n) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} \exp \left[-\frac{1}{2} (\mathbf{x} - \mu)' \Sigma^{-1} (\mathbf{x} - \mu) \right].$$

一个 n 维正态分布随机数生成过程为

生成上三角矩阵 C 使 $\Sigma = CC'$;

由 $\mathcal{N}(0, 1)$ 生成 $\mu_1, \mu_2, \dots, \mu_n$;

$$x_k = \mu_k + \sum_{i=1}^k c_{ki} \mu_i, \quad (k = 1, 2, \dots, n);$$

返回 $\mathbf{x} = (x_1, x_2, \dots, x_n)$.

复杂区域上的均匀分布: 下面讨论如何在空间 \mathbb{R}^n 的一个复杂区域 S 中产生均匀随机向量 (S 可以理解成数学规划中的一个可行集).

首先, 确定一个包含 S 的简单区域 Ω , 如, 一个 n 维超几何体

$$\Omega = \{(x_1, x_2, \dots, x_n) \in \mathbb{R}^n \mid a_i \leq x_i \leq b_i, i = 1, 2, \dots, n\}.$$

在超几何体中, 随机向量比较容易产生. 事实上, 只要 x_i 是 $[a_i, b_i]$ ($i = 1, 2, \dots, n$) 上的均匀随机变量, (x_1, x_2, \dots, x_n) 就是超几何体 Ω 上的均匀分布的随机向量. 生成的随机向量接受与否依赖于它是否在区域 S 内.

给定包含 S 的超几何体 Ω ;

重复以下过程

由 $\mathcal{U}(a_i, b_i)$ 分别生成 x_i , $i = 1, 2, \dots, n$;

置 $\mathbf{x} = (x_1, x_2, \dots, x_n)$;

直到 $(\mathbf{x} \in S)$

返回 \mathbf{x} .

3.2 随机模拟

随机模拟的应用范围非常广泛,如微积分、线性代数、优化问题、概率计算等.下面通过一些数值例子解释如何进行随机模拟.

例 1 在随机优化问题中,经常需要计算随机积分.现考虑积分

$$\theta = \int_{\mathbb{R}^n} g(\xi) d\Phi(\xi), \quad (3.21)$$

其中 $g(\xi)$ 是可积函数, $\Phi(\xi)$ 是随机向量 ξ 的分布函数,则 θ 显然是 $g(\xi)$ 的期望值,于是可以用下式估计 θ 的值

$$\theta = \frac{1}{N} \sum_{i=1}^N g(\xi_i). \quad (3.22)$$

其中 $\xi_1, \xi_2, \dots, \xi_N$ 为由概率分布函数 $\Phi(\xi)$ 产生的随机向量, N 为抽样个数.

例 2 考虑下面的无约束优化问题

$$\max_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x}). \quad (3.23)$$

当由梯度法求解此问题时,假定目标函数 f 有一阶连续偏导数,以 \mathbf{x}_i 表示极大点的第 i 次近似,采用下式求其第 $i+1$ 次近似点 \mathbf{x}_{i+1} ,

$$\mathbf{x}_{i+1} = \mathbf{x}_i + a_i \nabla f(\mathbf{x}_i), \quad a_i > 0, \quad (3.24)$$

这里

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f(\mathbf{x})}{\partial x_1}, \frac{\partial f(\mathbf{x})}{\partial x_2}, \dots, \frac{\partial f(\mathbf{x})}{\partial x_n} \right),$$

是 $f(\mathbf{x})$ 的梯度, $\partial f(\mathbf{x})/\partial x_j, j=1, 2, \dots, n$ 是偏导数, a_i 是第 i 次步长.

众所周知, 这种算法经常会收敛到局部最优解而不是全局最优解, 尤其目标函数 $f(x)$ 是多峰的情况. 为了避免收敛到局部最优解, 可以采用下面的迭代公式

$$x_{i+1} = x_i + \frac{a_i}{2b_i} [f(x_i + b_i \cdot d_i) - f(x_i - b_i \cdot d_i)] \cdot d_i, \quad (3.25)$$

其中 $a_i, b_i > 0$, d_i 是第 i 次迭代过程中从 n 维单位球体中均匀产生的随机向量.

这种求解优化问题的方法称为随机模拟法.

例 3 估计事件发生的概率

$$\theta = \Pr \{g_j(\xi) \leq 0, j = 1, 2, \dots, k\}, \quad (3.26)$$

这里 ξ 为随机变量, 其分布为 $\Phi(\xi)$, $g_j(\xi)$ 是实值函数, $j = 1, 2, \dots, k$. 为估计上述事件发生的概率, 从概率分布 $\Phi(\xi)$ 中, 独立产生 N 个随机变量 $\xi_i, i = 1, 2, \dots, N$. 设 N 次中共有 N' 个 ξ_i 满足

$$g_j(\xi_i) \leq 0, \quad j = 1, 2, \dots, k,$$

$i = 1, 2, \dots, N$. 由大数定律, 可以用下式估计 θ 的值,

$$\theta = \frac{N'}{N}. \quad (3.27)$$

考虑一个具体的例子, 例如

$$\theta = \Pr \{\xi_1 + \xi_2^2 \geq 3, \xi_3 + \xi_4^2 \leq 9\}, \quad (3.28)$$

其中 ξ_1 服从均匀分布 $U(2, 5)$, ξ_2 服从指数分布 $EXP(3)$, ξ_3 和 ξ_4 分别服从正态分布 $N(3, 2)$ 和 $N(1, 1)$, 模拟的结果如图 3.1, 直线代表已知的概率 0.85, 曲线表示不同的循环次数所模拟出的结果.

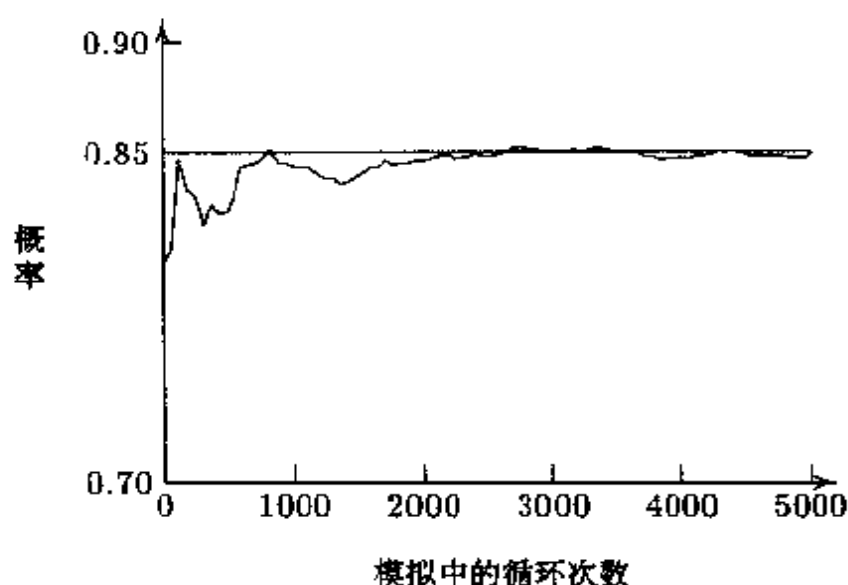


图 3.1 例 3 的随机模拟

从图 3.1 中可以看到, 经过 1000 次循环所得到结果的相对误差小于 1%, 说明对这类问题, 随机模拟是成功的.

例 4 考虑问题

$$\Pr\{f(\xi) \geq \bar{f}\} \geq \alpha, \quad (3.29)$$

其中 ξ 为随机变量, 其概率分布函数为 $\Phi(\xi)$, α 为给定的置信水平. 我们的目的是要找到使不等式 (3.29) 成立的最大的 \bar{f} . 首先从概率分布 $\Phi(\xi)$ 中产生 N 个独立的随机变量 $\xi_i, i = 1, 2, \dots, N$, 由 $f_i = f(\xi_i), i = 1, 2, \dots, N$, 得到序列 $\{f_1, f_2, \dots, f_N\}$, 取 N' 为 αN 的整数部分, 则由大数定律, $\{f_1, f_2, \dots, f_N\}$ 中第 N' 个最大的元素可以作为 \bar{f} 的估计.

例如, 求使下式成立的最大的 \bar{f}

$$\Pr\{\xi_1 + \xi_2^2 + \xi_3^3 \geq \bar{f}\} \geq 0.8,$$

其中 ξ_1 服从均匀分布 $U(1, 3)$, ξ_2 服从指数分布 $EXP(1)$, ξ_3 服从正态分布 $N(2, 1)$. 抽样 1000 次, 其第 800 个最大元素为 4.988,

于是得到的 \bar{f} 的估计值为 4.988.

3.3 模糊集合理论

从 1965 年 Zadeh^[143] 提出模糊集理论至今, 模糊集理论的研究已受到广泛的重视, 其应用范围遍及理、工、农、医及社会科学等众多的领域. 本节简单介绍一下模糊集理论的基本概念和本书后面将要用到的一些模糊集理论的结果.

下面的模糊集的概念由 Zadeh^[143] 提出.

定义 3.1 给定了论域 X , 其上的一个模糊集 A 是指, 对任何 $x \in X$, 都指定了一个数 $\mu_A(x) \in [0, 1]$ 与 x 对应, 它叫做 x 对 A 的隶属度. 这意味着作了一个映射

$$\mu_A: X \rightarrow [0, 1], \quad (3.30)$$

这个映射称为 A 的隶属函数.

也就是说, 一个模糊集 A 可以表示成序偶

$$A = \{(x, \mu_A(x)) \mid x \in X\}. \quad (3.31)$$

其中第一分量 x 表示论域中的元素, 第二分量 $\mu_A(x)$ 为相应的隶属度. 也可以对此概念作一些推广, 把区间 $[0, 1]$ 换成一般的区间.

考虑模糊集合 $A, B \subset X$, 隶属函数分别为 $\mu_A(x)$ 和 $\mu_B(x)$.

(a) 称 A 为 B 的子集, 记为 $A \subset B$, 如果

$$\mu_A(x) \leq \mu_B(x), \quad \forall x \in X. \quad (3.32)$$

(b) 称 A 与 B 相等, 记为 $A = B$, 如果

$$\mu_A(x) = \mu_B(x), \quad \forall x \in X. \quad (3.33)$$

(c) A 的补集 \bar{A} 的隶属函数定义为

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x), \quad \forall x \in X. \quad (3.34)$$

(d) A 和 B 的并 $A \cup B$ 的隶属函数定义为

$$\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x), \quad \forall x \in X. \quad (3.35)$$

(e) A 与 B 的交 $A \cap B$ 的隶属函数定义为

$$\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x), \quad \forall x \in X. \quad (3.36)$$

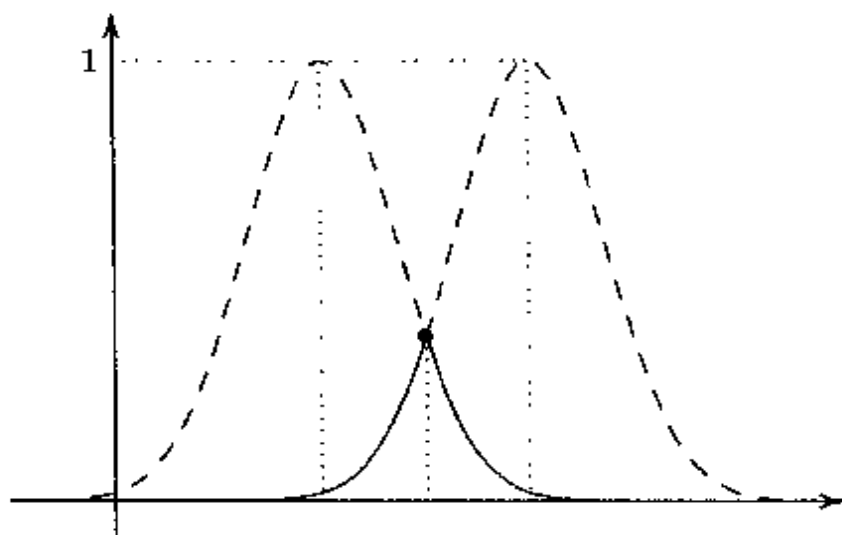


图 3.2 实线和虚线分别表示两个模糊集的和和交的隶属函数

定义3.2 对任意给定的 $\alpha \in [0, 1]$, 记

$$A_\alpha = \{x \in X \mid \mu_A(x) \geq \alpha\}. \quad (3.37)$$

称 A_α 为 A 的 α 水平截集.

特别地, 我们称实数域 \mathfrak{R} 上的模糊子集 \tilde{a} 为模糊数, 记 \tilde{a} 的隶属函数为 $\mu_{\tilde{a}}: \mathfrak{R} \rightarrow [0, 1]$. 假设 \tilde{a} 和 \tilde{b} 为两个模糊数, 隶属函数

分别为 $\mu_{\tilde{a}}$ 和 $\mu_{\tilde{b}}$, 由模糊数的定义和 Zadeh^[145] 的可能性理论, $\tilde{a} \leq \tilde{b}$ 的可能性定义为 (参看文献 [34][35]),

$$\text{Pos}\{\tilde{a} \leq \tilde{b}\} = \sup \{ \min(\mu_{\tilde{a}}(x), \mu_{\tilde{b}}(y)) \mid x, y \in \mathfrak{R}, x \leq y \}, \quad (3.38)$$

其中 Pos 表示可能性. 类似地, $\tilde{a} < \tilde{b}$ 的可能性定义为

$$\text{Pos}\{\tilde{a} < \tilde{b}\} = \sup \{ \min(\mu_{\tilde{a}}(x), \mu_{\tilde{b}}(y)) \mid x, y \in \mathfrak{R}, x < y \}. \quad (3.39)$$

$\tilde{a} = \tilde{b}$ 的可能性定义为

$$\text{Pos}\{\tilde{a} = \tilde{b}\} = \sup \{ \min(\mu_{\tilde{a}}(x), \mu_{\tilde{b}}(x)) \mid x \in \mathfrak{R} \}. \quad (3.40)$$

特别地, 当 \tilde{b} 为清晰数 b 时, 有

$$\begin{cases} \text{Pos}\{\tilde{a} \leq b\} = \sup \{ \mu_{\tilde{a}}(x) \mid x \in \mathfrak{R}, x \leq b \}, \\ \text{Pos}\{\tilde{a} < b\} = \sup \{ \mu_{\tilde{a}}(x) \mid x \in \mathfrak{R}, x < b \}, \\ \text{Pos}\{\tilde{a} = b\} = \mu_{\tilde{a}}(b). \end{cases} \quad (3.41)$$

设 $f: \mathfrak{R} \times \mathfrak{R} \rightarrow \mathfrak{R}$ 为实数域上的二元函数, 那么它可以推广到模糊数领域. 对两个模糊数 \tilde{a}, \tilde{b} , 记 $\tilde{c} = f(\tilde{a}, \tilde{b})$, 那么隶属函数 $\mu_{\tilde{c}}$ 为

$$\mu_{\tilde{c}}(z) = \sup \{ \min(\mu_{\tilde{a}}(x), \mu_{\tilde{b}}(y)) \mid x, y \in \mathfrak{R}, z = f(x, y) \}. \quad (3.42)$$

其中 $z \in \mathfrak{R}$.

一般地, 设 $f: \mathfrak{R}^n \rightarrow \mathfrak{R}$ 为 n 维欧几里得空间上的实函数, $\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n \in \mathfrak{R}$ 为模糊数, 记 $\tilde{c} = f(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n)$, 那么其隶属函数 $\mu_{\tilde{c}}$ 为

$$\mu_{\tilde{c}}(z) = \sup \left\{ \min_{1 \leq i \leq n} \mu_{\tilde{a}_i}(x_i) \mid \begin{array}{l} x_i \in \mathfrak{R}, i = 1, 2, \dots, n \\ z = f(x_1, x_2, \dots, x_n) \end{array} \right\}. \quad (3.43)$$

因此, $f(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) \leq b$ 成立的可能性为

$$\text{Pos} \{f(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) \leq b\} = \sup \{ \mu_{\tilde{c}}(z) \mid z \in \mathfrak{R}, z \leq b \}, \quad (3.44)$$

其中隶属函数 $\mu_{\tilde{c}}$ 由 (3.43) 给出, 即可能性可以表示为

$$\sup \left\{ \min_{1 \leq i \leq n} \mu_{\tilde{a}_i}(x_i) \mid \begin{array}{l} x_i \in \mathfrak{R}, i = 1, 2, \dots, n \\ f(x_1, x_2, \dots, x_n) \leq b \end{array} \right\}. \quad (3.45)$$

更一般的情况, 假设 $f_j: \mathfrak{R}^n \rightarrow \mathfrak{R}$ 是 n 维欧几里得空间上的实值函数, $j = 1, 2, \dots, m$. 则不等式组

$$f_j(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) \leq b_j, \quad j = 1, 2, \dots, m \quad (3.46)$$

成立的可能性为

$$\begin{aligned} & \text{Pos} \{f_j(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) \leq b_j, j = 1, 2, \dots, m\} \\ &= \sup_{x_1, x_2, \dots, x_n} \left\{ \min_{1 \leq i \leq n} \mu_{\tilde{a}_i}(x_i) \mid \begin{array}{l} f_j(x_1, x_2, \dots, x_n) \leq b_j \\ j = 1, 2, \dots, m \end{array} \right\}. \end{aligned}$$

其中 $b_j, j = 1, 2, \dots, m$ 为情晰数.

类似地, 我们有

$$\begin{aligned} & \text{Pos} \{f_j(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) < b_j, j = 1, 2, \dots, m\} \\ &= \sup_{x_1, x_2, \dots, x_n} \left\{ \min_{1 \leq i \leq n} \mu_{\tilde{a}_i}(x_i) \mid \begin{array}{l} f_j(x_1, x_2, \dots, x_n) < b_j \\ j = 1, 2, \dots, m \end{array} \right\} \end{aligned}$$

和

$$\begin{aligned} & \text{Pos} \{f_j(\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_n) = b_j, j = 1, 2, \dots, m\} \\ &= \sup_{x_1, x_2, \dots, x_n} \left\{ \min_{1 \leq i \leq n} \mu_{\tilde{a}_i}(x_i) \mid \begin{array}{l} f_j(x_1, x_2, \dots, x_n) = b_j \\ j = 1, 2, \dots, m \end{array} \right\}. \end{aligned}$$

而等式与不等式混合的情况也与之类似.

下面用梯形模糊数解释上述结果. 梯形模糊数记为 (r_1, r_2, r_3, r_4) , 其中 r_1, r_2, r_3, r_4 分别为清晰数, 且 $r_1 < r_2 \leq r_3 < r_4$, 其隶属函数为

$$\mu(x) = \begin{cases} \frac{x - r_1}{r_2 - r_1}, & r_1 \leq x \leq r_2 \\ 1, & r_2 \leq x \leq r_3 \\ \frac{x - r_4}{r_3 - r_4}, & r_3 \leq x \leq r_4 \\ 0, & \text{其它} \end{cases} \quad (3.47)$$

若 $r_2 = r_3$, 此时梯形模糊数称为三角模糊数, 记为 (r_1, r_2, r_4) . 考虑两个梯形模糊数 $\tilde{r} = (r_1, r_2, r_3, r_4)$ 和 $\tilde{b} = (b_1, b_2, b_3, b_4)$, 如图 3.3 所示.

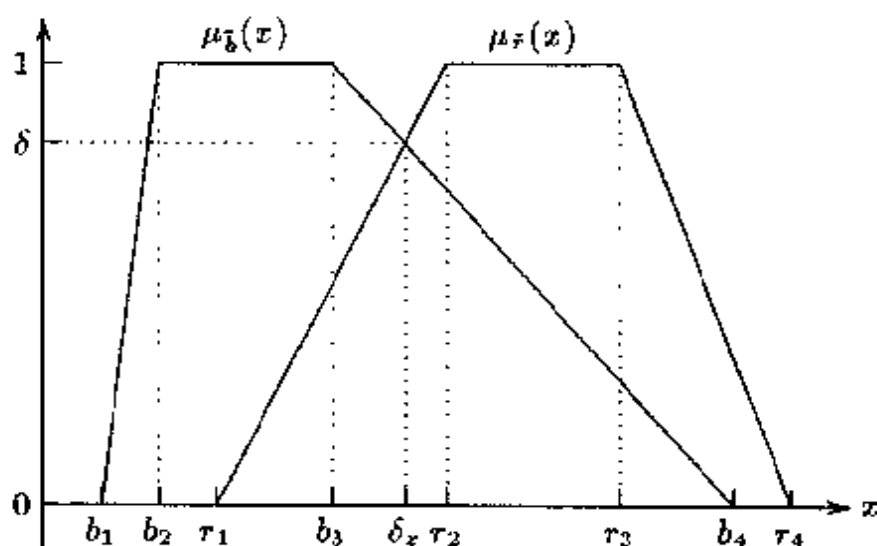


图 3.3 两个梯形模糊数 \tilde{r} 和 \tilde{b}

若 $r_2 \leq b_3$, 则

$$\begin{aligned}\text{Pos}\{\tilde{r} \leq \tilde{b}\} &= \sup\{\min\{\mu_{\tilde{r}}(x), \mu_{\tilde{b}}(y)\} \mid x \leq y\} \\ &\geq \min\{\mu_{\tilde{r}}(r_2), \mu_{\tilde{b}}(b_3)\} \\ &= \min\{1, 1\} = 1,\end{aligned}$$

若 $r_2 > b_3$ 且 $r_1 \leq b_4$, 则在隶属函数 $\mu_{\tilde{r}}(x)$ 和 $\mu_{\tilde{b}}(x)$ 的交点 δ_x 处得到最大值, 于是

$$\text{Pos}\{\tilde{r} \leq \tilde{b}\} = \delta = \frac{b_4 - r_1}{(b_4 - b_3) + (r_2 - r_1)},$$

且

$$\delta_x = r_1 + (r_2 - r_1)\delta.$$

若 $r_1 > b_4$, 则对任意的 $x < y$, 下面两个等式中至少有一个等式成立,

$$\mu_{\tilde{r}}(x) = 0, \quad \mu_{\tilde{b}}(y) = 0,$$

所以, 我们有 $\text{Pos}\{\tilde{r} \leq \tilde{b}\} = 0$. 总结以上结果, 有

$$\text{Pos}\{\tilde{r} \leq \tilde{b}\} = \begin{cases} 1, & r_2 \leq b_3 \\ \delta, & r_2 \geq b_3, r_1 \leq b_4 \\ 0, & r_1 \geq b_4. \end{cases} \quad (3.48)$$

尤其是当 \tilde{b} 为清晰数 0 时, 有

$$\text{Pos}\{\tilde{r} \leq 0\} = \begin{cases} 1, & r_2 \leq 0 \\ \delta, & r_1 \leq 0 \leq r_2 \\ 0, & r_1 \geq 0, \end{cases} \quad (3.49)$$

其中

$$\delta = \frac{r_1}{r_1 - r_2}. \quad (3.50)$$

下面再介绍一个引理.

引理3.1 设梯形模糊数为 $\tilde{r} = (r_1, r_2, r_3, r_4)$, 则对任意给定的置信水平 α , $0 \leq \alpha \leq 1$, $\text{Pos}\{\tilde{r} \leq 0\} \geq \alpha$ 当且仅当 $(1-\alpha)r_1 + \alpha r_2 \leq 0$.

证明 若 $\text{Pos}\{\tilde{r} \leq 0\} \geq \alpha$, 则 $r_2 \leq 0$ 和 $r_1/(r_1 - r_2) \geq \alpha$ 二者之中必有一者成立, 若 $r_2 \leq 0$, 则 $r_1 < r_2 \leq 0$, 所以 $(1-\alpha)r_1 + \alpha r_2 \leq 0$. 若 $r_1/(r_1 - r_2) \geq \alpha$, 由 $r_1 < r_2$ 可得 $r_1 \leq \alpha(r_1 - r_2)$. 故无论何种情况, 均有 $(1-\alpha)r_1 + \alpha r_2 \leq 0$.

若 $(1-\alpha)r_1 + \alpha r_2 \leq 0$, 分两种情况. 当 $r_2 \leq 0$ 时, 有 $\text{Pos}\{\tilde{r} \leq 0\} = 1$, 这意味着 $\text{Pos}\{\tilde{r} \leq 0\} \geq \alpha$. 当 $r_2 > 0$ 时, 有 $r_1 - r_2 < 0$, 重排 $(1-\alpha)r_1 + \alpha r_2 \leq 0$ 得到 $r_1/(r_1 - r_2) \geq \alpha$, 即 $\text{Pos}\{\tilde{r} \leq 0\} \geq \alpha$. \square

由 (3.42), 可以得到梯形模糊数 $\tilde{a} = (a_1, a_2, a_3, a_4)$ 与 $\tilde{b} = (b_1, b_2, b_3, b_4)$ 之和 $\tilde{a} + \tilde{b}$ 的隶属函数为

$$\begin{aligned} \mu_{\tilde{a}+\tilde{b}}(z) &= \sup \{ \min \{ \mu_{\tilde{a}}(x), \mu_{\tilde{b}}(y) \} \mid z = x + y \} \\ &= \begin{cases} \frac{z - (a_1 + b_1)}{(a_2 + b_2) - (a_1 + b_1)}, & a_1 + b_1 \leq z \leq a_2 + b_2 \\ 1, & a_2 + b_2 \leq z \leq a_3 + b_3 \\ \frac{z - (a_4 + b_4)}{(a_3 + b_3) - (a_4 + b_4)}, & a_3 + b_3 \leq z \leq a_4 + b_4 \\ 0, & \text{其它,} \end{cases} \end{aligned}$$

即梯形模糊数的和还是梯形模糊数, 且

$$\tilde{a} + \tilde{b} = (a_1 + b_1, a_2 + b_2, a_3 + b_3, a_4 + b_4). \quad (3.51)$$

下面考虑梯形模糊数 \tilde{a} 与纯量 λ 的积, 由定义

$$\mu_{\lambda \cdot \tilde{a}}(z) = \sup \{ \mu_{\tilde{a}}(x) \mid z = \lambda x \}$$

得到

$$\lambda \cdot \tilde{a} = \begin{cases} (\lambda a_1, \lambda a_2, \lambda a_3, \lambda a_4), & \lambda \geq 0 \\ (\lambda a_4, \lambda a_3, \lambda a_2, \lambda a_1), & \lambda < 0, \end{cases} \quad (3.52)$$

即梯形模糊数与纯量 λ 的积也是梯形模糊数. 因此梯形模糊数的加权和仍是梯形模糊数. 例如, 假设 \tilde{a}_i 是梯形模糊数 $(a_{i1}, a_{i2}, a_{i3}, a_{i4})$, λ_i 为纯量, $i = 1, 2, \dots, n$. 定义

$$\lambda_i^+ = \begin{cases} \lambda_i, & \text{若 } \lambda_i \geq 0 \\ 0, & \text{其它,} \end{cases} \quad \lambda_i^- = \begin{cases} 0, & \text{若 } \lambda_i \geq 0 \\ -\lambda_i, & \text{其它,} \end{cases}$$

$i = 1, 2, \dots, n$, 则 λ_i^+ 和 λ_i^- 非负且 $\lambda_i = \lambda_i^+ - \lambda_i^-$. 由模糊数的“积”与“和”运算, 有

$$\tilde{a} = \sum_{i=1}^n \lambda_i \cdot \tilde{a}_i = \begin{pmatrix} \sum_{i=1}^n (\lambda_i^+ a_{i1} - \lambda_i^- a_{i4}) \\ \sum_{i=1}^n (\lambda_i^+ a_{i2} - \lambda_i^- a_{i3}) \\ \sum_{i=1}^n (\lambda_i^+ a_{i3} - \lambda_i^- a_{i2}) \\ \sum_{i=1}^n (\lambda_i^+ a_{i4} - \lambda_i^- a_{i1}) \end{pmatrix}^T.$$

3.4 模糊模拟

对模糊系统, Liu 和 Iwamura^[87, 88] 提出可以用计算机模糊模拟技术检验下列不等式

$$\text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\} \geq \alpha$$

是否成立, 并且可以计算诸如下面的模糊事件的可能性,

$$G = \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\},$$

以及寻找最大的 f 值使得

$$\text{Pos} \{g(\xi) \geq f\} \geq \alpha,$$

其中 ξ 为已知模糊向量, 其隶属函数为 $\mu(\cdot)$.

下面介绍一些用来处理此类模糊系统的计算机模糊模拟技术.

情况 I 检验 $\text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\} \geq \alpha$

由模糊数的运算,

$$\text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\} \geq \alpha$$

当且仅当存在一个清晰向量 ξ^0 使得

$$g_i(\xi^0) \leq 0, \quad i = 1, 2, \dots, k,$$

且 $\mu(\xi^0) \geq \alpha$. 为了能够用计算机检验这个条件, 由模糊向量 ξ 随机生成一个清晰向量 ξ^0 使 $\mu(\xi^0) \geq \alpha$, 即在模糊向量 ξ 的 α 水平数集中随机抽取一个向量. 如果模糊向量的 α 水平截集过于复杂, 可以从包含 α 水平截集的超几何体 Ω 中抽取向量 ξ^0 , 然后检验 ξ^0 是否在 α 水平截集中, 拒绝与否依赖于 $\mu(\xi^0) \geq \alpha$ 是否成立. 如果向量 ξ^0 满足 $g_i(\xi^0) \leq 0, i = 1, 2, \dots, k$, 则认为 $\text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\} \geq \alpha$ 成立. 否则, 按同样方法, 由模糊向量 ξ 重新生成清晰向量 ξ^0 并检验约束条件. 经过给定的 N 次循环以后, 如果没有生成可行的向量 ξ^0 , 则认为

$$\text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\} \geq \alpha$$

不成立. 上述过程可以归纳为

步骤 1 随机地从模糊向量 ξ 的 α 水平截集中生成清晰向量 ξ^0 ;

步骤 2 若 $g_i(\xi^0) \leq 0, i = 1, 2, \dots, k$, 返回“成立”;

步骤 3 重复步骤 1 和 2 共 N 次;

步骤 4 返回“不成立”.

情况 II 计算 $G = \text{Pos} \{g_i(\xi) \leq 0, i = 1, 2, \dots, k\}$

首先置 $G = 0$. 其次由模糊向量 ξ 生成清晰向量 ξ^0 . 在实际操作过程中, 一般对可能性较低的决策向量不感兴趣, 所以可以事先置一水平, 如 α_0 , 然后从模糊向量 ξ 的 α_0 水平截集中随机地生成清晰向量 ξ^0 . 如果这个集合难以由计算机描述, 可以给出一个大的区域, 如包含所有感兴趣的样本的超几何体 Ω . 当然, 区域越小, 模糊模拟效率越高. 若 $g_i(\xi^0) \leq 0, i = 1, 2, \dots, k$, 且 $G < \mu(\xi^0)$, 则置 $G = \mu(\xi^0)$. 重复以上过程直到给定的次数 N 完成为止. 我们把值 G 作为可能性的估计值. 总结以上过程如下

步骤 1 置 $G = 0$;

步骤 2 从模糊向量 ξ 的 α_0 水平截集中随机地生成清晰向量 ξ^0 ;

步骤 3 若 $g_i(\xi^0) \leq 0, i = 1, 2, \dots, k$, 且 $G < \mu(\xi^0)$, 则置

$$G = \mu(\xi^0);$$

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 返回 G .

情况 III 处理 $\text{Pos} \{g(\xi) \geq f\} \geq \beta$

对模糊约束如

$$\text{Pos} \{g(\xi) \geq f\} \geq \beta, \quad (3.53)$$

我们的目的是找出最大的 f 使不等式 (3.53) 成立. 首先, 置 $f = -\infty$. 由模糊向量 ξ 随机生成清晰向量 ξ^0 , 使 $\mu(\xi^0) \geq \beta$, 即在模糊向量 ξ 的 β 水平截集中随机抽取一个向量, 如果 $f < g(\xi^0)$, 则置 $f = g(\xi^0)$. 重复以上过程 N 次, 所得的值 f 即作为估计. 总结以上过程如下

步骤 1 置 $f = -\infty$;

步骤 2 从模糊向量 ξ 的 β 水平截集中随机生成清晰向量 ξ^0 ;

步骤 3 若 $f < g(\xi^0)$, 则置 $f = g(\xi^0)$;

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 返回 f .

情况 IV 处理 $\text{Pos}\{g(\xi) \leq f\} \geq \beta$

对模糊约束如

$$\text{Pos}\{g(\xi) \leq f\} \geq \beta, \quad (3.54)$$

我们的目的是找到最小的 f 使不等式 (3.54) 成立. 首先, 置 $f = +\infty$. 然后由模糊向量 ξ 随机地生成清晰向量 ξ^0 使 $\mu(\xi^0) \geq \beta$, 即在模糊向量 ξ 的 β 水平截集中随机抽取. 如果 $f > g(\xi^0)$, 则置 $f = g(\xi^0)$. 重复以上过程 N 次, 所得值 f 作为估计值. 总结以上过程如下

步骤 1 置 $f = +\infty$;

步骤 2 从模糊向量 ξ 的 β 水平截集中随机地生成清晰向量 ξ^0 ;

步骤 3 若 $f > g(\xi^0)$, 则置 $f = g(\xi^0)$;

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 返回 f .

下面通过几个数值例子来解释模糊模拟技术.

例 1 考虑两个模糊数 \tilde{r} 和 \tilde{b} , 其隶属函数分别为

$$\mu_{\tilde{r}}(\xi) = \exp [-(\xi - 2)^2]$$

和

$$\mu_{\tilde{b}}(\xi) = \exp [-(\xi - 1)^2],$$

如图 3.4 所示.

已知 $\tilde{r} \leq \tilde{b}$ 的可能性为 $\text{Pos}\{\tilde{r} \leq \tilde{b}\} = \exp(-0.5^2) = 0.778$. 下面通过模糊模拟技术求得. 首先, 取区间 $[0, 3]$ 作为包含两个模糊数 \tilde{r} 和 \tilde{b} 的 α 水平截集的超几何体, 余下的部分由于可能性太低而不

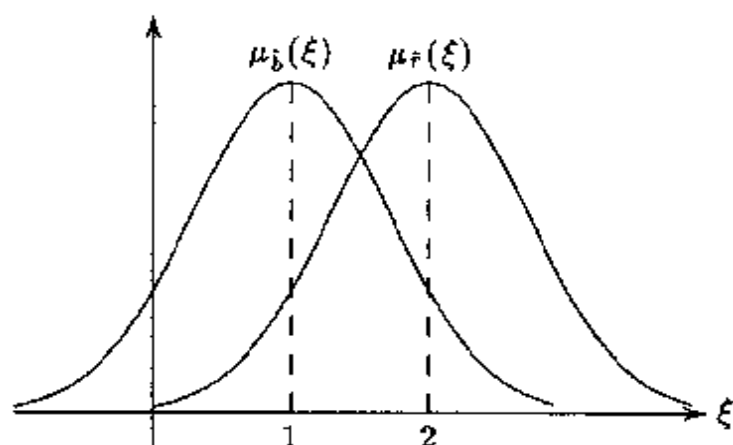


图 3.4 隶属函数为指数函数的两个模糊数

予以考虑. 从区间 $[0, 3]$ 中随机生成两个清晰向量 r 和 b . 若 $r > b$, 则交换它们的值使得 $r \leq b$. 置 $p = \min \{\mu_{\tilde{r}}(r), \mu_{\tilde{b}}(b)\}$. 再从区间 $[0, 3]$ 中重新生成两个清晰向量 r 和 b . 若 $r > b$, 则交换它们的值使得 $r \leq b$. 若 $p < \min \{\mu_{\tilde{r}}(r), \mu_{\tilde{b}}(b)\}$, 则置 $p = \min \{\mu_{\tilde{r}}(r), \mu_{\tilde{b}}(b)\}$. 重复以上过程直到给定次数完成为止. 最后, 得到的值 p 作为可能性 $\text{Pos}\{\tilde{r} \leq \tilde{b}\}$ 的估计.

经过 3000 次循环模拟, 所得的可能性的值为 0.760, 比较接近准确的值 0.778, 其相对误差不超过 2%. 这在实际决策问题中, 此精度已经足够了. 模拟结果如图 3.5.

例 2 考虑下面的更复杂的例子, 即计算可能性

$$\text{Pos} \left\{ \tilde{a}^2 + \tilde{b} \cdot \tilde{c} - \tilde{d}^{-1} \geq 4 \right\},$$

其中 \tilde{a} 和 \tilde{b} 为模糊数, 其隶属函数分别为

$$\mu_{\tilde{a}}(\xi) = \exp[-\xi^2], \quad \mu_{\tilde{b}}(\xi) = \exp[-|\xi - 2|],$$

\tilde{c} 是一个梯形模糊数 $(-1, 1, 2, 3)$, \tilde{d} 是三角模糊数 $(1, 2, 3)$, 取包含模糊数 $\tilde{a}, \tilde{b}, \tilde{c}$ 和 \tilde{d} 的 α 水平截集的超几何体分别为 $\mathcal{A} = [-1, 1]$, $\mathcal{B} = [1, 3]$, $\mathcal{C} = [-1, 3]$ 和 $\mathcal{D} = [1, 3]$. 从超几何体 $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ 中

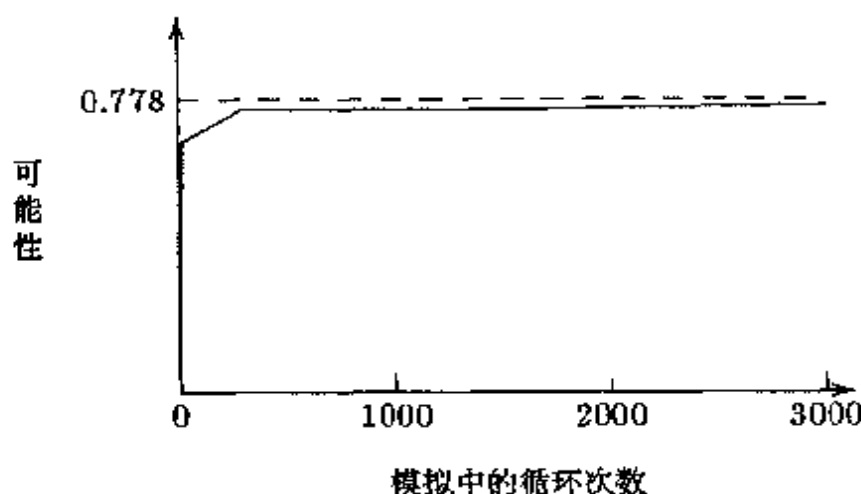


图 3.5 例 1 的模糊模拟

分别随机地生成四个清晰数 a, b, c, d . 若 $a^2 + b \cdot c - d^{-1} \geq 4$, 则置 $p = \min \{ \mu_{\tilde{a}}(a), \mu_{\tilde{b}}(b), \mu_{\tilde{c}}(c), \mu_{\tilde{d}}(d) \}$; 否则置 $p = 0$. 再从超几何体 $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ 中分别重新随机地生成四个清晰数 a, b, c, d . 若 $a^2 + b \cdot c - d^{-1} \geq 4$ 且 $p < \min \{ \mu_{\tilde{a}}(a), \mu_{\tilde{b}}(b), \mu_{\tilde{c}}(c), \mu_{\tilde{d}}(d) \}$, 则置 $p = \min \{ \mu_{\tilde{a}}(a), \mu_{\tilde{b}}(b), \mu_{\tilde{c}}(c), \mu_{\tilde{d}}(d) \}$. 重复以上过程直到完成给定次数为止. 所得值 p 作为可能性 $\text{Pos}\{\tilde{a}^2 + \tilde{b} \cdot \tilde{c} - \tilde{d}^{-1} \geq 4\}$ 的估计.

经过 5000 次循环以后, 得到

$$\text{Pos}\{\tilde{a}^2 + \tilde{b} \cdot \tilde{c} - \tilde{d}^{-1} \geq 4\} = 0.874,$$

它与准确值 0.908 相对误差不超过 3%.

例 3 假设 \tilde{a} 为三角模糊数 $(1, 2, 3)$, \tilde{b} 和 \tilde{c} 的隶属函数分别为

$$\mu_{\tilde{b}}(x) = \frac{1}{1 + (x - 3)^2}, \quad \mu_{\tilde{c}}(x) = \exp[-(x - 5)^2],$$

我们将由模糊模拟技术寻找最大的 f 使得

$$\text{Pos}\{\tilde{a} + (\tilde{b} - \tilde{c})^2 \geq f\} \geq 0.8.$$

已知 f 的最大值为 11.035. 经过 1000 次循环以后, 模糊模拟得到的估计值与已知值的相对误差小于 2%.

第 4 章

期望值模型

在期望约束下,使目标函数的期望值达到最优的数学规划,称为期望值模型.期望值模型是数学规划中常见的形式之一,如期望费用极小化问题、期望效益极大化问题等等.

首先考虑报童问题.报童需要每天提前到邮局订购报纸并确定所订购的报纸数量 x 份,每份价格为 c 元.已经知道每份报纸的售价为 a 元.如果报童没有卖完当天的报纸,则回收中心以极低的价格 b 元回收报纸.值设每天报纸的需求量为 ξ ,若 $x > \xi$,则每天报纸的剩余量为 $x - \xi$,否则为 0.这样报童的收益为

$$f(x, \xi) = \begin{cases} (a - c)x, & x \leq \xi \\ (b - c)x + (a - b)\xi, & x > \xi. \end{cases} \quad (4.1)$$

在实际问题中,报纸的需求量 ξ 通常是随机变量,从而导致效益函数 $f(x, \xi)$ 也是随机变量.既然不能准确地预测出订购 x 份报纸的实际收益,一个自然的方法就是考虑期望收益

$$\begin{aligned} E[f(x, \xi)] &= \int_0^x [(b - c)x + (a - b)\xi] \phi(\xi) d\xi \\ &\quad + \int_x^{+\infty} (a - c)x \phi(\xi) d\xi, \end{aligned} \quad (4.2)$$

其中 E 表示期望值算子, $\phi(\xi)$ 表示需求量 ξ 的概率密度函数.报童问题就是寻找最优的订购数量 x 使期望收益 $E[f(x, \xi)]$ 达到最大值.这是一个典型的期望值模型.

在本章里,主要介绍单目标期望值模型、多目标期望值模型以及期望值目标规划模型.同时也讨论一些期望值模型的数学性

质. 为了求解这些模型, 设计了基于随机模拟的遗传算法, 并给出了一些数值例子.

4.1 期望值算子

假设 t 维随机向量 ξ 的概率密度函数为 $\phi(\xi)$, 则随机向量 ξ 的期望值定义为

$$E[\xi] = \int_{\mathbb{R}^t} \xi \phi(\xi) d\xi. \quad (4.3)$$

通常也称其为均值.

设 f 为定义在 \mathbb{R}^t 上的实函数, 则 $f(\xi)$ 是一个随机量, 其期望值 $E[f(\xi)]$ 可以通过

$$E[f(\xi)] = \int_{\mathbb{R}^t} f(\xi) \phi(\xi) d\xi \quad (4.4)$$

计算.

下面是期望值算子的一些基本性质. 若 $\eta = a\xi + b$, 其中 a 和 b 是常数, 则

$$E[\eta] = aE[\xi] + b. \quad (4.5)$$

更一般的情况, 设 $\xi_1, \xi_2, \dots, \xi_n$ 是 n 个随机向量, 且期望值 $E[\xi_i]$ ($i = 1, 2, \dots, n$) 存在, 则有

$$E[\xi_1 + \xi_2 + \dots + \xi_n] = E[\xi_1] + E[\xi_2] + \dots + E[\xi_n]. \quad (4.6)$$

若 $\xi_1, \xi_2, \dots, \xi_n$ 是相互独立的, 则有

$$E[\xi_1 \cdot \xi_2 \cdots \xi_n] = E[\xi_1] \cdot E[\xi_2] \cdots E[\xi_n]. \quad (4.7)$$

4.2 期望值模型

单目标期望值模型的一般形式可以表示如下

$$\begin{cases} \max E[f(\mathbf{x}, \xi)] \\ \text{s.t.} \\ E[g_j(\mathbf{x}, \xi)] \leq 0, \quad j = 1, 2, \dots, p \\ E[h_k(\mathbf{x}, \xi)] = 0, \quad k = 1, 2, \dots, q, \end{cases} \quad (4.8)$$

其中 \mathbf{x} 是一个 n 维决策向量, ξ 是一个 t 维随机向量, 其概率密度函数为 $\phi(\xi)$, $f(\mathbf{x}, \xi)$ 是目标函数, $g_j(\mathbf{x}, \xi)$ 和 $h_k(\mathbf{x}, \xi)$ 是随机约束函数, $j = 1, 2, \dots, p$, $k = 1, 2, \dots, q$, E 表示期望值算子, 从而有

$$\begin{aligned} E[f(\mathbf{x}, \xi)] &= \int_{\mathbb{R}^t} f(\mathbf{x}, \xi) \phi(\xi) d\xi, \\ E[g_j(\mathbf{x}, \xi)] &= \int_{\mathbb{R}^t} g_j(\mathbf{x}, \xi) \phi(\xi) d\xi, \quad j = 1, 2, \dots, p, \\ E[h_k(\mathbf{x}, \xi)] &= \int_{\mathbb{R}^t} h_k(\mathbf{x}, \xi) \phi(\xi) d\xi, \quad k = 1, 2, \dots, q. \end{aligned} \quad (4.9)$$

若 ξ 是一个离散型随机变量且分布函数为 $\Pr(\xi = \xi_i) = \theta_i$, $i \in I$, 其中 I 序号集, 则有

$$\begin{aligned} E[f(\mathbf{x}, \xi)] &= \sum_{i \in I} \theta_i f(\mathbf{x}, \xi_i), \\ E[g_j(\mathbf{x}, \xi)] &= \sum_{i \in I} \theta_i g_j(\mathbf{x}, \xi_i), \quad j = 1, 2, \dots, p, \\ E[h_k(\mathbf{x}, \xi)] &= \sum_{i \in I} \theta_i h_k(\mathbf{x}, \xi_i), \quad k = 1, 2, \dots, q. \end{aligned} \quad (4.10)$$

因此, 一个解 \mathbf{x} 是可行的当且仅当 $E[g_j(\mathbf{x}, \xi)] \leq 0$, $j = 1, 2, \dots, p$ 和 $E[h_k(\mathbf{x}, \xi)] = 0$, $k = 1, 2, \dots, q$. 一个可行解 \mathbf{x}^* 是期望值模型

(4.8) 的最优解, 如果对任意的可行解 \mathbf{x} , $E[f(\mathbf{x}^*, \xi)] \geq E[f(\mathbf{x}, \xi)]$ 成立.

作为单目标期望值模型的推广, 多目标期望值模型可以写成如下形式

$$\begin{cases} \max [E[f_1(\mathbf{x}, \xi)], E[f_2(\mathbf{x}, \xi)], \dots, E[f_m(\mathbf{x}, \xi)]] \\ \text{s.t.} \\ E[g_j(\mathbf{x}, \xi)] \leq 0, \quad j = 1, 2, \dots, p \\ E[h_k(\mathbf{x}, \xi)] = 0, \quad k = 1, 2, \dots, q, \end{cases} \quad (4.11)$$

其中 $f_i(\mathbf{x}, \xi)$, $i = 1, 2, \dots, m$ 是目标函数.

根据决策者给定的优先结构和目标值, 也可以把一个随机决策系统转化为一个期望值目标规划.

$$\begin{cases} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ E[f_i(\mathbf{x}, \xi)] + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ E[g_j(\mathbf{x}, \xi)] \leq 0, \quad j = 1, 2, \dots, p \\ E[h_k(\mathbf{x}, \xi)] = 0, \quad k = 1, 2, \dots, q, \end{cases} \quad (4.12)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差, 定义为

$$d_i^+ = \begin{cases} E[f_i(\mathbf{x}, \xi)] - b_i, & E[f_i(\mathbf{x}, \xi)] > b_i \\ 0, & E[f_i(\mathbf{x}, \xi)] \leq b_i \end{cases}$$

d_i^- = 目标 i 偏离目标值的负偏差, 定义为

$$d_i^- = \begin{cases} 0, & E[f_i(\mathbf{x}, \xi)] \geq b_i \\ b_i - E[f_i(\mathbf{x}, \xi)], & E[f_i(\mathbf{x}, \xi)] < b_i, \end{cases}$$

f_i = 目标约束中的函数,

g_j, h_k = 系统约束中的函数,

b_i = 目标 i 的目标值,

ξ = 随机向量,

l = 优先级个数,

m = 目标约束个数,

p = 系统约束个数.

4.3 凸 性

凸性是优化理论中经常讨论的课题. 如果一个数学规划模型的目标函数和可行集均为凸的, 则称该模型为凸规划. 对期望值模型, 在凸性方面有如下的结论.

定理4.1 假设对每个给定的 ξ , 函数 $f(\mathbf{x}, \xi)$, $g_j(\mathbf{x}, \xi)$ 关于 \mathbf{x} 是凸的, $j = 1, 2, \dots, p$, 并且函数 $h_k(\mathbf{x}, \xi)$, $k = 1, 2, \dots, q$ 关于 \mathbf{x} 是线性的, 则期望值模型

$$\begin{cases} \min E[f(\mathbf{x}, \xi)] \\ \text{s.t.} \\ E[g_j(\mathbf{x}, \xi)] \leq 0, \quad j = 1, 2, \dots, p \\ E[h_k(\mathbf{x}, \xi)] = 0, \quad k = 1, 2, \dots, q \end{cases} \quad (4.13)$$

是凸规划.

证明 对每一个固定的 ξ , 函数 $f(\mathbf{x}, \xi)$ 关于 \mathbf{x} 是凸的, 从而对

任意给定的解 $\mathbf{x}_1, \mathbf{x}_2$ 和纯量 $\lambda \in [0, 1]$, 有

$$f(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi) \leq \lambda f(\mathbf{x}_1, \xi) + (1 - \lambda) f(\mathbf{x}_2, \xi).$$

由期望值算子的定义, 有

$$E[f(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi)] \leq \lambda E[f(\mathbf{x}_1, \xi)] + (1 - \lambda) E[f(\mathbf{x}_2, \xi)].$$

因此目标函数 $E[f(\mathbf{x}, \xi)]$ 关于 \mathbf{x} 也是凸的.

考虑约束条件

$$\begin{cases} E[g_j(\mathbf{x}, \xi)] \leq 0, & j = 1, 2, \dots, p \\ E[h_k(\mathbf{x}, \xi)] = 0, & k = 1, 2, \dots, q, \end{cases}$$

下面证明可行集的凸性, 即证明对任意可行解 \mathbf{x}_1 和 \mathbf{x}_2 , 及纯量 $\lambda \in [0, 1]$, $\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2$ 还是可行解. 根据函数 $g_j(\mathbf{x}, \xi)$, $j = 1, 2, \dots, p$ 的凸性, 我们知道

$$g_j(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi) \leq \lambda g_j(\mathbf{x}_1, \xi) + (1 - \lambda) g_j(\mathbf{x}_2, \xi), \quad j = 1, 2, \dots, p,$$

从而有

$$\begin{aligned} E[g_j(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi)] \\ \leq \lambda E[g_j(\mathbf{x}_1, \xi)] + (1 - \lambda) E[g_j(\mathbf{x}_2, \xi)] \leq 0, \end{aligned} \quad (4.14)$$

$j = 1, 2, \dots, p$. 由于函数 $h_k(\mathbf{x}, \xi)$, $k = 1, 2, \dots, q$ 是线性的, 有

$$h_k(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi) = \lambda h_k(\mathbf{x}_1, \xi) + (1 - \lambda) h_k(\mathbf{x}_2, \xi), \quad k = 1, 2, \dots, q.$$

这意味着

$$\begin{aligned} E[h_k(\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2, \xi)] \\ = \lambda E[h_k(\mathbf{x}_1, \xi)] + (1 - \lambda) E[h_k(\mathbf{x}_2, \xi)] = 0, \end{aligned} \quad (4.15)$$

$k = 1, 2, \dots, q$. 由式 (4.14) 和 (4.15), 可知 $\lambda \mathbf{x}_1 + (1 - \lambda) \mathbf{x}_2$ 是可行解. 因此可行集是凸的.

从而证明期望值模型是凸规划. \square

4.4 补 偿 模 型

对随机决策问题, 很多情况下, 可以先制定一个初始决策 x , 使目标函数 $f(x, \xi)$ 极小化, 待随机向量 ξ 实现之后, 仍有机会采取应急策略 y . 这就导致一个额外费用 $Q(x, \xi)$, 通常称为补偿函数. 这样, 有带有补偿函数的随机规划

$$\min_x E_{\xi} [f(x, \xi) + Q(x, \xi)]. \quad (4.16)$$

显然, 这是一个特殊的期望值模型, 其中 $Q(x, \xi)$ 可以表示为

$$Q(x, \xi) = \min_y \{q(y) \mid G_j(y|x, \xi) \leq 0, j = 1, 2, \dots, p\}, \quad (4.17)$$

这里 $q(y)$ 是一个费用函数, $G_j(y|x, \xi)$ 是给定决策 x 和参数 ξ 实现后关于应急策略 y 的一个约束函数, $j = 1, 2, \dots, p$.

补偿模型也可以看成是一个二阶段动态规划.

4.5 基于随机模拟的遗传算法

从数学观点来看, 确定性优化问题和期望值模型并没有什么区别, 唯一的区别在于后者存在多重积分. 本节里, 介绍基于随机模拟的遗传算法, 并给出一些数值例子来解释这种算法.

我们采用随机模拟技术来计算多重积分. 例如, 为计算积分

$$\int_{\mathbb{R}^t} f(x, \xi) \phi(\xi) d\xi,$$

设 $\xi_1, \xi_2, \dots, \xi_N$ 是从概率密度函数为 $\phi(\xi)$ 中抽出的向量, 那么

$$\frac{1}{N} \sum_{i=1}^N f(x, \xi_i)$$

是该积分的一个估计, 而误差为 $O(N^{-\frac{1}{2}})$. 与经典迭代方法求多重积分相比, 为达到给定精度的抽样次数与维数无关. 因此, 随机模拟技术更适合维数较高的积分.

在基于随机模拟的遗传算法中, 初始过程、选择过程、交叉过程和变异过程与第2章所描述的方法基本相同, 唯一的差异是在期望值模型中使用随机模拟技术计算积分.

遗传算法

步骤 0 输入参数 pop_size , P_c 和 P_m ;

步骤 1 初始产生 pop_size 个染色体, 其中可能使用随机模拟技术计算约束函数中的多重积分;

步骤 2 对染色体进行交叉和变异运算, 其中可能使用随机模拟技术检验后代的可行性;

步骤 3 使用随机模拟技术计算所有染色体的目标值;

步骤 4 根据目标值, 使用基于序的评价函数计算适应度;

步骤 5 旋转赌轮, 选择染色体;

步骤 6 重复步骤 2 到步骤 5, 直到完成给定的循环次数;

步骤 7 找出最好的染色体作为最优解.

求解期望值模型的基于随机模拟的遗传算法已写成 C 语言程序. 下面通过一些例子证实遗传算法的有数性, 所得结果是成功的. 这些例子都是在 PC 机上实现的, 所使用的参数为: 种群规模为 30, 交叉概率为 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 基于序的评价函数中的参数 $\alpha = 0.05$.

例 1 考虑本章开始所讨论的报童问题. 假设每份报纸售价 $a = 6$ 元, 回收中心的回收价格 $b = 0.2$ 元, 订购价格 $c = 4$ 元, 需求量 ξ 服从正态分布 $N(100, 100)$. 对遗传算法来说, 此例比较简单. 经过 100 代循环以后, 基于随机模拟的遗传算法给出的最优解为

$$x^* = 96,$$

而期望收益为 178.76 元.

例 2 考虑随机规划问题^[47],

$$\left\{ \begin{array}{l} \max f(\mathbf{x}) = \int_{\mathbb{R}^3} [(x_1 - \xi_1) \sin(\pi x_1) + (x_2 - \xi_2) \sin(4\pi x_2) \\ \quad + (x_3 - \xi_3) \sin(10\pi x_3)] \phi(\xi_1, \xi_2, \xi_3) d\xi_1 d\xi_2 d\xi_3 \\ \text{s.t.} \\ 0 \leq x_1 \leq 5, \quad 0 \leq x_2 \leq 5, \quad 0 \leq x_3 \leq 5, \end{array} \right.$$

其中 $\phi(\xi_1, \xi_2, \xi_3)$ 为联合正态密度函数, 即

$$\phi(\xi_1, \xi_2, \xi_3) = (2\pi)^{-\frac{3}{2}} \|C\|^{\frac{1}{2}} \exp \left\{ -\frac{1}{2} \sum_{i=1}^3 \sum_{j=1}^3 c_{ij} (\xi_i - u_i)(\xi_j - u_j) \right\},$$

而 $u_1 = 1, u_2 = 2, u_3 = 3, C = (c_{ij})$ 为正定矩阵,

$$c_{ij} = \begin{cases} 2, & i = j \\ 1, & i \neq j, \end{cases} \quad i = 1, 2, 3, j = 1, 2, 3,$$

$\|C\|$ 表示 C 的行列式.

经过 1000 次迭代以后, 基于随机模拟的遗传算法给出的最好的解为 8.0146; 经过 2000 次迭代以后, 所得的最好的解为 8.9584; 经过 2500 迭代以后, 最好的解为 8.9747; 经过 3000 次迭代以后, 所得的最好的解为 8.9906, 且

$$x_1^* = 4.5065, \quad x_2^* = 4.6279, \quad x_3^* = 0.1459.$$

这个例子所花费的时间包括随机模拟所花费的时间和遗传算法所花费的时间, 其中随机模拟所花费的时间与抽样个数成正比. 经 1000 次迭代, 用于遗传算法所花费的 CPU 时间为 58.3 秒.

4.6 注

期望值模型的确是随机优化问题中常用的且有效的方法，但我们并不总是关心极大化期望值效益问题或极小化期望值费用问题。实际上，有时可能更要考虑所谓的风险问题。

给定两种不同的投资方案，它们期望效益相同，而风险不同。一些人（称为喜好风险者）可能为追求最大效益而选择风险较大的方案，而另一些人（称为厌恶风险者）可能为躲避风险而选择风险较小的投资方案。也可能有些人不太在乎风险，认为哪一种方案都可以接受，这也是期望值模型的理论基础。

也有些情况，使用期望值模型显得不太合理。例如，众所周知，许多人喜欢购买彩票，显然彩票的期望值总是负的。若根据期望值模型的结果，人们不应参与购买彩票活动。而实际上，恰恰相反，造成这种情况的主要原因是较小的付出可能获得巨大的收益（尽管可能性是如此之小）。

继期望值模型之后，本书的后面部分将介绍机会约束规划和相关机会规划。

第 5 章

机会约束规划

作为第二种随机规划, 机会约束规划 (Chance Constrained Programming) 由 Charnes 和 Cooper^[20] 提出, 主要针对约束条件中含有随机变量, 且必须在观测到随机变量的实现之前作出决策的情况. 考虑到所做决策在不利情况发生时可能不满足约束条件, 而采取一种原则: 即允许所做决策在一定程度上不满足约束条件, 但该决策应使约束条件成立的概率不小于某一置信水平 α .

求解机会约束规划的传统方法是根据事先给定的置信水平, 把机会约束转化为各自的确定等价类, 然后用传统的方法求解其等价的确定性模型. 对一些特殊情况, 机会约束规划问题确实可以转化为确定性数学规划问题, 但对较复杂的机会约束规划问题, 通常很难作到这一点. 然而, 随着计算机的高速发展, 一些革新算法如遗传算法的提出, 使得复杂的机会约束规划可以不必通过转化为确定性数学规划而直接得到解决.

机会约束规划也可以推广为机会约束目标规划和机会约束多目标规划.

本章里我们主要介绍机会约束规划的一般理论, 提供机会约束规划的使用范围, 并列举一些已经知道的确定等价类, 同时设计出一个求解机会约束规划的基于随机模拟的遗传算法, 以及提供一些数值例子.

5.1 机会约束规划模型

考虑带有随机参数的数学规划模型,

$$\begin{cases} \max f(\mathbf{x}, \boldsymbol{\xi}) \\ \text{s.t.} \\ g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (5.1)$$

其中 \mathbf{x} 是一个 n 维决策向量, $\boldsymbol{\xi}$ 是一个随机向量, $f(\mathbf{x}, \boldsymbol{\xi})$ 是目标函数, $g_j(\mathbf{x}, \boldsymbol{\xi})$ 是随机约束函数, $j = 1, 2, \dots, p$. 实际上, 这个数学规划模型是没有定义的, 因为随机变量 $\boldsymbol{\xi}$ 使得 \max 和约束条件的意义并不明确.

一种有意义的随机规划是如下的机会约束规划模型 (CCP):

$$\begin{cases} \max \bar{f} \\ \text{s.t.} \\ \Pr \{f(\mathbf{x}, \boldsymbol{\xi}) \geq \bar{f}\} \geq \beta \\ \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (5.2)$$

其中 $\Pr\{\cdot\}$ 表示 $\{\cdot\}$ 中的事件成立的概率, α 和 β 分别是事先给定的约束条件和目标函数的置信水平.

一个点 \mathbf{x} 是可行的当且仅当事件 $\{\boldsymbol{\xi} | g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\}$ 的概率测度不小于 α , 即违反约束条件的概率小于 $(1 - \alpha)$.

无论何种随机参数 $\boldsymbol{\xi}$ 和何种函数形式 f , 对每一个给定的决策 \mathbf{x} , $f(\mathbf{x}, \boldsymbol{\xi})$ 是随机变量, 其概率密度函数用 $\phi_{f(\mathbf{x}, \boldsymbol{\xi})}(f)$ 表示, 这样可能有多个 \bar{f} 使得 $\Pr \{f(\mathbf{x}, \boldsymbol{\xi}) \geq \bar{f}\} \geq \beta$ 成立. 从极大化目标值 \bar{f} 的观点来看, 我们所需要的目标值 \bar{f} 应该是目标函数 $f(\mathbf{x}, \boldsymbol{\xi})$ 在保证置信水平至少是 β 时所取的最大值, 即

$$\bar{f} = \max \{f \mid \Pr \{f(\mathbf{x}, \boldsymbol{\xi}) \geq f\} \geq \beta\}. \quad (5.3)$$

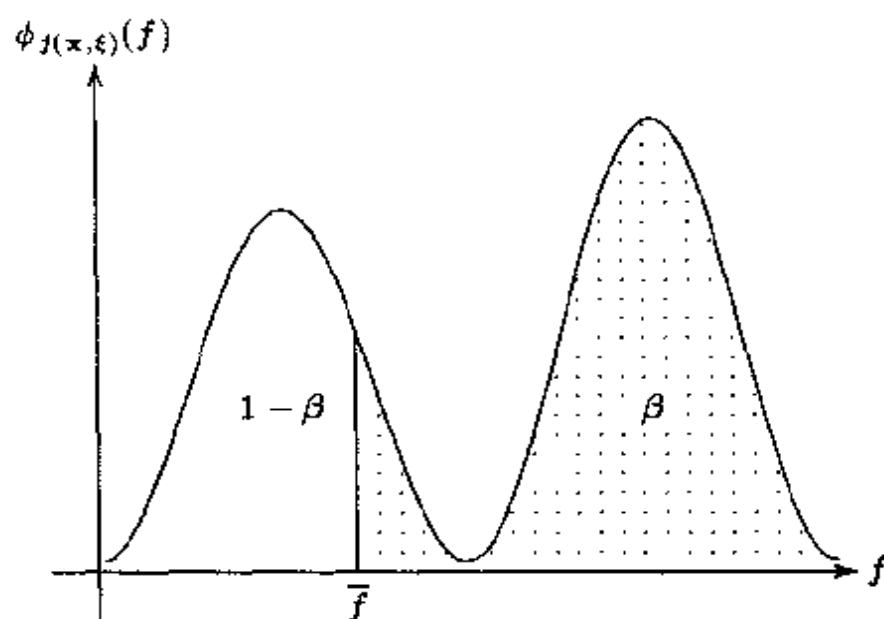


图 5.1 概率密度函数 $\phi_{J(x, \xi)}(f)$ 和目标值 \bar{f}

如图 5.1 所示.

机会约束规划 CCP(5.2) 中的概率约束称为联合机会约束. 另外, 联合机会约束有时可以分成几个独立的机会约束, 如

$$\Pr\{g_j(x, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \quad (5.4)$$

其中 α_j 是给定的置信水平, $j = 1, 2, \dots, p$. 更一般的情况是, 我们也可以采用混合机会约束

$$\begin{cases} \Pr\{g_j(x, \xi) \leq 0, j = 1, 2, \dots, k_1\} \geq \alpha_1 \\ \Pr\{g_j(x, \xi) \leq 0, j = k_1 + 1, k_1 + 2, \dots, k_2\} \geq \alpha_2 \\ \dots \\ \Pr\{g_j(x, \xi) \leq 0, j = k_{t-1} + 1, k_{t-1} + 2, \dots, p\} \geq \alpha_t, \end{cases} \quad (5.5)$$

其中 $1 \leq k_1 < k_2 < \dots < k_{t-1} < p$.

类似地, 对极小化问题, 有

$$\begin{cases} \min \bar{f} \\ \text{s.t.} \\ \Pr \{f(\mathbf{x}, \boldsymbol{\xi}) \leq \bar{f}\} \geq \beta \\ \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\} \geq \alpha. \end{cases} \quad (5.6)$$

其中 \bar{f} 是目标函数 $f(\mathbf{x}, \boldsymbol{\xi})$ 在置信水平为 β 时所取的最小值.

作为单目标机会约束规划的推广, 多目标机会约束规划可以表示成如下形式:

$$\begin{cases} \max [\bar{f}_1, \bar{f}_2, \dots, \bar{f}_m] \\ \text{s.t.} \\ \Pr \{f_i(\mathbf{x}, \boldsymbol{\xi}) \geq \bar{f}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (5.7)$$

其中 α_j 和 β_i 分别是第 j 个约束和第 i 个目标的置信水平, \bar{f}_i 分别是目标函数 $f_i(\mathbf{x}, \boldsymbol{\xi})$ 在概率水平至少为 β_i 时所取的是大值, $j = 1, 2, \dots, p, i = 1, 2, \dots, m$.

根据决策者给定的优先结构和目标值, 我们也可以为随机决策系统构造如下的机会约束目标规划

$$\begin{cases} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ \Pr \{f_i(\mathbf{x}, \boldsymbol{\xi}) + d_i^- - d_i^+ = b_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{cases} \quad (5.8)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

w_{ij} = 对应优先因子 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

f_i = 目标约束中的函数,

g_j = 系统约束中的函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

p = 系统约束个数.

注 1 对每一个固定的可行解 x , 即满足条件 $\Pr\{g_j(x, \xi) \leq 0\} \geq \alpha_j, j = 1, 2, \dots, p$ 的 x , 由于 ξ 是一个随机变量, 所以函数 $f_i(x, \xi)$ 也是一个随机变量. 对于目标约束 $\Pr\{f_i(x, \xi) + d_i^- - d_i^+ = b_i\} \geq \beta_i$. 正偏差 d_i^+ 和负偏差 d_i^- 将被分别理解为满足

$$\begin{aligned} \Pr\{d_i^- \geq b_i - f_i(x, \xi)\} &\geq \beta_i, \\ \Pr\{d_i^+ \geq f_i(x, \xi) - b_i\} &\geq \beta_i, \end{aligned} \quad (5.9)$$

$i = 1, 2, \dots, m$ 时所取得的最小非负值 d_i^+ 和 d_i^- .

注 2 在确定性目标规划中, d_i^- 和 d_i^+ 中至多有一个为正, 但对机会约束目标规划, d_i^- 和 d_i^+ 有可能都是正的.

注 3 一般情况下, 机会约束规划不一定是凸的, 除非附加一定条件.

注 4 有时决策变量 (或其中一部分) 可能要求是整数, 这种机会约束规划称为机会约束整数规划 (或机会约束混合整数规划).

5.2 确定性等价类

传统的处理机会约束规划的方法是把机会约束规划转化为它们各自的等价类. 我们知道, 对一些较复杂的问题通常很难作到这一点, 只有在一些特殊情况下才能做到. 我们将考虑下述形式的机会约束

$$\Pr \{g(\mathbf{x}, \xi) \leq 0\} \geq \alpha. \quad (5.10)$$

显然机会约束 (5.4) 是这种形式的组合. 若定义 $g(\mathbf{x}, \xi) = \bar{f} - f(\mathbf{x}, \xi)$, 则机会目标约束 $\Pr\{f(\mathbf{x}, \xi) \geq \bar{f}\} \geq \beta$ 也与这种形式吻合. 既然目标约束 $\Pr\{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \beta$ 可以理解为 $\Pr\{d^- \geq b - f(\mathbf{x}, \xi)\} \geq \beta$ 和 $\Pr\{d^+ \geq f(\mathbf{x}, \xi) - b\} \geq \beta$, 通过定义 $g(\mathbf{x}, \xi) = b - f(\mathbf{x}, \xi) - d^-$ 和 $g(\mathbf{x}, \xi) = f(\mathbf{x}, \xi) - b - d^+$, 可知前者 and 后者均与形式 (5.10) 一致. 本节将总结一些已知的结果.

情况 I 假设函数 $g(\mathbf{x}, \xi)$ 的形式为 $g(\mathbf{x}, \xi) = h(\mathbf{x}) - \xi$, 则机会约束 (5.10) 可以表示成如下形式:

$$\Pr \{h(\mathbf{x}) \leq \xi\} \geq \alpha, \quad (5.11)$$

其中 $h(\mathbf{x})$ 是决策向量 \mathbf{x} 的线性或非线性函数, ξ 是随机变量, 概率分布函数为 $\Phi(\cdot)$.

显然, 对每一个给定的置信水平 $\alpha (0 \leq \alpha \leq 1)$, 必存在一个数 K_α (可能多个或 ∞) 使得

$$\Pr \{K_\alpha \leq \xi\} = \alpha. \quad (5.12)$$

若用一个较小的数代替 K_α , 则左端的概率将变大. 因此

$$\Pr \{h(\mathbf{x}) \leq \xi\} \geq \alpha$$

当且仅当 $h(\mathbf{x}) \leq K_\alpha$.

注意到等式 $\Pr\{K_\alpha \leq \xi\} = 1 - \Phi(K_\alpha)$ 总是成立, 由 (5.12), 有

$$K_\alpha = \Phi^{-1}(1 - \alpha),$$

其中 Φ^{-1} 是 Φ 的逆函数. 有时, 等式 (5.12) 的解不唯一, 等价地, 函数 Φ^{-1} 是多值的. 对这种情况, 我们将选择最大的那一个, 即

$$K_\alpha = \sup\{K \mid K = \Phi^{-1}(1 - \alpha)\}.$$

这样就可以得到 (5.11) 的等价类, 其形式如下:

$$h(\mathbf{x}) \leq K_\alpha. \quad (5.13)$$

例如, 考虑机会约束

$$\begin{cases} \Pr\{3x_1 + 4x_2 \leq \xi_1\} \geq 0.80 \\ \Pr\{x_1^2 - x_2^3 \leq \xi_2\} \geq 0.90, \end{cases} \quad (5.14)$$

其中 ξ_1 是指数分布 $\mathcal{E}\mathcal{X}\mathcal{P}(2)$, 其概率分布记为 Φ_1 , ξ_2 是正态分布变量 $\mathcal{N}(2, 1)$, 其概率分布记为 Φ_2 . 则由公式 (5.13), 我们知道机会约束 (5.14) 等价于

$$\begin{cases} 3x_1 + 4x_2 \leq \Phi_1^{-1}(1 - 0.80) = 0.446 \\ x_1^2 - x_2^3 \leq \Phi_2^{-1}(1 - 0.90) = 0.719. \end{cases} \quad (5.15)$$

情况 II 类似于情况 I, 假设函数 $g(\mathbf{x}, \xi) = \xi - h(\mathbf{x})$, 则机会约束 (5.10) 可以写成如下形式

$$\Pr\{h(\mathbf{x}) \geq \xi\} \geq \alpha, \quad (5.16)$$

其中 $h(\mathbf{x})$ 是决策向量 \mathbf{x} 的 (线性或非线性) 函数, ξ 是随机变量, 概率分布函数为 $\Phi(\cdot)$.

显然, 对每一个给定的置信水平 $\alpha (0 \leq \alpha \leq 1)$, 存在一个数 K_α 使得

$$\Pr \{K_\alpha \geq \xi\} = \alpha. \quad (5.17)$$

若用一个较大的数代替 K_α , 左端的概率将变大. 因此,

$$\Pr \{h(\mathbf{x}) \geq \xi\} \geq \alpha$$

当且仅当 $h(\mathbf{x}) \geq K_\alpha$.

注意到等式 $\Pr \{K_\alpha \geq \xi\} = \Phi(K_\alpha)$ 总是成立, 因此, 我们有

$$K_\alpha = \Phi^{-1}(\alpha)$$

其中 Φ^{-1} 是函数 Φ 的逆函数. 有时, 等式 (5.17) 的解不唯一, 等价地, 函数 Φ^{-1} 是多值的. 对这种情况, 我们将选择最小的那一个, 即

$$K_\alpha = \inf \{K \mid K = \Phi^{-1}(\alpha)\}.$$

这样可以得到 (5.16) 的如下形式的等价类, 其形式如下:

$$h(\mathbf{x}) \geq K_\alpha. \quad (5.18)$$

情况 III 假设函数 $g(\mathbf{x}, \xi)$ 有形式

$$g(\mathbf{x}, \xi) = a_1 x_1 + a_2 x_2 + \cdots + a_n x_n - b,$$

其中 $\xi = (a_1, a_2, \cdots, a_n, b)$, a_i 和 b 为相互独立的正态随机变量, $i = 1, 2, \cdots, n$. 于是机会约束 (5.10) 可以写成如下形式

$$\Pr \left\{ \sum_{i=1}^n a_i x_i \leq b \right\} \geq \alpha. \quad (5.19)$$

则函数

$$y(\mathbf{x}) = \sum_{i=1}^n a_i x_i - b$$

也是正态随机变量, 且有

$$\begin{aligned} E(y(\mathbf{x})) &= \sum_{i=1}^n E(a_i)x_i - E(b), \\ V(y(\mathbf{x})) &= \sum_{i=1}^n V(a_i)x_i^2 + V(b), \end{aligned}$$

其中 $E(\cdot)$ 和 $V(\cdot)$ 分别为期望值和方差. 注意到变量

$$\frac{\sum_{i=1}^n a_i x_i - b - \left(\sum_{i=1}^n E(a_i)x_i - E(b) \right)}{\sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)}}$$

一定服从标准正态分布 $\mathcal{N}(0, 1)$, 而不等式 $\sum_{i=1}^n a_i x_i \leq b$ 等价于

$$\frac{\sum_{i=1}^n a_i x_i - b - \left(\sum_{i=1}^n E(a_i)x_i - E(b) \right)}{\sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)}} \leq - \frac{\sum_{i=1}^n E(a_i)x_i - E(b)}{\sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)}}.$$

因此, 机会约束 (5.19) 等价于

$$\Pr \left\{ \eta \leq - \frac{\sum_{i=1}^n E(a_i)x_i - E(b)}{\sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)}} \right\} \geq \alpha, \quad (5.20)$$

其中 η 服从标准正态分布, 其概率分布函数形式为

$$\Phi(\eta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\eta} \exp\left(-\frac{t^2}{2}\right) dt.$$

则机会约束 (5.20) 成立当且仅当

$$\Phi^{-1}(\alpha) \leq - \frac{\sum_{i=1}^n E(a_i)x_i - E(b)}{\sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)}}, \quad (5.21)$$

即

$$\sum_{i=1}^n E(a_i)x_i + \Phi^{-1}(\alpha) \sqrt{\sum_{i=1}^n V(a_i)x_i^2 + V(b)} \leq E(b). \quad (5.22)$$

这个约束就是机会约束 (5.19) 的等价类.

例如, 假设机会约束有如下形式

$$\Pr \{a_1x_1 + a_2x_2 + a_3x_3 \leq b\} \geq 95\%, \quad (5.23)$$

其中 a_1, a_2, a_3 和 b 分别服从正态分布 $\mathcal{N}(1, 1)$, $\mathcal{N}(2, 1)$, $\mathcal{N}(3, 1)$ 和 $\mathcal{N}(4, 1)$. 则由公式 (5.22) 和 $\Phi^{-1}(95\%) = 1.645$, 导出 (5.23) 的等价类为

$$x_1 + 2x_2 + 3x_3 + 1.645 \sqrt{x_1^2 + x_2^2 + x_3^2 + 1} \leq 4. \quad (5.24)$$

5.3 一些性质

本节里, 我们准备介绍机会约束规划的一些基本性质. 考虑如下的一个特殊的机会约束规划,

$$\begin{cases} \min E f(\mathbf{x}, \boldsymbol{\xi}) \\ \text{s.t.} \\ \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (5.25)$$

其中 \mathbf{x} 是一个决策向量, $\boldsymbol{\xi}$ 是一个随机向量, 其联合概率密度函数为 $\phi(\boldsymbol{\xi})$, E 是关于 $\boldsymbol{\xi}$ 的期望算子.

下面定义一个新的函数

$$h(\mathbf{x}, \xi) = \begin{cases} \alpha - 1, & \text{若 } g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p \\ \alpha, & \text{其它,} \end{cases} \quad (5.26)$$

即对每个固定的决策变量 \mathbf{x} , $h(\mathbf{x}, \xi)$ 在区域

$$\Xi = \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p\}$$

上为 $\alpha - 1$, 在其补集 Ξ^c 上为 α . 则由期望值定义, 有

$$\begin{aligned} & Eh(\mathbf{x}, \xi) \\ &= \int h(\mathbf{x}, \xi) \phi(\xi) d\xi \\ &= (\alpha - 1) \int_{\Xi} \phi(\xi) d\xi + \alpha \int_{\Xi^c} \phi(\xi) d\xi \\ &= (\alpha - 1) \Pr \left\{ \begin{array}{l} g_j(\mathbf{x}, \xi) \leq 0 \\ 1 \leq j \leq p \end{array} \right\} + \alpha \left[1 - \Pr \left\{ \begin{array}{l} g_j(\mathbf{x}, \xi) \leq 0 \\ 1 \leq j \leq p \end{array} \right\} \right] \\ &= \alpha - \Pr \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p\}. \end{aligned}$$

因此, $Eh(\mathbf{x}, \xi) \leq 0$ 当且仅当 (5.25) 中的机会约束成立. 从而为期望值模型和机会约束规划提供了一种联系.

定理5.1 机会约束规划

$$\begin{cases} \min Ef(\mathbf{x}, \xi) \\ \text{s.t.} \\ \Pr \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{cases}$$

等价于期望值模型

$$\begin{cases} \min Ef(\mathbf{x}, \xi) \\ \text{s.t.} \\ Eh(\mathbf{x}, \xi) \leq 0, \end{cases}$$

其中函数 $h(\mathbf{x}, \xi)$ 由 (5.26) 定义.

机会约束集合一般情况下是非凸的, 即使我们假定所有的函数 $g_j(\mathbf{x}, \xi)$, $j = 1, 2, \dots, p$ 都是凸的, 也不能保证机会约束集合的凸性. 为了保证机会约束规划的凸性, 我们需添加一些其它条件.

首先, 介绍一些概念. 一个实函数称为伪凹的, 如果对所有的 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ 和 $\lambda \in [0, 1]$, 有

$$f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{y}) \geq \min\{f(\mathbf{x}), f(\mathbf{y})\}. \quad (5.27)$$

对任意给定常数 c , \mathbf{x}_1 和 \mathbf{x}_2 为满足条件 $f(\mathbf{x}_1) \geq c$ 和 $f(\mathbf{x}_2) \geq c$ 的任意两点, 那么由函数 $f(\mathbf{x})$ 的伪凹性, 有 $f(\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2) \geq \min\{f(\mathbf{x}_1), f(\mathbf{x}_2)\} \geq c$ 对任意的 $\lambda \in [0, 1]$ 成立, 这意味着集合 $\{\mathbf{x} | f(\mathbf{x}) \geq c\}$ 对任意的 c 是凸的. 反之, 若集合 $\{\mathbf{x} | f(\mathbf{x}) \geq c\}$ 对任意的 c 是凸的, 那么对任意给定的两点 \mathbf{x}_1 和 \mathbf{x}_2 , 取 $c = \min\{f(\mathbf{x}_1), f(\mathbf{x}_2)\}$, 集合 $\{\mathbf{x} | f(\mathbf{x}) \geq c\}$ 也是凸的, 并且包含两点 \mathbf{x}_1 和 \mathbf{x}_2 , 所以对任意数 $\lambda \in [0, 1]$, 我们有 $\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2 \in \{\mathbf{x} | f(\mathbf{x}) \geq c\}$, 这意味着 $f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{y}) \geq \min\{f(\mathbf{x}), f(\mathbf{y})\}$. 因此, 函数 f 是伪凹的, 换句话说, 函数 f 的伪凹性等价于对任意数 c , 集合 $\{\mathbf{x} | f(\mathbf{x}) \geq c\}$ 是凸的.

定义在直线 \mathbb{R} 上的实函数 $f(x)$ 称为关于 v 单峰的, 如果当 x 远离 v 时, 函数 $f(x)$ 是非增的. 若维数 $n = 1$, 则伪凹性等价于单峰性.

一个非负函数 f 称为对数凹的, 若 $\log f$ 是凹的, 即

$$\log f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{y}) \geq \lambda \log f(\mathbf{x}) + (1 - \lambda) \log f(\mathbf{y}), \quad (5.28)$$

等价地,

$$f(\lambda \mathbf{x} + (1 - \lambda)\mathbf{y}) \geq [f(\mathbf{x})]^\lambda [f(\mathbf{y})]^{1-\lambda}, \quad (5.29)$$

其中 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$, $\lambda \in [0, 1]$. 可以证明对数凹的函数一定是伪凹的.

假设集合 \mathcal{A} 是凸集的集合, F 是一个定义在 \mathcal{A} 上的函数. F 称为伪凹的, 如果

$$F(\lambda A_1 + (1 - \lambda)A_2) \geq \min \{F(A_1), F(A_2)\}, \quad (5.30)$$

其中 $A_1, A_2 \in \mathcal{A}$, $\lambda \in [0, 1]$.

引理5.1 如果 $g_j(x, \xi)$ 关于 (x, ξ) 是联合凸的, $j = 1, 2, \dots, p$, 并且概率测度 Pr 是伪凹的, 则对任意置信水平 α , $0 \leq \alpha \leq 1$, 机会约束集合

$$\mathcal{X} = \{x \mid \text{Pr} \{g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha\}$$

是凸的.

证明 首先, 定义

$$\Xi(x) = \{\xi \mid g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\}.$$

为了证明 \mathcal{X} 的凸性, 设 $x_1, x_2 \in \mathcal{X}$ 和 $\lambda \in [0, 1]$ 给定, 则对任意的 $\xi_1 \in \Xi(x_1)$ 及 $\xi_2 \in \Xi(x_2)$, 有

$$g_j(x_i, \xi_i) \leq 0, \quad j = 1, 2, \dots, p, \quad i = 1, 2.$$

由函数 $g_j(x, \xi)$ 关于 (x, ξ) 是联合凸性, $j = 1, 2, \dots, p$, 我们有

$$\begin{aligned} g_j[\lambda x_1 + (1 - \lambda)x_2, \lambda \xi_1 + (1 - \lambda)\xi_2] \\ \leq \lambda g_j(x_1, \xi_1) + (1 - \lambda)g_j(x_2, \xi_2) \leq 0, \end{aligned}$$

$j = 1, 2, \dots, p$. 即

$$\lambda \xi_1 + (1 - \lambda)\xi_2 \in \Xi(\lambda x_1 + (1 - \lambda)x_2),$$

这意味着

$$\lambda \Xi(x_1) + (1 - \lambda)\Xi(x_2) \subset \Xi(\lambda x_1 + (1 - \lambda)x_2).$$

根据概率测度 \Pr 的伪凹性, 有

$$\begin{aligned}\Pr[\Xi(\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2)] &\geq \Pr[\lambda \Xi(\mathbf{x}_1) + (1 - \lambda)\Xi(\mathbf{x}_2)] \\ &\geq \min\{\Pr[\Xi(\mathbf{x}_1)], \Pr[\Xi(\mathbf{x}_2)]\} \\ &\geq \alpha,\end{aligned}$$

从而推出

$$\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2 \in \mathcal{X}.$$

即机会约束集合 \mathcal{X} 的凸性成立. \square

定理5.2 假设函数 $f(\mathbf{x}, \xi)$ 对每一个固定的 ξ 关于 \mathbf{x} 是凸的, 函数 $g_j(\mathbf{x}, \xi), j = 1, 2, \dots, p$ 关于 (\mathbf{x}, ξ) 是联合凸的. 如果概率测度 \Pr 是伪凹的, 则机会约束规划模型

$$\begin{cases} \min & Ef(\mathbf{x}, \xi) \\ \text{s.t.} & \\ & \Pr\{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{cases}$$

是凸规划.

证明 由已知条件, 函数 $g_j(\mathbf{x}, \xi)$ 关于 (\mathbf{x}, ξ) 的联合凸性, $j = 1, 2, \dots, p$, 及概率测度 \Pr 的伪凹性, 根据引理 5.1, 可行集是凸的. 所以只要证明目标函数 $Ef(\mathbf{x}, \xi)$ 关于 \mathbf{x} 是凸的, 则定理得证. 对任意给定的 $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$ 和 $\lambda \in [0, 1]$, 根据函数 $f(\mathbf{x}, \xi)$ 对每一个 ξ 关于 \mathbf{x} 是凸的, 有

$$\begin{aligned}Ef[\lambda \mathbf{x}_1 + (1 - \lambda)\mathbf{x}_2, \xi] &\leq E[\lambda f(\mathbf{x}_1, \xi) + (1 - \lambda)f(\mathbf{x}_2, \xi)] \\ &= \lambda Ef(\mathbf{x}_1, \xi) + (1 - \lambda)Ef(\mathbf{x}_2, \xi),\end{aligned}$$

从而 $Ef(\mathbf{x}, \xi)$ 关于 \mathbf{x} 是凸的. \square

注 我们列举了机会约束规划模型的一些性质, 事实上, 由于概率测度 \Pr 的伪凹性难于验证, 因此, 这些性质只是理论上的一些结果而已.

5.4 随 机 模 拟

机会约束规划与确定性规划的区别在于前者存在机会约束. 因此我们讨论的重点在于如何处理机会约束上.

如果机会约束比较容易处理, 我们可以把机会约束转化为它们各自的确定性等价类. 否则, 我们可以使用随机模拟技术处理复杂的机会约束.

检验随机系统约束

考虑机会约束

$$\Pr \{g_j(x, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha, \quad (5.31)$$

其中 ξ 是随机向量, 其累积概率分布为 $\Phi(\xi)$. 对任意给定的决策变量 x , 我们使用如下的随机模拟技术 (即 Monte Carlo 模拟) 检验机会约束 (5.31) 成立与否. 首先从概率分布 $\Phi(\xi)$ 中产生 N 个独立的随机变量 $\xi_1, \xi_2, \dots, \xi_N$, 这里产生随机变量的方法在第 3 章中已作过详细介绍. 设 N' 是 N 次实验中

$$g_j(x, \xi_i) \leq 0, \quad j = 1, 2, \dots, k,$$

$i = 1, 2, \dots, N$ 成立的次数, 即所产生的随机变量中满足约束的个数. 由大数定律, 可以用频率 N'/N 估计概率. 因此, 机会约束 (5.31) 成立当且仅当频率 $N'/N \geq \alpha$.

步骤 1 置 $N' = 0$;

步骤 2 由概率分布 $\Phi(\xi)$ 生成随机变量 ξ ;

步骤 3 如果 $g_j(x, \xi) \leq 0$, $j = 1, 2, \dots, k$, 则 $N'++$;

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 如果 $N'/N \geq \alpha$, 返回“成立”, 否则返回“不成立”.

计算目标值

考虑带有随机参数 ξ 的目标函数

$$\Pr\{f(\mathbf{x}, \xi) \geq \bar{f}\} \geq \beta, \quad (5.32)$$

有时, 我们需要找到使 (5.32) 成立的最大值 \bar{f} . 随机模拟对这类问题是切实可行的. 为了加快模拟过程, 我们从概率分布 $\Phi(\xi)$ 中生成 N 个独立的随机向量 $\xi_i, i = 1, 2, \dots, N$, 这样得到序列 $\{f_1, f_2, \dots, f_N\}$, 其中 $f_i = f(\mathbf{x}, \xi_i), i = 1, 2, \dots, N$. 取 N' 为 βN 的整数部分. 由大数定律, 序列 $\{f_1, f_2, \dots, f_N\}$ 中第 N' 个最大的元素可以作为 \bar{f} 的估计.

步骤 1 从概率分布 $\Phi(\xi)$ 中生成 N 个随机向量 $\{\xi_1, \xi_2, \dots, \xi_N\}$;

步骤 2 置 $f_i = f(\mathbf{x}, \xi_i), i = 1, 2, \dots, N$;

步骤 3 置 N' 为 βN 整数部分;

步骤 4 返回序列 $\{f_1, f_2, \dots, f_N\}$ 中第 N' 个最大的元素.

处理目标约束

考虑带有随机参数 ξ 的目标约束

$$\Pr\{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \beta. \quad (5.33)$$

根据目标约束的意义, 我们需要找到使

$$\Pr\{d^- \geq b - f(\mathbf{x}, \xi)\} \geq \beta, \quad \Pr\{d^+ \geq f(\mathbf{x}, \xi) - b\} \geq \beta$$

成立的 d^- 和 d^+ 的最小的非负值. 类似于计算目标值的模拟过程, 我们同样可以得到序列 $\{f_1^-, f_2^-, \dots, f_N^-\}$, 其中 $f_i^- = b - f(\mathbf{x}, \xi_i), \xi_i$ 是从概率分布 $\Phi(\xi)$ 中生成的随机向量, $i = 1, 2, \dots, N$. 取 N' 作为 βN 的整数部分, 则序列 $\{f_1^-, f_2^-, \dots, f_N^-\}$ 中第 N' 个最小的元素作为负偏差 d^- 的估计. 如果它是负的, 则置 $d^- = 0$. 类

似地, 我们也可以得到序列 $\{f_1^+, f_2^+, \dots, f_N^+\}$, 其中 $f_i^+ = f(x, \xi_i) - b$, ξ_i 是从概率分布 $\Phi(\xi)$ 中生成的随机向量, $i = 1, 2, \dots, N$. 取 N' 作为 βN 的整数部分, 则序列 $\{f_1^+, f_2^+, \dots, f_N^+\}$ 中第 N' 个最小的元素作为正偏差 d^+ 的估计. 若 $d^+ < 0$, 则置 $d^+ = 0$.

步骤 1 从概率分布 $\Phi(\xi)$ 中生成 N 个随机向量 $\{\xi_1, \xi_2, \dots, \xi_N\}$;

步骤 2 置 $f_i^- = b - f(x, \xi_i)$ 和 $f_i^+ = f(x, \xi_i) - b$, $i = 1, 2, \dots, N$;

步骤 3 置 N' 为 βN 的整数部分;

步骤 4 置 f^- 和 f^+ 分别为序列 $\{f_i^-\}_{i=1}^N$ 和 $\{f_i^+\}_{i=1}^N$ 中第 N' 个最小的元素;

步骤 5 $d^- = f^- \vee 0$, $d^+ = f^+ \vee 0$.

5.5 基于随机模拟的遗传算法

我们知道处理机会约束规划的传统方法是根据事先给定置信水平, 把随机约束转化为它们各自的确定性等价类. 既然这个过程仅对一些特殊情况成立, Iwamura 和 Liu^[60] 建议使用基于随机模拟的遗传算法求解一般的机会约束规划以及机会约束多目标规划和机会约束目标规划.

本节里, 我们介绍基于随机模拟的遗传算法, 并给出一些数值例子解释该算法. 除了使用随机模拟检验后代的可行性, 计算目标值和处理目标约束 (正如在第 5.4 节中讨论的那样) 以外, 本节所使用的初始化过程、选择过程、交叉操作和变异操作与第 2 章中所介绍的基本相同.

遗传算法程序

步骤 0 输入参数 pop_size , P_c 及 P_m ;

步骤 1 初始产生 pop_size 个染色体, 其中可能使用随机模拟技术检验染色体的可行性;

步骤 2 对染色体进行交叉和变异操作, 其中可能使用随机模拟技

术检验后代的可行性;

步骤 3 使用随机模拟技术计算所有染色体的目标值;

步骤 4 根据目标值,使用基于序的评价函数计算每个染色体的适应度;

步骤 5 旋转赌轮,选择染色体;

步骤 6 重复步骤 2 到步骤 5,直到完成给定的循环次数;

步骤 7 给出最好的染色体作为最优解.

解决机会约束规划的遗传算法已编成 C 语言程序.为了解释遗传算法的有数性,本节将给出一些数值例子,计算结果是成功的.下面的例子均在 PC 机上实现,所使用的参数为:种群规模为 30,交叉概率 $P_c = 0.3$,变异概率 $P_m = 0.2$,评价函数中的参数 $\alpha = 0.05$.

例 1 考虑一个简单的机会约束规划模型

$$\begin{cases} \max & x_1 x_2 (x_3 + x_4) \\ \text{s.t.} & \\ & \Pr \{ \xi_1 x_1^2 + \xi_2 x_2^2 + \xi_3 x_3^2 + \xi_4 x_4^2 \leq 100 \} \geq 0.90 \\ & x_1, x_2, x_3, x_4 \geq 0, \end{cases} \quad (5.34)$$

其中 ξ_1 服从均匀分布 $U(2, 3)$, ξ_2 服从正态分布 $N(2, 1)$, ξ_3 服从指数分布 $EXP(3)$, 而 ξ_4 服从柯西分布 $C(3, 2)$.

经过 100 次迭代以后,基于随机模拟的遗传算法所给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*) = (3.1751, 3.1208, 2.2650, 1.7028),$$

其对应的目标值为 39.316. 并有

$$\Pr \{ \xi_1 x_1^{*2} + \xi_2 x_2^{*2} + \xi_3 x_3^{*2} + \xi_4 x_4^{*2} \leq 100 \} \approx 0.90.$$

例 2 考虑单目标机会约束规划

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \Pr \{ \xi_1 x_1 + \xi_2 x_2 + \xi_3 x_3 \geq \bar{f} \} \geq 0.90 \\ \Pr \{ \eta_1 x_1^2 + \eta_2 x_2^2 + \eta_3 x_3^2 \leq 8 \} \geq 0.80 \\ \Pr \{ \tau_1 x_1^3 + \tau_2 x_2^3 + \tau_3 x_3^3 \leq 15 \} \geq 0.85 \\ x_1, x_2, x_3 \geq 0, \end{array} \right. \quad (5.35)$$

其中 ξ_1, η_1 和 τ_1 分别服从均匀分布 $\mathcal{U}(1, 2)$, $\mathcal{U}(2, 3)$ 和 $\mathcal{U}(3, 4)$, ξ_2, η_2 和 τ_2 分别服从正态分布 $\mathcal{N}(1, 1)$, $\mathcal{N}(2, 1)$ 和 $\mathcal{N}(3, 1)$, ξ_3, η_3 和 τ_3 分别服从指数分布 $\mathcal{EXP}(1)$, $\mathcal{EXP}(2)$ 和 $\mathcal{EXP}(3)$.

每一次随机模拟将执行 2000 次循环, 经过 2000 代以后, 基于随机模拟的遗传算法所给出的最优解为

$$(x_1^*, x_2^*, x_3^*) = (1.5296, 0.4226, 0.6608),$$

而目标值 $\bar{f}^* = 2.386$. 更进一步, 我们有

$$\left\{ \begin{array}{l} \Pr \{ \xi_1 x_1^* + \xi_2 x_2^* + \xi_3 x_3^* \geq \bar{f}^* \} \approx 0.90 \\ \Pr \{ \eta_1 x_1^{*2} + \eta_2 x_2^{*2} + \eta_3 x_3^{*2} \leq 8 \} \approx 0.80 \\ \Pr \{ \tau_1 x_1^{*3} + \tau_2 x_2^{*3} + \tau_3 x_3^{*3} \leq 15 \} \approx 0.85. \end{array} \right.$$

例 3 考虑机会约束目标规划 (CCGP)

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^-, d_2^-, d_3^-\} \\ \text{s.t.} \\ \Pr \{ \tau_1 x_1 + \tau_2 x_2 + \tau_3 x_3 + d_1^- - d_1^+ = 10 \} \geq 0.90 \\ x_1 + d_2^- - d_2^+ = 4 \\ x_2 + x_3 + d_3^- - d_3^+ = 2 \\ \Pr \left\{ \begin{array}{l} \eta_1 x_1 + x_2 + x_3 \leq \xi_1 \\ x_1 + \eta_2 x_2 + x_3 \leq \xi_2 \\ x_1 + x_2 + \eta_3 x_3 \leq \xi_3 \end{array} \right\} \geq 0.8 \\ x_1, x_2, x_3 \geq 0, \end{array} \right. \quad (5.36)$$

其中参数 $\xi_1, \xi_2, \xi_3, \eta_1$ 和 τ_1 分别服从均匀分布 $\mathcal{U}(8, 10)$, $\mathcal{U}(10, 12)$, $\mathcal{U}(5, 7)$, $\mathcal{U}(1, 2)$ 和 $\mathcal{U}(2, 3)$, η_2 和 τ_2 分别服从指数分布 $\mathcal{EXP}(2)$ 和 $\mathcal{EXP}(3)$, η_3 和 τ_3 分别服从正态分布 $\mathcal{N}(2, 1)$ 和 $\mathcal{N}(3, 1)$.

经过 400 代以后, 基于随机模拟的遗传算法所给出的最优解为

$$(x_1^*, x_2^*, x_3^*) = (4.0001, 1.0291, 0.1045),$$

其满足第 1 个和第 2 个目标, 但第 3 个目标的负偏差为 0.8664. 更进一步, 我们有

$$\Pr \{ \tau_1 x_1^* + \tau_2 x_2^* + \tau_3 x_3^* \geq 10 \} \approx 0.90$$

和

$$\Pr \left\{ \begin{array}{l} \eta_1 x_1^* + x_2^* + x_3^* \leq \xi_1 \\ x_1^* + \eta_2 x_2^* + x_3^* \leq \xi_2 \\ x_1^* + x_2^* + \eta_3 x_3^* \leq \xi_3 \end{array} \right\} \approx 0.8.$$

5.6 注

本章里，我们介绍了机会约束规划以及机会约束多目标规划和机会约束目标规划模型的一般形式，并介绍了机会约束规划的一些确定等价类及性质，本章也设计了用来检验随机系统约束以及计算目标值和处理目标约束的随机模拟技术。

基于随机模拟的遗传算法是解决复杂的机会约束规划模型(包括机会约束多目标规划和机会约束目标规划)的有力工具。这种方法的突出优点就是能够很好地得到全局最优解。另一个优点就是不需要把机会约束转化为它们各自的确定性等价类，从而保证我们能够处理更一般的机会约束规划模型。基于随机模拟的遗传算法的有效性已被大量实验所证实，本章最后给出一些数值例子对此也进行了解释。

第 6 章

机会约束规划的应用

本章里，我们提供一些实际例子，以便解释机会约束规划和机会约束多目标规划以及机会约束目标规划的建模思想。介绍的实际问题有：

- (a) 生产过成；
- (b) 饲料混合问题；
- (c) 随机资源分配；
- (d) 开放存储网络；
- (e) 资金预算。

6.1 生 产 过 程

本节所介绍的是 Kall 和 Wallace^[65] 所给出的例子。某炼油厂冶炼两种原油（分别记为 raw_1 和 raw_2 ），需要提前制定一周的生产计划，以便为天然气公司提供天然气（记为 $prod_1$ ）和为火力发电厂提供燃料用油（记为 $prod_2$ ）。

假定原料 raw_1 所能生产出的天然气的产量 $\pi(raw_1, prod_1)$ 和原料 raw_2 所能生产出的燃料用油的产量 $\pi(raw_2, prod_2)$ 是随机变化的，而所生产的其它产品的产量却是固定的。这里假设

$$\begin{aligned}\pi(raw_1, prod_1) &= 2 + \eta_1, & \pi(raw_1, prod_2) &= 3, \\ \pi(raw_2, prod_1) &= 6, & \pi(raw_2, prod_2) &= 3.4 - \eta_2,\end{aligned}$$

其中 η_1 为服从均匀分布 $U(-0.8, 0.8)$ 的随机变量， η_2 为服从指数分布 $EXP(0.4)$ 的随机变量。

我们知道, 用户 (天然气公司和火力发电厂) 对天然气一周的需求量 h_1 和对燃料的需求量 h_2 也是随机变化的, 可以分别表示为

$$h_1 = 180 + \xi_1, \quad h_2 = 162 + \xi_2,$$

这里 ξ_1 和 ξ_2 是分别服从正态分布 $\mathcal{N}(0, 12)$ 和 $\mathcal{N}(0, 9)$ 的随机变量.

假设 x_1 和 x_2 分别是原料 raw_1 和 raw_2 的一周使用量. 原料 raw_1 的单位价格为 $c_1 = 2$, 原料 raw_2 的单位价格为 $c_2 = 3$. 这样总费用为 $2x_1 + 3x_2$.

生产能力 (即原材料的最大消耗量) 假设为 100, 这样, 对生产计划来说就有一个约束 $x_1 + x_2 \leq 100$.

由于每周的生产计划 (x_1, x_2) 是提前制定的, 并且一周之内不能改变. 同时, 在相应的周内, 客户希望他们的实际需要得到满足, 即

$$(2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1, \quad 3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2.$$

由于此约束中含有随机变量, 因此, 它的数学意义并不明确. 一种可行的方法是使用机会约束, 对两个用户分别给予置信水平 α_1 和 α_2 ,

$$\Pr\{(2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1\} \geq \alpha_1,$$

$$\Pr\{3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2\} \geq \alpha_2.$$

我们希望在满足用户需求的机会约束下尽可能地降低总费用. 区别于文献 [65] 中所使用的补偿模型, 我们把这个生产过程问题

建模为机会约束规划模型,

$$\left\{ \begin{array}{l} \min f(\mathbf{x}) = 2x_1 + 3x_2 \\ \text{s.t.} \\ \Pr\{(2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1\} \geq \alpha_1 \\ \Pr\{3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2\} \geq \alpha_2 \\ x_1 + x_2 \leq 100 \\ x_1, x_2 \geq 0. \end{array} \right. \quad (6.1)$$

当置信水平 α_1 和 α_2 分别取为 0.8 和 0.7 时, 使用基于随机模拟的遗传算法, 经过 500 代以后, 得到的最优生产计划为

$$(x_1^*, x_2^*) = (31.95, 22.65),$$

其总费用为 $f(x_1^*, x_2^*) = 131.85$. 更进一步, 我们有

$$\Pr\{(2 + \eta_1)x_1^* + 6x_2^* \geq 180 + \xi_1\} \approx 0.88,$$

$$\Pr\{3x_1^* + (3.4 - \eta_2)x_2^* \geq 162 + \xi_2\} \approx 0.70.$$

6.2 饲料混合问题

Van de Panne 和 Popp^[133] 为饲料混合问题提出了一个机会约束规划模型. 选择四种原料进行混合得到一种混合饲料, 在满足蛋白质和脂肪含量约束下, 使总的费用达到最小.

设 x_1, x_2, x_3 和 x_4 是四种原料所占的百分比, 因此首先我们有约束 $x_1 + x_2 + x_3 + x_4 = 1$.

假设四种原料中每个单位的蛋白质含量分别为 2.3, 5.6, 11.1 和 1.3. 如果规定饲料的蛋白质含量不低于 5, 则有约束

$$2.3x_1 + 5.6x_2 + 11.1x_3 + 1.3x_4 \geq 5.$$

四种原料中每个单位脂肪含量分别为 η_1, η_2, η_3 和 η_4 ，并假定它们分别服从正态分布 $\mathcal{N}(12.0, 0.2809^2)$ ， $\mathcal{N}(11.9, 0.1936^2)$ ， $\mathcal{N}(41.8, 20.25^2)$ 和 $\mathcal{N}(52.1, 0.6241^2)$ 。如果饲料中脂肪含量要求大于 21 的概率为 $\alpha (0 < \alpha < 1)$ ，则有机会约束

$$\Pr \{ \eta_1 x_1 + \eta_2 x_2 + \eta_3 x_3 + \eta_4 x_4 \geq 21 \} \geq \alpha.$$

四种原料的价格分别假设为 24.55，26.75，39.00 和 40.50。则饲料的单位费用为

$$f(x) = 24.55x_1 + 26.75x_2 + 39.00x_3 + 40.50x_4.$$

这样，对饲料混合问题，机会约束规划模型为

$$\left\{ \begin{array}{l} \min f(x) = 24.55x_1 + 26.75x_2 + 39.00x_3 + 40.50x_4 \\ \text{s.t.} \\ x_1 + x_2 + x_3 + x_4 = 1 \\ 2.3x_1 + 5.6x_2 + 11.1x_3 + 1.3x_4 \geq 5 \\ \Pr \{ \eta_1 x_1 + \eta_2 x_2 + \eta_3 x_3 + \eta_4 x_4 \geq 21 \} \geq \alpha \\ x_1, x_2, x_3, x_4 \geq 0. \end{array} \right. \quad (6.2)$$

当 $\alpha = 0.8$ 时，经过 1000 代以后，基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*) = (0.001, 0.739, 0.053, 0.199),$$

其单位混合饲料费用为 $f(x_1^*, x_2^*, x_3^*, x_4^*) = 30.25$ ，而脂肪的实际含量满足要求的概率恰为 80%。

6.3 随机资源分配

考虑随机资源分配问题，其中有多处资源和多个用户。随机资源分配的任务是确定各种资源的组合，使其满足给定的目标。

例如, 下面提到的水资源供给系统, 有 3 处水资源和 4 个用户. 水资源供给网络如图 6.1 所示.

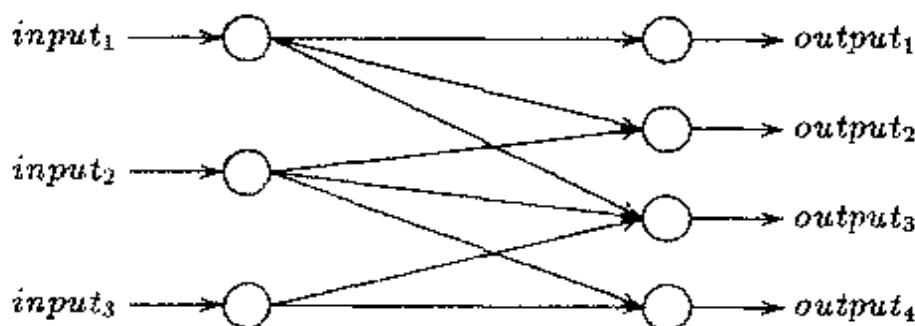


图 6.1 水资源供给系统

为了确定最优水资源分配计划, 用 8 个决策变量 x_1, x_2, \dots, x_8 表示一个行动方案, 其中 x_1, x_2, x_3 表示用户 1, 2, 3 分别从资源 $input_1$ 订购的数量, x_4, x_5, x_6 表示用户 2, 3, 4 分别从资源 $input_2$ 订购的数量, x_7, x_8 是用户 3, 4 分别从资源 $input_3$ 订购的数量.

注意到我们所使用的外部资源有它们自己的性质. 例如, 三处资源所能提供的最大供水量表示为 ξ_1, ξ_2 和 ξ_3 , 它们分别服从具有两个参数的对数正态分布 $LOGN(1.56, 0.56^2)$, $LOGN(1.36, 0.45^2)$ 和 $LOGN(0.95, 0.38^2)$. 这样, 首先有下面的随机约束,

$$\begin{aligned} x_1 + x_2 + x_3 &\leq \xi_1, \\ x_4 + x_5 + x_6 &\leq \xi_2, \\ x_7 + x_8 &\leq \xi_3. \end{aligned} \quad (6.3)$$

为了处理随机约束 (6.3), 设三个约束的置信水平分别为 $\alpha_1 = 0.9$, $\alpha_2 = 0.7$ 和 $\alpha_3 = 0.8$, 这样我们有机会约束

$$\begin{cases} \Pr\{x_1 + x_2 + x_3 \leq \xi_1\} \geq 0.9, \\ \Pr\{x_4 + x_5 + x_6 \leq \xi_2\} \geq 0.7, \\ \Pr\{x_7 + x_8 \leq \xi_3\} \geq 0.8. \end{cases}$$

另一个明显的约束是 $x_i \geq 0, i = 1, 2, \dots, 8$, 即从资源订购的数量必须是非负的.

管理目标是满足四个用户的需求, 其需求量分别为 3, 1, 2 和 3. 同时假设管理者有如下的目标值和优先结构.

第 1 优先级, 尽可能满足用户 $output_1$ 的需求, 即

$$x_1 + d_1^- - d_1^+ = 3,$$

其中 d_1^- 将被极小化.

第 2 优先级, 尽可能满足用户 $output_2$ 的需求, 即

$$x_2 + x_4 + d_2^- - d_2^+ = 1,$$

其中 d_2^- 将被极小化.

第 3 优先级, 尽可能满足用户 $output_3$ 的需求, 即

$$x_3 + x_5 + x_7 + d_3^- - d_3^+ = 2,$$

其中 d_3^- 将被极小化.

最后一个优先级是尽可能满足用户 $output_4$ 的需求, 即

$$x_6 + x_8 + d_4^- - d_4^+ = 3,$$

其中 d_4^- 将被极小化.

那么, 有关此问题的机会约束目标规划可以表示为

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, d_3^-, d_4^-\} \\ \text{s.t.} \\ x_1 + d_1^- - d_1^+ = 3 \\ x_2 + x_4 + d_2^- - d_2^+ = 1 \\ x_3 + x_5 + x_7 + d_3^- - d_3^+ = 2 \\ x_6 + x_8 + d_4^- - d_4^+ = 3 \\ \Pr\{x_1 + x_2 + x_3 \leq \xi_1\} \geq 0.9 \\ \Pr\{x_4 + x_5 + x_6 \leq \xi_2\} \geq 0.7 \\ \Pr\{x_7 + x_8 \leq \xi_3\} \geq 0.8 \\ x_i \geq 0, \quad i = 1, 2, \dots, 8. \end{array} \right. \quad (6.4)$$

经过 3000 代运行以后, 基于随机模拟的遗传算法给出的最优解为

$$\mathbf{x} = (3.01, 0.00, 0.07, 1.31, 0.99, 0.15, 0.47, 1.07),$$

它满足前两个目标, 但第 3 个目标和第 4 个目标的负偏差分别为 0.47 和 1.78.

6.4 开放存储网络

通常, 一个网络可以用节点 $\{1, 2, \dots, p\}$ 和弧 $\{1, 2, \dots, n\}$ 表示. 一个弧 k 表示为 $k \sim (i, j)$, 表明它起始于节点 i , 结束于节点 j . 作为一个推广, Liu 和 Esogbue^[92] 提出了开放存储网络, 把外部世界看作一个节点, 称为外部节点, 表示为 ∞ .

作为开放存储网络的实际背景, 考虑随机存储问题和水库操作问题, 那么节点表示仓库或水库, 外部节点表示市场或用户, 弧表示订购量、运输量或放水量.

对每个节点 i (不包括 $\infty!$)，设 a_i 和 ξ_i 分别表示状态和随机扰动，则 $\xi_i > 0$ 表示输入， $\xi_i < 0$ 表示输出。对存储问题， a_i 表示存储量， ξ_i 表示随机需求；对水库操作问题， a_i 表示库容量， ξ_i 表示随机径流。

对每个弧 $k \sim (i, j)$ ，我们用 x_k 表示流量，它是一个决策变量。那么 $x_k > 0$ 表示从节点 i 流向 j ， $x_k < 0$ 表示从节点 j 流向 i 。对一个存储问题，一个决策表示从外部订购或从其它仓库运输商品的数量；对水库操作问题，一个决策表示从其它水库引水或弃水量。

我们称以上的网络为开放存储网络，其中弧对应着决策，节点对应着仓库或水库，外部节点对应外部世界。这里开放的意思是由于外部节点和随机扰动的引入，系统中总量是不守恒的。

一般情况下，一个网络称为有能力约束的，如果对所有的弧 k ， $-\infty < x_k < +\infty$ ， $k = 1, 2, \dots, n$ 。否则，称为无能力约束的。多数网络是两种情况的混合。这里流量 x_k 的约束条件为 $q_k \leq x_k \leq Q_k$ ， $k = 1, 2, \dots, n$ 。既然每个节点 i (不包括 $\infty!$) 是一个仓库或者一个水库，它们的状态 a_i 自然应该有一个约束，

$$v_i \leq a_i \leq V_i, \quad i = 1, 2, \dots, p. \quad (6.5)$$

例如， $v_i = 0$ ， $V_i =$ 最大库容量。

设 $\mathbf{a} = (a_1, a_2, \dots, a_p)$ 是状态向量， $\boldsymbol{\xi} = (\xi_1, \xi_2, \dots, \xi_p)$ 是随机向量，其联合概率密度函数为 $\phi(\boldsymbol{\xi})$ ， $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 是决策变量，其约束为

$$D = \{\mathbf{x} \in \Re^n \mid q_i \leq x_i \leq Q_i, \quad i = 1, 2, \dots, n\}, \quad (6.6)$$

其中 q_i 和 Q_i 不一定是正的或有限的。例如，在水库操作问题中， q_i 和 Q_i 分别表示水的需求量和下游允许的最大流量。

经过一个决策和一个随机扰动以后, 状态 \mathbf{a}' 可以用状态转移方程描述

$$\mathbf{a}' = \mathbf{a} + t(\mathbf{x}) + \xi, \quad (6.7)$$

其中 $t()$ 是一个 $\mathbb{R}^n \rightarrow \mathbb{R}^p$ 的函数.

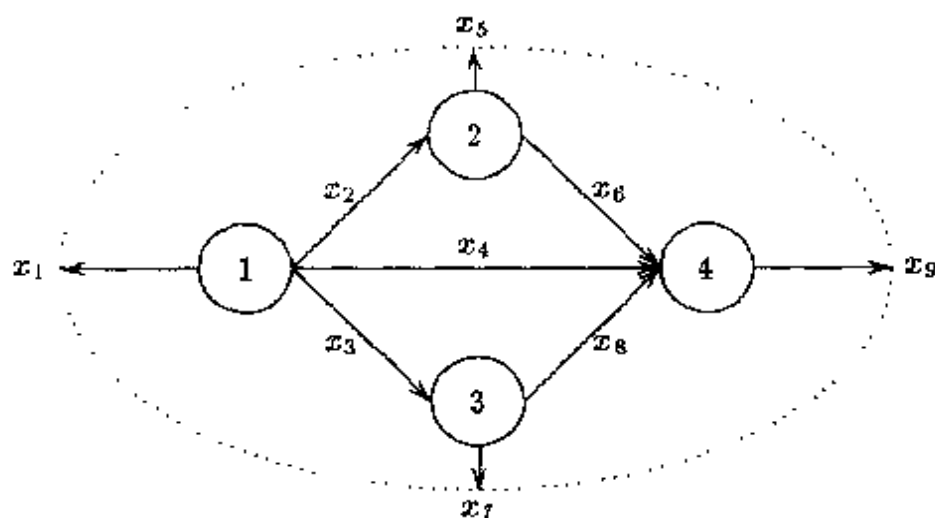


图 6.2 开放存储网络

在图 6.2 中, 有四个节点, 记为 1, 2, 3 和 4, 外部节点用虚线椭圆表示, 九个弧分别为 $(1, \infty)$, $(1, 2)$, $(1, 3)$, $(1, 4)$, $(2, \infty)$, $(2, 4)$, $(3, \infty)$, $(3, 4)$ 和 $(4, \infty)$. 在此问题中, 函数 $t()$ 为

$$t(\mathbf{x}) = \begin{pmatrix} -x_1 - x_2 - x_3 - x_4 \\ x_2 - x_5 - x_6 \\ x_3 - x_7 - x_8 \\ x_4 + x_6 + x_8 - x_9 \end{pmatrix}^T$$

如果记 $t_1(\mathbf{x}) = -x_1 - x_2 - x_3 - x_4$, $t_2(\mathbf{x}) = x_2 - x_5 - x_6$, $t_3(\mathbf{x}) = x_3 - x_7 - x_8$ 和 $t_4(\mathbf{x}) = x_4 + x_6 + x_8 - x_9$, 则有

$$t(\mathbf{x}) = (t_1(\mathbf{x}), t_2(\mathbf{x}), t_3(\mathbf{x}), t_4(\mathbf{x})).$$

一个最基本的网络优化问题是最短路问题, 即在网络中找到两个给定节点间的最短路径, 路径的测度可以是距离、时间、货币等等. 最小费用流问题主要考虑在网络中从起点运送一批货物到达指定终点, 每条弧上不但有流量问题, 而且要考虑单位流量费用问题, 即要求设计一个费用最小的运输方案. 最大流问题考虑的是在网络中从指定起点运送货物到指定终点, 每条弧上有流量限制, 要求确定其最大流量.

最短路问题和最小费用流问题实际上都假定了网络中起始状态和终止状态是已知的, 然后, 在一些最优准则下找到一个行动, 达到终止状态. 最大流问题试图找到最大流, 或等价地, 找到一个给定节点的最大终止状态. 然而, 开放存储网络中的最优操作问题并没有假定终止的状态最事先已知的. 实际上, 最优操作问题本身就是根据一些最优准则, 在弧和节点的容量约束下, 从给定的起始状态出发, 找到最好的终止状态. 对这种情况, 必须给出一个最优操作策略, 以到达最好的终止状态.

开放存储网络的一个最优操作模型是由 Liu 和 Esogbue^[84, 92]提出的基于动态规划的模糊准则模型. 如果制定决策时只考虑约束集 (6.6), 而忽略节点的实际约束, 那么就可以通过虚拟状态 (即经过决策和随机扰动之后每个的状态, 不包括外部节点) 来描述网络. 当虚拟状态小于死库容时, 说明没有满足需求, 差额部分表示缺货数; 当虚拟状态大于最大库容时, 总量超出了仓库或水库的容量, 差额部分表示超出的水平或水灾的程度. 通常, 存在一个最好的状态, 我们定义模糊准则的值为 1. 当存储量远离最好的状态时, 模糊准则的值随之减小. 这样, 所有满意状态构成的集合是一个模糊准则集合, 模糊准则函数是每个元素的满意程度. 这里模糊准则函数用 $\mu(\cdot)$ 表示.

对一个给定的 K 阶段决策过程, 设 $\mu_1, \mu_2, \dots, \mu_K$ 分别是阶段 $1, 2, \dots, K$ 所对应的模糊准则函数, $\rho_1, \rho_2, \dots, \rho_K$ 是凸组合的

系数, 表示 $\mu_1, \mu_2, \dots, \mu_K$ 之间的相对重要性. 假设目标函数是极大化所有阶段的模糊准则函数的加权和, 则模糊准则模型可以表示为,

$$\sup_{\mathbf{x}_k, k=1, 2, \dots, K} \sum_{k=1}^K \rho_k \int_{\mathbb{R}^p} \mu_k(\mathbf{a}_k + t(\mathbf{x}_k) + \boldsymbol{\xi}) \phi_k(\boldsymbol{\xi}) d\boldsymbol{\xi}, \quad (6.8)$$

其状态转移函数为

$$\mathbf{a}_{k+1} = \mathbf{a}_k + t(\mathbf{x}_k) + \boldsymbol{\xi}, \quad (6.9)$$

$k = 1, 2, \dots, K - 1$. 而它的动态模糊准则模型为

$$\begin{cases} f_K(\mathbf{a}) = \sup_{\mathbf{x} \in D_K} L_K(\mathbf{a} + t(\mathbf{x})) \\ f_k(\mathbf{a}) = \sup_{\mathbf{x} \in D_k} \left\{ \theta_k L_k(\mathbf{a} + t(\mathbf{x})) + (1 - \theta_k) \right. \\ \quad \left. \times \int_{\mathbb{R}^p} f_{k+1}(\mathbf{a} + t(\mathbf{x}) + \boldsymbol{\xi}) \phi_k(\boldsymbol{\xi}) d\boldsymbol{\xi} \right\} \\ k \leq K - 1, \end{cases} \quad (6.10)$$

其中

$$L_k(\mathbf{y}) = \int_{\mathbb{R}^p} \mu_k(\mathbf{y} + \boldsymbol{\xi}) \phi_k(\boldsymbol{\xi}) d\boldsymbol{\xi} \quad (6.11)$$

及 $\theta_k = \rho_k / (\rho_k + \rho_{k+1} + \dots + \rho_K)$, $k = 1, 2, \dots, K$.

通过模糊准则函数 μ_k , $k = 1, 2, \dots, K$, 模糊准则模型吸收了专家的知识 and 经验, 所以说, 这种模型符合现代决策理论的思想. 但模糊准则模型仅适用于制定长期的和动态的决策. 有时, 我们需要更细致地处理开放存储网络. 下面使用机会约束规划建立这种开放存储网络模型.

由于每一个节点 i 表示一个仓库或水库, 所以状态 a_i 是非负的. 对存储系统, 状态 $a_i \geq 0$; 对水资源系统, $a_i \geq v_i$, 其中 v_i 是节点 i 的死库容. 另一方面, 状态 a_i 一定小于最大库容 V_i , 即

无论什么节点, 都有 $a_i \leq V_i$, 因此所得出的决策 $\mathbf{x} \in D$ 必须使得经过决策和随机扰动之后的状态 a'_i 满足 $v_i \leq a'_i \leq V_i$, 即

$$v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, \quad i = 1, 2, \dots, p, \quad (6.12)$$

其中 $t(\mathbf{x}) = (t_1(\mathbf{x}), t_2(\mathbf{x}), \dots, t_p(\mathbf{x}))$, $t_i(\mathbf{x})$ 是实值函数. 既然等式 (6.12) 中存在随机扰动向量 $\xi = (\xi_1, \xi_2, \dots, \xi_p)$, 在随机向量实现之前, 我们并不知道给定决策 \mathbf{x} 是否满足约束. 但根据机会约束规划的思想, 我们可以对随机约束 (6.12) 事先确定一个置信水平 α , 即

$$\Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, i = 1, 2, \dots, p\} \geq \alpha, \quad (6.13)$$

这是一个联合机会约束, 其中 $\Pr\{\cdot\}$ 表示事件 $\{\cdot\}$ 的概率. 这个机会约束将被添加到约束集合 D 中. 所以一个决策 $\mathbf{x} \in D$ 是可行的当且仅当集合

$$\{\xi | v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, i = 1, 2, \dots, p\}$$

的概率测度不小于 α . 注意到等式 (6.12) 等价于

$$v_i - a_i - t_i(\mathbf{x}) \leq \xi_i \leq V_i - a_i - t_i(\mathbf{x}), \quad i = 1, 2, \dots, p,$$

即 $\xi = (\xi_1, \xi_2, \dots, \xi_p)$ 位于超几何体

$$[v_1 - a_1 - t_1(\mathbf{x}), V_1 - a_1 - t_1(\mathbf{x})] \times \dots \times [v_p - a_p - t_p(\mathbf{x}), V_p - a_p - t_p(\mathbf{x})]$$

中. 这样我们有

$$\begin{aligned} & \Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, i = 1, 2, \dots, p\} \\ &= \int_{v_1 - a_1 - t_1(\mathbf{x})}^{V_1 - a_1 - t_1(\mathbf{x})} \dots \int_{v_p - a_p - t_p(\mathbf{x})}^{V_p - a_p - t_p(\mathbf{x})} \phi(\xi_1, \dots, \xi_p) d\xi_1 \dots d\xi_p, \end{aligned}$$

从而导出机会约束 (6.13) 的确定性等价类为

$$\int_{v_1 - a_1 - t_1(\mathbf{x})}^{V_1 - a_1 - t_1(\mathbf{x})} \cdots \int_{v_p - a_p - t_p(\mathbf{x})}^{V_p - a_p - t_p(\mathbf{x})} \phi(\xi_1, \dots, \xi_p) d\xi_1 \cdots d\xi_p \geq \alpha.$$

有时联合机会约束可以分成几个独立的机会约束

$$\Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i\} \geq \alpha_i, \quad i = 1, 2, \dots, p, \quad (6.14)$$

其中 α_i , $i = 1, 2, \dots, p$ 是对各个约束事先给定的置信水平, 假定 $\phi(\xi_1, \xi_2, \dots, \xi_p)$ 是可分离的, 即

$$\phi(\xi_1, \xi_2, \dots, \xi_p) = \phi_1(\xi_1)\phi_2(\xi_2) \cdots \phi_p(\xi_p).$$

则有

$$\begin{aligned} & \Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i\} \\ &= \int_{v_i - a_i - t_i(\mathbf{x})}^{V_i - a_i - t_i(\mathbf{x})} \phi_i(\xi_i) d\xi_i \\ &= \Phi_i(V_i - a_i - t_i(\mathbf{x})) - \Phi_i(v_i - a_i - t_i(\mathbf{x})) \end{aligned}$$

其中 Φ_i 是 ξ_i , $i = 1, 2, \dots, p$ 的概率分布函数. 则机会约束 (6.14) 等价于

$$\Phi_i(V_i - a_i - t_i(\mathbf{x})) - \Phi_i(v_i - a_i - t_i(\mathbf{x})) \geq \alpha_i, \quad (6.15)$$

$i = 1, 2, \dots, p$. 这里, 机会约束集合不一定是凸的.

根据集合 D 的定义, 决策 $\mathbf{x} \in D$ 指决策 \mathbf{x} 满足用户的需求并且不超过弧的流量. 如果一个决策 \mathbf{x} 能够实现, 即满足约束 (6.12) 或机会约束 (6.14), 那么这个决策一般认为是可以接受的. 所以

可以把这个问题建模为如下的机会约束规划,

$$\left\{ \begin{array}{l} \text{找到 } D \text{ 中的点 } \mathbf{x} \\ \text{s.t.} \\ \Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i\} \geq \alpha_i \\ i = 1, 2, \dots, p, \end{array} \right. \quad (6.16)$$

其中 v_i, V_i, a_i 和 α_i 是给定参数, $i = 1, 2, \dots, p$, 决策向量为 $\mathbf{x} = (x_1, x_2, \dots, x_n)$, 等价地, 我们有

$$\left\{ \begin{array}{l} \text{找到 } D \text{ 中的点 } \mathbf{x} \\ \text{s.t.} \\ \Phi_i(V_i - a_i - t_i(\mathbf{x})) - \Phi_i(v_i - a_i - t_i(\mathbf{x})) \geq \alpha_i \\ i = 1, 2, \dots, p. \end{array} \right.$$

虽然机会约束规划 (6.16) 给定的任何决策 \mathbf{x} 都可以接受, 但决策者通常仅对其中一些特殊的决策感兴趣. 例如, 我们期望 (i) 所有流量的和尽可能地小, 即极小化函数 $f_1(\mathbf{x}) = |x_1| + |x_2| + \dots + |x_n|$; (ii) 总运输费用尽可能地小, 即极小化 $f_2(\mathbf{x}) = c_1|x_1| + c_2|x_2| + \dots + c_n|x_n|$ 等. 当选定目标函数 $f(\mathbf{x})$ (不失一般性, 假定极小化 $f(\mathbf{x})$) 以后, 我们把开放存储网络建模为如下的机会约束规划,

$$\left\{ \begin{array}{l} \min f(\mathbf{x}) \\ \text{s.t.} \\ \Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i\} \geq \alpha_i \\ i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{array} \right. \quad (6.17)$$

这个机会约束规划等价于

$$\left\{ \begin{array}{l} \min f(\mathbf{x}) \\ \text{s.t.} \\ \Phi_i(V_i - a_i - t_i(\mathbf{x})) - \Phi_i(v_i - a_i - t_i(\mathbf{x})) \geq \alpha_i \\ i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{array} \right.$$

既然开放存储网络可能有多个目标函数, 我们把单目标机会约束规划推广到多目标情况,

$$\left\{ \begin{array}{l} \min [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ \Pr \{v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i\} \geq \alpha_i \\ i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{array} \right. \quad (6.18)$$

该模型有如下的确定性等价类

$$\left\{ \begin{array}{l} \min [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ \Phi_i(V_i - a_i - t_i(\mathbf{x})) - \Phi_i(v_i - a_i - t_i(\mathbf{x})) \geq \alpha_i \\ i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{array} \right.$$

当管理目标值给定以后, 根据决策者的优先结构, 目标函数可以是极小化偏差. 这样我们可以把随机开放存储网络建模为机

会约束目标规划,

$$\left\{ \begin{array}{ll} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) & \\ \text{s.t.} & \\ f_i(x) + d_i^- - d_i^+ = b_i, & i = 1, 2, \dots, m \\ \Pr\{v_i \leq a_i + t_i(x) + \xi_i \leq V_i\} \geq \alpha_j, & j = 1, 2, \dots, p \\ x \in D & \\ d_i^-, d_i^+ \geq 0, & i = 1, 2, \dots, m, \end{array} \right.$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

$x = n$ 维决策向量,

f_i = 目标约束中的函数,

g_j = 系统约束中的函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

p = 系统约束个数.

下面回到图 6.2 所示的开放存储网络 (水资源系统). 值定九个弧上的所有流量构成集合 D , 其中元素 $x = (x_1, x_2, \dots, x_9)$ 的

约束集为

$$\begin{aligned} 10 \leq x_1 \leq 50, \quad 0 \leq x_2 \leq 10, \quad 0 \leq x_3 \leq 10, \\ 0 \leq x_4 \leq 15, \quad 15 \leq x_5 \leq 60, \quad -5 \leq x_6 \leq 5, \\ 15 \leq x_7 \leq 60, \quad -5 \leq x_8 \leq 5, \quad 20 \leq x_9 \leq 70. \end{aligned}$$

随机径流分别服从具有两参数的对数正态分布 $\text{LOGN}(\mu_i, \sigma_i^2)$, $i = 1, 2, 3, 4$. 假定四个水库的参数 (μ_i, σ_i) 的值分别为

$$(2.24, 1.12), \quad (1.60, 1.28), \quad (1.87, 1.45), \quad (1.30, 1.34),$$

同时还假定四个水库的参数 (v_i, V_i) 的值分别为 $(10, 120)$, $(20, 100)$, $(10, 80)$ 和 $(0, 90)$.

假定给定决策 $\mathbf{x} \in D$ 满足约束 (6.12) 的概率不小于置信水平 $\alpha = 0.90$, 目标函数是极小化所有流量之和. 则开放存储网络的优化问题可以建模为机会约束规划,

$$\left\{ \begin{array}{l} \min |x_1| + |x_2| + |x_3| + |x_4| + |x_5| + |x_6| + |x_7| + |x_8| + |x_9| \\ \text{s.t.} \\ 10 \leq x_1 \leq 50, \quad 0 \leq x_2 \leq 10, \quad 0 \leq x_3 \leq 10 \\ 0 \leq x_4 \leq 15, \quad 15 \leq x_5 \leq 60, \quad -5 \leq x_6 \leq 5 \\ 15 \leq x_7 \leq 60, \quad -5 \leq x_8 \leq 5, \quad 20 \leq x_9 \leq 70 \\ \Pr \left\{ \begin{array}{l} 10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120 \\ 20 \leq 80 + x_2 - x_5 - x_6 + \xi_2 \leq 100 \\ 10 \leq 60 + x_3 - x_7 - x_8 + \xi_3 \leq 80 \\ 0 \leq 50 + x_4 + x_6 + x_8 - x_9 + \xi_4 \leq 90 \end{array} \right\} \geq 0.90, \end{array} \right.$$

其中 70, 80, 60 和 50 是四个水库的初始库容量. 经过 3000 代运算以后, 基于随机模拟的遗传算法给出的最优解为

$$\mathbf{x}^* = (31.8, 0.0, 0.1, 0.1, 43.6, 1.3, 47.0, 0.0, 20.2),$$

其实际可靠性恰为 90%, 所有流量之和为 144.1.

6.5 资 金 预 算

最初的资金预算主要考虑的是在资金总量约束下, 通过选择适当的方案组合, 使总效益极大化. 出于未来需求的不确定性的考虑和多个冲突的目标的要求, Keown 和 Martin^[70] 以及 Keown 和 Taylor^[72] 分别在资金预算领域引进了机会约束整数目标规划. 另外, De 等^[29] 把机会约束目标规划推广到 0-1 规划并应用到资金预算问题.

考虑一个企业, 假定有 n 种型号的机器可以选择, 用 x_i 表示在型号为 i 的机器中确定使用的机器数, $i = 1, 2, \dots, n$. 那么所有的机器数 x_i 是非负整数. 设 a_i 是为型号 i 的机器分配的购买资金, a 是用来分配的资金总数, 则有,

$$a_1x_1 + a_2x_2 + \dots + a_nx_n \leq a, \quad (6.19)$$

另外一个约束是对机器的最大使用空间的限制. 假定 b_i 是型号为 i 的机器所需的空间, $i = 1, 2, \dots, n$. 若总有效空间为 b , 则有如下的约束

$$b_1x_1 + b_2x_2 + \dots + b_nx_n \leq b. \quad (6.20)$$

假定不同的机器生产不同的产品. η_i 是型号为 i 的机器生产产品 i 时的生产能力, 则产品 i 的总数为 η_ix_i , $i = 1, 2, \dots, n$. 同时假定产品 i 的未来需求量为 ξ_i , $i = 1, 2, \dots, n$. 既然产品应满足未来的需求, 因此有

$$\eta_ix_i \geq \xi_i, \quad i = 1, 2, \dots, n. \quad (6.21)$$

假设 c_i 是型号为 i 的机器的净效益, $i = 1, 2, \dots, n$, 那么总净效益为 $c_1x_1 + c_2x_2 + \dots + c_nx_n$. 假设目标函数是极大化总净效益, 即

$$\max c_1x_1 + c_2x_2 + \dots + c_nx_n, \quad (6.22)$$

则对资金预算问题,有如下的整数规划模型,

$$\left\{ \begin{array}{l} \max c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \\ \text{s.t.} \\ a_1 x_1 + a_2 x_2 + \cdots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \cdots + b_n x_n \leq b \\ \eta_i x_i \geq \xi_i, \quad i = 1, 2, \cdots, n \\ x_i, i = 1, 2, \cdots, n, \quad \text{非负整数.} \end{array} \right. \quad (6.23)$$

资金预算问题 (6.23) 显然是著名的背包问题的一般情况. 我们知道, 所谓背包问题就是填充背包使其总价值最大, 即

$$\left\{ \begin{array}{l} \max c_1 x_1 + c_2 x_2 + \cdots + c_n x_n \\ \text{s.t.} \\ a_1 x_1 + a_2 x_2 + \cdots + a_n x_n \leq a \\ x_i, i = 1, 2, \cdots, n, \quad \text{非负整数.} \end{array} \right. \quad (6.24)$$

其中 (c_1, c_2, \cdots, c_n) 和 (a_1, a_2, \cdots, a_n) 分别是 n 种物品的价值和体积, a 是背包的容量.

的确, 实际中的资金预算问题远比上面提到的模型复杂. 但是, 作为解释机会约束整数规划, 这些已经足够了. 本节里, 我们主要通过机会约束整数规划建模资金预算问题. 这方面的内容可以参阅文献 [70][72][29].

实际问题中, 产品总数 η_i 和未来需求 ξ_i 不一定是确定的. 这里, 我们假定它们是随机变量. 设 ψ_i 和 ϕ_i 分别表示随机变量 η_i 和 $\xi_i (i = 1, 2, \cdots, n)$ 的概率密度函数. 则约束 $\eta_i x_i \geq \xi_i$ 是随机的. 如果用 α_i 分别表示满足产品 i 需求的置信水平, $i = 1, 2, \cdots, n$, 则有如下的机会约束

$$\Pr \{ \eta_i x_i \geq \xi_i \} \geq \alpha_i, \quad i = 1, 2, \cdots, n, \quad (6.25)$$

其中 $\text{Pr}\{\cdot\}$ 表示事件 $\{\cdot\}$ 的概率. 这样, 可以得到如下的机会约束整数规划,

$$\left\{ \begin{array}{l} \max c_1x_1 + c_2x_2 + \cdots + c_nx_n \\ \text{s.t.} \\ a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq a \\ b_1x_1 + b_2x_2 + \cdots + b_nx_n \leq b \\ \text{Pr}\{\eta_ix_i \geq \xi_i\} \geq \alpha_i, \quad i = 1, 2, \cdots, n \\ x_i, i = 1, 2, \cdots, n, \quad \text{非负整数.} \end{array} \right. \quad (6.26)$$

我们可用分离的机会约束 $\text{Pr}\{c_ix_i \geq d_i\} \geq \alpha_i, i = 1, 2, \cdots, n$ 代替联合约束

$$\text{Pr}\{\eta_ix_i \geq \xi_i, i = 1, 2, \cdots, n\} \geq \alpha,$$

或者采用一些混合形式.

如果净数益 c_i 也是随机的, $i = 1, 2, \cdots, n$, 则总效益 $c_1x_1 + c_2x_2 + \cdots + c_nx_n$ 显然也是随机的. 我们的目标是在不小于概率 β 的情况下, 使总效益达到最大值, 即极大化目标值 \bar{f} 值

$$\text{Pr}\{c_1x_1 + c_2x_2 + \cdots + c_nx_n \geq \bar{f}\} \geq \beta.$$

这样机会约束整数规划为

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \text{Pr}\{c_1x_1 + c_2x_2 + \cdots + c_nx_n \geq \bar{f}\} \geq \beta \\ a_1x_1 + a_2x_2 + \cdots + a_nx_n \leq a \\ b_1x_1 + b_2x_2 + \cdots + b_nx_n \leq b \\ \text{Pr}\{\eta_ix_i \geq \xi_i\} \geq \alpha_i, \quad i = 1, 2, \cdots, n \\ x_i, i = 1, 2, \cdots, n, \quad \text{非负整数.} \end{array} \right. \quad (6.27)$$

现在我们考虑下面给定的目标值和优先结构:

优先级 1 资金目标, 机器所需的总费用尽可能的不超过所规定的数量, 即

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n + d_1^- - d_1^+ = a,$$

其中 d_1^+ 将被极小化.

优先级 2 空间目标, 机器所使用的空间尽可能的不超过实际有效的空间, 即

$$b_1x_1 + b_2x_2 + \cdots + b_nx_n + d_2^- - d_2^+ = b,$$

其中 d_2^+ 将被极小化.

优先级 3 效益目标, 总效益尽可能的达到给定水平 c (若 c 非常大, 则目标意味着尽可能使效益达到最大)

$$c_1x_1 + c_2x_2 + \cdots + c_nx_n + d_3^- - d_3^+ = c,$$

其中 d_3^- 将被极小化.

同时, 我们假设满足需求的概率不小于 α , 即

$$\Pr \{ \eta_i x_i \geq \xi_i, i = 1, 2, \cdots, n \} \geq \alpha,$$

其中所有变量 $x_i, i = 1, 2, \cdots, n$ 是非负整数. 则有机会约束目标规划如下,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^+, d_2^+, d_3^-\} \\ \text{s.t.} \\ a_1x_1 + a_2x_2 + \cdots + a_nx_n + d_1^- - d_1^+ = a \\ b_1x_1 + b_2x_2 + \cdots + b_nx_n + d_2^- - d_2^+ = b \\ c_1x_1 + c_2x_2 + \cdots + c_nx_n + d_3^- - d_3^+ = c \\ \Pr \{ \eta_i x_i \geq \xi_i, i = 1, 2, \cdots, n \} \geq \alpha \\ x_i, i = 1, 2, \cdots, n, \text{ 非负整数.} \end{array} \right. \quad (6.28)$$

理论上, 如果所有 a_i, b_i 和 c_i 均为随机变量, $i = 1, 2, \dots, n$, 则可以分别对预算、空间和效益目标置以置信水平 β_1, β_2 和 β_3 . 这样, 我们有如下的机会约束目标规划,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^+, d_2^+, d_3^-\} \\ \text{s.t.} \\ \Pr\{a_1x_1 + a_2x_2 + \dots + a_nx_n + d_1^- - d_1^+ = a\} \geq \beta_1 \\ \Pr\{b_1x_1 + b_2x_2 + \dots + b_nx_n + d_2^- - d_2^+ = b\} \geq \beta_2 \\ \Pr\{c_1x_1 + c_2x_2 + \dots + c_nx_n + d_3^- - d_3^+ = c\} \geq \beta_3 \\ \Pr\{\eta_ix_i \geq \xi_i, i = 1, 2, \dots, n\} \geq \alpha \\ x_i, i = 1, 2, \dots, n, \quad \text{非负整数.} \end{array} \right.$$

第 7 章

相关机会规划

有时，一个复杂的系统通常要承担多项任务，这里称之为事件，决策者往往希望这些事件实现的概率（即机会函数）尽可能地大。为了解决这类问题，文献 [85] 提出了第三类随机规划的理论框架，包括相关机会规划以及相关机会多目标规划和相关机会目标规划。

简单地说，相关机会规划是使事件的机会函数在不确定环境下达到最优的问题。在确定性规划以及期望值模型和机会约束规划中，当对实际问题建模以后，可行集本质上则已经确定，然而所给出的最优解在实际中可能根本无法执行。于是相关机会规划并不假定可行集是确定的，虽然相关机会规划也给出一个确定的解，但这个解只是要求在实际问题中尽可能地被执行。显然，相关机会规划的这一特点与前面所提到的两种随机规划是截然不同的。然而，这类问题在实际生活中是确实存在的，有关相关机会规划的实例可以参看 [82, 83, 86] 等文献。

本章里，将介绍相关机会规划中所涉及到的一些概念如不确定环境、事件、机会函数和诱导约束等。同时，提供相关机会规划以及相关机会多目标规划和相关机会目标规划的理论框架。最后，设计一个求解相关机会规划的基于随机模拟的遗传算法，并给出数值例子解释遗传算法的有效性。

7.1 背景：供给 - 分配系统

许多系统（包括供给 - 分配系统）可以表示成图 7.1 的形式。

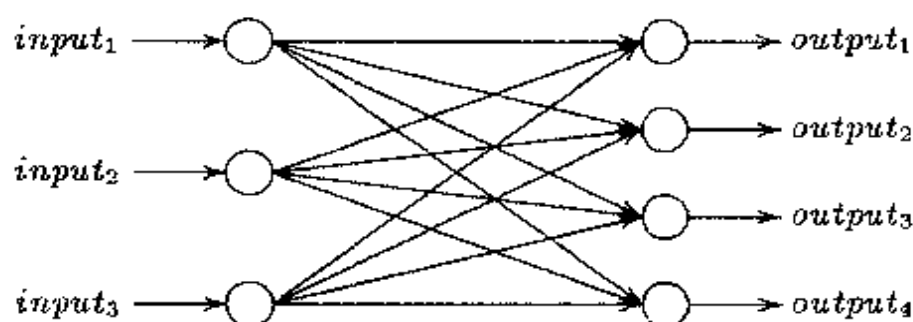


图 7.1 供给 - 分配系统

作为一个例子, 在图 7.1 中, 3 个输入代表 3 个水库, 4 个输出代表 4 个用户的需求. 对此系统, 我们面对的问题有两个: (a) 供给问题, 为了能够达到未来的给定目标, 决策者必须考虑现在和未来所要采取的行动, 即必须提前确定输入的最优组合, 如确定 3 个输入的最优量. (b) 分配问题, 一个最基本的分配问题是资源的最优分配, 这部分的任务是确定资源的最佳组合以达到确定的目标. 对于这个系统, 供给和分配问题当然是不可分割的.

为了回答以上两个问题, 使用 12 个变量 x_1, x_2, \dots, x_{12} 表示一个行动, 其中 x_1, x_2, x_3, x_4 分别表示 4 个用户从资源 $input_1$ 订购的数量, x_5, x_6, x_7, x_8 分别表示从资源 $input_2$ 订购的数量, $x_9, x_{10}, x_{11}, x_{12}$ 分别表示从资源 $input_3$ 订购的数量. 实际上, 由于一些客观存在的约束, 有些变量可能是不存在的 (如在图中某个连通不存在).

注意到我们所使用的外部资源有它们自己的性质. 例如, 用变量 ξ_1, ξ_2, ξ_3 表示 3 处资源所能提供的最大数量, 则它们有可能是随机变量, 不妨设其密度函数分别为 ϕ_1, ϕ_2, ϕ_3 . 这样, 首先得

到下面的约束,

$$\begin{cases} x_1 + x_2 + x_3 + x_4 \leq \xi_1 \\ x_5 + x_6 + x_7 + x_8 \leq \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \leq \xi_3. \end{cases} \quad (7.1)$$

如果 ξ_1, ξ_2 和 ξ_3 中至少有一个是随机变量, 那么这个约束的意义是不明确的, 因为无法在 ξ_1, ξ_2 和 ξ_3 实现之前知道所制定的决策是否是可行的. 我们称这类约束为不确定环境. 在本例中的不确定环境又称为随机环境 (另一种情况是模糊环境). 另外一个显然的约束是

$$x_i \geq 0, \quad i = 1, 2, \dots, 12, \quad (7.2)$$

表示从资源处订购的数量必须是非负的.

由于研究的问题是一个决策系统, 毫无疑问存在一些决策准则. 我们将把决策准则看成是对系统中事件的评价. 例如, 4 个用户的需求量分别记为 c_1, c_2, c_3 和 c_4 , 那么就有下列 4 个事件,

$$\begin{aligned} x_1 + x_5 + x_9 &= c_1, \\ x_2 + x_6 + x_{10} &= c_2, \\ x_3 + x_7 + x_{11} &= c_3, \\ x_4 + x_8 + x_{12} &= c_4. \end{aligned} \quad (7.3)$$

这些等式意味着决策应当满足用户的需求. 鉴于系统的不确定性, 在随机变量实现之前, 我们并不知道决策是否真的能够实现, 所以必须使用所谓的机会函数去评价这 4 个事件. 设

$$\begin{aligned} f_1(\mathbf{x}) &= \text{Pr}\{x_1 + x_5 + x_9 = c_1\}, \\ f_2(\mathbf{x}) &= \text{Pr}\{x_2 + x_6 + x_{10} = c_2\}, \\ f_3(\mathbf{x}) &= \text{Pr}\{x_3 + x_7 + x_{11} = c_3\}, \\ f_4(\mathbf{x}) &= \text{Pr}\{x_4 + x_8 + x_{12} = c_4\}, \end{aligned} \quad (7.4)$$

其中 $\text{Pr}\{\cdot\}$ 表示事件 $\{\cdot\}$ 的概率.

通常, 希望极大化这 4 个机会函数 $f_1(x)$, $f_2(x)$, $f_3(x)$ 和 $f_4(x)$, 即尽可能地提高这 4 个事件实现的可靠性水平. 这里提醒读者, 虽然决策向量 x 不是随机向量, 但它拥有一些不确定性质. 这一点有别于期望值模型和机会约束规划的思想. 事实上, 决策向量的随机性是由不确定环境中的随机参数 ξ_1, ξ_2, ξ_3 和 ξ_4 引起的.

另外, 由于一些经济或政治上的原因, 希望尽可能地少使用资源 input_3 . 这样, 第 5 个事件是极小化从资源 input_3 的订购数量, 即

$$\min f_5(x) = x_9 + x_{10} + x_{11} + x_{12}, \quad (7.5)$$

而这是一个确定的目标函数.

这样, 可以把不确定环境中的供给-分配问题建成如下的随机规划,

$$\left\{ \begin{array}{l} \max f_1(x) = \text{Pr}\{x_1 + x_5 + x_9 = c_1\} \\ \max f_2(x) = \text{Pr}\{x_2 + x_6 + x_{10} = c_2\} \\ \max f_3(x) = \text{Pr}\{x_3 + x_7 + x_{11} = c_3\} \\ \max f_4(x) = \text{Pr}\{x_4 + x_8 + x_{12} = c_4\} \\ \min f_5(x) = x_9 + x_{10} + x_{11} + x_{12} \\ \text{s.t.} \\ \quad x_1 + x_2 + x_3 + x_4 \leq \xi_1 \\ \quad x_5 + x_6 + x_7 + x_8 \leq \xi_2 \\ \quad x_9 + x_{10} + x_{11} + x_{12} \leq \xi_3 \\ \quad x_i \geq 0, \quad i = 1, 2, \dots, 12, \end{array} \right. \quad (7.6)$$

其中, ξ_1, ξ_2 和 ξ_3 为随机变量, 其概率密度函数分别为 ϕ_1, ϕ_2 和 ϕ_3 . 在这个随机规划模型中, 不能单独考虑每个决策分量, 例如,

必须同时考虑 x_1, x_2, x_3 和 x_4 . 这意味着有些变量以及它们的机会函数是随机相关的. 因此, 我们称如 (7.6) 的随机规划为相关机会规划.

使用相关机会规划这个术语的原因有: i) 一些约束是随机的; ii) 一些目标是事件的机会函数; iii) 一些决策变量是随机相关的; iv) 一些事件的机会函数是相关的. 机会相关意味着我们不能单独考虑每一个事件的可靠性.

7.2 随机集合

在确定性规划、期望值模型和机会约束规划中, 可行集通常表示成由 X 中的元素 x 构成的确定性集合. 每个元素即可以属于一个集合 A , 也可以不属于集合 A . 这样的确定性集合可以用不同的方法描述: 可以把集合中的每一个元素一一列出; 通过一组等式或不等式 (约束) 描述集合; 或通过定义特征函数, 表明元素与集合之间的隶属关系, 如 1 表示元素是集合中的或员, 0 表示不是集合中的成员. 作为一个推广, 使用概率函数描述一个可行集. 概率为 1 表示陈述 x 属于 A 为真, 0 表示陈述为假. 当可行集的界取决于随机因素时, 概率函数的值在 0 和 1 之间变化. 这种情况出现在随机规划的随机约束中. 使用随机集合这一概念来描述这种集合. 类似模糊集合的定义, 下面给出随机集合的定义.

定义7.1 假设 Ω 是由全体元素 x 构成的集合, 集合 A 称为 Ω 中的随机集合, 若

$$A = \{(x, \mu_A(x)) \mid x \in \Omega\}, \quad (7.7)$$

其中 $\mu_A(x)$ 称为元素 x 属于 A 的概率.

在上面的定义中, $\mu_A(x)$ 是元素 x 属于 A 的概率的意思是在某些给定的随机环境中 A 中的元素 x 能够实现的概率是 $\mu_A(x)$.

例如, 假设一座水库和一条河的蓄水量分别为随机变量 ξ_1 和 ξ_2 , 概率密度函数分别为 ϕ_1 和 ϕ_2 . 那么能够从水库和河中抽取多少水呢? 显然, 我们不能确定地回答这个问题. 假设 $x = (x_1, x_2)$ 为一向量, 第一个分量 x_1 表示从水库中抽取的水量, 第二个分量 x_2 表示从河中抽取的水量. 理论上, 向量 x 的所有可能实现值来构成欧几里德空间的第一象限. 设 $\Omega = \mathbb{R}_2^+$, 表示 \mathbb{R}^2 中的第一象限, A 表示满足条件 $1 \leq x_1 + x_2 \leq 5$ 的向量. 考虑到约束 $x_1 \leq \xi_1$, $x_2 \leq \xi_2$, 即所抽取的水量不能超出实际中存在的水量. 这样, A 是一个随机集合, 其概率函数为

$$\mu_A(x) = \Pr \{0 \leq x_1 \leq \xi_1; 0 \leq x_2 \leq \xi_2; 1 \leq x_1 + x_2 \leq 5\},$$

等价于

$$\mu_A(x) = \begin{cases} \int_{x_1}^{+\infty} \phi_1(\xi) d\xi \cdot \int_{x_2}^{+\infty} \phi_2(\xi) d\xi, & \begin{matrix} x_1 \geq 0, x_2 \geq 0 \\ 1 \leq x_1 + x_2 \leq 5 \end{matrix} \\ 0, & \text{其它.} \end{cases}$$

注意到概率函数 $\mu_A(x)$ 由两个因素确定, 一个来自于随机变量的实现, 如, $x_1 \leq \xi_1$ 和 $x_2 \leq \xi_2$, 另一个来自于决策者或实际系统给定的约束, 如 $x_1 \geq 0, x_2 \geq 0$ 和 $1 \leq x_1 + x_2 \leq 5$.

由随机集合的定义, 知道 $\mu_A(x) = 0$ 意味着决策 x 几乎是不可能实现的; $\mu_A(x) = 1$ 意味着对任意的随机变量, 决策 x 几乎均能实现; $\mu_A(x) = \alpha (0 < \alpha < 1)$ 表示决策 x 实现的概率为 α .

如果 $\mu_A(x)$ 仅把 Ω 映射到两点 0 和 1, 则 A 是一个通常的确定集合, 并且此时的概率函数 $\mu_A(x)$ 等同于特征函数.

类似于模糊集理论中的 α 水平截集的概念, 这里给出随机集合的 α 水平截集的定义.

定义7.2 由所有属于随机集合 A 且概率不小于 α 的元素构成的集合称为随机集合 A 的 α 水平截集, 表示为

$$A_\alpha = \{x \in \Omega \mid \mu_A(x) \geq \alpha\}. \quad (7.8)$$

考虑具有如下形式的机会约束规划,

$$\begin{cases} \max E f(x, \xi) \\ \text{s.t.} \\ \Pr\{g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (7.9)$$

其中, x 是决策向量, ξ 是随机向量, $f(x, \xi)$ 是目标函数, E 表示关于随机向量 ξ 的期望值算子, $g_j(x, \xi)$ 表示随机约束函数, $j = 1, 2, \dots, p$, $\Pr\{\cdot\}$ 表示事件 $\{\cdot\}$ 的概率, α 表示事先给定的置信水平. 所以一个点 x 是可行的当且仅当集合 $\{\xi \mid g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\}$ 的概率测度不小于 α . 换句话说, 违反约束的概率最大为 $(1 - \alpha)$. 显然, 可以用随机集合

$$S = \{(x, \mu_S(x)) \mid x \in R^n\} \quad (7.10)$$

表示随机约束 $g_j(x, \xi) \leq 0, j = 1, 2, \dots, p$, 其中概率函数 $\mu_S(x)$ 定义为

$$\mu_S(x) = \Pr\{g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\}, \quad (7.11)$$

则 (7.9) 中的约束集合是由 (7.10) 定义的随机集合 S 的 α 水平截集. 所以机会约束规划模型 (7.9) 又可以表示为

$$\max_{x \in S_\alpha} E f(x, \xi). \quad (7.12)$$

注意到当 $\alpha = 1$ 时, 既然 $x \in S_1$ 对任意随机变量 ξ 总是可行的, 所以此时的最优解是一个安全解, 然而, 这样的解在实际中一般是不可接受的. 这种形式的机会约束规划也并没有比 (7.9) 多做什么.

7.3 不确定环境

不同于确定性规划以及期望值模型和机会约束规划, 在一个不确定环境如

$$g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, \quad j = 1, 2, \dots, p \quad (7.13)$$

中讨论问题时, 我们不能说一个点 \mathbf{x} 是可行的还是不可行的, 只能说一个点 \mathbf{x} 可行的概率是 α , 其中 α 是概率函数

$$\mu_S(\mathbf{x}) = \Pr \{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, \quad j = 1, 2, \dots, p\} \quad (7.14)$$

的值. 设 $\mathbf{x} = (x_1, x_2, \dots, x_n)$. 为了方便起见, 在本书中, 我们称 \mathbf{x} 为决策向量, 而 $x_i, i = 1, 2, \dots, n$ 为决策分量.

在第 7.1 节中我们介绍一些决策分量可能是随机相关的. 我们说两个分量 x' 和 x'' 是随机相关的, 若 x' 的实现依赖于 x'' 的实现, 反之亦然. 这里相关的意思是 x' 和 x'' 中任何一个的机会函数至少是 x' 和 x'' 的函数. 然而, 概率函数本身并不能提供分量之间是否相关的信息. 所以, 必须从不确定环境中分析它们.

考虑不确定环境 (7.1), 可以发现 $\{x_1, x_2, x_3, x_4\}$ 是一组随机相关的分量, 它们中任意一个的实现都依赖于其它分量的实现. 当我们要同时实现它们的时候, 它们中的任意一个的机会都是 x_1, x_2, x_3 和 x_4 的函数. 更进一步, 有

$$f(x_i) = \int_{x_1+x_2+x_3+x_4}^{+\infty} \phi_1(\xi) d\xi, \quad i = 1, 2, 3, 4,$$

其中 $f(x_i)$ 是变量 x_i 能够实现的机会函数, $i = 1, 2, 3, 4$. 类似地, $\{x_5, x_6, x_7, x_8\}$ 和 $\{x_9, x_{10}, x_{11}, x_{12}\}$ 是另外的两组随机相关的变量. 另一方面, 上述三组之间不存在任何随机关系, 这一性质对于讨论多目标的情况是非常重要的.

为求解相关机会规划, 如下的决策变量之间的随机关系 [85] 应是已知的.

随机关系 假设一个决策向量的 n 个分量能够被划分成 k 组, 这 k 组之间是相互随机独立的, 而每组内部的分量却是随机相关的, 并且当每组的分量同时要求实现时, 则它们有相同的实现机会.

例如, 对供给 - 分配系统, 若仅仅需要满足用户 1 的需求量 $output_1$ 和用户 2 的需求量 $output_2$, 那么仅分量 x_1, x_2, x_5, x_6, x_9 和 x_{10} 要求同时实现. 根据随机关系, 所有的 12 个决策分量可以划分成 $\{x_1, x_2, x_3, x_4\}$, $\{x_5, x_6, x_7, x_8\}$ 和 $\{x_9, x_{10}, x_{11}, x_{12}\}$, 又由于决策分量 x_3 和 x_4 并没有要求实现, 我们知道 x_1 和 x_2 是随机相关的, 它们实现的机会是相同的且为

$$f(x_i) = \int_{x_1+x_2}^{+\infty} \phi_1(\xi) d\xi, \quad i = 1, 2.$$

对 $\{x_5, x_6\}$ 和 $\{x_9, x_{10}\}$, 情况是类似的.

实际上, 这个问题的随机关系是十分清楚的, 并且也比较容易得到. 如果我们不把随机集合上的这种随机关系引入计算机程序, 求解相关机会多目标规划和相关机会目标规划的算法将无法工作. 虽然不确定环境中的随机关系本身对人来说比较容易确认, 但目前还没有发现一种方法可以教会计算机如何自动识别相关机会规划模型中的随机关系.

7.4 事件和机会函数

对一个随机决策系统, 根据不同的评价准则, 有三种类型的目标函数.

- (a) 期望值目标. 例如, 期望费用极小化, 期望效益极大化等等.
- (b) 机会约束下的目标, 即在约束满足给定的置信水平前提条件下, 优化所给定的目标.
- (c) 事件的机会, 即给定一个事件, 优化这个事件实现的机会.

类型 (a) 和 (b) 已分别在期望值模型和机会约束规划中予以讨论. 这里, 将考虑类型 (c). 不失一般性, 假设我们的目标是极大化每一个给定事件的机会函数, 否则, 可以讨论此事件补集的机会函数. 前面已经说过, 机会函数是一个事件的概率. 例如, 下列等式

$$x_1 + x_5 + x_9 = c_1$$

表示满足供给 - 分配系统中用户 1 的需求, 用 E 表示这个事件, 则该事件的特征函数为

$$\mu_E(\mathbf{x}) = \begin{cases} 1, & x_1 + x_5 + x_9 = c_1 \\ 0, & \text{其它.} \end{cases}$$

考虑这样一个问题: 在不确定环境中如何确定一个事件的概率? 如何表示一个事件的机会函数?

回到我们的例子中, 事件 E 由所有可能满足需求的决策变量 $\mathbf{x} = (x_1, x_2, \dots, x_{12})$ 组成, 即 $x_1 + x_5 + x_9 = c_1$. 然而, 我们感兴趣的仅是其中的部分分量 x_1, x_5 和 x_9 , 即我们仅对满足事件 $x_1 + x_5 + x_9 = c_1$ 的变量 x_1, x_5 和 x_9 的实现感兴趣, 其它分量的实现不在考虑范围之内. 用 $V(E)$ 表示决策向量 \mathbf{x} 的所有分量中我们感兴趣的分量构成的集合. 上面我们提到, 事件 $x_1 + x_5 + x_9 = c_1$ 仅涉及到分量 x_1, x_5 和 x_9 , 因此, $V(E) = \{x_1, x_5, x_9\}$.

通常, 一个给定事件 E 的机会函数用 $f(\mathbf{x})$ 表示, 所给定的问题的不确定环境用

$$g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0, \quad j = 1, 2, \dots, p \quad (7.15)$$

表示, 其中 $\boldsymbol{\xi}$ 是随机向量. 用 $V(E)$ 表示决策向量 \mathbf{x} 的分量中与这个事件有关的分量所构成的集合. 既然我们仅对一个事件感兴趣, 称这个事件优先于其它所有可能的事件, 这样, 当 $\mathbf{x} \in E$ (意

思是 $V(E)$ 的分量满足此事件) 时, 机会函数 $f(x)$ 将是 $V(E)$ 中的分量能够实现的最大概率, 若不能满足此事件, 则定义机会函数为 0, 即

$$f(x) = \begin{cases} \max_{y \in E^*} \Pr \left\{ \begin{array}{l} g_j(y, \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\}, & x \in E \\ 0, & x \notin E, \end{cases} \quad (7.16)$$

其中

$$E^* = \left\{ y = (y_1, y_2, \dots, y_n) \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in V(E) \\ j = 1, \dots, n \end{array} \right\}. \quad (7.17)$$

注意对 E^* 中的任意元素 $y = (y_1, y_2, \dots, y_n)$, 其中 $V(E)$ 中的决策分量 y_j 是确定的, 并且等于 x_j , 而其它的分量是任意的.

在实际问题中, 对每一个 $x = (x_1, x_2, \dots, x_n) \in E$, 通常不难确定 $V(E)$ 以外的分量 x_j^* 的值使得 $y^*(x) = (y_1, y_2, \dots, y_n)$, 其中

$$y_j = \begin{cases} x_j, & y_j \in V(E) \\ x_j^*, & y_j \notin V(E), \end{cases} \quad 1 \leq j \leq n, \quad (7.18)$$

满足

$$f(x) = \Pr \{ g_j(y^*(x), \xi) \leq 0, j = 1, 2, \dots, p \}, \quad (7.19)$$

其中 $x \in E$. 因此, 事件 E 的机会函数为

$$f(x) = \begin{cases} \Pr \left\{ \begin{array}{l} g_j(y^*(x), \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\}, & x \in E \\ 0, & x \notin E. \end{cases} \quad (7.20)$$

通常, 点 $y^*(x)$ 一般在一些极端的位置取值. 例如, 在供给 - 分配系统中, 对 $x \in E$, 决策变量中 $V(E)$ 以外的分量 x_j^* 将取值 0. 我们称约束

$$g_j(y^*(x), \xi) \leq 0, \quad j = 1, 2, \dots, p \quad (7.21)$$

为关于事件 E 的诱导约束. 在使用随机模拟计算机机会函数时, 诱导约束起着关键的作用.

7.5 相关机会规划

本节里, 将考虑单目标相关机会规划, 其基本形式可以表示成如下形式:

$$\max_{x \in S} f(x) \quad (7.22)$$

或等价地表示成在不确定环境下, 极大化机会函数 $f(x)$, 即

$$\begin{cases} \max f(x) \\ \text{s.t.} \\ g_j(x, \xi) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (7.23)$$

其中 x 是 n 维决策向量, S 是 \mathbb{R}^n 上的随机集合, 概率函数为

$$\mu_S(x) = \Pr \{g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\},$$

这里并没有假设已知第 7.3 节中所提到的随机关系, 因为对求解单目标相关机会规划来说, 这种随机关系是不需要的, $f(x)$ 是某个事件的机会函数. 引进经典集合理论中的将号 \in , $x \in S$ 的意思是 x 可行的概率为 $\mu_S(x)$. 值得注意的是集合 S 是随机的而不是确定的. 一个点 $x^* \in S$ 称为 (7.22) 的最优解, 若对任意的 $x \in S$, 有 $f(x^*) \geq f(x)$. 用 E 表示 x 中满足事件的分量构成的集合, 则机会函数 $f(x)$ 可以由 (7.16) 或 (7.20) 确定.

现在回到供给-分配系统. 假设随机约束是 (7.1) 以及 $x_i \geq 0$, $i = 1, 2, \dots, 12$. 则这个约束可以由随机集合 S 及概率函数

$$\mu_S(\mathbf{x}) = \begin{cases} \prod_{k=1}^3 \int_{x^{(k)}}^{+\infty} \phi_k(\xi) d\xi, & x_i \geq 0, \forall i \\ 0, & \text{其它} \end{cases}$$

表示, 其中 $x^{(1)} = x_1 + x_2 + x_3 + x_4$, $x^{(2)} = x_5 + x_6 + x_7 + x_8$ 且 $x^{(3)} = x_9 + x_{10} + x_{11} + x_{12}$. 这里, 假设 ϕ_1, ϕ_2 和 ϕ_3 分别是具有两个参数的对数正态概率密度函数, 其参数分别为 $u = 1.56, 1.36, 0.95$ 和 $\sigma = 0.56, 0.45, 0.38$. 我们感兴趣的事件是满足用户 1 的需求 $c_1 = 6$, 即 $x_1 + x_5 + x_9 = c_1$. 这样事件 E 可以表示为

$$E = \{\mathbf{x} = (x_1, x_2, \dots, x_{12}) \mid x_1 + x_5 + x_9 = c_1 = 6\}.$$

于是事件 E 的机会函数为

$$f(\mathbf{x}) = \begin{cases} \max_{\mathbf{y} \in E^*} \Pr \left\{ \begin{array}{l} x_1 + x_2 + x_3 + x_4 \leq \xi_1 \\ x_5 + x_6 + x_7 + x_8 \leq \xi_2 \\ x_9 + x_{10} + x_{11} + x_{12} \leq \xi_3 \end{array} \right\}, & \mathbf{x} \in E \\ 0, & \mathbf{x} \notin E. \end{cases}$$

对任意给定决策的 \mathbf{x} , 既然仅有一个事件 E , 则事件 E 在点 \mathbf{x} 的诱导约束集应为

$$x_1 \leq \xi_1, \quad x_5 \leq \xi_2, \quad x_9 \leq \xi_3, \quad (7.24)$$

若 $i \neq 1, 5, 9$, 则取 $x_i = 0$, 并且机会函数 $f(\mathbf{x})$ 可以表示为

$$f(\mathbf{x}) = \begin{cases} \Pr \{x_1 \leq \xi_1, x_5 \leq \xi_2, x_9 \leq \xi_3\}, & \mathbf{x} \in E \\ 0, & \mathbf{x} \notin E. \end{cases}$$

注意到

$$\Pr \left\{ \begin{array}{l} x_1 \leq \xi_1 \\ x_5 \leq \xi_2 \\ x_9 \leq \xi_3 \end{array} \right\} = \int_{x_1}^{+\infty} \phi_1(\xi) d\xi \int_{x_5}^{+\infty} \phi_2(\xi) d\xi \int_{x_9}^{+\infty} \phi_3(\xi) d\xi.$$

从而该模型等价于

$$\left\{ \begin{array}{l} \max f(x) \\ \text{s.t.} \\ x_i \geq 0, i = 1, 2, \dots, 9 \end{array} \right.$$

这是一个常规的数学规划.

经过编程运算, 计算机对此问题给出的最优解为

$$x^* = (x_1^*, x_2^*, \dots, x_{12}^*),$$

其中除了

$$x_1^* = 2.6, \quad x_5^* = 2.1, \quad x_9^* = 1.3$$

以外, 其余的所有 x_i^* 均为零, 目标函数值为 $f(x^*) = 0.71$, 即在给定的 x^* 的优先结构的情况下, 此事件实现的概率是 71%.

7.6 相关机会多目标规划

既然一个复杂的决策系统通常需要从事多项任务, 毫无疑问, 决策过程中存在着多个可能的目标函数 (它们中间有些可能是机会函数, 例如, 极大化每个事件实现的机会). 一个典型的相关机会多目标规划可以表示成如下的形式,

$$\max_{x \in S} f(x) = [f_1(x), f_2(x), \dots, f_m(x)], \quad (7.25)$$

或表示成在不确定环境下, 极大化机会函数向量, 即

$$\begin{cases} \max [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ g_j(\mathbf{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (7.26)$$

其中 \mathbf{x} 是一个 n 维决策向量, S 是一个随机集合, 其概率函数为 $\mu_S(\mathbf{x})$, 并假设随机关系是已知的, $f(\mathbf{x})$ 是由实函数 f_i , $i = 1, 2, \dots, m$ 构成的向量, 其中有些是机会函数, 符号 $\mathbf{x} \in S$ 的意思是决策 \mathbf{x} 实现的概率是 $\mu_S(\mathbf{x})$.

上面给出了计算单目标相关机会规划的计算过程. 下面, 考虑多目标规划的情况. 如果 $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})$ 有些不是机会函数, 很容易得到它们的值. 不失一般性, 假设所有的

$$f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})$$

均为机会函数. 每一个机会函数 $f_i(\mathbf{x})$ 表示成一个确定事件

$$E_i = \{\mathbf{x} = (x_1, x_2, \dots, x_n) \mid \mathbf{x} \text{ 满足事件 } i\} \quad (7.27)$$

的概率, $1 \leq i \leq m$. 记

$$E = E_1 \cap E_2 \cap \dots \cap E_m, \quad (7.28)$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_m), \quad (7.29)$$

其中, $\mathbf{x} \in E$ 意味着决策向量 \mathbf{x} 满足所有的事件 E_1, E_2, \dots, E_m , $V(E_i)$ 分别表示满足事件 i 的所必须的分量构成的集合, $i = 1, 2, \dots, m$. 这样, $V(E)$ 表示满足 m 个事件所必须的分量构成的集合. 同时, 定义 E^*, E_i^* 如下,

$$E^* = \{\mathbf{y} = (y_1, y_2, \dots, y_n) \mid y_j = x_j \text{ 若 } y_j \in V(E), 1 \leq j \leq n\}.$$

$$E_i^* = \{\mathbf{y} = (y_1, y_2, \dots, y_n) \mid y_j = x_j \text{ 若 } y_j \in V(E_i), 1 \leq j \leq n\},$$

$i = 1, 2, \dots, m$, 即决策分量 $y_j \in V(E)$ 是固定的, 并且等于 x_j , 其余的分量是任意的.

根据随机关系, 设 $D(E)$ 和 $D(E_i)$ 分别表示与集合 $V(E)$ 和 $V(E_i)$ ($i = 1, 2, \dots, m$) 中至少一个元素随机相关的分量构成的集合. 这样, 有 $V(E) \subset D(E)$ 和 $V(E_i) \subset D(E_i)$, $i = 1, 2, \dots, m$.

通常, 有

$$\max_{y \in E_i^*} \mu_S(y) \geq f_i(x) \geq \max_{y \in E^*} \mu_S(y), \quad i = 1, 2, \dots, m, \quad (7.30)$$

其中第一个不等式的意思是事件 E_i 在多个事件的情况下实现的概率不超过只有一个事件的情况, 这是因为我们不能以牺牲其它事件作为代价; 第二个不等式的意思是在多个事件的情况下, 事件 E_i 实现的概率不小于所有事件实现的概率, 因为有些事件可能是随机独立的, 即有时一个事件的实现并不依赖于其它事件的实现.

现在讨论决策向量与它们各自机会函数之间的关系. 事实上, 我们的方法是尽可能实现每一个事件如 E_i , 而不牺牲其它事件实现的机会. 所以必须等同对待 $V(E_i)$ 的随机相关集合 $D(E_i)$ 中的每一个元素. 另一方面, 我们并不关心 $V(E)$ 以外的元素, 因为它们对我们所关心的事件并不起任何作用. 也就是说, 对事件 E_i , 必须同时考虑而且仅需考虑 $D(E_i) \cap V(E)$ 中的所有元素. 根据随机关系, 知道 $D(E_i) \cap V(E)$ ($V(E)$ 的子集) 中的所有元素随机独立于 $V(E)$ 中其它元素. 因此, 我们可以尽可能地实现 $D(E_i) \cap V(E)$ 中的元素. 从而有

$$f_i(x) = \begin{cases} \max_{y \in E_i^*} \Pr \{g_j(y, \xi) \leq 0, j = 1, 2, \dots, p\}, & x \in E_i \\ 0, & x \notin E_i, \end{cases}$$

其中集合 E_i^* 定义为

$$E_i^* = \left\{ y = (y_1, \dots, y_n) \in E_i \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in D(E_i) \cap V(E) \\ j = 1, \dots, n \end{array} \right\},$$

$i = 1, 2, \dots, m$. 通常, 可以确定 $y_i^*(x)$, 使事件 E_i 在点 x 的诱导约束为

$$g_j(y_i^*(x), \xi) \leq 0, \quad j = 1, 2, \dots, p,$$

从而机会函数 $f_i(x)$ 为

$$f_i(x) = \begin{cases} \Pr \{g_j(y_i^*(x), \xi) \leq 0, j = 1, 2, \dots, p\}, & x \in E_i \\ 0, & x \notin E_i, \end{cases}$$

$i = 1, 2, \dots, m$.

经过建立决策向量与机会函数之间的关系, 我们已可以 (对复杂的随机约束通过随机模拟) 计算机会函数. 如果决策者能够提供完备的偏好信息, 可以求解相关机会多目标规划; 如果决策者不能提供这方面有效的信息, 我们不得不试图寻找所有有效解. 而在实际问题中, 决策者一般只能提供部分信息, 于是可以使用所谓的交互方法, 如可行区域简化法、加权向量空间简化法、准则锥压缩或线性搜索 (见文献 [128]).

7.7 相关机会目标规划

相关机会目标规划可以看成是复杂随机决策系统中目标规划的推广. 当一些管理目标确定以后, 根据决策者给定的优先结构, 目标函数可以是极小化偏差 (包括正偏差、负偏差或正负偏差).

可以把随机决策系统建模为如下的相关机会目标规划,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m \\ \mathbf{x} \in S \end{array} \right. \quad (7.31)$$

或等价地表示成

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ g_j(\mathbf{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{array} \right. \quad (7.32)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

\mathbf{x} = n 维决策向量,

f_i = 目标约束中的机会函数或普通的是实值函数 $\mathbb{R}^n \rightarrow \mathbb{R}^1$,

g_j = 不确定环境中实值函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

S = 定义在 \mathcal{R}^n 上概率函数为 $\mu_S(\mathbf{x})$ 的随机集合, 且已知第 7.3 节给出的随机关系.

解决相关机会目标规划的关键问题是对给定的 \mathbf{x} , 计算所有的 $f_i(\mathbf{x})$. 在第一个优先级中, 假定有 t 个目标, 分别为

$$f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, t.$$

如果 $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_t(\mathbf{x})$ 中有些函数是常规实值函数, 可以立即得到它们的偏差. 不失一般性, 假设它们均为机会函数. 类似于相关机会多目标规划, 机会函数 $f_i(\mathbf{x})$ 表示事件 E_i 的概率. 记

$$E = E_1 \cap E_2 \cap \dots \cap E_t$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_t).$$

考虑到优先级高的目标应尽可能地得到满足, 因此, 有

$$f_i(\mathbf{x}) = \begin{cases} \max_{\mathbf{y} \in E_i^*} \Pr \{g_j(\mathbf{y}, \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\}, & \mathbf{x} \in E_i \\ 0, & \mathbf{x} \notin E_i, \end{cases}$$

其中集合 E_i^* 定义为

$$E_i^* = \left\{ \mathbf{y} = (y_1, \dots, y_n) \in E_i \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in D(E_i) \cap V(E) \\ j = 1, \dots, n \end{array} \right\},$$

$i = 1, 2, \dots, t$. 通常, 能够确定 $\mathbf{y}_i^*(\mathbf{x})$ 使事件 E_i 在点 \mathbf{x} 的诱导约束为

$$g_j(\mathbf{y}_i^*(\mathbf{x}), \boldsymbol{\xi}) \leq 0, \quad j = 1, 2, \dots, p.$$

从而得到机会函数 $f_i(\mathbf{x})$,

$$f_i(\mathbf{x}) = \begin{cases} \Pr \{g_j(\mathbf{y}_i^*(\mathbf{x}), \boldsymbol{\xi}) \leq 0, j = 1, 2, \dots, p\}, & \mathbf{x} \in E_i \\ 0, & \mathbf{x} \notin E_i, \end{cases}$$

$i = 1, 2, \dots, t$.

在第二个优先级中, 假设机会函数为 $f_{t+1}(\mathbf{x}), f_{t+2}(\mathbf{x}), \dots, f_{t'}(\mathbf{x})$, 除了用

$$E = E_1 \cap \dots \cap E_t \cap E_{t+1} \cap \dots \cap E_{t'}$$

和

$$V(E) = V(E_1) \cup \dots \cup V(E_t) \cup V(E_{t+1}) \cup \dots \cup V(E_{t'})$$

取代前面的 E 以外, 类似上面的过程可以得到其它偏差. 这里 $E_{t+1}, E_{t+2}, \dots, E_{t'}$ 是决策向量 \mathbf{x} 中所有满足机会函数

$$f_{t+1}(\mathbf{x}), f_{t+2}(\mathbf{x}), \dots, f_{t'}(\mathbf{x})$$

对应的事件的分量构成的集合. 因为在第一优先级中已经得到满足事件 E_1, E_2, \dots, E_t 的决策分量, 所以在当前决策中必须继承较高一级的决策分量.

重复以上过程, 直到所有的常规实值函数和机会函数计算完为止. 这样, 就能够得到任何给定决策 \mathbf{x} 的所有偏差.

通过以上过程并结合求解非线性目标规划的技术, 就可以求解相关机会目标规划模型.

7.8 执行最优解

在确定性情况以及期望值模型和机会约束规划中, 总是假设解在实际问题中能够实现 (虽然这一假设存在着问题), 所以既可

以一个一个地也可以同时执行解 x 的每一个分量. 理论上, 如何执行是没有什么区别的.

但是, 在随机环境中, 我们不能保证这一点, 因为可行集是随机的而不是确定的. 在随机环境中, 一个重要的问题就是如何执行解 x^* . 为了符合相关机会目标规划的思想, 类似目标规划的优先结构, 在一个解向量的分量中存在着等级之分, 其中等级结构等同于目标结构. 因此级别高的分量要优先于级别低的分量得到满足. 这意味着在第一优先级中首先执行 $V(E_1) \cup V(E_2) \cup \dots \cup V(E_t)$ 中的分量, 然后再执行其余的分量. 在水资源管理系统中, 决策者就是按这种方法执行他们的计划的.

7.9 机会函数的随机模拟

根据上面的分析, $f_i(x)$ 是 $D(E_i) \cap V(E)$ 中决策分量实现的概率. 实际上, 把它转化为不确定环境中的诱导约束并不困难. 设 $g_j(y^*(x), \xi) \leq 0 (j = 1, 2, \dots, p)$ 是事件 E_i 在点 x 的诱导约束. 则机会函数 $f_i(x)$ 由下式给出,

$$f_i(x) = \Pr \{g_j(y^*(x), \xi) \leq 0, j = 1, 2, \dots, p\}.$$

当诱导约束容易处理的时候, 可以使用数值积分的方法. 但如果诱导约束过于复杂而难于处理, 不得不使用随机模拟技术处理机会函数. 不失一般性, 假设 ξ 是随机向量, 其概率分布为 Φ . 对任意给定的决策 x , 使用随机模拟技术估计 $f_i(x)$. 首先, 从概率分布 Φ 中, 生成 N 个独立的随机向量 $\xi_k, k = 1, 2, \dots, N$ (产生随机向量的方法参见第3章). 假设 N 个随机向量中有 N' 个随机向量满足约束

$$g_j(y^*(x), \xi_k) \leq 0, \quad j = 1, 2, \dots, p,$$

$k = 1, 2, \dots, N$. 根据大数定律, $f_i(\mathbf{x})$ 可以用

$$f_i(\mathbf{x}) = \frac{N'}{N} \quad (7.33)$$

估计.

步骤 1 置 $N' = 0$;

步骤 2 从概率分布 $\Phi(\xi)$ 中生成随机向量 ξ ;

步骤 3 若 $g_j(\mathbf{x}, \xi) \leq 0$, $j = 1, 2, \dots, p$, 则 $N'++$;

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 返回 N'/N .

7.10 基于随机模拟的遗传算法

可以把随机模拟技术与现有的计算方法相结合求解相关机会规划、相关机会多目标规划和相关机会目标规划. 考虑到目标函数是非常复杂的, 遗传算法将被采用作为求解技术. 一般地, 基于随机模拟的遗传算法的步骤如下:

遗传算法过程

步骤 0 输入参数 pop_size , P_c 和 P_m ;

步骤 1 初始产生 pop_size 个染色体;

步骤 2 对染色体进行交叉和变异操作;

步骤 3 使用随机模拟方法计算所有染色体的目标值;

步骤 4 根据目标值, 使用基于序的评价函数计算每个染色体的适应度;

步骤 5 旋转赌轮, 选择染色体;

步骤 6 重复步骤 2 到步骤 5, 直到完成给定的循环次数;

步骤 7 所得到的最好的染色体作为最优解.

用来求解相关机会规划的基于随机模拟的遗传算法已编写成 C 语言程序. 这里给出几个在 PC 机上实现的数值例子, 其中所使

用的参数分别为: 种群规模 $pop_size = 30$, 交叉概率 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 基于序的评价函数中参数 $\alpha = 0.05$. 进化过程中每次随机模拟执行 3000 次循环.

例 1 考虑单目标相关机会规划模型

$$\begin{cases} \max f(\mathbf{x}) = \Pr \{x_1 + x_2^2 + x_3^3 = 10\} \\ \text{s.t.} \\ x_1^2 + x_2^2 + x_3^2 + x_4^2 \leq \xi_1 \\ \xi_2(x_1 x_2 x_3) \geq \xi_3 \\ x_1, x_2, x_3, x_4 \geq 0, \end{cases}$$

其中 ξ_1 , ξ_2 和 ξ_3 分别服从均匀分布 $\mathcal{U}[8, 20]$, 正态分布 $\mathcal{N}(5, 1)$ 和指数分布 $\mathcal{EXP}(2)$.

显然, 唯一的任务是在条件 $x_1, x_2, x_3 \geq 0$ 下, 完成 $x_1 + x_2^2 + x_3^3 = 10$. 既然决策变量 x_4 对该事件没有任何贡献, 取 $x_4 = 0$, 因此这个事件的诱导约束为

$$\begin{cases} x_1^2 + x_2^2 + x_3^2 \leq \xi_1 \\ \xi_2(x_1 x_2 x_3) \geq \xi_3. \end{cases}$$

这样, 这个事件的机会函数 $f(\mathbf{x})$ 可以表示为

$$f(\mathbf{x}) = \begin{cases} \Pr \left\{ \begin{array}{l} x_1^2 + x_2^2 + x_3^2 \leq \xi_1 \\ \xi_2(x_1 x_2 x_3) \geq \xi_3 \end{array} \right\}, & \begin{array}{l} x_1 + x_2^2 + x_3^3 = 10 \\ x_1, x_2, x_3 \geq 0 \end{array} \\ 0, & \text{其它.} \end{cases}$$

可以用随机模拟技术对此进行估计. 从超几何体 $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ 中抽取染色体 $V = (x_1, x_2, x_3)$, 通过变换

$$x'_1 = \frac{x_1}{Q}, \quad x'_2 = \frac{x_2}{\sqrt{Q}}, \quad x'_3 = \frac{x_3}{\sqrt[3]{Q}},$$

把它转化为满足事件的可行解, 其中 $Q = (x_1 + x_2^2 + x_3^3)/10$. 显然这能够保证 $x_1' + x_2'^2 + x_3'^3 = 10$.

经过 300 代以后, 基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*) = (1.6304, 1.9400, 1.6638, 0),$$

其可靠性水平为 $f(\mathbf{x}^*) = 89.7\%$.

例 2 考虑相关机会目标规划

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, d_3^-, d_4^+\} \\ \text{s.t.} \\ f_1(\mathbf{x}) + d_1^- - d_1^+ = 0.96 \\ f_2(\mathbf{x}) + d_2^- - d_2^+ = 0.90 \\ f_3(\mathbf{x}) + d_3^- - d_3^+ = 0.90 \\ f_4(\mathbf{x}) + d_4^- - d_4^+ = 0 \\ x_1 + x_2 + x_5 \leq \xi_1 \\ x_3 + x_4 \leq \xi_2 \\ x_6 + x_7 \leq \xi_3 \\ x_i \geq 0, \quad i = 1, 2, \dots, 7 \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, 4, \end{array} \right.$$

其中 ξ_1 , ξ_2 和 ξ_3 分别服从均匀分布 $\mathcal{U}[5, 7]$, 正态分布 $\mathcal{N}(4, 1)$ 和指数分布 $\mathcal{EXP}(3)$, 且

$$\left\{ \begin{array}{l} f_1(\mathbf{x}) = \Pr\{x_1 + x_3^2 = 4\} \\ f_2(\mathbf{x}) = \Pr\{x_2^2 + x_4 = 2\} \\ f_3(\mathbf{x}) = \Pr\{x_5^2 + x_6^2 = 2\} \\ f_4(\mathbf{x}) = x_4 + x_6. \end{array} \right.$$

注意决策变量 x_7 在目标约束中没有出现.

由随机环境, 知道如下的随机关系: 7 个决策变量可以分成 3 组 $\{x_1, x_2, x_5\}$, $\{x_3, x_4\}$ 和 $\{x_6, x_7\}$, 这 3 组之间是相互随机独立的, 而每组内分量是随机相关的且有相同的实现机会.

在第一优先级中, 只有一个事件 $x_1 + x_3^2 = 4$, 记为 E_1 . 显然 $V(E_1) = \{x_1, x_3\}$, 并且事件 E_1 的诱导约束为

$$\begin{cases} x_1 \leq \xi_1 \\ x_3 \leq \xi_2. \end{cases}$$

在当前优先级中, 对事件 E_1 来说, 其它变量可以暂时认为是 0. 这样, E_1 的机会函数为

$$\begin{aligned} f_1(\mathbf{x}) &= \Pr \{x_1 + x_3^2 = 4\} \\ &= \begin{cases} \Pr \{x_1 \leq \xi_1, x_3 \leq \xi_2\}, & x_1 + x_3^2 = 4, \mathbf{x} \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

在第二个优先级中, 有两个事件 $x_2^2 + x_4 = 2$ 和 $x_5^2 + x_6^2 = 2$, 分别记为 E_2 和 E_3 . 同时有 $V(E_2) = \{x_2, x_4\}$ 和 $V(E_3) = \{x_5, x_6\}$, 记 $E = E_1 \cap E_2 \cap E_3$, 则 $V(E) = V(E_1) \cup V(E_2) \cup V(E_3) = \{x_1, x_2, x_3, x_4, x_5, x_6\}$. 既然

$$\begin{aligned} D(E_2) \cap V(E) &= \{x_1, x_2, x_5, x_3, x_4\} \cap \{x_1, x_2, x_3, x_4, x_5, x_6\} \\ &= \{x_1, x_2, x_3, x_4, x_5\}, \end{aligned}$$

事件 E_2 的诱导约束为

$$\begin{cases} x_1 + x_2 + x_5 \leq \xi_1 \\ x_3 + x_4 \leq \xi_2, \end{cases}$$

于是事件 E_2 的机会函数 $f_2(\mathbf{x})$ 为

$$\begin{aligned} f_2(\mathbf{x}) &= \Pr\{x_2^2 + x_4 = 2\} \\ &= \begin{cases} \Pr\left\{\begin{array}{l} x_1 + x_2 + x_5 \leq \xi_1 \\ x_3 + x_4 \leq \xi_2 \end{array}\right\}, & x_2^2 + x_4 = 2, \mathbf{x} \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

由

$$\begin{aligned} D(E_3) \cap V(E) &= \{x_1, x_2, x_5, x_6, x_7\} \cap \{x_1, x_2, x_3, x_4, x_5, x_6\} \\ &= \{x_1, x_2, x_5, x_6\}, \end{aligned}$$

事件 E_3 的诱导约束为

$$\begin{cases} x_1 + x_2 + x_5 \leq \xi_1 \\ x_6 \leq \xi_3, \end{cases}$$

从而事件 E_3 的机会函数 $f_3(\mathbf{x})$ 为

$$\begin{aligned} f_3(\mathbf{x}) &= \Pr\{x_5^2 + x_6^2 = 2\} \\ &= \begin{cases} \Pr\left\{\begin{array}{l} x_1 + x_2 + x_5 \leq \xi_1 \\ x_6 \leq \xi_3 \end{array}\right\}, & x_5^2 + x_6^2 = 2, \mathbf{x} \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

在第三优先级中, 没有机会函数, 所以, 可以立即得到它们的偏差.

从超几何体 $[0, 10]^7$ 中抽取染色体 $V = (x_1, x_2, \dots, x_7)$, 并按下列方法把它转化为满足需求的可行解,

$$x'_1 = \frac{x_1}{Q_1}, \quad x'_2 = \frac{x_2}{\sqrt{Q_2}}, \quad x'_3 = \frac{x_3}{\sqrt{Q_1}},$$

$$x'_4 = \frac{x_4}{Q_2}, \quad x'_5 = \frac{x_5}{\sqrt{Q_3}}, \quad x'_6 = \frac{x_6}{\sqrt{Q_3}}, \quad x'_7 = x_7,$$

其中 $Q_1 = (x_1 + x_3^2)/4$, $Q_2 = (x_2^2 + x_4)/2$, $Q_3 = (x_5^2 + x_6^2)/2$.

经过 500 代以后, 基于随机模拟的遗传算法给出的最优解为

$$\mathbf{x}^* = (2.4071, 1.3862, 1.2621, 0.0786, 1.4142, 0.0004, 0.0000)$$

满足前两个目标但第三个目标值为 0.0790. 更进一步, 有

$$\begin{aligned} f_1(\mathbf{x}^*) &\approx 96\%, & f_2(\mathbf{x}^*) &\approx 90\%, \\ f_3(\mathbf{x}^*) &\approx 90\%, & f_4(\mathbf{x}^*) &= 0.0790. \end{aligned}$$

7.11 注

继期望值模型和机会约束规划之后, 本章介绍了第三类随机规划, 即相关机会规划, 相关机会多目标规划和相关机会目标规划. 对含有随机因素并且多个事件的可靠性水平需要优化的决策系统来说, 相关机会规划是有效的. 同时, 我们也提供了在任何给定决策下计算机会函数的随机模拟方法. 这样, 可以把随机模拟技术和现有的算法相结合求解相关机会规划模型. 考虑到目标函数的复杂性, 给出了基于随机模拟的遗传算法搜索最优解, 并给出了几个数值例子.

其次, 我们需要更多的实验和理论, 以确定决策分量之间的随机关系. 也要把相关机会规划运用到更加广泛的随机环境中的实际管理问题上去. 同时, 设计更加有效的算法以求解大规模决策系统.

第 8 章

随机决策系统建模

在这一章里将提供一些例子，解释如何应用相关机会规划、相关机会多目标规划和相关机会目标规划对复杂的随机决策系统进行建模。将要讨论的问题有：

- (a) 水资源供给 - 分配问题；
- (b) 生产过程；
- (c) 开放存储网络；
- (d) 资金预算。

8.1 水资源供给 - 分配问题

作为一种随机资源分配问题，考虑水资源供给 - 分配问题，如图 8.1 所示。

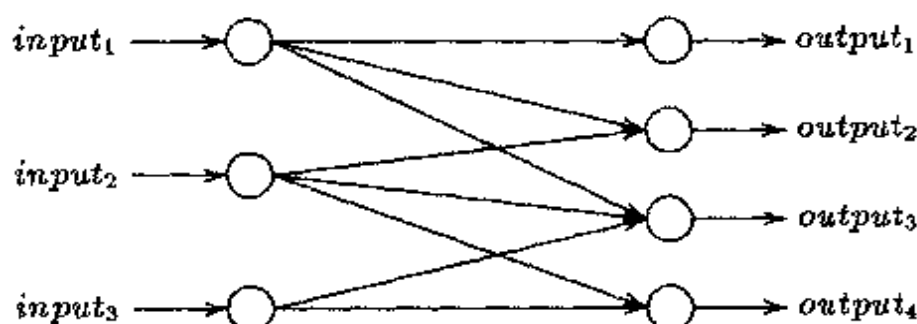


图 8.1 供给 - 分配系统

在图 8.1 中，有 3 处资源 (记为 $input_i, i = 1, 2, 3$) 分别表示 3 个水库，4 个输出 (记为 $output_j, j = 1, 2, 3, 4$) 分别表示 4 个用户

的需求. 我们必须考虑下面两个问题. (a) 供给问题, 为了达到将来预定的目标, 必须提前确定所有资源的最优组合, 即确定从3个水库的引水量; (b) 分配问题, 即确定各种资源的组合, 以满足预定的目标. 当然, 在这个系统中, 供给和分配问题是不可分离的.

为了回答以上两个问题, 使用8个决策变量 x_1, x_2, \dots, x_8 表示一个行动, 其中 x_1, x_2, x_3 分别表示用户 1, 2, 3 从资源 $input_1$ 处的引水量; x_4, x_5, x_6 分别表示用户 2, 3, 4 从资源 $input_2$ 处的引水量; x_7, x_8 分别表示用户 3, 4 从资源 $input_3$ 处的引水量.

注意到我们所使用外部资源有它们自己的特性. 例如三个资源能够提供的最大供水量分别表示为 ξ_1, ξ_2, ξ_3 , 可以合理地假设三个变量是随机变量, 其概率密度函数分别为 ϕ_1, ϕ_2, ϕ_3 . 这样, 首先有下面的随机环境,

$$x_1 + x_2 + x_3 \leq \xi_1, x_4 + x_5 + x_6 \leq \xi_2, x_7 + x_8 \leq \xi_3. \quad (8.1)$$

另一个明显的约束为 $\{x_i \geq 0, i = 1, 2, \dots, 8\}$, 表示引水量是非负的.

既然研究的目标是一个决策系统, 一些决策准则毫无疑问是存在的. 我们认为每一个准则是对这个系统中每一个确定事件的评价. 例如, 4个用户有它们各自的需求, 分别表示为 c_1, c_2, c_3 和 c_4 . 因此有下面的4个事件:

$$x_1 = c_1, x_2 + x_4 = c_2, x_3 + x_5 + x_7 = c_3, x_6 + x_8 = c_4. \quad (8.2)$$

这些等式说明决策应该满足需求. 根据这个系统的不确定性, 在随机变量实现以前, 并不能确定一个决策是否真的能够实现, 所以必须使用机会函数去评价这4个事件. 设

$$\begin{aligned} f_1(x) &= \Pr\{x_1 = c_1\}, & f_2(x) &= \Pr\{x_2 + x_4 = c_2\}, \\ f_3(x) &= \Pr\{x_3 + x_5 + x_7 = c_3\}, & f_4(x) &= \Pr\{x_6 + x_8 = c_4\}, \end{aligned}$$

其中 Pr 表示在 $\{\cdot\}$ 中的事件的概率. 通常, 希望极大化这 4 个机会函数, 即尽可能地提高这 4 个事件实现的概率.

到此为止, 已经把不确定环境下的供给 - 分配问题建模为一个相关机会多目标规划,

$$\left\{ \begin{array}{l} \max f_1(\mathbf{x}) = \text{Pr}\{x_1 = c_1\} \\ \max f_2(\mathbf{x}) = \text{Pr}\{x_2 + x_4 = c_2\} \\ \max f_3(\mathbf{x}) = \text{Pr}\{x_3 + x_5 + x_7 = c_3\} \\ \max f_4(\mathbf{x}) = \text{Pr}\{x_6 + x_8 = c_4\} \\ \text{s.t.} \\ \quad x_1 + x_2 + x_3 \leq \xi_1 \\ \quad x_4 + x_5 + x_6 \leq \xi_2 \\ \quad x_7 + x_8 \leq \xi_3 \\ \quad x_i \geq 0, \quad i = 1, 2, \dots, 8. \end{array} \right. \quad (8.3)$$

于是随机可行集 S 的概率函数定义为

$$\mu_S(\mathbf{x}) = \text{Pr} \left\{ \begin{array}{l} x_1 + x_2 + x_3 \leq \xi_1 \\ x_4 + x_5 + x_6 \leq \xi_2 \\ x_7 + x_8 \leq \xi_3 \\ x_i \geq 0, i = 1, \dots, 8 \end{array} \right\}. \quad (8.4)$$

容易得到下面随机关系: 决策分量分成 3 组, 为 $\{x_1, x_2, x_3\}$, $\{x_4, x_5, x_6\}$ 和 $\{x_7, x_8\}$, 则它们是相互随机独立的, 每一组中元素是随机相关的, 并且有相同的实现机会. 同时也有

$$\begin{aligned} V(E_1) &= \{x_1\}, & V(E_2) &= \{x_2, x_4\}, \\ V(E_3) &= \{x_3, x_5, x_7\}, & V(E_4) &= \{x_6, x_8\}. \end{aligned}$$

现定义 $E = E_1 \cap E_2 \cap E_3 \cap E_4$, 显然有 $V(E) = \{x_1, x_2, \dots, x_8\}$.

由随机关系, 得到

$$D(E_1) = \{x_1, x_2, x_3\}.$$

$$D(E_2) = \{x_1, x_2, x_3, x_4, x_5, x_6\}.$$

$$D(E_3) = \{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\},$$

$$D(E_4) = \{x_4, x_5, x_6, x_7, x_8\}.$$

事件 E_1 的诱导约束为 $\{x_1 + x_2 + x_3 \leq \xi_1\}$. 事件 E_2 的诱导约束为 $\{x_1 + x_2 + x_3 \leq \xi_1, x_4 + x_5 + x_6 \leq \xi_2\}$. 事件 E_3 的诱导约束为 $\{x_1 + x_2 + x_3 \leq \xi_1, x_4 + x_5 + x_6 \leq \xi_2, x_7 + x_8 \leq \xi_3\}$ 以及事件 E_4 的诱导约束为 $\{x_4 + x_5 + x_6 \leq \xi_2, x_7 + x_8 \leq \xi_3\}$. 因此, 对 $\mathbf{x} \in E_1 \cap E_2 \cap E_3 \cap E_4$, 有

$$\begin{cases} f_1(\mathbf{x}) = \Pr \{x_1 + x_2 + x_3 \leq \xi_1\}, \\ f_2(\mathbf{x}) = \Pr \left\{ \begin{array}{l} x_1 + x_2 + x_3 \leq \xi_1 \\ x_4 + x_5 + x_6 \leq \xi_2 \end{array} \right\}, \\ f_3(\mathbf{x}) = \Pr \left\{ \begin{array}{l} x_1 + x_2 + x_3 \leq \xi_1 \\ x_4 + x_5 + x_6 \leq \xi_2 \\ x_7 + x_8 \leq \xi_3 \end{array} \right\}, \\ f_4(\mathbf{x}) = \Pr \{x_4 + x_5 + x_6 \leq \xi_2, x_7 + x_8 \leq \xi_3\}. \end{cases} \quad (8.5)$$

既然 ξ_1 , ξ_2 和 ξ_3 是相互独立的随机变量, 等价地, 对每一个

$x \in E_1 \cap E_2 \cap E_3 \cap E_4$, 有

$$\left\{ \begin{array}{l} f_1(x) = \int_{x^{(1)}}^{+\infty} \phi_1(\xi) d\xi, \\ f_2(x) = \int_{x^{(1)}}^{+\infty} \phi_1(\xi) d\xi \cdot \int_{x^{(2)}}^{+\infty} \phi_2(\xi) d\xi, \\ f_3(x) = \int_{x^{(1)}}^{+\infty} \phi_1(\xi) d\xi \cdot \int_{x^{(2)}}^{+\infty} \phi_2(\xi) d\xi \cdot \int_{x^{(3)}}^{+\infty} \phi_3(\xi) d\xi, \\ f_4(x) = \int_{x^{(2)}}^{+\infty} \phi_2(\xi) d\xi \cdot \int_{x^{(3)}}^{+\infty} \phi_3(\xi) d\xi. \end{array} \right. \quad (8.6)$$

其中 $x^{(1)} = x_1 + x_2 + x_3$, $x^{(2)} = x_4 + x_5 + x_6$, $x^{(3)} = x_7 + x_8$.

根据 (8.5), 通过随机模拟 (或根据 (8.6), 通过数值积分方法), 可以计算机会函数 $f_1(x)$, $f_2(x)$, $f_3(x)$ 和 $f_4(x)$.

假设 4 个用户的需求量分别为 $(c_1, c_2, c_3, c_4) = (1, 2, 2, 3)$, 同时也假设随机参数 ξ_1, ξ_2 和 ξ_3 分别服从双参数对数正态分布 $\text{LOGN}(1.56, 0.56^2)$, $\text{LOGN}(1.36, 0.45^2)$ 和 $\text{LOGN}(0.95, 0.38^2)$. 若 4 个目标的权 (可以解释为决策者的偏好) 分别为 w_1, w_2, w_3, w_4 , 则相关机会多目标规划可以转变为单目标优化问题. 假设

$$(w_1, w_2, w_3, w_4) = (3, 1, 2, 3),$$

基于随机模拟的遗传算法给出的最优解为

$$x^* = (1.00, 1.90, 1.44, 0.10, 0.54, 1.63, 0.02, 1.37),$$

满足 4 个用户需求的机会分别为 62%, 55%, 52% 和 84%.

下面从几个不同的角度考虑此问题. 首先考虑下面决策者给定的优先结构和目标值:

优先级 1 满足第一个用户需求的机会尽可能地达到 90%, 即

$$\Pr\{x_1 = c_1\} + d_1^- - d_1^+ = 0.90,$$

其中 d_1^- 将被极小化.

优先级 2 满足第二个用户需求的机会尽可能地达到 85%, 即

$$\Pr\{x_2 + x_4 = c_2\} + d_2^- - d_2^+ = 0.85,$$

其中 d_2^- 将被极小化.

优先级 3 满足第三个和第四个用户需求的机会尽可能地达到 70% 和 50%, 权系数分别为 2 和 1, 即

$$\Pr\{x_3 + x_5 + x_7 = c_3\} + d_3^- - d_3^+ = 0.70,$$

$$\Pr\{x_6 + x_8 = c_4\} + d_4^- - d_4^+ = 0.50,$$

其中 $2d_3^- + d_4^-$ 将被极小化.

优先级 4 从资源 3 处尽可能地少引水, 即

$$x_7 + x_8 + d_5^- - d_5^+ = 0,$$

其中 d_5^+ 将被极小化.

因此上述问题的相关机会目标规划模型为

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, 2d_3^- + d_4^-, d_5^+\} \\ \text{s.t.} \\ \Pr\{x_1 = c_1\} + d_1^- - d_1^+ = 0.90 \\ \Pr\{x_2 + x_4 = c_2\} + d_2^- - d_2^+ = 0.85 \\ \Pr\{x_3 + x_5 + x_7 = c_3\} + d_3^- - d_3^+ = 0.70 \\ \Pr\{x_6 + x_8 = c_4\} + d_4^- - d_4^+ = 0.5 \\ x_7 + x_8 + d_5^- - d_5^+ = 0 \\ x_1 + x_2 + x_3 \leq \xi_1 \\ x_4 + x_5 + x_6 \leq \xi_2 \\ x_7 + x_8 \leq \xi_3 \\ x_i \geq 0, \quad i = 1, 2, \dots, 8 \\ d_j^-, d_j^+ \geq 0, \quad j = 1, 2, \dots, 5. \end{array} \right. \quad (8.7)$$

在第一优先级中, 仅存在一个目标: $\text{Pr}\{x_1 = c_1\} + d_1^- - d_1^+ = 0.90$. 对这种情况, 有

$$E_1 = \{(x_1, x_2, \dots, x_8) \mid x_1 = c_1\}.$$

这样有 $V(E) = V(E_1) = \{x_1\}$ 及 $D(E_1) = \{x_1, x_2, x_3\}$ 和 $D(E_1) \cap V(E) = \{x_1\}$. 于是事件 E_1 的诱导约束为 $x_1 \leq \xi_1$. 因此, 第一个事件的机会函数为

$$\text{Pr}\{x_1 = c_1\} = \text{Pr}\{x_1 = c_1 \leq \xi_1\}. \quad (8.8)$$

在第二优先级中, 也只有一个目标: $\text{Pr}\{x_2 + x_4 = c_2\} + d_2^- - d_2^+ = 0.85$. 这样, 有

$$E_2 = \{(x_1, x_2, \dots, x_8) \mid x_2 + x_4 = c_2\}$$

以及

$$V(E) = V(E_1) \cup V(E_2) = \{x_1, x_2, x_4\},$$

$$D(E_2) = \{x_1, x_2, x_3, x_4, x_5, x_6\},$$

并且 $D(E_2) \cap V(E) = \{x_1, x_2, x_4\}$. 则事件 E_2 的诱导约束为 $\{x_1 + x_2 \leq \xi_1, x_4 \leq \xi_2\}$. 那么, 第二个事件的机会函数为

$$\text{Pr}\{x_2 + x_4 = c_2\} = \text{Pr}\{x_1 + x_2 \leq \xi_1, x_4 \leq \xi_2\}. \quad (8.9)$$

在第三优先级中, 有两个目标, $\text{Pr}\{x_3 + x_5 + x_7 = c_3\} + d_3^- - d_3^+ = 0.70$ 和 $\text{Pr}\{x_6 + x_8 = c_4\} + d_4^- - d_4^+ = 0.5$. 有

$$E_3 = \{(x_1, x_2, \dots, x_8) \mid x_3 + x_5 + x_7 = c_3\},$$

$$E_4 = \{(x_1, x_2, \dots, x_8) \mid x_6 + x_8 = c_4\}$$

和

$$V(E_3) = \{x_3, x_5, x_7\},$$

$$V(E_4) = \{x_6, x_8\},$$

$$V(E) = V(E_1) \cup V(E_2) \cup V(E_3) \cup V(E_4) = \{x_1, x_2, \dots, x_8\},$$

$$D(E_3) = \{x_1, x_2, \dots, x_8\},$$

$$D(E_4) = \{x_4, x_5, \dots, x_8\}.$$

由 $D(E_3) \cap V(E) = \{x_1, x_2, \dots, x_8\}$ ，得出事件 E_3 的诱导约束为

$$x_1 + x_2 + x_3 \leq \xi_1, \quad x_4 + x_5 + x_6 \leq \xi_2, \quad x_7 + x_8 \leq \xi_3,$$

又由 $D(E_4) \cap V(E) = \{x_4, x_5, \dots, x_8\}$ ，事件 E_4 的诱导约束为

$$x_4 + x_5 + x_6 \leq \xi_2, \quad x_7 + x_8 \leq \xi_3.$$

因此，第3和第4个事件的机会函数分别为

$$\Pr\{x_3 + x_5 + x_7 = c_3\} = \Pr \left\{ \begin{array}{l} x_1 + x_2 + x_3 \leq \xi_1 \\ x_4 + x_5 + x_6 \leq \xi_2 \\ x_7 + x_8 \leq \xi_3 \end{array} \right\} \quad (8.10)$$

和

$$\Pr\{x_6 + x_8 = c_4\} = \Pr \left\{ \begin{array}{l} x_4 + x_5 + x_6 \leq \xi_2 \\ x_7 + x_8 \leq \xi_3 \end{array} \right\}. \quad (8.11)$$

在最后一个优先级中，没有机会函数，所以立即得到偏差的值。在与前面相同的随机参数假设下，基于对 (8.8), (8.9), (8.10) 和 (8.11) 的随机模拟，遗传算法给出的最优解为

$$\mathbf{x}^* = (1.00, 1.67, 1.96, 0.33, 0.03, 1.28, 0.01, 1.72),$$

其满足第一、第二和第四个目标的机会分别为 99%，85% 和 85%，但第三个目标的机会仅为 47%，小于希望的水平 70%，而从资源 3 处的引水总量为 1.72。

8.2 生 产 过 程

重新考虑第 6.1 节的生产过程. 一个冶炼厂制定一周冶炼原油 (分别为 raw_1 和 raw_2) 的计划, 并考虑为天然气公司提供天然气 ($prod_1$) 和为火力发电厂提供燃料 ($prod_2$).

假定产品 $\pi(raw_1, prod_1)$ 和 $\pi(raw_2, prod_2)$ (即来自原油 raw_1 的天然气产量和来自原油 raw_2 的燃料产量) 是随机变化的, 而其它产品的产量是确定的,

$$\begin{aligned}\pi(raw_1, prod_1) &= 2 + \eta_1, & \pi(raw_1, prod_2) &= 3, \\ \pi(raw_2, prod_1) &= 6, & \pi(raw_2, prod_2) &= 3.4 - \eta_2,\end{aligned}$$

其中 η_1 服从均匀分布 $U[-0.8, 0.8]$, η_2 服从指数分布 $\mathcal{EXP}(0.4)$.

用户 (天然气公司和火力发电厂) 一周的需求量分别为天然气 h_1 和燃料 h_2 , 它们也是随机变化的, 分别表示为

$$h_1 = 180 + \xi_1, \quad h_2 = 162 + \xi_2,$$

其中 ξ_1 和 ξ_2 分别服从正态分布 $\mathcal{N}(0, 12)$ 和 $\mathcal{N}(0, 9)$.

原料 raw_1 的单位价格是 $c_1 = 2$, 原料 raw_2 的单位价格是 $c_2 = 3$, 即总的价格为 $2x_1 + 3x_2$, 其中 x_1 和 x_2 是两种原油的数量, 为决策变量.

又假设生产能力, 即原料的最大加工数量为 100, 这样有 $x_1 + x_2 \leq 100$.

如果一周的生产计划 (x_1, x_2) 必须提前制定, 一周之内不能改变, 并且用户希望他们的实际需求在该周内得到满足, 即 $(2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1$ 和 $3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2$. 显然, 这是一个不确定环境.

如果我们希望总费用小于 140 的机会尽可能地高. 那么, 什么样的生产计划才能保证总费用不超过 140 并且满足用户实际需

求的机会尽可能地高呢? 为此, 构造如下的相关机会规划模型,

$$\begin{cases} \max f(\mathbf{x}) = \Pr\{2x_1 + 3x_2 \leq 140\} \\ \text{s.t.} \\ x_1 + x_2 \leq 100 \\ (2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1 \\ 3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2 \\ x_1, x_2 \geq 0. \end{cases} \quad (8.12)$$

对这个单目标相关机会规划, 不需要讨论决策变量之间的随机关系. 所求的最优解 (x_1, x_2) 首先应满足

$$x_1 + x_2 \leq 100, \quad x_1, x_2 \geq 0,$$

然后极大化机会函数

$$f(\mathbf{x}) = \begin{cases} p(\mathbf{x}), & 2x_1 + 3x_2 \leq 140 \\ 0, & 2x_1 + 3x_2 > 140, \end{cases}$$

其中

$$p(\mathbf{x}) = \Pr \left\{ \begin{array}{l} (2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1 \\ 3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2 \end{array} \right\}.$$

基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*) = (34.74, 23.51),$$

其最大机会为 84%, 总生产费用恰为 140.

如果管理者想分别考虑用户的需求实现的水平, 所给定的优先结构和目标值分别为:

优先级 1 总的生产费用尽可能地不超过 140;

优先级 2 满足天然气公司需求的机会尽可能地超过 95%;

优先级 3 满足火力发电厂需求的机会尽可能地超过 90%.

则此问题可以建模为如下的相关机会目标规划,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^+, d_2^-, d_3^-\} \\ \text{s.t.} \\ 2x_1 + 3x_2 + d_1^- - d_1^+ = 140 \\ \Pr\{(2 + \eta_1)x_1 + 6x_2 \geq 180 + \xi_1\} + d_2^- - d_2^+ = 0.95 \\ \Pr\{3x_1 + (3.4 - \eta_2)x_2 \geq 162 + \xi_2\} + d_3^- - d_3^+ = 0.90 \\ x_1 + x_2 \leq 100 \\ x_1, x_2 \geq 0. \end{array} \right.$$

由于机会函数可以通过随机模拟求得, 因此这个模型比较容易处理. 经过 300 代进化, 基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*) = (35.774, 22.817).$$

该解满足前两个目标, 而第三个目标的可靠性为 85%, 小于希望的目标值 90%.

8.3 开放存储网络

回顾第 6.4 节讨论的开放存储网络. 基于相关机会规划的开放存储网络优化操作的最简单模型是

$$\left\{ \begin{array}{l} \max f(\mathbf{x}) = \Pr\{\mathbf{x}\} \\ \text{s.t.} \\ v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, \quad i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{array} \right. \quad (8.13)$$

即寻找一个决策 $\mathbf{x} \in D$ 值决策实现的概率最大. 注意到一个决策 $\mathbf{x} \in D$ 实际是否能够实现当且仅当 ξ 的实现值是否满足 $v_i \leq$

$a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, i = 1, 2, \dots, p$. 事实上, 此模型的目的在于找到最安全的解. 这里值得注意的是, 可行集是随机的而不是确定的, 因此决策 \mathbf{x} 具有某些不确定性, 尽管它本身不是随机向量. 有时可能希望分别极大化 4 个水库的可靠性水平. 一般来说, 我们有 m 个可能的目标 (有些可能是机会函数), 那么我们的问题可以转化为相关机会多目标规划模型,

$$\begin{cases} \max[f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_m(\mathbf{x})] \\ \text{s.t.} \\ v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, \quad i = 1, 2, \dots, p \\ \mathbf{x} \in D. \end{cases} \quad (8.14)$$

为了平衡多个不相容的目标, 根据决策者给定的优先结构和目标值, 开放存储网络也可以建模为相关机会目标规划,

$$\begin{cases} \min \sum_{j=1}^i P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ v_i \leq a_i + t_i(\mathbf{x}) + \xi_i \leq V_i, \quad i = 1, 2, \dots, p \\ \mathbf{x} \in D \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m \end{cases} \quad (8.15)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子的 j 第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子的 j 第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

$\mathbf{x} = n$ 维决策向量,

$f_i =$ 目标约束中的机会函数或普通的实值函数 $\Re^n \rightarrow \Re^1$,

$g_j =$ 不确定环境中的实值函数,

$b_i =$ 目标 i 的目标值,

$l =$ 优先级个数,

$m =$ 目标约束个数.

现在, 应用基于随机模拟的遗传算法求解开放存储网络最优化操作问题的相关机会规划模型. 在遗传算法中, 所使用的参数分别为: 种群规模为 30, 交叉概率 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 基于序的评价函数中的参数 $\alpha = 0.05$, 随机模拟的循环次数为 5000.

现在考虑图 6.2 中的开放存储网络, 假设 9 个弧的流量向量构成的集合 D , 其中元素 $\mathbf{x} = (x_1, x_2, \dots, x_9)$ 的约束为

$$\begin{aligned} 10 \leq x_1 \leq 50, \quad 0 \leq x_2 \leq 10, \quad 0 \leq x_3 \leq 10, \\ 0 \leq x_4 \leq 15, \quad 15 \leq x_5 \leq 60, \quad -5 \leq x_6 \leq 5, \\ 15 \leq x_7 \leq 60, \quad -5 \leq x_8 \leq 5, \quad 20 \leq x_9 \leq 70. \end{aligned}$$

随机径流假设服从两个参数的对数正态分布 $\text{LOGN}(\mu_i, \sigma_i^2)$, $i = 1, 2, 3, 4$, 参数 (μ_i, σ_i) 的值分别为

$$(2.24, 1.12), \quad (1.60, 1.28), \quad (1.87, 1.45), \quad (1.30, 1.34).$$

同时也假设 4 个水库的最小最大库容参数 (v_i, V_i) 分别为 $(10, 120)$, $(20, 100)$, $(10, 80)$ 和 $(0, 90)$.

如果决策者关心的不只是每一个水库的有数性, 而是整个系统的可靠性, 那么管理目标就是找到最安全的解 $\mathbf{x} \in D$. 为解决

此问题, 可以把问题转化为单目标相关机会规划模型,

$$\left\{ \begin{array}{l} \max f(\mathbf{x}) = \Pr\{\mathbf{x}\} \\ \text{s.t.} \\ 10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120 \\ 20 \leq 80 + x_2 - x_5 - x_6 + \xi_2 \leq 100 \\ 10 \leq 60 + x_3 - x_7 - x_8 + \xi_3 \leq 80 \\ 0 \leq 50 + x_4 + x_6 + x_8 - x_9 + \xi_4 \leq 90 \\ (x_1, x_2, \dots, x_9) \in D. \end{array} \right. \quad (8.16)$$

显然, 如果 $\mathbf{x} \in D$, 则机会函数为

$$f(\mathbf{x}) = \Pr \left\{ \begin{array}{l} 10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120 \\ 20 \leq 80 + x_2 - x_5 - x_6 + \xi_2 \leq 100 \\ 10 \leq 60 + x_3 - x_7 - x_8 + \xi_3 \leq 80 \\ 0 \leq 50 + x_4 + x_6 + x_8 - x_9 + \xi_4 \leq 90 \end{array} \right\},$$

否则, $f(\mathbf{x}) = 0$. 经过 1000 代运算以后, 基于随机模拟的遗传算法给出的最优解 (即最安全的解) 为

$$\mathbf{x}^* = (37.1, 2.8, 4.6, 9.2, 54.7, 3.4, 50.7, 3.3, 65.3),$$

其可靠性为 92.7%, 所有流量的总和为 231.1.

现我们在相同的不确定环境下从不同角度考虑这个问题. 设决策者给出的管理目标和优先结构如下:

优先级 1 水库 1 的可靠性尽可能达到希望的水平 96%;

优先级 2 整个网络的可靠性尽可能达到希望的水平 90%;

优先级 3 总流量尽可能地小.

为了满足管理目标, 使用相关机会目标规划模型,

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, d_3^+\} \\ \text{s.t.} \\ f_1(\mathbf{x}) + d_1^- - d_1^+ = 0.96 \\ f_2(\mathbf{x}) + d_2^- - d_2^+ = 0.90 \\ \sum_{i=1}^9 x_i + d_3^- - d_3^+ = 0 \\ 10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120 \\ 20 \leq 80 + x_2 - x_5 - x_6 + \xi_2 \leq 100 \\ 10 \leq 60 + x_3 - x_7 - x_8 + \xi_3 \leq 80 \\ 0 \leq 50 + x_4 + x_6 + x_8 - x_9 + \xi_4 \leq 90 \\ (x_1, x_2, \dots, x_9) \in D \\ d_i^+, d_i^- \geq 0, \quad i = 1, 2, 3, \end{array} \right. \quad (8.17)$$

其中机会函数 $f_1(\mathbf{x})$ 和 $f_2(\mathbf{x})$ 分别表示水库和整个网络正常运转的可靠性. 由随机关系, 容易导出

$$f_1(\mathbf{x}) = \text{Pr} \{10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120\} \quad (8.18)$$

及

$$f_2(\mathbf{x}) = \text{Pr} \left\{ \begin{array}{l} 10 \leq 70 - x_1 - x_2 - x_3 - x_4 + \xi_1 \leq 120 \\ 20 \leq 80 + x_2 - x_5 - x_6 + \xi_2 \leq 100 \\ 10 \leq 60 + x_3 - x_7 - x_8 + \xi_3 \leq 80 \\ 0 \leq 50 + x_4 + x_6 + x_8 - x_9 + \xi_4 \leq 90 \end{array} \right\}. \quad (8.19)$$

经过 2000 代运算以后, 基于随机模拟的遗传算法给出的最优解为

$$\mathbf{x}^* = (31.3, 2.3, 0.0, 3.3, 37.3, -1.0, 48.5, 0.1, 30.1),$$

该解满足前两个目标, 而总流量为 153.9.

8.4 资金预算

在第6章,已经把资金预算问题建模为机会约束整数规划模型.本节里,把此问题建模成相关机会规划模型.有关资金预算的最简单的相关机会规划整数模型为

$$\left\{ \begin{array}{l} \max \Pr \{ \eta_i x_i \geq \xi_i, i = 1, 2, \dots, n \} \\ \text{s.t.} \\ a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \dots + b_n x_n \leq b \\ x_i, i = 1, 2, \dots, n, \text{ 非负整数.} \end{array} \right.$$

我们的目的是找到最安全的解,即使产品满足需求的概率达到最大的解,其中 a 是用来分配的总资金, b 表示总空间, a_i 表示需要分配给类型 i 的机器的资金量, b_i 表示类型 i 的机器需要的空间, η_i 是类型 i 的机器生产产品 i 的生产能力, ξ_i 是产品 i 的未来需求, x_i 分别表示类型 i 的机器被选择的个数, $i = 1, 2, \dots, n$. 所以所有决策变量 x_i 是非负的整数.

有时,希望分别极大化满足各种需求的可靠性水平,这时,资金预算问题可以建模为相关机会多目标规划模型,

$$\left\{ \begin{array}{l} \max [\Pr \{ \eta_1 x_1 \geq \xi_1 \}, \Pr \{ \eta_2 x_2 \geq \xi_2 \}, \dots, \Pr \{ \eta_n x_n \geq \xi_n \}] \\ \text{s.t.} \\ a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \dots + b_n x_n \leq b \\ x_i, i = 1, 2, \dots, n, \text{ 非负整数.} \end{array} \right.$$

为了平衡多个不相容的目标,根据决策者的目标值和优先结

构, 资金预算问题可以建模成下面的相关机会目标规划,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^n (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ \Pr\{\eta_i x_i \geq \xi_i\} + d_i^- - d_i^+ = \alpha_i, \quad i = 1, 2, \dots, n \\ a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \dots + b_n x_n \leq b \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, n \\ x_i, i = 1, 2, \dots, n, \quad \text{非负整数.} \end{array} \right. \quad (8.20)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

x = n 维决策向量,

l = 优先级个数,

α_i = 目标 i 的目标值, 及所希望的可靠性水平.

如果在 (8.20) 中, 参数 a, b, a_i 和 $b_i (i = 1, 2, \dots, n)$ 有些是随机变量, 则可行集

$$\left\{ \begin{array}{l} a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \dots + b_n x_n \leq b \\ x_i, i = 1, 2, \dots, n, \quad \text{非负整数} \end{array} \right.$$

是随机环境.

考虑一个具体问题, 设有 5 种类型的机器, 用来生产 5 种不同的产品. 值设类型 i 的机器的生产能力为服从对数正态分布的

随机变量, 该随机变量的概率密度函数 $\phi_i(\eta_i)$ 为

$$\phi_i(\eta_i) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_i\eta_i} \exp\left[-\frac{(\ln \eta_i - \mu_i)^2}{2\sigma_i^2}\right], & 0 \leq c_i < \infty \\ 0, & \text{其它,} \end{cases}$$

其中 (μ_i, σ_i) 分别为 $(3.0, 1.0)$, $(4.0, 1.6)$, $(5.0, 1.6)$, $(4.0, 1.2)$ 和 $(3.0, 0.8)$, $i = 1, 2, 3, 4, 5$. 同时也假设产品 i 的需求量 ξ_i 分别为服从指数分布的随机变量, 即它们的概率密度函数分别为

$$\psi_i(\xi_i) = \begin{cases} \frac{1}{\beta_i} \exp\left(-\frac{\xi_i}{\beta_i}\right), & 0 \leq d_i < \infty \\ 0, & \text{其它,} \end{cases} \quad (8.21)$$

其中 β_i 分别为 10, 15, 20, 18 和 16, $i = 1, 2, 3, 4, 5$.

分配给类型 i 的机器的资金 a_i 分别为 300, 800, 700, 900 和 1000, $i = 1, 2, \dots, 5$, 用于分配的资金总数 a 为 12500. 另外, 类型为 i 的机器所使用的空间 b_i 分别为 20, 30, 50, 30 和 10, $i = 1, 2, \dots, 5$, 并且最大有效空间 b 为 500.

若我们的目的是找到使产品满足需求的概率最大的资金预算方案, 那么可以把此问题建模为相关机会整数规划模型,

$$\begin{cases} \max \Pr \{ \eta_i x_i \geq \xi_i, i = 1, 2, \dots, 5 \} \\ \text{s.t.} \\ 300x_1 + 800x_2 + 700x_3 + 900x_4 + 1000x_5 \leq 12500 \\ 20x_1 + 30x_2 + 50x_3 + 30x_4 + 10x_5 \leq 500 \\ x_i, i = 1, 2, \dots, 5, \text{ 非负整数,} \end{cases}$$

其中 $x_i, i = 1, 2, \dots, 5$ 为决策变量, 表示类型 i 的机器被选中的台数. 经过 300 代进化以后, 基于随机模拟的遗传算法给出的最优资金预算方案为

$$(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (6, 3, 2, 3, 4),$$

其可靠性水平为 85.06%.

如果考虑下面目标值和优先结构:
严格约束 空间限制,

$$20x_1 + 30x_2 + 50x_3 + 30x_4 + 10x_5 \leq 500.$$

优先级 1 满足需求 ξ_1 的概率水平尽可能地达到 97%, 即

$$\Pr\{\eta_1 x_1 \geq \xi_1\} + d_1^- - d_1^+ = 0.97,$$

其中 d_1^- 将被极小化;

优先级 2 满足需求 ξ_2 的概率水平尽可能地达到 95%, 即

$$\Pr\{\eta_2 x_2 \geq \xi_2\} + d_2^- - d_2^+ = 0.95,$$

其中 d_2^- 将被极小化;

优先级 3 满足需求 ξ_3, ξ_4 和 ξ_5 的概率水平尽可能地达到 90%,
即

$$\Pr\{\eta_i x_i \geq \xi_i, i = 3, 4, 5\} + d_3^- - d_3^+ = 0.90,$$

其中 d_3^- 将被极小化;

优先级 4 购买机器资金总数尽可能地不超过 12500, 即

$$300x_1 + 800x_2 + 700x_3 + 900x_4 + 1000x_5 + d_4^- - d_4^+ = 12500,$$

其中 d_4^+ 将被极小化.

根据以上提到的优先结构和目标值, 可以建立下面的相关机会目

标规划模型,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^-, d_2^-, d_3^-, d_4^+\} \\ \text{s.t.} \\ \Pr\{\eta_1 x_1 \geq \xi_1\} + d_1^- - d_1^+ = 0.97 \\ \Pr\{\eta_2 x_2 \geq \xi_2\} + d_2^- - d_2^+ = 0.95 \\ \Pr\{\eta_i x_i \geq \xi_i, i = 3, 4, 5\} + d_3^- - d_3^+ = 0.90 \\ 300x_1 + 800x_2 + 700x_3 + 900x_4 + 1000x_5 \\ \quad + d_4^- - d_4^+ = 12500 \\ 20x_1 + 30x_2 + 50x_3 + 30x_4 + 10x_5 \leq 500 \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, 4 \\ x_i, i = 1, 2, \dots, 5, \quad \text{非负整数.} \end{array} \right.$$

经过 400 代进化以后, 基于随机模拟的遗传算法给出的最优资金预算方案为

$$(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (5, 4, 2, 4, 4),$$

该解满足前三个目标, 但第 4 个目标为 1200. 实际上, 我们也有

$$\Pr\{\eta_1 x_1^* \geq \xi_1\} = 97.65\%,$$

$$\Pr\{\eta_2 x_2^* \geq \xi_2\} = 95.34\%,$$

$$\Pr\{\eta_i x_i^* \geq \xi_i, i = 3, 4, 5\} = 91.89\%,$$

要分配的资金总数为 13700.

第 9 章

模糊机会约束规划

在理论上存在着两种熟知的不确定环境：随机环境和模糊环境。随机规划为解决带有随机参数的优化问题提供了有力的工具。作为随机规划的发展，机会约束规划由 Charnes 和 Cooper^[20] 提出，主要针对约束条件中含有随机变量，且必须在观测到随机变量的实现之前作出决策的情况。考虑到所作决策在不利情况发生时可能不满足约束条件，而采用一种原则：即允许所作决策在一定程度上不满足约束条件，但该决策应使约束条件成立的概率不小于某一置信水平 α 。这种情况下，机会的意思表示满足约束的概率。带有随机参数的机会约束规划在第 5 章中已经讨论。

模糊规划过去有几种不同的定义方法。这里把含有模糊参数的数学规划称为模糊规划。类似于含有随机参数的机会约束规划的思想，在模糊决策系统中，假设约束成立的可能性至少是 α ，而这里机会的意思表示约束得到满足的可能性。

本章里，主要介绍由 Liu 和 Iwamura^[87, 88] 建立的含有模糊参数的机会约束规划和机会约束多目标规划以及机会约束目标规划。对一些特殊的机会约束，提供相应的清晰等价类。对难以转化为它们各自清晰等价类的机会约束，将采用模糊模拟技术。最后，设计一个基于模糊模拟的遗传算法，用于求解一般模糊机会约束规划，并给出一些数值例子。

9.1 机会约束规划模型

带有模糊参数的数学规划可以写成如下形式

$$\begin{cases} \max f(x, \xi) \\ \text{s.t.} \\ g_j(x, \xi) \leq 0, \quad j = 1, 2, \dots, p. \end{cases} \quad (9.1)$$

其中 x 是决策向量, ξ 是模糊参数向量, $f(x, \xi)$ 是目标函数, $g_j(x, \xi)$ 是约束函数 $j = 1, 2, \dots, p$. 但是, 这个模糊规划的数学意义并不明确, 这是因为 ξ 为模糊向量而导致符号 \max 以及约束没有定义.

因此, 我们必须考虑其它一些有意义的模糊规划形式. 类似于 Charnes 和 Cooper^[20] 提出的带有随机参数的机会约束规划的思想, Liu 和 Iwamura^[87, 88] 提出了带有模糊参数的机会约束规划的理论框架. 带有模糊参数的单目标机会约束规划可以表示成如下的形式,

$$\begin{cases} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(x, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{cases} \quad (9.2)$$

其中 α 和 β 分别是事先给定的对约束和目标的置信水平, $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性.

所以一个点 x 是可行的当且仅当集合

$$\{\xi | g_j(x, \xi) \leq 0, j = 1, 2, \dots, p\}$$

的可能性至少是 α . 对任意给定的决策 x , $f(x, \xi)$ 显然是一个模糊数. 这样存在多个可能的 \bar{f} 使得 $\text{Pos} \{f(x, \xi) \geq \bar{f}\} \geq \beta$. 我们

的目的是极大化目标值 \bar{f} , 因此目标值 \bar{f} 应该是目标函数 $f(x, \xi)$ 在置信水平 β 下所取得的最大值, 即

$$\bar{f} = \max_f \{f \mid \text{Pos} \{f(x, \xi) \geq f\} \geq \beta\}. \quad (9.3)$$

当然, 也可以把模糊决策问题建成如下的机会约束规划模型,

$$\begin{cases} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(x, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(x, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (9.4)$$

其中 α_j 分别是各个约束事先给定的置信水平, $j = 1, 2, \dots, p$.

对极小化问题, 目标值 \bar{f} 是目标函数 $f(x, \xi)$ 在可能性至少为 β 时所取得的最小值, 这样, 有下面形式的模型,

$$\begin{cases} \min \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(x, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(x, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p. \end{cases} \quad (9.5)$$

作为带有模糊参数的单目标机会约束规划的推广, 机会约束多目标规划可以写成如下形式,

$$\begin{cases} \max [\bar{f}_1, \bar{f}_2, \dots, \bar{f}_m] \\ \text{s.t.} \\ \text{Pos} \{f_i(x, \xi) \geq \bar{f}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(x, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (9.6)$$

其中 β_i 是第 i 目标的置信水平, 目标值 \bar{f}_i 是目标函数 $f(x, \xi)$ 在可能性至少为 β_i 时所取得的最大值, $i = 1, 2, \dots, m$.

根据决策者给定的优先结构和目标值, 也可以把模糊决策系统转化为机会约束目标规划,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ \text{Pos} \{ f_i(\mathbf{x}, \boldsymbol{\xi}) + d_i^- - d_i^+ = b_i \} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{ g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0 \} \geq \alpha_j, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{array} \right. \quad (9.7)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

\mathbf{x} = n 维决策向量,

f_i = 目标约束中的函数,

g_j = 机会约束中的实值函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数,

$\boldsymbol{\xi}$ = 模糊参向量,

p = 系统约束个数.

对每一个固定的可行解 \mathbf{x} , 即满足约束条件 $\text{Pos}\{g_j(\mathbf{x}, \boldsymbol{\xi}) \leq 0\} \geq \alpha_j, j = 1, 2, \dots, p$ 的 \mathbf{x} , 由于 $\boldsymbol{\xi}$ 是一个模糊向量, 因此 $f_i(\mathbf{x}, \boldsymbol{\xi})$ 也是一个模糊集合. 这样, 有多个 d_i^- 和 d_i^+ 值得 $\text{Pos}\{f_i(\mathbf{x}, \boldsymbol{\xi}) +$

$d_i^- - d_i^+ = b_i\} \geq \beta_i$. 对这种情况, 正、负偏差被认为分别是使

$$\begin{aligned} \text{Pos}\{d_i^- \geq b_i - f_i(\mathbf{x}, \xi)\} &\geq \beta_i, \\ \text{Pos}\{d_i^+ \geq f_i(\mathbf{x}, \xi) - b_i\} &\geq \beta_i \end{aligned} \quad (9.8)$$

成立的 d_i^+ 和 d_i^- 的最小非负值, $i = 1, 2, \dots, m$.

若模糊向量 ξ 退化成为清晰向量, 只要 $\beta_i > 0$, 则可能性

$$\text{Pos}\{d_i^- \geq b_i - f_i(\mathbf{x}, \xi)\}$$

和

$$\text{Pos}\{d_i^+ \geq f_i(\mathbf{x}, \xi) - b_i\}$$

将永远为 1, 并且 (9.8) 等价于

$$d_i^- = [b_i - f_i(\mathbf{x}, \xi)] \vee 0, \quad d_i^+ = [f_i(\mathbf{x}, \xi) - b_i] \vee 0.$$

这与清晰目标规划吻合.

9.2 清晰等价类

求解带有模糊参数的机会约束规划的一个传统方法是把机会约束

$$\text{Pos}\{g(\mathbf{x}, \xi) \leq 0\} \geq \alpha \quad (9.9)$$

转化为它的清晰等价类, 然后, 用传统的方法求解. 注意模糊系统约束 $\text{Pos}\{g_j(\mathbf{x}, \xi) \leq 0\} \geq \alpha_j (j = 1, 2, \dots, p)$ 是一组符合这种形式的机会约束. 若定义 $g(\mathbf{x}, \xi) = \bar{f} - f(\mathbf{x}, \xi)$, 则模糊目标函数 $\text{Pos}\{f(\mathbf{x}, \xi) \geq \bar{f}\} \geq \beta$ 也符合这种形式. 既然模糊目标约束 $\text{Pos}\{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \beta$ 可以理解为 $\text{Pos}\{d^- \geq b - f(\mathbf{x}, \xi)\} \geq \beta$ 和 $\text{Pos}\{d^+ \geq f(\mathbf{x}, \xi) - b\} \geq \beta$, 若定义 $g(\mathbf{x}, \xi) = b - f(\mathbf{x}, \xi) - d^-$ 和 $g(\mathbf{x}, \xi) = f(\mathbf{x}, \xi) - b - d^+$, 则前者和后者均符合形式 (9.9). 本节介绍一些已知的结果.

情况 I 假设函数 $g(\mathbf{x}, \xi)$ 的形式为 $g(\mathbf{x}, \xi) = h(\mathbf{x}) - \xi$, 则机会约束 (9.9) 可以表示成如下的形式,

$$\text{Pos} \{h(\mathbf{x}) \leq \xi\} \geq \alpha. \quad (9.10)$$

其中 $h(\mathbf{x})$ 是决策变量 \mathbf{x} 的 (线性或非线性) 函数, ξ 是一个模糊数, 隶属函数为 $\mu(\xi)$.

显然, 对任意给定的置信水平 α ($0 \leq \alpha \leq 1$), 必存在值 K_α (可能是 $+\infty$ 或 $-\infty$) 使得

$$\text{Pos} \{K_\alpha \leq \xi\} = \alpha. \quad (9.11)$$

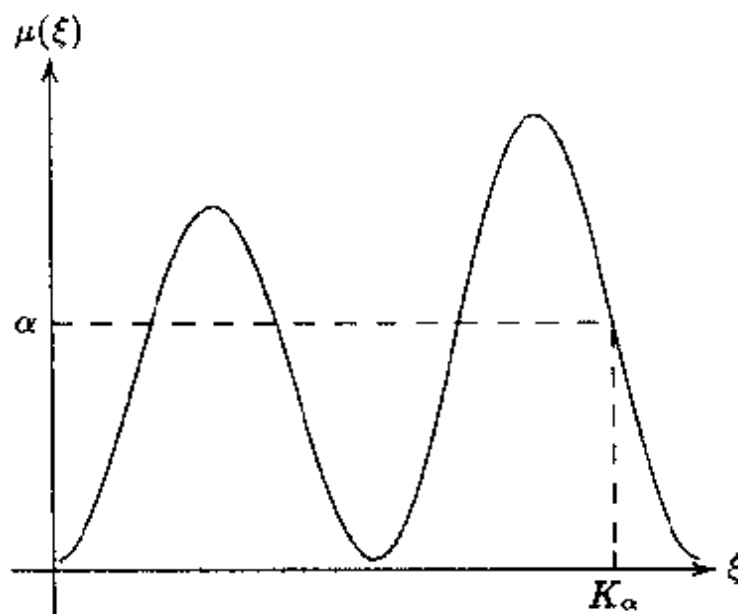


图 9.1 隶属函数 $\mu(\xi)$ 以及 K_α 和 α

如果用一个较小的数 K'_α 代替 K_α , 则可能性 $\text{Pos}\{K_\alpha \leq \xi\}$ 将随之增加, 这是因为

$$\begin{aligned} \text{Pos} \{K_\alpha \leq \xi\} &= \sup \{ \mu(\xi) \mid \xi \geq K_\alpha \} \\ &\leq \sup \{ \mu(\xi) \mid \xi \geq K'_\alpha \} \\ &= \text{Pos} \{K'_\alpha \leq \xi\}. \end{aligned}$$

这样, 机会约束 (9.10) 等价于

$$h(\mathbf{x}) \leq K_\alpha, \quad (9.12)$$

其中 K_α 是满足 (9.11) 的最大值.

事实上, 最大的值 K_α 可由下式确定,

$$K_\alpha = \sup \{K \mid K = \mu^{-1}(\alpha)\}, \quad (9.13)$$

其中 μ^{-1} 是 μ 的反函数.

情况 II 假设 $g(\mathbf{x}, \xi) = \xi - h(\mathbf{x})$, 则机会约束 (9.9) 可以表示成

$$\text{Pos} \{h(\mathbf{x}) \geq \xi\} \geq \alpha, \quad (9.14)$$

其中 $h(\mathbf{x})$ 是决策变量 \mathbf{x} 的 (线性或非线性) 函数, ξ 是一个模糊数, 隶属函数为 $\mu(\xi)$.

显然, 对任意给定的置信水平 α ($0 \leq \alpha \leq 1$), 存在一些值 K_α (可能为 $+\infty$ 或 $-\infty$) 使得

$$\text{Pos} \{K_\alpha \geq \xi\} = \alpha. \quad (9.15)$$

如果用一个较大的数 K'_α 代替 K_α , 则可能性 $\text{Pos} \{K_\alpha \geq \xi\}$ 将随之递增, 这是由于

$$\begin{aligned} \text{Pos} \{K_\alpha \geq \xi\} &= \sup \{\mu(\xi) \mid \xi \leq K_\alpha\} \\ &\leq \sup \{\mu(\xi) \mid \xi \leq K'_\alpha\} \\ &= \text{Pos} \{\xi \mid K'_\alpha \geq \xi\}. \end{aligned}$$

于是机会约束 (9.14) 的清晰等价类为

$$h(\mathbf{x}) \geq K_\alpha, \quad (9.16)$$

其中 K_α 是满足 (9.15) 的最小值, 即

$$K_\alpha = \inf \{K \mid K = \mu^{-1}(\alpha)\},$$

这里 μ^{-1} 是函数 μ 的反函数.

情况 III 假设在机会约束 (9.9) 中, 函数 $g(\mathbf{x}, \xi)$ 是 ξ 的分量的线性函数. 例如, 假设 $\xi = (\xi_1, \xi_2, \dots, \xi_t)$ 是 t 维向量. 由于 $g(\mathbf{x}, \xi)$ 是 ξ 的分量的线性函数, 说明 $g(\mathbf{x}, \xi)$ 可以表示成下面的形式,

$$g(\mathbf{x}, \xi) = h_1(\mathbf{x})\xi_1 + h_2(\mathbf{x})\xi_2 + \dots + h_t(\mathbf{x})\xi_t + h_0(\mathbf{x}), \quad (9.17)$$

其中函数 $h_k(\mathbf{x})$ 不一定是线性的或非负的, $k = 0, 1, 2, \dots, t$. 若 $\xi_1, \xi_2, \dots, \xi_t$ 是一般的模糊数, 那么 $g(\mathbf{x}, \xi)$ 的隶属函数将变得非常复杂. 这里假定所有的 ξ_k 是梯形模糊数, 记为

$$\xi_k = (r_{k1}, r_{k2}, r_{k3}, r_{k4}), \quad k = 1, 2, \dots, t.$$

定义两个函数

$$h_k^+(\mathbf{x}) = \begin{cases} h_k(\mathbf{x}), & h_k(\mathbf{x}) \geq 0 \\ 0, & h_k(\mathbf{x}) < 0 \end{cases}$$

和

$$h_k^-(\mathbf{x}) = \begin{cases} 0, & h_k(\mathbf{x}) \geq 0 \\ -h_k(\mathbf{x}), & h_k(\mathbf{x}) < 0, \end{cases}$$

$k = 1, 2, \dots, t$, 则函数 $h_k^+(\mathbf{x})$ 和 $h_k^-(\mathbf{x})$ 是非负的, 并且 $h_k(\mathbf{x}) =$

$h_k^+(\mathbf{x}) - h_k^-(\mathbf{x})$. 这样有

$$\begin{aligned} g(\mathbf{x}, \xi) &= \sum_{k=1}^t h_k(\mathbf{x}) \xi_k + h_0(\mathbf{x}) \\ &= \sum_{k=1}^t [h_k^+(\mathbf{x}) - h_k^-(\mathbf{x})] \xi_k + h_0(\mathbf{x}) \\ &= \sum_{k=1}^t [h_k^+(\mathbf{x}) \xi_k + h_k^-(\mathbf{x}) \xi'_k] + h_0(\mathbf{x}), \end{aligned}$$

其中 ξ'_k 也是梯形模糊数,

$$\xi'_k = (-r_{k4}, -r_{k3}, -r_{k2}, -r_{k1}), \quad k = 1, 2, \dots, t.$$

根据梯形模糊数的加法和乘法运算, 函数 $g(\mathbf{x}, \xi)$ 也是梯形模糊数, 并且可以表示成如下的形式,

$$g(\mathbf{x}, \xi) = \left(\begin{array}{c} \sum_{k=1}^t [r_{k1} h_k^+(\mathbf{x}) - r_{k4} h_k^-(\mathbf{x})] + h_0(\mathbf{x}) \\ \sum_{k=1}^t [r_{k2} h_k^+(\mathbf{x}) - r_{k3} h_k^-(\mathbf{x})] + h_0(\mathbf{x}) \\ \sum_{k=1}^t [r_{k3} h_k^+(\mathbf{x}) - r_{k2} h_k^-(\mathbf{x})] + h_0(\mathbf{x}) \\ \sum_{k=1}^t [r_{k4} h_k^+(\mathbf{x}) - r_{k1} h_k^-(\mathbf{x})] + h_0(\mathbf{x}) \end{array} \right)^r$$

由引理 3.1, 我们知道, 机会约束 (9.9) 的清晰等价类为

$$\begin{aligned} (1 - \alpha) \sum_{k=1}^t [r_{k1} h_k^+(\mathbf{x}) - r_{k4} h_k^-(\mathbf{x})] \\ + \alpha \sum_{k=1}^t [r_{k2} h_k^+(\mathbf{x}) - r_{k3} h_k^-(\mathbf{x})] + h_0(\mathbf{x}) \leq 0. \end{aligned} \quad (9.18)$$

9.3 模糊模拟

虽然有些机会约束可以转化为解析的清晰等价类,但对一般的情况,还需要其它的方法. Liu 和 Iwamura^[87, 88] 提出了使用模糊模拟检验模糊系统约束

$$\text{Pos} \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha$$

和处理模糊目标函数

$$\text{Pos} \{f(\mathbf{x}, \xi) \geq \bar{f}\} \geq \beta$$

以及处理模糊目标约束

$$\text{Pos} \{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \beta.$$

本节里, 介绍一些模糊模拟技术.

检验模糊系统约束

由模糊数运算的定义, 我们说对任意给定的决策向量 \mathbf{x} ,

$$\text{Pos} \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha \quad (9.19)$$

成立当且仅当存在一个清晰向量 ξ^0 使得 $g_j(\mathbf{x}, \xi^0) \leq 0, j = 1, 2, \dots, k$ 且 $\mu(\xi^0) \geq \alpha$. 这样, 可以由模糊向量 ξ 均匀地生成一个清晰向量 ξ^0 使得 $\mu(\xi^0) \geq \alpha$, 即在模糊数 ξ 的 α 水平截集中抽取一个向量. 若模糊向量 ξ 的 α 水平截集过于复杂难于确定, 可以从包含 α 水平截集的超几何体 Ω 中抽取向量 ξ^0 , 接受与否依赖于 $\mu(\xi^0) \geq \alpha$ 是否成立. 为了加快模拟速度, 超几何体 Ω 一般应设计得尽可能小. 如果 ξ^0 满足 $g_j(\mathbf{x}, \xi^0) \leq 0, j = 1, 2, \dots, k$, 那么, 我们确信 $\text{Pos} \{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha$. 否则, 从模

向量 ξ 的 α 水平截集中重新抽取清晰向量 ξ^0 ，并检验约束是否成立。经过给定的次数 N 以后，如果没有生成清晰向量 ξ^0 使得 $g_j(x, \xi^0) \leq 0, j = 1, 2, \dots, k$ 成立，则我们认为给定的决策向量 x 是不可行的。

步骤 1 从模糊向量 ξ 的 α 水平截集中均匀地生成向量 ξ^0 ；

步骤 2 若 $g_j(x, \xi^0) \leq 0, j = 1, 2, \dots, k$ ，返回“可行”；

步骤 3 重复步骤 1 和 2 共 N 次；

步骤 4 返回“不可行”。

处理模糊目标函数

对含有模糊参数 ξ 的模糊目标函数

$$\text{Pos} \{f(x, \xi) \geq \bar{f}\} \geq \beta, \quad (9.20)$$

出于极大化 \bar{f} 的目的，对给定的决策向量 x ，我们应找到最大的值 \bar{f} 使得 (9.20) 成立。首先，置 $\bar{f} = -\infty$ 。然后，由模糊向量 ξ 均匀地生成清晰向量 ξ^0 使得 $\mu(\xi^0) \geq \beta$ ，即在模糊向量 ξ 的 β 水平截集中抽取一个向量。若 $\bar{f} < f(x, \xi^0)$ ，置 $\bar{f} = f(x, \xi^0)$ 。重复以上过程 N 次，则认为值 \bar{f} 是在点 x 处的目标值。

步骤 1 置 $\bar{f} = -\infty$ ；

步骤 2 从模糊向量 ξ 的 α 水平截集中均匀地生成向量 ξ^0 ；

步骤 3 若 $\bar{f} < f(x, \xi^0)$ ，则置 $\bar{f} = f(x, \xi^0)$ ；

步骤 4 重复步骤 2 和 3 共 N 次；

步骤 5 返回 \bar{f} 。

处理模糊目标约束

在带有模糊系数的机会约束目标规划中，可能存在下面形式

的目标约束,

$$\text{Pos}\{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \beta, \quad (9.21)$$

我们的目的是对每个固定的决策向量 \mathbf{x} , 找到使得

$$\text{Pos}\{d^- \geq b - f(\mathbf{x}, \xi)\} \geq \beta,$$

$$\text{Pos}\{d^+ \geq f(\mathbf{x}, \xi) - b\} \geq \beta$$

成立的 d^- 和 d^+ 的最小非负值. 首先, 置 $d^- = d^+ = +\infty$. 然后从模糊向量 ξ 的 β 水平截集中均匀地生成向量 ξ^0 . 若 $d^- > b - f(\mathbf{x}, \xi^0)$, 则置 $d^- = [b - f(\mathbf{x}, \xi^0)] \vee 0$. 类似地, 若 $d^+ > f(\mathbf{x}, \xi^0) - b$, 则置 $d^+ = [f(\mathbf{x}, \xi^0) - b] \vee 0$. 通过生成新的清晰向量 ξ^0 去重复以上的过程 N 次, 所得的值 d^+ 和 d^- 分别作为在点 \mathbf{x} 处正、负偏差的估计值.

步骤 1 置 $d^+ = d^- = +\infty$;

步骤 2 从模糊向量 ξ 的 β 水平截集中均匀地生成清晰向量 ξ^0 ;

步骤 3 若 $d^+ > f(\mathbf{x}, \xi^0) - b$, 则置 $d^+ = [f(\mathbf{x}, \xi^0) - b] \vee 0$;

步骤 4 若 $d^- > b - f(\mathbf{x}, \xi^0)$, 则置 $d^- = [b - f(\mathbf{x}, \xi^0)] \vee 0$;

步骤 5 重复步骤 2 到步骤 4 直到给定的次数 N ;

步骤 6 返回 d^+ 和 d^- .

9.4 基于模糊模拟的遗传算法

对带有模糊系数的机会约束规划, 如果机会约束可以转化为它们各自的清晰等价类, 那么机会约束规划可以用处理清晰数学规划的方法得到解决.

本节里, 将求解那些无法转化成清晰等价类的复杂的机会约束规划. 所使用的方法就是基于模糊模拟的遗传算法, 除了使用第 9.3 节中介绍的模糊模拟检验模糊系统约束和处理模糊目标函

数以及模糊目标约束以外,与第2章中所介绍的遗传算法的过程类似.

遗传算法程序

步骤0 输入参数 pop_size , P_c 和 P_m ;

步骤1 初始产生 pop_size 个染色体,可能使用模糊模拟技术检验染色体的可行性;

步骤2 对染色体进行交叉和变异操作,可能使用模糊模拟检验后代的可行性;

步骤3 使用模糊模拟计算所有染色体的目标值;

步骤4 根据目标值,使用基于序的评价函数计算每个染色体的适应度;

步骤5 通过赌轮选择染色体;

步骤6 重复步骤2到步骤5直到完成给定的次数;

步骤7 将最好的染色体作为最优解.

基于模糊模拟的遗传算法的计算机代码已编写成C语言程序.为了解释该遗传算法的有效性,这里给出了一些数值例子,计算结果是成功的.所给出的例子都是在PC机上实现的,所使用的参数分别为:种群规模为30,交叉概率 $P_c = 0.2$,变异概率 $P_m = 0.3$,评价函数中的参数 $\alpha = 0.05$.

例1 考虑下面带有模糊系数的单目标机会约束规划,

$$\left\{ \begin{array}{l} \max 2x_1 + 3x_2 \\ \text{s.t.} \\ \text{Pos} \left\{ \tilde{a}x_1 + x_2 \leq \tilde{b} \right\} \geq 0.90 \\ \text{Pos} \left\{ x_1 + \tilde{c}x_2 \leq \tilde{d} \right\} \geq 0.85 \\ x_1 + x_2 \leq 100 \\ x_1, x_2 \geq 0, \end{array} \right. \quad (9.22)$$

其中 \tilde{a} 为梯形模糊数 $(1, 1.5, 2.5, 3)$, \tilde{b} 是模糊数, 其隶属函数为

$$\mu_{\tilde{b}}(\xi) = \exp \left[-\frac{1}{10} |\xi - 150| \right],$$

\tilde{c} 是三角模糊数 $(1, 2, 3)$, \tilde{d} 是模糊数, 其隶属函数为

$$\mu_{\tilde{d}}(\xi) = \exp \left[-\frac{1}{10} |\xi - 130| \right].$$

为了使用模糊模拟检验一个给定的解 x 是否可行, 取 $\tilde{a}, \tilde{b}, \tilde{c}$ 和 \tilde{d} 的 α 水平截集为 $[1, 3], [140, 160], [1, 3]$ 和 $[120, 140]$. 经过 200 代运算以后, 基于模糊模拟的遗传算法所给出的最好的解为

$$(x_1^*, x_2^*) = (62.56, 37.33),$$

其目标值为 237.11. 更进一步, 有

$$\text{Pos} \left\{ \tilde{a}x_1^* + x_2^* \leq \tilde{b} \right\} = 1.00, \quad \text{Pos} \left\{ x_1^* + \tilde{c}x_2^* \leq \tilde{d} \right\} = 0.85.$$

例 2 考虑带有模糊系数的机会约束目标规划,

$$\begin{cases} \text{lexmin} \{d_1^-, d_2^-\} \\ \text{s.t.} \\ x_1 + x_2 + d_1^- - d_1^+ = 2 \\ x_3 + d_2^- - d_2^+ = 3 \\ \text{Pos} \left\{ \tilde{a}_1 x_1 + \tilde{a}_2 x_2 + \tilde{a}_3 x_3 \leq \tilde{b} \right\} \geq 0.80 \\ x_1, x_2, x_3, d_1^-, d_1^+, d_2^-, d_2^+ \geq 0, \end{cases} \quad (9.23)$$

其中 \tilde{a}_1 是梯形模糊数 $(1, 2, 3, 4)$, \tilde{a}_2 是模糊数, 其隶属函数为

$$\mu_{\tilde{a}_2}(\xi) = \exp \left[-(\xi - 2)^2 \right].$$

\tilde{a}_3 是模糊数, 其隶属函数为

$$\mu_{\tilde{a}_3}(\xi) = \begin{cases} \frac{1}{\xi}, & \xi \geq 1 \\ 0, & \text{其它,} \end{cases}$$

\tilde{b} 是三角模糊数 $(3, 4, 5)$. 进化 600 代以后, 基于模糊模拟的遗传算法所给出的最好的解为

$$(x_1^*, x_2^*, x_3^*) = (0.045, 1.957, 1.073),$$

它满足第 1 个目标, 但第 2 个目标的负偏差为 1.927. 同时, 有

$$\text{Pos} \{ \tilde{a}_1 x_1^* + \tilde{a}_2 x_2^* + \tilde{a}_3 x_3^* \leq \tilde{b} \} = 0.80.$$

例 3 考虑如下的系统约束和目标约束中同时带有模糊系数的机会约束目标规划,

$$\left\{ \begin{array}{l} \text{lexmin} \{ d_1^-, d_2^-, d_3^- + d_3^+ \} \\ \text{s.t.} \\ \text{Pos} \{ \tilde{c}_1 x_1 x_3 + d_1^- - d_1^+ = 2 \} \geq 0.9 \\ \text{Pos} \{ x_2 + d_2^- - d_2^+ = \tilde{c}_2 \} \geq 0.8 \\ x_4 + d_3^- - d_3^+ = 4 \\ \text{Pos} \{ (\tilde{a}_1 x_1 + \tilde{a}_2 x_2)(\tilde{a}_3 x_3 + \tilde{a}_4 x_4) \leq 25 \} \geq 0.8 \\ x_1, x_2, x_3, x_4, d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, \end{array} \right.$$

其中 $\tilde{a}_1 = (1, 2, 3)$ 是三角模糊数, \tilde{a}_2 的隶属函数为

$$\mu_{\tilde{a}_2}(z) = \exp [-(z-1)^2],$$

$\tilde{a}_3 = (2, 2.5, 3.5, 4)$ 是梯形模糊数, \tilde{a}_4 的隶属函数为

$$\mu_{\tilde{a}_4}(z) = \frac{1}{1 + (z-2)^2},$$

目标约束中的模糊系数 $\tilde{c}_1 = (0.5, 1, 1.5)$ 是三角模糊数, \tilde{c}_2 的隶属函数为

$$\mu_{\tilde{c}_2}(z) = \exp[-(z-2)^2].$$

为了使用模糊模拟处理模糊系统约束和模糊目标约束, 分别取包含模糊数 $\tilde{a}_1, \tilde{a}_2, \tilde{a}_3, \tilde{a}_4, \tilde{c}_1$ 和 \tilde{c}_2 的 α 水平截集的超几何体为 $[1.5, 2.5], [0.5, 1.5], [2, 4], [1, 3], [0.8, 1.2]$ 和 $[1.5, 2.5]$. 本例中, 每次模糊模拟执行 2000 次. 基于模糊模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*) = (1.1511, 1.5394, 1.6603, 3.0555),$$

它满足前两个目标, 但第三个目标为 0.9445. 同时有

$$\text{Pos}\{\tilde{c}_1 x_1^* x_3^* \geq 2\} = 0.904 \approx 0.9,$$

$$\text{Pos}\{x_2^* \geq \tilde{c}_2\} = 0.807 \approx 0.8.$$

$$\text{Pos}\{(\tilde{a}_1 x_1^* + \tilde{a}_2 x_2^*)(\tilde{a}_3 x_3^* + \tilde{a}_4 x_4^*) \leq 25\} = 0.801 \approx 0.8.$$

这里, 基于模糊模拟的遗传算法所花费的时间由两部分构成, 模糊模拟花费的时间和遗传算法花费的时间, 其中模糊模拟的时间是必须的, 因为我们已假设没有可以替代模糊模拟的方法.

9.5 资金预算

回顾第 6 章中所讨论资金预算问题, 我们已经讨论了机会约束

$$\text{Pr}\{\eta_i x_i \geq \xi_i\} \geq \alpha_i, \quad i = 1, 2, \dots, n,$$

其中 η_i 和 ξ_i 假设是具有已知概率密度函数的随机变量, α_i 是事先给定的置信水平, $i = 1, 2, \dots, n$. 很多情况下, 概率密度函数是未知的, 需要通过多次重复实验进行估计. 然而, 有些时候, 我

们无法进行这样的重复实验. 在这种情况下, 可以把 η_i 和 ξ_i 视为模糊数, 通过一些专家知识和经验建立它们的隶属函数.

本节里, 假定 η_i 和 ξ_i 的隶属函数均已给定. 如果希望满足需求 ξ_i 的可能性分别为 α_i , $i = 1, 2, \dots, n$, 那么有下面模糊环境下的机会约束

$$\text{Pos} \{ \eta_i x_i \geq \xi_i \} \geq \alpha_i, \quad i = 1, 2, \dots, n$$

其中 $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性. 更进一步, 假设每台机器均可生产其它类型的产品, 例如有 p 类产品需求, 需求量分别记为 ξ_j , 对类型 j 的产品, 类型 i 的机器的生产能力分别为 η_{ij} , $i = 1, 2, \dots, n$, $j = 1, 2, \dots, p$, 因此机会约束可以表示成

$$\text{Pos} \{ \eta_{1j} x_1 + \eta_{2j} x_2 + \dots + \eta_{nj} x_n \geq \xi_j \} \geq \alpha_i, \quad (9.24)$$

$j = 1, 2, \dots, p$, 或写成联合的机会约束形式

$$\text{Pos} \{ \eta_{1j} x_1 + \eta_{2j} x_2 + \dots + \eta_{nj} x_n \geq \xi_j, j = 1, 2, \dots, p \} \geq \alpha, \quad (9.25)$$

其中 α 是事先给定的置信水平. 有些情况下, 所有的模糊数均为梯形模糊数, 这时, 机会约束 (9.24) 可以转化为清晰等价类.

对资金预算问题, 最简单的带有模糊系数的机会约束整数规划为

$$\left\{ \begin{array}{l} \max c_1 x_1 + c_2 x_2 + \dots + c_n x_n \\ \text{s.t.} \\ a_1 x_1 + a_2 x_2 + \dots + a_n x_n \leq a \\ b_1 x_1 + b_2 x_2 + \dots + b_n x_n \leq b \\ \text{Pos} \{ \eta_{1j} x_1 + \eta_{2j} x_2 + \dots + \eta_{nj} x_n \geq \xi_j, j = 1, 2, \dots, p \} \geq \alpha \\ x_i, i = 1, 2, \dots, n. \quad \text{非负整数.} \end{array} \right.$$

为了平衡多个不相容的目标, 根据决策者的优先结构和目标值, 资金问题可以建模为如下的机会约束目标规划,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^+, d_2^+, d_3^-\} \\ \text{s.t.} \\ a_1x_1 + a_2x_2 + \cdots + a_nx_n + d_1^- - d_1^+ = a \\ b_1x_1 + b_2x_2 + \cdots + b_nx_n + d_2^- - d_2^+ = b \\ c_1x_1 + c_2x_2 + \cdots + c_nx_n + d_3^- - d_3^+ = c \\ \text{Pos} \{ \eta_{1j}x_1 + \eta_{2j}x_2 + \cdots + \eta_{nj}x_n \geq \xi_j, j = 1, 2, \cdots, p \} \geq \alpha \\ x_i, i = 1, 2, \cdots, n, \text{ 非负整数.} \end{array} \right.$$

下面考虑一些数值例子. 假设有 5 台机器, 我们的目的是极大化总效益, 则资金预算问题可以建模为

$$\left\{ \begin{array}{l} \max 3x_1 + x_2 + 2x_3 + 3x_4 + x_5 \\ \text{s.t.} \\ 2x_1 + x_2 + 3x_3 + 6x_4 + 4x_5 \leq 50 \\ 7x_1 + 6x_2 + 4x_3 + 8x_4 + x_5 \leq 100 \\ \text{Pos} \left\{ \begin{array}{l} \eta_{11}x_1 + \eta_{21}x_2 + \eta_{31}x_3 \geq \xi_1 \\ \eta_{32}x_3 + \eta_{42}x_4 + \eta_{52}x_5 \geq \xi_2 \end{array} \right\} \geq 0.9 \\ x_1, x_2, x_3, x_4, x_5, \text{ 非负整数,} \end{array} \right.$$

其中 η_{11} 是三角模糊数 (13,14,15), η_{21} 的隶属函数为

$$\mu_{\eta_{21}}(u) = \exp [-(u-8)^2],$$

η_{31} 的隶属函数为

$$\mu_{\eta_{31}}(u) = \begin{cases} \frac{1}{u-9}, & u \geq 10 \\ 0, & u < 10, \end{cases}$$

第一类产品的需求 ξ_1 是模糊数, 隶属函数为

$$\mu_{\xi_1}(u) = \exp[-|u - 50|].$$

η_{32} 是梯形模糊数 (8,9,10,11), η_{42} 是三角模糊数 (10,11,12), η_{52} 的隶属函数为

$$\mu_{\eta_{52}}(u) = \exp[-|u - 10|],$$

第二类产品的需求 ξ_2 是三角模糊数 (30,40,50). 经过 300 代进化以后, 基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (10, 0.7, 0, 1),$$

其总效益为 45.

现在考虑下面给定的目标值和优先结构:

优先级 1 预算目标, 即每类机器所分配的总费用尽可能地不超过已有数目, 即

$$2x_1 + x_2 + 3x_3 + 6x_4 + 4x_5 + d_1^- - d_1^+ = 50,$$

其中 d_1^+ 将被极小化;

优先级 2 空间目标, 即机器所使用的总的空间尽可能地不超过已有空间, 即

$$7x_1 + 6x_2 + 4x_3 + 8x_4 + x_5 + d_2^- - d_2^+ = 100.$$

其中 d_2^+ 将被极小化;

优先级 3 效益目标, 即总数益尽可能地达到给定的水平, 即

$$3x_1 + x_2 + 2x_3 + 3x_4 + x_5 + d_3^- - d_3^+ = 50,$$

其中 d_3^+ 将被极小化.

那么资金预算问题可以建模为带有模糊系数的机会约束整数目标规划,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^+, d_2^+, d_3^-\} \\ \text{s.t.} \\ 2x_1 + x_2 + 3x_3 + 6x_4 + 4x_5 + d_1^- - d_1^+ = 50 \\ 7x_1 + 6x_2 + 4x_3 + 8x_4 + x_5 + d_2^- - d_2^+ = 100 \\ 3x_1 + x_2 + 2x_3 + 3x_4 + x_5 + d_3^- - d_3^+ = 50 \\ \text{Pos} \left\{ \begin{array}{l} \eta_{11}x_1 + \eta_{21}x_2 + \eta_{31}x_3 \geq \xi_1 \\ \eta_{32}x_3 + \eta_{42}x_4 + \eta_{52}x_5 \geq \xi_2 \end{array} \right\} \geq 0.9 \\ x_1, x_2, x_3, x_4, x_5, \quad \text{非负整数,} \end{array} \right.$$

这里所使用的模糊参数同上. 经过 400 代进化以后, 基于随机模拟的遗传算法给出的最优解为

$$(x_1^*, x_2^*, x_3^*, x_4^*, x_5^*) = (10, 0, 7, 0, 1),$$

它满足前两个目标, 但第三个目标的负偏差为 5.

9.6 注

根据带有随机参数的机会约束规划的思想, 本章介绍了带有模糊系数的机会约束规划和机会约束多目标规划以及机会约束目标规划. 同时, 也列举了检验模糊系统约束和处理模糊目标约束所需的模糊模拟技术, 并设计了求解一般机会约束规划模型的基于模糊模拟的遗传算法. 为了解释所设计的遗传算法的有效性, 同时给出了一些数值例子.

第 10 章

模糊环境下的相关机会规划

遵循随机环境下的相关机会规划的思想, Liu^[89] 提出了模糊环境下的相关机会规划以及相关机会多目标规划和相关机会目标规划的理论框架.

本章里, 将介绍模糊情况下的不确定环境、事件、机会函数和诱导约束等概念, 并建立模糊环境下相关机会规划的理论. 最后, 设计了基于模糊模拟的遗传算法, 并给出一些数值例子解释如何应用该算法求解模糊相关机会规划模型.

10.1 相关机会规划

不确定环境、事件、机会函数和诱导约束是随机环境下的相关机会规划的理论框架中的关键术语. 下面给出它们在模糊环境下的定义.

所谓的不确定环境 (此时为模糊环境), 是指解集由一些不确定条件约束定义, 这些约束通常表示为

$$g_j(x, \xi) \leq 0, \quad j = 1, 2, \dots, p, \quad (10.1)$$

其中 x 是决策向量, ξ 是模糊参数向量. 虽然 x 是决策向量, 但由于 ξ 的不确定性, 决策变量 x 的实现依赖于模糊参数向量 ξ 的实现, 从而决策变量 x 也具有某些不确定性, 它的不确定性可以由 (10.1) 和 ξ 的性质导出.

一个复杂的决策系统通常要从事多项任务, 并满足一些系统内部的要求. 用事件表示这些任务和内部要求, 通常表示成一个满足

预定目标的决策变量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 构成的集合, 如 $x_1 + x_2 = 10$ 是一个事件, 表示顾客的需求应该得到满足.

机会函数表示事件的可能性测度, 通常表示成满足某个事件的可能性, 如 $f(\mathbf{x}) = \text{Pos}\{x_1 + x_2 = 10\}$, 这里 $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性.

现在的问题是如何通过不确定环境表示机会函数. 设 E 是模糊决策系统中唯一的事件, 则事件 E 可以由适当的决策向量 $\mathbf{x} = (x_1, x_2, \dots, x_n)$ 来达到. 一般情况下, E 的实现仅与 $\{x_1, x_2, \dots, x_n\}$ 中的部分分量有关, 而不是全部. 用符合 $V(E)$ 表示决策向量 \mathbf{x} 的分量中对事件 E 是必要的那部分分量构成的集合. 这样, 事件 E 的实现仅与 $V(E)$ 中的元素有关. 既然模糊决策系统中仅有一个事件 E , 事件 E 的机会函数就是使当 $\mathbf{x} \in E$ 时 $V(E)$ 中的元素实现的最大可能性, 否则认为机会函数为零, 即

$$f(\mathbf{x}) = \begin{cases} \max_{\mathbf{y} \in E^*} \text{Pos} \left\{ \begin{array}{l} g_j(\mathbf{y}, \boldsymbol{\xi}) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\}, & \mathbf{x} \in E \\ 0, & \mathbf{x} \notin E, \end{cases} \quad (10.2)$$

其中 $\mathbf{x} \in E$ 表示决策向量 \mathbf{x} 满足事件 E ,

$$E^* = \left\{ \mathbf{y} = (y_1, y_2, \dots, y_n) \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in V(E) \\ j = 1, 2, \dots, n \end{array} \right\}. \quad (10.3)$$

注意对 E^* 中的任何元素 $\mathbf{y} = (y_1, y_2, \dots, y_n)$, 决策分量 $y_j \in V(E)$ 是固定的, 并等于 x_j , 其余的分量则是任意的. 通常, 对每个给定的决策 \mathbf{x} , 不难确定 (10.2) 的最优解 $\mathbf{y}^*(\mathbf{x})$ 中 $V(E)$ 以外的决策分量 x_j^* 的值使得 $\mathbf{y}^*(\mathbf{x}) = (y_1, y_2, \dots, y_n)$. 其中

$$y_j = \begin{cases} x_j, & y_j \in V(E) \\ x_j^*, & y_j \notin V(E), \end{cases} \quad j = 1, 2, \dots, n.$$

事实上, 决策分量 x_j^* 一般在一些极端的地方取值. 因此, 事件 E 的机会函数可以表示为

$$f(x) = \begin{cases} \text{Pos} \left\{ \begin{array}{l} g_j(y^*(x), \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\}, & x \in E \\ 0, & x \notin E, \end{cases} \quad (10.4)$$

其中约束

$$g_j(y^*(x), \xi) \leq 0, \quad j = 1, 2, \dots, p \quad (10.5)$$

称为在点 x 上关于事件 E 的诱导约束. 到此为止, 已经解释清楚了如何表示不确定环境下一个事件的机会函数.

于是模糊环境下典型的单目标相关机会规划可以表示为

$$\begin{cases} \max f(x) \\ \text{s.t.} \\ g_j(x, \xi) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (10.6)$$

其中 x 是一个 n 维决策向量, ξ 是一个模糊向量, $f(x)$ 是一个事件的机会函数, $g_j(x, \xi) \leq 0, j = 1, 2, \dots, p$ 是不确定环境. 值得注意的是, 在相关机会规划模型中, 不确定环境所描述的可行集并不假定是清晰的.

10.2 相关机会多目标规划

既然一个复杂的决策系统通常执行着多项任务, 毫无疑问, 在决策过程中存在多个可能的目标 (有些是机会函数). 一个典型的模糊相关机会多目标规划可以写成如下的形式,

$$\begin{cases} \max f(x) = [f_1(x), f_2(x), \dots, f_m(x)] \\ \text{s.t.} \\ g_j(x, \xi) \leq 0, \quad j = 1, 2, \dots, p \end{cases} \quad (10.7)$$

其中 $f(x)$ 是由 m 个函数 $f_i(x)$ (有些是机会函数) 构成的向量值函数.

为了表示多个事件的机会函数, 下面介绍决策分量 x_1, x_2, \dots, x_n 之间模糊关系的定义. 首先回忆第 7 章中所介绍的决策分量之间的随机关系: 决策向量的 n 个分量可以划分成 k 组, 这 k 组元素是相互随机独立的, 而每组内的元素却是随机相关的, 并且有相同的实现机会. 类似地, 对模糊情况, 假设存在下面的模糊关系.

模糊关系 决策向量的 n 个分量之间可以划分成 k 组, 这 k 组元素之间是相互模糊独立的, 而每组内的元素却是模糊相关的, 并且有相同的实现机会.

根据模糊关系, 对事件 E , 用符合 $D(E)$ 表示由所有与 $V(E)$ 中至少一个元素模糊相关的分量构成的集合. 注意在实际问题中集合 $D(E)$ 是容易确定的.

对多目标的情况, 不失一般性, 假设存在 m 个事件 E_i , 机会函数分别为 $f_i(x)$, $i = 1, 2, \dots, m$. 记

$$E = E_1 \cap E_2 \cap \dots \cap E_m$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_m),$$

其中 $x \in E$ 的意思是决策向量 x 满足所有的事件 E_1, E_2, \dots, E_m , $V(E)$ 是由所有对 m 个事件必要的分量构成的集合. 根据模糊关系, $D(E), D(E_i) (i = 1, 2, \dots, m)$ 分别是由至少与 $V(E), V(E_i) (i = 1, 2, \dots, m)$ 中一个元素模糊相关的分量构成的集合. 由于所有事件在同一优先级中, 因此, 只能在不牺牲其它事件实现机会的前提下, 尽可能地实现每一个事件 E_i . 一方面, 同等对待 $D(E_i)$ 中的每一元素; 另一方面, 对 $V(E)$ 以外的元素不感兴趣, 因为它们对这 m 个事件无任何贡献. 由模糊关系, 知道, 在 $D(E_i) \cap V(E) (V(E)$

的子集) 中所有元素与 $V(E)$ 的其它元素是模糊独立的. 因此, 对事件 E_i , $i = 1, 2, \dots, m$, 可以尽可能地实现 $D(E_i) \cap V(E)$ 中的元素. 这样, 有

$$f_i(\mathbf{x}) = \begin{cases} \max_{\mathbf{y} \in E_i^*} \text{Pos} \left\{ \begin{array}{l} g_j(\mathbf{y}, \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\}, & \mathbf{x} \in E_i \\ 0, & \mathbf{x} \notin E_i, \end{cases} \quad (10.8)$$

其中集合 E_i^* 由

$$E_i^* = \left\{ \mathbf{y} = (y_1, y_2, \dots, y_n) \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in D(E_i) \cap V(E) \\ j = 1, 2, \dots, n \end{array} \right\}$$

定义, $i = 1, 2, \dots, m$. 也就是说, 对 E_i^* 中任意元素

$$\mathbf{y} = (y_1, y_2, \dots, y_n).$$

决策分量 $y_j \in D(E_i) \cap V(E)$ 是固定的, 并等于 x_j , 而其余的分量是任意的. 对任意给定的决策 \mathbf{x} , 容易确定 $\mathbf{y}_i^*(\mathbf{x})$ 使得事件 E_i 的诱导约束有下列形式,

$$g_j(\mathbf{y}_i^*(\mathbf{x}), \xi) \leq 0, \quad j = 1, 2, \dots, p, \quad (10.9)$$

并且事件 E_i 的机会函数分别为

$$f_i(\mathbf{x}) = \begin{cases} \text{Pos} \{g_j(\mathbf{y}_i^*(\mathbf{x}), \xi) \leq 0, j = 1, 2, \dots, p\}, & \mathbf{x} \in E_i \\ 0, & \mathbf{x} \notin E_i, \end{cases}$$

$i = 1, 2, \dots, m$.

10.3 相关机会目标规划

模糊环境下的相关机会目标规划是复杂的不确定决策系统中的目标规划的一个推广. 当管理目标给定时, 在给定的优先结构

下, 目标函数可以是极小化偏差 (包括正偏差、负偏差或正负偏差). 这样, 根据决策者的目标值, 可以把一个模糊决策系统建模为相关机会目标规划,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ f_i(\mathbf{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ g_j(\mathbf{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{array} \right. \quad (10.10)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

f_i = 目标约束中的机会函数或普通的实值函数: $\Re^n \rightarrow \Re^1$,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数.

对相关机会目标规划模型, 既然存在多个优先级, 可以依次由高到低考虑目标.

在第一优先级中, 假设存在 t 个事件, 分别表示为 E_1, E_2, \dots, E_t . 定义

$$E = E_1 \cap E_2 \cap \dots \cap E_t$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_t).$$

类似于多目标的情况, 对 t 个事件可以产生诱导约束和机会函数 $f_1(x), f_2(x), \dots, f_t(x)$, 因此, 有

$$f_i(x) = \begin{cases} \max_{y \in E_i^*} \text{Pos} \{g_j(y, \xi) \leq 0, j = 1, 2, \dots, p\}, & x \in E_i \\ 0, & x \notin E_i, \end{cases}$$

其中集合 E_i^* 分别由

$$E_i^* = \left\{ y = (y_1, y_2, \dots, y_n) \mid \begin{array}{l} y_j = x_j \text{ 若 } y_j \in D(E_i) \cap V(E) \\ j = 1, 2, \dots, n \end{array} \right\}$$

定义, $i = 1, 2, \dots, t$. 即对 E_i^* 中任意元素 $y = (y_1, y_2, \dots, y_n)$, 决策变量 $y_j \in D(E_i) \cap V(E)$ 是固定的, 并等于 x_j , 而其余元素则是任意的. 对任何给定的决策 x , 容易确定 $y_i^*(x)$ 使得事件 E_i 的诱导约束有下面的形式,

$$g_j(y_i^*(x), \xi) \leq 0, \quad j = 1, 2, \dots, p. \quad (10.11)$$

事件 E_i 的机会函数分别为

$$f_i(x) = \begin{cases} \text{Pos} \{g_j(y_i^*(x), \xi) \leq 0, j = 1, 2, \dots, p\}, & x \in E_i \\ 0, & x \notin E_i, \end{cases}$$

$i = 1, 2, \dots, t$. 这样, 可以计算所有的机会函数 $f_i(x)$, 并得到所有正负偏差 d_i^+ 和 d_i^- , $i = 1, 2, \dots, t$.

在第二个优先级中, 假设有事件 $E_{t+1}, E_{t+2}, \dots, E_{t'}$, 只要继承第一优先级所做的决策, 通过定义

$$E = E_1 \cap E_2 \cap \dots \cap E_t \cap E_{t+1} \cap E_{t+2} \cap \dots \cap E_{t'}$$

和

$$V(E) = V(E_1) \cup \dots \cup V(E_t) \cup V(E_{t+1}) \cup \dots \cup V(E_{t'}),$$

就可类似地分别导出事件 $E_{t+1}, E_{t+2}, \dots, E_{t'}$ 的诱导约束和机会函数 $f_{t+1}(\mathbf{x}), f_{t+2}(\mathbf{x}), \dots, f_{t'}(\mathbf{x})$. 一般地, 只要继承前面优先级所做的决策, 按同样的方法就可讨论下一级的目标.

10.4 机会函数的模糊模拟

在文献 [87, 88] 中, 模糊模拟已经用来处理模糊系统约束

$$\text{Pos}\{g_j(\mathbf{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha$$

和模糊目标函数

$$\text{Pos}\{f(\mathbf{x}, \xi) \geq \bar{f}\} \geq \alpha$$

以及模糊目标约束

$$\text{Pos}\{f(\mathbf{x}, \xi) + d^- - d^+ = b\} \geq \alpha.$$

本节里, 将介绍如何使用模糊模拟技术计算给定事件的机会函数. 设 E 是模糊决策系统中的一个事件. 又假设在点 \mathbf{x} 处事件 E 的诱导约束为

$$g_j(\mathbf{y}, \xi) \leq 0, \quad j = 1, 2, \dots, p,$$

其中 \mathbf{y} 由点 \mathbf{x} 导出. 则事件 E 的机会函数为

$$f(\mathbf{x}) = \begin{cases} \text{Pos}\{g_j(\mathbf{y}, \xi) \leq 0, j = 1, 2, \dots, p\}, & \mathbf{x} \in E \\ 0, & \mathbf{x} \notin E. \end{cases}$$

若 $\mathbf{x} \notin E$, 可以立即得到 $f(\mathbf{x}) = 0$. 对每一个固定的决策 $\mathbf{x} \in E$, 首先置 $f(\mathbf{x}) = 0$. 然后由模糊向量 ξ 随机生成一个清晰向量 ξ^0 . 实际上, 对可能性较低的决策向量不感兴趣. 所以, 可以事先置一水平 α_0 , 然后, 从模糊向量 ξ 的 α_0 水平截集中随机产生清晰

向量 ξ^0 . 如果模糊向量 ξ 的 α_0 水平截集过于复杂难于处理, 可以从包含模糊向量 ξ 的 α_0 水平截集的超几何体 Ω 中抽取向量, 接受与否依赖于 $\mu(\xi^0) \geq \alpha_0$ 是否成立. 当然, 超几何体越小, 模糊模拟越有效. 如果 $g_j(y, \xi^0) \leq 0, j = 1, 2, \dots, p$ 及 $f(x) < \mu(\xi^0)$, 则置 $f(x) = \mu(\xi^0)$. 重复以上的过程 N 次, 则值 $f(x)$ 可以作为它的估计值.

步骤 1 置 $f(x) = 0$;

步骤 2 从模糊向量 ξ 的 α_0 水平截集中随机生成向量 ξ^0 ;

步骤 3 若 $g_j(y, \xi^0) \leq 0, j = 1, 2, \dots, p$ 和 $f(x) < \mu(\xi^0)$, 则置

$$f(x) = \mu(\xi^0);$$

步骤 4 重复步骤 2 和 3 共 N 次;

步骤 5 返回 $f(x)$.

10.5 基于模糊模拟的遗传算法

对随机机会约束规划, Iwamura 和 Liu^[60] 提出了基于随机模拟的遗传算法; 对模糊机会约束规划, Liu 和 Iwamura^[87, 88] 提出了基于模糊模拟的遗传算法. 随机环境下的相关机会规划可以通过基于随机模拟的遗传算法得到解决 (见文献 [83, 85, 86]).

事实上, 只要使用模糊模拟计算模糊环境下的机会函数, 基于模糊模拟的遗传算法对求解模糊环境下的相关机会规划模型也是适用的.

遗传算法程序

步骤 0 输入参数 pop_size 、 P_c 和 P_m ;

步骤 1 初始产生 pop_size 个染色体;

步骤 2 对染色体进行交叉和变异操作;

步骤 3 使用模糊模拟计算所有染色体的目标值;

步骤 4 根据目标值使用基于序的评价函数计算每个染色体的适应度;

步骤 5 旋转赌轮选择染色体;

步骤 6 重复步骤 2 到步骤 5 直到给定的次数完成为止;

步骤 7 把最好的染色体作为最优解.

求解模糊环境下的相关机会规划的基于模糊模拟的遗传算法已编写成 C 语言程序. 这里所给出的数值例子均是在 PC 机上实现的, 所使用的参数分别为: 种群规模为 30, 交叉概率 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 基于序的评价函数中的参数 $\alpha = 0.05$. 进化过程中的每次模糊模拟执行 2000 次.

例 1 考虑下面的单目标相关机会规划模型,

$$\begin{cases} \max f(x) = \text{Pos} \{x_1^2 + x_2^2 + x_3^2 = 1\} \\ \text{s.t.} \\ x_1 + x_2 + x_3 \leq \tilde{a} \\ \tilde{b}(x_1 + x_2)x_3 \geq \tilde{c} \\ x_1, x_2, x_3 > 0 \end{cases}$$

其中 \tilde{a} 和 \tilde{b} 是模糊数, 其隶属函数分别为

$$\mu_{\tilde{a}}(\xi) = \exp(-|\xi - 1|), \quad \mu_{\tilde{b}}(\xi) = \frac{1}{1 + (\xi - 10)^2},$$

\tilde{c} 是梯形模糊数 $(0, 1, 2, 3)$.

在非负条件 $x_1, x_2, x_3 > 0$ 下, 仅有一个事件 $x_1^2 + x_2^2 + x_3^2 = 1$, 它的诱导约束为

$$\begin{cases} x_1 + x_2 + x_3 \leq \tilde{a} \\ \tilde{b}(x_1 + x_2)x_3 \geq \tilde{c}. \end{cases}$$

这样, 事件的机会函数 $f(x)$ 可以表示为

$$f(x) = \begin{cases} \text{Pos} \left\{ \begin{array}{l} x_1 + x_2 + x_3 \leq \tilde{a} \\ \tilde{b}(x_1 + x_2)x_3 \geq \tilde{c} \end{array} \right\}, & \begin{array}{l} x_1^2 + x_2^2 + x_3^2 = 1 \\ x_1, x_2, x_3 > 0 \end{array} \\ 0, & \text{其它.} \end{cases}$$

可以通过模糊模拟计算机机会函数. 首先, 从超几何体 $[0, 1] \times [0, 1] \times [0, 1]$ 中抽取染色体 $V = (x_1, x_2, x_3)$, 并按下面方法把它转化为满足约束 $x_1^2 + x_2^2 + x_3^2 = 1$ 的一个解,

$$x'_i = \frac{x_i}{\sqrt{x_1^2 + x_2^2 + x_3^2}}, \quad i = 1, 2, 3,$$

显然有 $x_1'^2 + x_2'^2 + x_3'^2 = 1$.

经过 400 代进化以后, 基于模糊模拟的遗传算法所给出的最优解为

$$(x_1^*, x_2^*, x_3^*) = (0.08656, 0.00331, 0.99624),$$

这个解保证满足事件的可能性达到 $f(x^*) = 0.9175$.

例 2 考虑相关机会目标规划

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, d_3^-, d_4^+\} \\ \text{s.t.} \\ f_1(x) + d_1^- - d_1^+ = 0.98 \\ f_2(x) + d_2^- - d_2^+ = 0.95 \\ f_3(x) + d_3^- - d_3^+ = 0.90 \\ f_4(x) + d_4^- - d_4^+ = 0 \\ x_1 + x_2 + x_5 \leq \tilde{a} \\ x_3 + x_4 \leq \tilde{b} \\ x_6 + x_7 \leq \tilde{c} \\ x_i \geq 0, \quad i = 1, 2, \dots, 7 \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, 4 \end{array} \right.$$

其中 \tilde{a} 是梯形模糊数 $(3, 4, 5, 6)$, \tilde{b} 是三角模糊数 $(2, 3, 4)$, 模糊数 \tilde{c} 的隶属函数为 $\mu_{\tilde{c}}(\xi) = 1/[1 + (\xi - 2)^2]$, 并且

$$\begin{cases} f_1(\mathbf{x}) = \text{Pos}\{x_1^2 + x_3^2 = 9\}, \\ f_2(\mathbf{x}) = \text{Pos}\{x_2 + x_4 = 3\}, \\ f_3(\mathbf{x}) = \text{Pos}\{x_5^2 + x_6 = 7\}, \\ f_4(\mathbf{x}) = x_4 + x_6. \end{cases}$$

注意决策分量 x_7 在目标约束中并没有出现.

由模糊环境可以得到下面的模糊关系: 7 个决策分量分成 3 组, 分别为 $\{x_1, x_2, x_5\}$, $\{x_3, x_4\}$ 和 $\{x_6, x_7\}$, 这 3 组之间是模糊独立的, 每组内部任何元素都是模糊相关的, 并且有相同的实现机会.

在第一优先级中, 有一个事件, 记为 E_1 , 由 $x_1^2 + x_3^2 = 9$ 表示. 显然 $V(E_1) = \{x_1, x_3\}$, 并且事件 E_1 的诱导约束为

$$\begin{cases} x_1 \leq \tilde{a} \\ x_3 \leq \tilde{b}, \end{cases}$$

这是因为在当前的优先级中, 其余的分量可以暂时看成是零, 这样, 事件 E_1 的机会函数 $f_1(\mathbf{x})$ 为

$$\begin{aligned} f_1(\mathbf{x}) &= \text{Pos}\{x_1^2 + x_3^2 = 9\} \\ &= \begin{cases} \text{Pos}\{x_1 \leq \tilde{a}, x_3 \leq \tilde{b}\}, & x_1^2 + x_3^2 = 9, \mathbf{x} \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

在第二优先级中, 有两个事件 E_2 和 E_3 , 分别由 $x_2 + x_4 = 3$ 和 $x_5^2 + x_6 = 7$ 表示. 同时也有 $V(E_2) = \{x_2, x_4\}$ 和 $V(E_3) = \{x_5, x_6\}$. 记 $E = E_1 \cap E_2 \cap E_3$, 则 $V(E) = V(E_1) \cup V(E_2) \cup V(E_3) =$

$\{x_1, x_2, x_3, x_4, x_5, x_6\}$. 既然

$$\begin{aligned} D(E_2) \cap V(E) &= \{x_1, x_2, x_5, x_3, x_4\} \cap \{x_1, x_2, x_3, x_4, x_5, x_6\} \\ &= \{x_1, x_2, x_3, x_4, x_5\}, \end{aligned}$$

事件 E_2 的诱导约束为

$$\begin{cases} x_1 + x_2 + x_5 \leq \tilde{a} \\ x_3 + x_4 \leq \tilde{b}, \end{cases}$$

从而导出事件 E_2 的机会函数 $f_2(x)$ 如下,

$$\begin{aligned} f_2(x) &= \text{Pos}\{x_2 + x_4 = 3\} \\ &= \begin{cases} \text{Pos}\left\{\begin{array}{l} x_1 + x_2 + x_5 \leq \tilde{a} \\ x_3 + x_4 \leq \tilde{b} \end{array}\right\}, & x_2 + x_4 = 3, x \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

另外, 由

$$\begin{aligned} D(E_3) \cap V(E) &= \{x_1, x_2, x_5, x_6, x_7\} \cap \{x_1, x_2, x_3, x_4, x_5, x_6\} \\ &= \{x_1, x_2, x_5, x_6\} \end{aligned}$$

得到事件 E_3 上的诱导约束为

$$\begin{cases} x_1 + x_2 + x_5 \leq \tilde{a} \\ x_6 \leq \tilde{c}, \end{cases}$$

从而导出事件 E_3 的机会函数 $f_3(x)$ 为

$$\begin{aligned} f_3(x) &= \text{Pos}\{x_5^2 + x_6 = 7\} \\ &= \begin{cases} \text{Pos}\left\{\begin{array}{l} x_1 + x_2 + x_5 \leq \tilde{a} \\ x_6 \leq \tilde{c} \end{array}\right\}, & x_5^2 + x_6 = 7, x \geq 0 \\ 0, & \text{其它.} \end{cases} \end{aligned}$$

在第三优先级中, 不存在机会函数, 所以可以立即得到偏差.

可以从超几何体 $[0, 1]^6$ 中抽取染色体 $V = (x_1, x_2, \dots, x_6)$, 并按下面的方法把它转化为可行解

$$\begin{aligned} x'_1 &= \frac{x_1}{\sqrt{Q_1}}, & x'_2 &= \frac{x_2}{Q_2}, & x'_3 &= \frac{x_3}{\sqrt{Q_1}}, \\ x'_4 &= \frac{x_4}{Q_2}, & x'_5 &= \frac{x_5}{\sqrt{Q_3}}, & x'_6 &= \frac{x_6}{Q_3}, \end{aligned}$$

其中 $Q_1 = (x_1^2 + x_3^2)/9$, $Q_2 = (x_2 + x_4)/3$ 和 $Q_3 = (x_5^2 + x_6^2)/7$, 并有 $x'^2_1 + x'^2_3 = 9$, $x'_2 + x'_4 = 3$ 和 $x'^2_5 + x'_6 = 7$.

经过 1000 代进化以后, 基于模糊模拟的遗传算法给出的最优解为

$$\mathbf{x}^* = (0.0000, 2.9186, 3.0000, 0.0814, 2.1611, 2.3296, 0.0000),$$

此解满足第一个目标, 但第二个和第三个目标分别为 0.0315 和 2.4110. 事实上, 有

$$\begin{aligned} f_1(\mathbf{x}^*) &= 1.0000, & f_2(\mathbf{x}^*) &= 0.9185, \\ f_3(\mathbf{x}^*) &= 0.9020, & f_4(\mathbf{x}^*) &= 2.4110. \end{aligned}$$

10.6 注

本章里, 提供了模糊环境下的相关机会规划以及相关机会多目标规划和相关机会目标规划. 同时也介绍了模糊环境下的不确定环境、事件、机会函数和诱导约束以及模糊关系等概念. 最后, 通过一些数值例子解释了基于模糊模拟的遗传算法的有效性.

第 11 章

带有模糊决策的模糊规划

传统的数学规划模型提供的清晰决策向量使得所要考虑的目标达到最优值。然而，出于对实际应用的考虑，有时，我们提供的应是模糊决策而不是清晰决策。Bouchon-Meunier 等^[15]总结了在一个模糊集上极大化一个数值函数的各种各样的方法。Buckley 和 Hayashi^[16]介绍了通过选择一个最优模糊集去极大化一个实值函数的模糊遗传算法，并应用到模糊优化问题、模糊极大流问题和模糊回归以及模糊控制等领域。

更一般地，Liu 和 Iwamura^[93]提出了带有模糊决策的机会约束规划以及机会约束多目标规划和机会约束目标规划。另外，Liu^[94]建立了带有模糊决策的相关机会规划并推广到相关机会多目标规划和相关机会目标规划。

本章里，我们首先介绍带有模糊决策的机会约束规划和相关机会规划的基本理论，并为求解这类模型设计了一个基于模糊模拟的遗传算法，最后给出一些数值例子解释如何应用遗传算法求解此类问题。

11.1 模 糊 决 策

一方面，实际控制器并不一定是精确的，所以 Zadeh^[144]提出模糊逻辑控制系统，并应用到各种生产领域，另一方面，即使我们提供了清晰值，管理决策有时也不可能在实际中准确地执行。于是，我们可以做一更合理的假设，即不但为控制问题所提供的是模糊决策，而且为决策问题所提供的也是模糊决策，而不是清

晰决策. 更进一步, 假定每一个决策限制在模糊集 (不一定是有限个或可数个) 构成的参考类中, 其中模糊集由决策系统的性质确定. 例如, (a) 一个决策可以表示成语言变量, 如较大、中等、较小、零等; (b) 一个决策也可以是由下面的隶属函数所描述的类中的一个模糊集,

$$\mu(x) = \frac{1}{1 + \|x - t\|}, \quad (11.1)$$

其中 $t \in [0, 10]$.

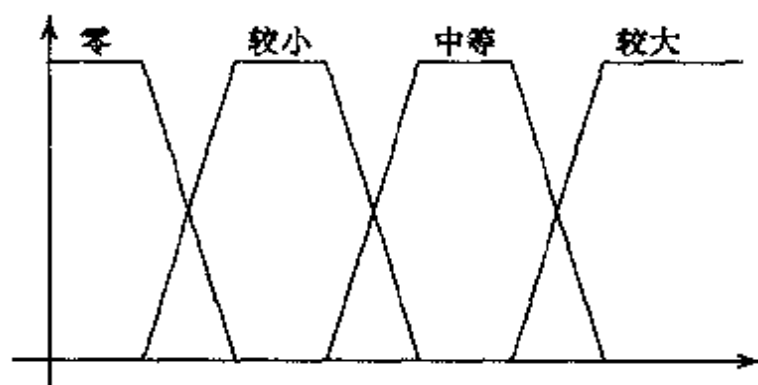


图 11.1 较大、中等、较小和零的隶属函数

假设一个模糊决策为一个 n 维向量 $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$, 其中每一个模糊分量 \tilde{x}_i 分别在模糊集所构成的参考类 $X_i (i = 1, 2, \dots, n)$ 中取值, 而所有参考类是预先给定的. 这样模糊决策取值于参考类的 Cartesian 积

$$\mathcal{X} = X_1 \otimes X_2 \otimes \dots \otimes X_n. \quad (11.2)$$

11.2 机会约束规划模型

带有模糊决策的单目标机会约束规划具有下面的形式,

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{array} \right. \quad (11.3)$$

其中 \bar{x} 是 (11.2) 所定义的参考类 \mathcal{X} 中的一个模糊向量, ξ 是一个已知的模糊参数向量, $f(\bar{x}, \xi)$ 是目标函数, $g_j(\bar{x}, \xi)$ 是约束函数, $j = 1, 2, \dots, p$, α 和 β 分别是模糊约束和模糊目标给定的置信水平, $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性.

所以一个模糊解 \bar{x} 是可行的当且仅当事件

$$\{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\}$$

的可能性至少是 α . 对每一个固定的模糊解 \bar{x} , 目标值 \bar{f} 是目标函数 $f(\bar{x}, \xi)$ 在可能性至少为 β 时所取的最大值.

有时, 我们可以把模糊决策问题建模为如下形式的机会约束规划,

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\bar{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{array} \right. \quad (11.4)$$

其中 α_j 分别是第 j 个约束的事先给定的置信水平, $j = 1, 2, \dots, p$.

若我们的目的是极小化目标函数 $f(\tilde{x}, \xi)$, 则相应的机会约束规划可以表示成如下的形式,

$$\begin{cases} \min \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\tilde{x}, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (11.5)$$

其中目标值 \bar{f} 是目标函数 $f(\tilde{x}, \xi)$ 在可能性至少为 β 时所取的最小值.

作为单目标机会约束规划的一个推广, 机会约束多目标规划可以表示成,

$$\begin{cases} \max [\bar{f}_1, \bar{f}_2, \dots, \bar{f}_m] \\ \text{s.t.} \\ \text{Pos} \{f_i(\tilde{x}, \xi) \geq \bar{f}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (11.6)$$

其中 $f_i(\tilde{x}, \xi)$ 是目标函数, β_i 分别是第 i 个目标的置信水平, 目标值 \bar{f}_i 是目标函数 $f_i(\tilde{x}, \xi)$ 在可能性至少为 β_i 时所取的最大值, $i = 1, 2, \dots, m$.

根据决策者给定的优先结构和目标值, 也可以把模糊决策系统建模为机会约束目标规划,

$$\begin{cases} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ \text{Pos} \{f_i(\tilde{x}, \xi) + d_i^- - d_i^+ = b_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{cases} \quad (11.7)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数.

注意在带有模糊决策的机会约束目标规划模型中, 正偏差和负偏差可以理解为使得下式成立的 d_i^+ 和 d_i^- 的最小非负值,

$$\begin{aligned} \text{Pos} \{d_i^- \geq b_i - f_i(\bar{x}, \xi)\} &\geq \beta_i, \\ \text{Pos} \{d_i^+ \geq f_i(\bar{x}, \xi) - b_i\} &\geq \beta_i, \end{aligned} \quad (11.8)$$

$i = 1, 2, \dots, m$.

11.3 相关机会规划模型

不确定环境、事件、机会函数和诱导约束是相关机会规划理论模架中关键的术语. 本节里, 我们将重新介绍这些概念, 并提供一系列带有模糊决策的相关机会规划模型.

所谓的不确定环境, 我们认为是由一些不确定条件约束的解集合, 这些约束条件一般可以表述为

$$g_j(\bar{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p \quad (11.9)$$

其中 \bar{x} 为一个模糊决策向量, ξ 为模糊参数向量. 一个决策向量 \bar{x} 可行与否的模键性不仅由它本身引起而且与不确定参数向量 ξ 有关.

一个复杂的决策系统通常要执行多项任务, 并满足一些系统内部的要求. 我们用事件表示这些任务和内部要求, 通常表示成满足预定目标的模糊解 $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ 的集合, 如, $\tilde{x}_1 + \tilde{x}_2 = c$ 表示用户的需求 c 应该得到满足.

机会函数表示满足事件的可能性, 如 $f(\tilde{x}) = \text{Pos}\{\tilde{x}_1 + \tilde{x}_2 = c\}$, 其中 $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性.

现在的问题是如何通过不确定环境表示机会函数. 在第 7 章和第 10 章对清晰决策情况已予讨论. 这里, 我们分析模糊的情况. 设 E 是不确定决策系统中唯一的一个事件. 我们通过一个合适的模糊决策向量 $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ 使得该事件 E 得到满足. 通常, E 的实现仅与 $\{\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n\}$ 中部分分量有关而不是全部. 用 $V(E)$ 表示 \tilde{x} 的分量中对事件 E 必需的分量构成的集合. 这样, 事件 E 实现的可能性仅与 $V(E)$ 中的元素的实现有关. 若不确定决策系统中仅有一个事件 E , 则事件 E 的机会函数是 $V(E)$ 中元素实现的量大可能性, 即

$$f(\tilde{x}) = \max_{\tilde{y} \in E^*} \text{Pos}\{g_j(\tilde{y}, \xi) \leq 0, j = 1, 2, \dots, p \ \& \ \tilde{x} \in E\}, \quad (11.10)$$

或等价地

$$f(\tilde{x}) = \max_{\tilde{y} \in E^*} \text{Pos}\left\{ \begin{array}{l} g_j(\tilde{y}, \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\} \wedge \text{Pos}\{\tilde{x} \in E\}, \quad (11.11)$$

其中 $\tilde{x} \in E$ 的意思是模糊决策向量 \tilde{x} 满足事件 E , 并且

$$E^* = \left\{ \tilde{y} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n) \mid \begin{array}{l} \tilde{y}_j = \tilde{x}_j, \text{ 若 } \tilde{y}_j \in V(E) \\ j = 1, 2, \dots, n \end{array} \right\}. \quad (11.12)$$

注意对 E^* 中的任何元素 $\tilde{y} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n)$, 模糊决策分量 $\tilde{y}_j \in V(E)$ 是固定的, 并且等于 \tilde{x}_j , 其余的分量是任意的. 通常, 对

每一个给定的决策 \tilde{x} , 不难确定 (11.10) 的最优解 $\tilde{y}^*(\tilde{x})$ 中 $V(E)$ 以外的决策分量 \tilde{x}_j^* 值, 使得 $\tilde{y}^*(\tilde{x}) = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n)$, 其中

$$\tilde{y}_j = \begin{cases} \tilde{x}_j, & \tilde{y}_j \in V(E) \\ \tilde{x}_j^*, & \tilde{y}_j \notin V(E), \end{cases} \quad j = 1, 2, \dots, n.$$

事实上, 模糊决策分量 x_j^* 通常取值于一些极端的地方, 因此, 事件 E 的机会函数为

$$f(\tilde{x}) = \text{Pos} \left\{ \begin{array}{l} g_j(\tilde{y}^*(\tilde{x}), \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\} \wedge \text{Pos} \{ \tilde{x} \in E \}, \quad (11.13)$$

其中约束

$$g_j(\tilde{y}^*(\tilde{x}), \xi) \leq 0, \quad j = 1, 2, \dots, p \quad (11.14)$$

称为在点 \tilde{x} 处事件 E 的诱导约束. 到此为止, 我们已经介绍清楚了如何表示不确定环境下一个事件的机会函数.

带有模糊决策的单目标相关机会规划可以表示成

$$\begin{cases} \max f(\tilde{x}) \\ \text{s.t.} \\ g_j(\tilde{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (11.15)$$

其中 \tilde{x} 是一个 n 维模糊决策向量, ξ 是一个模糊向量参数, $f(\tilde{x})$ 是一个事件的机会函数, 不确定环境为 $g_j(\tilde{x}, \xi) \leq 0, j = 1, 2, \dots, p$.

既然一个复杂的决策系统通常执行多项任务, 毫无疑问, 在决策过程中存在多个可能的目标 (它们中间有些可能是机会函数). 一个带有模糊决策的典型的相关机会多目标规划可以表示为

$$\begin{cases} \max f(\tilde{x}) = [f_1(\tilde{x}), f_2(\tilde{x}), \dots, f_m(\tilde{x})] \\ \text{s.t.} \\ g_j(\tilde{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p, \end{cases} \quad (11.16)$$

其中 $f(\bar{x})$ 为由 m 个函数 $f_i(\bar{x})$ 构成的向量值函数, $f_i(\bar{x})$ 中有些是机会函数.

为了表示多个事件的机会函数, 假设已经知道在第 10.2 节第 187 页中定义的模糊决策分量 $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n$ 之间的模糊关系, 即决策向量的 n 个分量可以划分成 k 组, 这 k 组之间是相互模糊独立的, 而每组内部元素之间是模糊相关的, 并有相同的实现机会.

根据模糊关系, 对事件 E , 用 $D(E)$ 表示与 $V(E)$ 中至少一个元素模糊相关的元素构成的集合. 对实际问题, 一般容易确定集合 $D(E)$.

对多目标的情况, 不失一般性, 假设存在 m 个事件 E_i , 机会函数分别为 $f_i(\bar{x})$, $i = 1, 2, \dots, m$. 记

$$E = E_1 \cap E_2 \cap \dots \cap E_m$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_m),$$

其中 $\bar{x} \in E$ 表示模糊决策向量 \bar{x} 满足所有事件 E_1, E_2, \dots, E_m , $V(E)$ 表示 m 个事件所必需的分量构成的集合. 模据模糊关系, 用 $D(E), D(E_i), i = 1, 2, \dots, m$ 分别表示所有与 $V(E), V(E_i) (i = 1, 2, \dots, m)$ 中至少一个元素模糊相关的元素构成的集合. 既然所有事件在同一优先级中, 因此我们应在不牺牲其它事件实现的机会的前提下, 尽可能地实现每一个事件 E_i . 一方面, 我们平等地对待 $D(E_i)$ 中每一个元素; 另一方面, 我们对 $V(E)$ 以外的其它元素不感兴趣, 因为它们对这 m 个事件无任何贡献. 模据模糊关系, 我们知道 $D(E_i) \cap V(E) (V(E) \text{ 的子集})$ 中任何元素与 $V(E)$ 的其它元素是模糊独立的. 因此, 对事件 E_i , 我们应当尽可能地实

现 $D(E_i) \cap V(E)$ 中的元素, 于是有

$$f_i(\bar{x}) = \max_{\tilde{y} \in E_i^*} \text{Pos} \left\{ \begin{array}{l} g_j(\tilde{y}, \xi) \leq 0 \\ j = 1, 2, \dots, p \end{array} \right\} \wedge \text{Pos} \{ \bar{x} \in E_i \}, \quad (11.17)$$

其中集合 E_i^* 定义为

$$E_i^* = \left\{ \tilde{y} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n) \mid \begin{array}{l} \tilde{y}_j = \tilde{x}_j, \text{ 若 } \tilde{y}_j \in D(E_i) \cap V(E) \\ j = 1, 2, \dots, n \end{array} \right\},$$

$i = 1, 2, \dots, m$. 对每一个给定的模糊解 \bar{x} , 假设我们可以确定 $\tilde{y}_i^*(\bar{x})$, 使得事件 E_i 的诱导约束有下面的形式,

$$g_j(\tilde{y}_i^*(\bar{x}), \xi) \leq 0, \quad j = 1, 2, \dots, p, \quad (11.18)$$

则事件 E_i 的机会函数分别为

$$f_i(\bar{x}) = \text{Pos} \{ g_j(\tilde{y}_i^*(\bar{x}), \xi) \leq 0, j = 1, 2, \dots, p \} \wedge \text{Pos} \{ \bar{x} \in E_i \},$$

$i = 1, 2, \dots, m$.

带有模糊决策的相关机会目标规划可以看成是一个复杂的不确定决策系统中目标规划的一个推广. 当管理目标给定以后, 在一定的优先结构下, 目标函数可以是极小化偏差 (包括正偏差、负偏差或正负偏差). 因此, 根据决策者的优先结构和目标值, 一个模糊决策系统可以建模为相关机会目标规划模型,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ f_i(\bar{x}) + d_i^- - d_i^+ = b_i, \quad i = 1, 2, \dots, m \\ g_j(\bar{x}, \xi) \leq 0, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m. \end{array} \right. \quad (11.19)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

f_i = 目标约束中的机会函数或普通的实值函数,

b_i = 目标 i 的目标值,

l = 优先级个数,

m = 目标约束个数.

对相关机会目标规划模型, 既然存在多个优先级, 我们可以依次由高到低考虑目标.

在第一优先级中, 假设有 t 个事件, 分别表示为 E_1, E_2, \dots, E_t . 定义

$$E = E_1 \cap E_2 \cap \dots \cap E_t$$

和

$$V(E) = V(E_1) \cup V(E_2) \cup \dots \cup V(E_t).$$

类似于多目标的情况, 可以导出 t 个事件的诱导约束和机会函数 $f_1(\bar{x}), f_2(\bar{x}), \dots, f_t(\bar{x})$. 因此, 首先有

$$f_i(\bar{x}) = \max_{\tilde{y} \in E_i^*} \text{Pos} \{g_j(\tilde{y}, \xi) \leq 0, j = 1, 2, \dots, p\} \wedge \text{Pos} \{\bar{x} \in E_i\},$$

其中集合 E_i^* 定义为

$$E_i^* = \left\{ \tilde{y} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_n) \mid \begin{array}{l} \tilde{y}_j = \bar{x}_j, \text{ 若 } \tilde{y}_j \in D(E_i) \cap V(E) \\ j = 1, 2, \dots, n \end{array} \right\},$$

$i = 1, 2, \dots, t$. 对每一个给定的模糊解 \bar{x} , 假设我们能够确定 $\bar{y}_i^*(\bar{x})$, 使得事件 E_i 的诱导约束分别有下面的形式,

$$g_j(\bar{y}_i^*(\bar{x}), \xi) \leq 0, \quad j = 1, 2, \dots, p, \quad (11.20)$$

则事件 E_i 的机会函数为

$$f_i(\bar{x}) = \text{Pos} \{g_j(\bar{y}_i^*(\bar{x}), \xi) \leq 0, j = 1, 2, \dots, p\} \wedge \text{Pos} \{\bar{x} \in E_i\},$$

$i = 1, 2, \dots, t$.

在第二优先级中, 我们假设有事件 $E_{t+1}, E_{t+2}, \dots, E_{t'}$, 只要我们继承第一优先级所做的决策, 即定义

$$E = E_1 \cap E_2 \cap \dots \cap E_t \cap E_{t+1} \cap E_{t+2} \cap \dots \cap E_{t'}$$

和

$$V(E) = V(E_1) \cup \dots \cup V(E_t) \cup V(E_{t+1}) \cup \dots \cup V(E_{t'}).$$

就可以类似地导出事件 $E_{t+1}, E_{t+2}, \dots, E_{t'}$ 的诱导约束和机会函数 $f_{t+1}(\bar{x}), f_{t+2}(\bar{x}), \dots, f_{t'}(\bar{x})$.

一般地, 只要我们继承前面优先级所做的决策, 按类似的方法可以讨论接下来的目标.

11.4 模糊模拟

Liu 和 Iwamura^[93] 介绍了一种模糊模拟技术, 用以检验模糊系统约束

$$\text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha$$

以及处理模糊目标函数

$$\text{Pos} \{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta$$

和模糊目标约束

$$\text{Pos} \{f(\bar{x}, \xi) + d^- - d^+ = b\} \geq \beta,$$

另外, Liu^[94] 为计算机会函数

$$f(\bar{x}) = \text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\}$$

提供了模糊模拟技术, 其中 \bar{x} 和 ξ 假设为模糊向量, 其隶属函数分别为 $\nu(\cdot)$ 和 $\mu(\cdot)$. 本节里, 我们将介绍这些技术.

检验模糊系统约束

由模糊数运算的定义, 对一个给定的决策向量 \bar{x} , 我们说

$$\text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha \quad (11.21)$$

当且仅当存在两个清晰向量 ξ^0 和 \bar{x}^0 使得 $g_j(\bar{x}^0, \xi^0) \leq 0, j = 1, 2, \dots, k$ 和 $\mu(\xi^0) \wedge \nu(\bar{x}^0) \geq \alpha$. 这样, 我们可以由模糊向量 ξ 和 \bar{x} 生成清晰向量 ξ^0 和 \bar{x}^0 使得 $\mu(\xi^0) \geq \alpha$ 和 $\nu(\bar{x}^0) \geq \alpha$, 即分别在模糊向量 ξ 和 \bar{x} 的 α 水平截集抽取向量. 若模糊向量 ξ 的 α 水平截集过于复杂, 我们可以从包含 α 水平截集的超几何体 Ω 中抽取一个向量 ξ^0 , 接收与否依赖于 $\mu(\xi^0) \geq \alpha$ 是否成立. 当然, 为了加快模拟速度, 超几何体 Ω 应设计得尽可能小. 对模糊决策 \bar{x} , 抽样过程类似于 ξ . 如果两个清晰向量 ξ^0 和 \bar{x}^0 满足条件 $g_j(\bar{x}^0, \xi^0) \leq 0, j = 1, 2, \dots, k$, 则我们可以确信 $\text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, k\} \geq \alpha$. 否则从模糊向量 ξ 和 \bar{x} 的 α 水平截集中分别重新生成两个清晰向量 ξ^0 和 \bar{x}^0 , 并重新检验约束条件. 经过给定的次数 N 之后, 如果没有清晰向量 ξ^0 和 \bar{x}^0 满足 $g_j(\bar{x}^0, \xi^0) \leq 0, j = 1, 2, \dots, k$, 则认为给定的决策向量 \bar{x} 对机会约束 (11.21) 是不可行的. 以上过程可以总结如下,

- 步骤 1 分别从模糊向量 ξ 和 \bar{x} 的 α 水平截集中随机地生成 ξ^0 和 \bar{x}^0 ;
- 步骤 2 若 $g_j(\bar{x}^0, \xi^0) \leq 0, j = 1, 2, \dots, k$, 返回“可行”;
- 步骤 3 重复步骤 1 和步骤 2 共 N 次;
- 步骤 4 返回“不可行”.

处理模糊目标函数

对一个模糊目标函数

$$\text{Pos} \{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta, \quad (11.22)$$

出于极大化 \bar{f} 的目的, 对给定的模糊决策向量 \bar{x} , 我们必须找到最大的值 \bar{f} 使得 (11.22) 成立. 首先, 置 $\bar{f} = -\infty$. 我们分别从模糊向量 ξ 和 \bar{x} 的 α 水平模集中随机地生成 ξ^0 和 \bar{x}^0 , 使得 $\mu(\xi^0) \wedge \nu(\bar{x}^0) \geq \beta$, 即分别在模糊向量 ξ 和 \bar{x} 的 α 水平截集中抽取向量. 只要 $\bar{f} < f(\bar{x}^0, \xi^0)$, 则置 $\bar{f} = f(\bar{x}^0, \xi^0)$. 重复这个过程 N 次, 所得的值 \bar{f} 作为在模糊决策 \bar{x} 处的目标值. 总结以上过程如下,

- 步骤 1 置 $\bar{f} = -\infty$;
- 步骤 2 分别从模糊向量 ξ 和 \bar{x} 的 α 水平截集中随机地生成 ξ^0 和 \bar{x}^0 ;
- 步骤 3 若 $\bar{f} < f(\bar{x}^0, \xi^0)$, 则置 $\bar{f} = f(\bar{x}^0, \xi^0)$;
- 步骤 4 重复步骤 2 和步骤 3 共 N 次;
- 步骤 5 返回 \bar{f} .

处理模糊目标约束

带有模糊决策的机会约束目标规划可能具有下面形式的模糊目标约束,

$$\text{Pos} \{f(\bar{x}, \xi) + d^- - d^+ = b\} \geq \beta, \quad (11.23)$$

我们的目的是对给定的模糊决策 \bar{x} 找到最小的非负值 d^- 和 d^+ 值得

$$\text{Pos}\{d^- \geq b - f(\bar{x}, \xi)\} \geq \beta, \quad \text{Pos}\{d^+ \geq f(\bar{x}, \xi) - b\} \geq \beta.$$

首先, 置 $d^- = d^+ = +\infty$. 然后, 我们分别从模糊向量 ξ 和 \bar{x} 的 α 水平截集中随机地生成 ξ^0 和 \bar{x}^0 , 若 $d^- > b - f(\bar{x}^0, \xi^0)$, 则置 $d^- = [b - f(\bar{x}^0, \xi^0)] \vee 0$. 同样, 若 $d^+ > f(\bar{x}^0, \xi^0) - b$, 则置 $d^+ = [f(\bar{x}^0, \xi^0) - b] \vee 0$. 重复以上生成清晰向量 ξ^0 和 \bar{x}^0 的过程 N 次. 所得的值 d^+ 和 d^- 分别作为在模糊决策 \bar{x} 处正负偏差的估计值. 总结以上的过程如下,

步骤 1 置 $d^+ = d^- = +\infty$;

步骤 2 分别从模糊向量 ξ 和 \bar{x} 的 α 水平截集中随机地生成 ξ^0 和 \bar{x}^0 ;

步骤 3 若 $d^+ > f(\bar{x}^0, \xi^0) - b$, 则置 $d^+ = [f(\bar{x}^0, \xi^0) - b] \vee 0$;

步骤 4 若 $d^- > b - f(\bar{x}^0, \xi^0)$, 则置 $d^- = [b - f(\bar{x}^0, \xi^0)] \vee 0$;

步骤 5 重复步骤 2 到步骤 4 共 N 次;

步骤 6 返回 d^+ 和 d^- .

计算机会函截

假设 E 是模糊决策系统中的一个事件. 在模糊解 \bar{x} 处事件 E 的诱导约束为 $g_j(\bar{y}, \xi) \leq 0, j = 1, 2, \dots, p$, 其中模糊点 \bar{y} 是由模糊点 \bar{x} 导出的. 则事件 E 的机会函数为

$$f(\bar{x}) = \text{Pos}\{g_j(\bar{y}, \xi) \leq 0, j = 1, 2, \dots, p\} \wedge \text{Pos}\{\bar{x} \in E\}.$$

一般情况下, 我们无法用解析的方法估计机会函数 $f(\bar{x})$ 的值. 所以我们必须使用下面的模糊模拟技术. 对每一个固定的模糊决策 \bar{x} , 首先, 置 $f(\bar{x}) = 0$. 然后, 分别从模糊向量 ξ 和 \bar{x} 的 α 水平截集中随机地生成 ξ^0 和 \bar{x}^0 . 实际上, 我们对可能性太低的决策

向量并不感兴趣, 所以, 一般事先给定一个水平, 如 α_0 , 然后, 分别从模糊向量 ξ^0 和 \tilde{y}^0 的 α_0 水平截集中随机地生成 ξ^0 和 \tilde{x}^0 . 若 $g_j(\tilde{y}^0, \xi^0) \leq 0, j = 1, 2, \dots, p$ 和 $f(\tilde{x}) < \mu(\xi^0) \wedge \nu(\tilde{y}^0) \wedge \text{Pos}\{\tilde{x} \in E\}$, 则置 $f(\tilde{x}) = \mu(\xi^0) \wedge \nu(\tilde{y}^0) \wedge \text{Pos}\{\tilde{x} \in E\}$. 重复这个过程直到给定的次数 N , 值 $f(\tilde{x})$ 作为解 \tilde{x} 的可能性的估计. 以上的过程可以总结如下,

步骤 1 置 $f(\tilde{x}) = 0$;

步骤 2 分别从模糊向量 ξ 和 \tilde{y} 的 α_0 水平截集中随机地生成 ξ^0 和 \tilde{x}^0 ;

步骤 3 若 $g_j(\tilde{y}^0, \xi^0) \leq 0, j = 1, 2, \dots, p$ 和 $f(\tilde{x}) < \mu(\xi^0) \wedge \nu(\tilde{y}^0) \wedge \text{Pos}\{\tilde{x} \in E\}$, 则置 $f(\tilde{x}) = \mu(\xi^0) \wedge \nu(\tilde{y}^0) \wedge \text{Pos}\{\tilde{x} \in E\}$;

步骤 4 重复步骤 2 和步骤 3 共 N 次;

步骤 5 返回 $f(\tilde{x})$.

11.5 基于模糊模拟的遗传算法

本节里, 我们设计一个基于模糊模拟的遗传算法, 以求解带有模糊决策的机会约束规划和相关机会规划模型.

表示结构

传统上, 可以用二进制向量或浮点数向量代码表示一个解. 但对模糊解 (决策), 我们必须使用模糊染色体, 其中每一个基因是一个模糊集而不是清晰数.

这里我们使用模糊向量 $V = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ 作为一个染色体以表示机会约束规划或相关机会规划模型的一个模糊解, 其中, n 为维数, 所有分量 \tilde{x}_i 分别取自于模糊集合构成的参考类 $X_i (i = 1, 2, \dots, n)$ 中.

初始化过程

对每一个模糊基因 \tilde{x}_i ，从参考类 X_i 中随机抽取模糊集，形成一个模糊染色体 $V = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ 。

对机会约束规划模型，如果模糊模拟证明模糊染色体是不可行的，那么我们重新抽取模糊集，直到得到可行的染色体为止。重复以上过程 pop_size 次，我们可以得到 pop_size 个初始可行的染色体 $V_1, V_2, \dots, V_{pop_size}$ 。

对相关机会规划模型，既然我们对可能性太低的决策向量不感兴趣，因此，我们可以确定一个超几何体 Ω (若我们不能确定这样的超几何体 Ω ，则置集合 $\Omega = X_1 \otimes X_2 \otimes \dots \otimes X_n$)，我们仅从这个超几何体 Ω 中搜索最优解。超几何体 Ω 越小，初始化过程越快。因此，若产生的染色体证明不在 Ω 中，则重新抽取模糊集直到得到 Ω 中的模糊染色体为止。重复以上的过程 pop_size 次，可以得到 pop_size 个 Ω 中的初始染色体 $V_1, V_2, \dots, V_{pop_size}$ 。

评价函数

评价函数 (用 $eval(V)$ 表示)，用来对种群中的每个染色体 V 确定一个概率，以使该染色体被选中的可能性与它相对种群中其它染色体的适应性成比例，即通过赌轮，适应性强的染色体被选中产生后代的机会要大。

设目前该代中的染色体为 $V_1, V_2, \dots, V_{pop_size}$ ，我们根据染色体的序进行再生实验，而不是实际的目标值。无论何种数学规划 (单目标、多目标、或目标规划)，都可以做一合理假设，即在染色体 $V_1, V_2, \dots, V_{pop_size}$ 中，决策者可以给出一个序的关系，值这些染色体可以由好到坏进行重排，也就是说，一个染色体越好，其序号越小。通常根据模糊模拟所得到目标值进行重排。在进化系统中给定一个参数 $\alpha \in (0, 1)$ ，则基于序的评价函数的定义

如下,

$$\text{eval}(V_i) = a(1-a)^{i-1}, \quad i = 1, 2, \dots, \text{pop_size}, \quad (11.24)$$

其中 $i = 1$ 意味着染色体最好, $i = \text{pop_size}$ 说明染色体最坏.

选择过程

旋转赌轮 pop_size 次, 每次为新的种群选择一个模糊染色体, 具体方法如下:

步骤 1 对每一个模糊染色体 V_i , 计算累积概率 q_i ,

$$\begin{aligned} q_0 &= 0, \\ q_i &= \sum_{j=1}^i \text{eval}(V_j), \quad i = 1, 2, \dots, \text{pop_size}; \end{aligned} \quad (11.25)$$

步骤 2 从区间 $(0, q_{\text{pop_size}}]$ 中产生一个随机数 r ;

步骤 3 选择第 i 个染色体 V_i ($1 \leq i \leq \text{pop_size}$) 使得

$$q_{i-1} < r \leq q_i;$$

步骤 4 重复步骤 2 和步骤 3 共 pop_size 次, 这样得到 pop_size 个复制的模糊染色体.

交叉操作

定义一个参数 P_c 作为交叉操作的概率. 为了确定用来交叉操作的父代, 从 $i = 1$ 到 pop_size 重复以下的过程: 从区间 $[0, 1]$ 中生成随机数 r , 若 $r < P_c$, 则选择 V_i 作为一个父代.

用 V'_1, V'_2, V'_3, \dots 表示以上选择的父代, 并把它们分成下面的对:

$$(V'_1, V'_2), \quad (V'_3, V'_4), \quad (V'_5, V'_6), \quad \dots$$

下面用 (V'_1, V'_2) 解释对每对所进行的交叉操作. 首先, 在 1 到 n 之间随机生成一个整数作为交叉点, 记为 n' , 然后交换模糊染色体 V'_1 和 V'_2 第 n' 个基因以后的所有基因, 得到两个后代. 如果证明对于机会约束规划两个后代是可行的, 或对于相关机会规划在超几何体 Ω 中, 将用这两个后代取代它们的父代, 否则, 重新进行交叉操作.

变异操作

定义参数 P_m 作为变异概率. 类似于交叉操作中选择父代的过程, 从 $i = 1$ 到 pop_size 重复以下的过程: 从区间 $[0, 1]$ 中生成随机数 r , 若 $r < P_m$, 则选择 V_i 作为一个父代.

用 $V = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$ 表示以上选择的父代, 按下面的方法进行变异. 首先在 1 和 n 之间随机选择一个变异位置 n' , 然后从模糊集合构成的类 $X_{n'}$ 中抽取新的模糊集, 取代 V 的第 n' 个基因, 形成一个新的染色体 V' . 如果证明染色体 V' 对机会约束规划是不可行的或对相关机会规划不在超几何体 Ω 中, 则重新进行变异, 直到得到可行的染色体为止.

遗传算法程序

继选择、交叉和变异之后, 得到一个新的种群, 准备进行下一轮评价. 经过给定的循环次数以后遗传算法终止运算. 我们可以总结求解带有模糊决策的机会约束规划和相关机会规划模型的遗传算法如下:

步骤 0 输入参数 pop_size 、 a 、 P_c 和 P_m ;

步骤 1 初始产生 pop_size 个模糊染色体, 其中可能使用模糊模拟检验染色体可行性;

步骤 2 对模糊染色体进行交叉和变异操作, 其中可能使用模糊模拟检验后代的可行性;

步骤 3 使用模糊模拟计算所有模糊染色体的目标值;

步骤 4 根据目标值使用基于序的评价函数计算所有模糊染色体的适应度;

步骤 5 旋转赌轮选择模糊染色体;

步骤 6 重复步骤 2 到步骤 5 直到给定的循环次数;

步骤 7 把最好的染色体作为最优模糊解.

我们知道, 最好的染色体不一定出现在最后一代中, 所以在进化开始的时候, 必须把最好的染色体保留下来, 如果在新的种群中又发现更好的染色体, 则用它代替原来的染色体. 在进化完成之后, 这个染色体就看成是最优模糊解.

11.6 数值例子

求解带有模糊决策的模糊规划模型的基于模糊模拟的遗传算法已用 C 语言编成程序. 这里我们给出一些在 PC 机上实现的数值例子, 所使用参数分别为: 种群规模为 30, 交叉概率 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 评价函数中的参数 $\alpha = 0.05$, 每次模糊模拟执行 2000 次循环.

例 1 考虑下面的带有模糊决策的单目标机会约束规划模型,

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{ \tilde{c}_1 \tilde{x}_1 + \tilde{c}_2 \tilde{x}_2 + \tilde{c}_3 \tilde{x}_3 \geq \bar{f} \} \geq 0.9 \\ \text{Pos} \{ \tilde{a}_1 \tilde{x}_1^2 + \tilde{a}_2 \tilde{x}_2^2 + \tilde{a}_3 \tilde{x}_3^2 \leq 100 \} \geq 0.7 \\ \text{Pos} \{ \tilde{x}_1 \tilde{x}_2 \tilde{x}_3 \leq \tilde{b} \} \geq 0.8, \end{array} \right.$$

其中模糊决策 $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3)$ 是从 Cartesian 积 $X_1 \otimes X_2 \otimes X_3$ 中生成的, 参考类 X_1 由所有的三角模糊数 $(t-1, t, t+1)$ 组成, $t \in [1, 5]$, X_2 由所有隶属函数为 $\mu_2(x) = \exp(-|x-t|)$ 的模糊数组成, $t \in [2, 10]$, X_3 由所有隶属函数为 $\mu_3(x) = 1/[1+(x-t)^2]$ 的模糊数组成, $t \in [5, 8]$, 三角模糊参数 $\tilde{a}_1 = (0, 1, 2)$, $\tilde{a}_2 = (1, 2, 3)$, $\tilde{a}_3 = (2, 3, 4)$, 梯形模糊数 $\tilde{b} = (20, 25, 28, 30)$, 模糊数 \tilde{c}_1 , \tilde{c}_2 及 \tilde{c}_3 的隶属函数分别为 $\mu_{\tilde{c}_1}(x) = 1/[1+|x-1|]$, $\mu_{\tilde{c}_2}(x) = 1/[1+|x-2|]$ 和 $\mu_{\tilde{c}_3}(x) = 1/[1+|x-3|]$.

经过 600 代进化以后, 基于模糊模拟的遗传算法给出的最优模糊解 $(\tilde{x}_1^*, \tilde{x}_2^*, \tilde{x}_3^*)$ 为

$$\begin{cases} \tilde{x}_1^* = (0.369, 1.369, 2.369) \\ \mu_{\tilde{x}_2^*}(x) = \exp[-|x-5.336|] \\ \mu_{\tilde{x}_3^*}(x) = \frac{1}{1+(x-5.251)^2} \end{cases}$$

其目标值为 $\bar{f}^* = 30.4$.

例 2 考虑下面的带有模糊决策的机会约束目标规划,

$$\left\{ \begin{array}{l} \text{lexmin } \{d_1^-, d_2^-, d_3^-\} \\ \text{s.t.} \\ \text{Pos} \{ \tilde{x}_1^2 + \tilde{x}_2^2 + d_1^- - d_1^+ = \tilde{b} \} \geq 0.95 \\ \text{Pos} \{ \tilde{x}_3 + d_2^- - d_2^+ = \tilde{c} \} \geq 0.90 \\ \text{Pos} \{ \tilde{x}_4 + d_3^- - d_3^+ = 4 \} \geq 0.85 \\ \text{Pos} \{ \tilde{a}_1 \tilde{x}_1 + \tilde{a}_2 \tilde{x}_2 + \tilde{a}_3 \tilde{x}_3 + \tilde{a}_4 \tilde{x}_4 \leq 10 \} \geq 0.80 \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, \end{array} \right.$$

其中 $\tilde{a}_1, \tilde{a}_2, \tilde{a}_3$ 及 \tilde{a}_4 分别是三角模糊数 $(0, 1, 2), (1, 2, 3), (2, 3, 4)$ 和 $(3, 4, 5)$, 模糊数 \tilde{b} 和 \tilde{c} 的隶属函数分别为

$$\mu_{\tilde{b}}(x) = \frac{1}{1 + (x - 10)^2}, \quad \mu_{\tilde{c}}(x) = \frac{1}{1 + |x - 3|},$$

模糊数 $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4)$ 是从 Cartesian 积 $X_1 \otimes X_2 \otimes X_3 \otimes X_4$ 中生成的, 其中参考类 X_1 为清晰区间 $[1, 10]$ (一种退化情况), X_2 由所有三角模糊数 $(t-1, t, t+1)$ 构成, 这里 $t \in [1, 10]$, X_3 由所有隶属函数为 $\mu_3(x) = \exp(-|x - t|)$ 的模糊数构成, 这里 $t \in [1, 10]$, X_4 由所有隶属函数为 $\mu_4(x) = 1/[1 + |x - t|]$ 的模糊数构成, 这里 $t \in [1, 10]$.

为了应用模糊模拟技术处理模糊系统约束和模糊目标约束, 分别取包含模糊数 $\tilde{a}_1, \tilde{a}_2, \tilde{a}_3, \tilde{a}_4, \tilde{b}$ 和 \tilde{c} 的 α 水平截集的超几何体 $[0.5, 1.5], [1.5, 2.5], [2.5, 3.5], [3.5, 4.5], [9, 11]$ 及 $[2, 4]$, 经过 1000 代进化以后, 基于模糊模拟的遗传算法所给出的最优模糊解 $(\tilde{x}_1^*, \tilde{x}_2^*, \tilde{x}_3^*, \tilde{x}_4^*)$ 为

$$\left\{ \begin{array}{l} \tilde{x}_1^* = 2.951 \\ \tilde{x}_2^* = (0, 1, 2) \\ \mu_{\tilde{x}_3^*}(x) = \exp(-|x - 1.420|) \\ \mu_{\tilde{x}_4^*}(x) = \frac{1}{1 + |x - 1.001|} \end{array} \right.$$

它满足第一个目标, 但第二个和第三个目标分别为 1.37 和 2.82.

例 3 考虑一个简单的带有模糊决策的相关机会规划,

$$\left\{ \begin{array}{l} \max f(\tilde{x}) = \text{Pos} \{ \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 + \tilde{x}_4 = 8 \} \\ \text{s.t.} \\ \tilde{x}_1^2 + \tilde{x}_2^2 \leq \tilde{a} \\ \tilde{x}_3^3 + \tilde{x}_4^3 \leq \tilde{b}, \end{array} \right.$$

其中 \tilde{a} 和 \tilde{b} 分别是三角模糊数 (6,7,8) 和梯形模糊数 (7,8,9,10), 模糊决策 $(\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4)$ 是从 Cartesian 积 $X_1 \otimes X_2 \otimes X_3 \otimes X_4$ 中生成的, 其中参考类 X_1 由所有三角模糊数 $(t-1, t, t+1)$ 构成, 这里 $t \in [1, 10]$, X_2 由所有隶属函数为 $\mu_2(\xi) = 1/[1+(\xi-t)^2]$ 的模糊数构成, 这里 $t \in [2, 4]$, X_3 由所有隶属函数为 $\mu_3(\xi) = \exp[-|\xi-t|]$ 的模糊数构成, 这里 $t \in [1, 5]$, X_4 为闭区间 $[1, 10]$ (一种退化的情况).

对这种情况我们并不需要讨论决策分量之间的模糊关系. 既然决策系统中仅有一个事件, 并且对此事件所有分量都是必要的, 于是诱导约束是不确定环境本身. 这样, 机会函数 $f(\tilde{x})$ 为

$$\begin{aligned} f(\tilde{x}) &= \text{Pos} \left\{ \begin{array}{l} \tilde{x}_1^2 + \tilde{x}_2^2 \leq \tilde{a} \\ \tilde{x}_3^3 + \tilde{x}_4^3 \leq \tilde{b} \end{array} \right\} \wedge \text{Pos} \{ \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 + \tilde{x}_4 = 8 \} \\ &= \text{Pos} \left\{ \begin{array}{l} \tilde{x}_1^2 + \tilde{x}_2^2 \leq \tilde{a}, \tilde{x}_3^3 + \tilde{x}_4^3 \leq \tilde{b} \\ \tilde{x}_1 + \tilde{x}_2 + \tilde{x}_3 + \tilde{x}_4 = 8 \end{array} \right\}. \end{aligned}$$

经过 200 代进化以后, 基于模糊模拟的遗传算法所给出的最优模糊解 $\tilde{x}^* = (\tilde{x}_1^*, \tilde{x}_2^*, \tilde{x}_3^*, \tilde{x}_4^*)$ 为

$$\tilde{x}_1^* = (1.0265, 2.0265, 3.0265), \quad \mu_{\tilde{x}_2^*}(\xi) = \frac{1}{1 + (\xi - 2.1438)^2},$$

$$\mu_{\tilde{x}_3^*}(\xi) = \exp[-|\xi - 1.6784|], \quad \tilde{x}_4^* = 1.6367,$$

其可能性水平为 90.3%.

例 4 考虑下面模糊环境下的带有模糊决策的相关机会目标规划,

$$\left\{ \begin{array}{l} \text{leximin} \{d_1^-, d_2^-, d_3^-\} \\ \text{s.t.} \\ f_1(\tilde{\mathbf{x}}) + d_1^- - d_1^+ = 0.95 \\ f_2(\tilde{\mathbf{x}}) + d_2^- - d_2^+ = 0.92 \\ f_3(\tilde{\mathbf{x}}) + d_3^- - d_3^+ = 0.90 \\ \tilde{x}_1 + \tilde{x}_2 \leq \tilde{a} \\ \tilde{x}_3 + \tilde{x}_4 + \tilde{x}_5 \leq \tilde{b} \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3, \end{array} \right.$$

其中机会函数为

$$\left\{ \begin{array}{l} f_1(\tilde{\mathbf{x}}) = \text{Pos}\{\tilde{x}_1^2 + \tilde{x}_3^2 = 10\} \\ f_2(\tilde{\mathbf{x}}) = \text{Pos}\{\tilde{x}_2 = 7\} \\ f_3(\tilde{\mathbf{x}}) = \text{Pos}\{\tilde{x}_4^2 + \tilde{x}_5^2 = 25\}. \end{array} \right.$$

\tilde{a} 是梯形模糊数 $(7, 8, 9, 10)$, \tilde{b} 是三角模糊数 $(4, 5, 6)$, 模糊决策 $\tilde{\mathbf{x}} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5)$ 的每一个分量 \tilde{x}_i 从所有隶属函数为 $\mu(\xi) = 1/[1 + (\xi - t)^2]$ 的模糊数构成的类中生成, 这里 $t \in [1, 8]$.

由模糊环境, 我们知道下面的模糊关系: 5 个决策分量可以分成两组 $\{\tilde{x}_1, \tilde{x}_2\}$ 及 $\{\tilde{x}_3, \tilde{x}_4, \tilde{x}_5\}$, 并且这两组相互模糊独立, 而每组内任何元素都是模糊相关的, 并有相同的实现机会.

在第一优先级中, 模糊环境中仅有一个事件 $\tilde{x}_1^2 + \tilde{x}_3^2 = 10$, 表示为 E_1 . 显然, $V(E_1) = \{\tilde{x}_1, \tilde{x}_3\}$, 并且事件 E_1 的诱导约束为

$$\begin{cases} \tilde{x}_1 + \tilde{1} \leq \tilde{a} \\ \tilde{x}_3 + \tilde{1} + \tilde{1} \leq \tilde{b}. \end{cases}$$

在当前的优先级中, 对事件 E_1 , 其余的分量可以暂时认为是 $\hat{1}$, 其中 $\hat{1}$ 是隶属函数为 $\mu(\xi) = 1/[1 + (\xi - 1)^2]$ 的最小模糊数. 这样事件 E_1 的机会函数为

$$f_1(\tilde{x}) = \text{Pos} \{ \tilde{x}_1 + \hat{1} \leq \tilde{a}, \tilde{x}_3 + \hat{1} + \hat{1} \leq \tilde{b} \} \wedge \text{Pos} \{ \tilde{x}_1^2 + \tilde{x}_3^2 = 10 \}.$$

在第二优先级中, 仅有一个事件 $\tilde{x}_2 = 7$, 记为 E_2 . 我们有 $V(E_2) = \{\tilde{x}_2\}$. 记 $E = E_1 \cap E_2$, 则 $V(E) = V(E_1) \cup V(E_2) = \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3\}$. 既然

$$D(E_2) \cap V(E) = \{\tilde{x}_1, \tilde{x}_2\} \cap \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3\} = \{\tilde{x}_1, \tilde{x}_2\},$$

于是事件 E_2 的诱导约束为 $\tilde{x}_1 + \tilde{x}_2 \leq \tilde{a}$, 从而事件 E_2 的机会函数 $f_2(\tilde{x})$ 如下,

$$f_2(\tilde{x}) = \text{Pos} \{ \tilde{x}_1 + \tilde{x}_2 \leq \tilde{a} \} \wedge \text{Pos} \{ \tilde{x}_2 = 7 \}.$$

在第三个优先级中, 也仅有一个事件 $\tilde{x}_4^2 + \tilde{x}_5^2 = 25$, 记为 E_3 . 从而有 $V(E_3) = \{\tilde{x}_4, \tilde{x}_5\}$ 和 $V(E) = V(E_1) \cup V(E_2) \cup V(E_3) = \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5\}$. 由

$$D(E_3) \cap V(E) = \{\tilde{x}_3, \tilde{x}_4, \tilde{x}_5\} \cap \{\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4, \tilde{x}_5\} = \{\tilde{x}_3, \tilde{x}_4, \tilde{x}_5\}$$

可以导出事件 E_3 的诱导约束为 $\tilde{x}_3 + \tilde{x}_4 + \tilde{x}_5 \leq \tilde{b}$, 从而事件 E_3 的机会函数 $f_3(\tilde{x})$ 如下,

$$f_3(\tilde{x}) = \text{Pos} \{ \tilde{x}_3 + \tilde{x}_4 + \tilde{x}_5 \leq \tilde{b} \} \wedge \text{Pos} \{ \tilde{x}_4^2 + \tilde{x}_5^2 = 25 \}.$$

经过 1000 代进化以后, 基于模糊模拟的遗传算法给出的最优模糊解为 $\bar{x}^* = (\bar{x}_1^*, \bar{x}_2^*, \bar{x}_3^*, \bar{x}_4^*, \bar{x}_5^*)$, 其中 \bar{x}_i^* 分别是隶属函数 $\mu(\xi) = 1/[1 + (\xi - t_i^*)^2]$ 的模糊数, 这里 t_i^* 分别为

$$2.8107, \quad 6.7051, \quad 1.0038, \quad 1.0022, \quad 4.3799,$$

$i = 1, 2, 3, 4, 5$. 此模糊解满足前两个目标, 但第三个目标为 0.063. 同时, 有

$$f_1(\bar{x}^*) = 0.981, \quad f_2(\bar{x}^*) = 0.920, \quad f_3(\bar{x}^*) = 0.837.$$

11.7 注

本章介绍了带有模糊决策的机会约束规划和相关机会规划理论, 同时设计了求解这些模型的基于模糊模拟的遗传算法. 最后, 通过一些数值例子解释了遗传算法的有数性. 虽然基于模糊模拟的遗传算法有速度慢、费用高的缺点, 但对求解那些复杂而又无法转化为清晰等价类的问题, 的确是一种有数而且容易值用的方法.

第 12 章

Minimax 机会约束规划模型

在第 9 和第 11 章, 讨论了模糊决策系统的机会约束规划模型. 实际上, 这两章所讨论的本质上是 Maximax 模型 (乐观模型), 即极大化可能达到的最大收益.

本章将介绍由 Liu^[90] 建立的 Minimax 模型, 该模型的基本思想是基于前面所介绍的机会约束规划以及机会约束多目标规划和机会约束目标规划, 而目标是极大化可能达到的最小收益. 最后设计一个求解 Minimax 模型的基于模糊模拟的遗传算法, 并给出一些数值例子说明算法的有效性.

12.1 Maximax 模型

Liu 和 Iwamura^[88, 93] 建立了带有模糊系数和模糊决策的非线性机会约束规划以及机会约束多目标规划和机会约束目标规划. 事实上, 它们本质上是 Maximax 模型 (乐观模型). 为了介绍 Minimax 机会约束规划模型, 再次讨论 Maximax 模型.

文献 [93] 给出的单目标 Maximax 机会约束规划模型的形式如下:

$$\left\{ \begin{array}{l} \max \bar{f} \\ \text{s.t.} \\ \text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f} \} \geq \beta \\ \text{Pos} \{ g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p \} \geq \alpha, \end{array} \right. \quad (12.1)$$

其中模糊决策 \tilde{x} 是一个 n 维向量 $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$, 其分量 \tilde{x}_i 分别是模糊集合构成的参考类 X_i 中元素, $i = 1, 2, \dots, n$, 这样每一个模糊决策取值于参考类的 Cartesian 积

$$\mathcal{X} = X_1 \otimes X_2 \otimes \dots \otimes X_n, \quad (12.2)$$

而 ξ 是一个模糊参数向量, $f(\tilde{x}, \xi)$ 为目标函数, $g_j(\tilde{x}, \xi)$ 为约束函数, $j = 1, 2, \dots, p$, α 和 β 是事先给定的模糊约束和模糊目标的置信水平, $\text{Pos}\{\cdot\}$ 表示 $\{\cdot\}$ 中事件的可能性.

对每一个固定的模糊解 \tilde{x} , 既然我们的目的是极大化 \bar{f} , 目标值 \bar{f} 应当是目标函数 $f(\tilde{x}, \xi)$ 在可能性至少为 β 时所取的最大值, 即

$$\bar{f} = \max\{f \mid \text{Pos}\{f(\tilde{x}, \xi) \geq f\} \geq \beta\}. \quad (12.3)$$

因此, 上面提到的机会约束规划模型等价于下面的模型

$$\begin{cases} \max_{\tilde{x}} \max_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos}\{f(\tilde{x}, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos}\{g_j(\tilde{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (12.4)$$

显然, 这是一个 Maximax 模型.

类似地, 极小化目标函数的机会约束规划模型

$$\begin{cases} \min \bar{f} \\ \text{s.t.} \\ \text{Pos}\{f(\tilde{x}, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos}\{g_j(\tilde{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (12.5)$$

等价于下面的 Minimin 形式,

$$\begin{cases} \min_{\tilde{x}} \min_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\tilde{x}, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{cases} \quad (12.6)$$

其中 $\min \bar{f}$ 是目标函数 $f(\tilde{x}, \xi)$ 在可能性至少为 β 时所取的最小值.

下面给出的机会约束多目标规划模型,

$$\begin{cases} \max [\bar{f}_1, \bar{f}_2, \dots, \bar{f}_m] \\ \text{s.t.} \\ \text{Pos} \{f_i(\tilde{x}, \xi) \geq \bar{f}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p, \end{cases} \quad (12.7)$$

其中 α_j 和 β_i 分别为约束和目标的置信水平, \bar{f}_i 是目标函数 $f_i(\tilde{x}, \xi)$ 分别在可能性至少为 β_i 时所取的最大值, $i = 1, 2, \dots, m$, $j = 1, 2, \dots, p$. 类似于单目标情况, 此模型等价于 Maximax 模型,

$$\begin{cases} \max_{\tilde{x}} \left[\max_{\bar{f}_1} \bar{f}_1, \max_{\bar{f}_2} \bar{f}_2, \dots, \max_{\bar{f}_m} \bar{f}_m \right] \\ \text{s.t.} \\ \text{Pos} \{f_i(\tilde{x}, \xi) \geq \bar{f}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(\tilde{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p. \end{cases} \quad (12.8)$$

根据决策者给定的优先结构和目标值, 机会约束目标规划的模型为

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij}d_i^+ + v_{ij}d_i^-) \\ \text{s.t.} \\ \text{Pos} \{f_i(\bar{x}, \xi) + d_i^- - d_i^+ = \tilde{b}_i\} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{g_j(\bar{x}, \xi) \leq 0\} \geq \alpha_j, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m, \end{array} \right. \quad (12.9)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

\tilde{b}_i = 目标 i 的模糊目标值,

l = 优先级个数,

m = 目标约束个数.

在这个机会约束目标规划模型中, 正偏差和负偏差理解为分别使得

$$\text{Pos} \{d_i^- \geq \tilde{b}_i - f_i(\bar{x}, \xi)\} \geq \beta_i, \quad \text{Pos} \{d_i^+ \geq f_i(\bar{x}, \xi) - \tilde{b}_i\} \geq \beta_i$$

成立的最小非负值 d_i^+ 和 d_i^- , $i = 1, 2, \dots, m$. 这样, 机会约束目标规划本质上是 Minimin 模型.

12.2 Minimax 模型

本节将介绍 Minimax 机会约束规划模型^[90]，其基本思想是优化可能达到的最坏的目标值。

由于存在多个值 \bar{f} 使得 $\text{Pos}\{f(\bar{x}, \xi) \leq \bar{f}\} \geq \beta$ ，这里取 \bar{f} 的极小值作为目标函数并极大化。这样，单目标 Minimax 机会约束规划模型可以写成如下形式，

$$\left\{ \begin{array}{l} \max_{\bar{x}} \min_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos}\{f(\bar{x}, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos}\{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{array} \right. \quad (12.10)$$

其中 \bar{f} 是目标函数 $f(\bar{x}, \xi)$ 在可能性至少为 β 时所取的最小值。

如果我们的问题是极小化目标函数，则有下列的 Minimax 模型，

$$\left\{ \begin{array}{l} \min_{\bar{x}} \max_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos}\{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos}\{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{array} \right. \quad (12.11)$$

其中 $\max \bar{f}$ 是目标函数 $f(\bar{x}, \xi)$ 在可能性至少为 β 时所取的最大值。

作为单目标情况的推广，Minimax 机会约束多目标规划模型

可以表示成如下形式,

$$\left\{ \begin{array}{l} \max_{\bar{x}} \left[\min_{\bar{f}_1}, \min_{\bar{f}_2}, \dots, \min_{\bar{f}_m} \right] \\ \text{s.t.} \\ \text{Pos} \{ f_i(\bar{x}, \xi) \leq \bar{f}_i \} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{ g_j(\bar{x}, \xi) \leq 0 \} \geq \alpha_j, \quad j = 1, 2, \dots, p \end{array} \right. \quad (12.12)$$

其中 α_j 和 β_i 分别是约束和目标的置信水平, \bar{f}_i 分别是目标函数 $f_i(\bar{x}, \xi)$ 在可能性至少为 β_i 时所取的最大值, $i = 1, 2, \dots, m$, $j = 1, 2, \dots, p$.

根据决策者给定的优先结构和目标值, 机会约束目标规划模型可以表示成下面的形式,

$$\left\{ \begin{array}{l} \min \sum_{j=1}^l P_j \sum_{i=1}^m (u_{ij} d_i^+ + v_{ij} d_i^-) \\ \text{s.t.} \\ \text{Pos} \{ f_i(\bar{x}, \xi) + d_i^- - d_i^+ = \tilde{b}_i \} \geq \beta_i, \quad i = 1, 2, \dots, m \\ \text{Pos} \{ g_j(\bar{x}, \xi) \leq 0 \} \geq \alpha_j, \quad j = 1, 2, \dots, p \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, \dots, m \end{array} \right. \quad (12.13)$$

其中

P_j = 优先因子, 表示各个目标的相对重要性, 且对所有的 j , 有 $P_j \gg P_{j+1}$,

u_{ij} = 对应优先因子 j 的第 i 个目标正偏差的权重因子,

v_{ij} = 对应优先因子 j 的第 i 个目标负偏差的权重因子,

d_i^+ = 目标 i 偏离目标值的正偏差,

d_i^- = 目标 i 偏离目标值的负偏差,

\tilde{b}_i = 目标 i 的模糊目标值,

$l =$ 优先级个数,

$m =$ 目标约束个数.

如果正偏差和负偏差理解为分别使得

$$\text{Pos} \left\{ d_i^- \leq \tilde{b}_i - f_i(\bar{x}, \xi) \right\} \geq \beta_i, \quad \text{Pos} \left\{ d_i^+ \leq f_i(\bar{x}, \xi) - \tilde{b}_i \right\} \geq \beta_i$$

成立得最大非负值 d_i^+ 和 d_i^- , $i = 1, 2, \dots, m$, 则机会约束目标规划模型本质上是一个 Minimax 模型. 这里, 最大非负值的意思是若无正的 d_i^+ 和 d_i^- 满足以上不等式, 则置 $d_i^+ = 0$ 和 $d_i^- = 0$.

这里, 值得一提的是, Minimin 和 Minimax 机会约束目标规划模型并没有什么本质的区别, 二者的差异只在于偏差理解为极大值还是极小值而已.

12.3 Minimax 与 Maximax

对于 Minimax 和 Maximax 机会约束规划模型来说, 有下面两个要考虑的子优化问题,

$$\begin{aligned} \bar{f}_{\min} &= \min \{ \bar{f} \mid \text{Pos} \{ f(\bar{x}, \xi) \leq \bar{f} \} \geq \beta \}, \\ \bar{f}_{\max} &= \max \{ \bar{f} \mid \text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f} \} \geq \beta \}, \end{aligned} \quad (12.14)$$

无论什么样的模糊决策 \bar{x} 、模糊参向量 ξ 和函数形式 $f(\cdot, \cdot)$, $f(\bar{x}, \xi)$ 都可以认为是一个模糊数, 其隶属函数表示为 $\mu_{f(\bar{x}, \xi)}(\cdot)$, 理论上它可以由 \bar{x} 和 ξ 的隶属函数导出.

由 Dubois 和 Prade^[34, 35] 给出的可能性定义, 有

$$\text{Pos} \{ f(\bar{x}, \xi) \leq \bar{f} \} = \sup_f \left\{ \mu_{f(\bar{x}, \xi)}(f) \mid f \leq \bar{f} \right\}$$

和

$$\text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f} \} = \sup_f \left\{ \mu_{f(\bar{x}, \xi)}(f) \mid f \geq \bar{f} \right\}.$$

既然 $\bar{f}_{\min} = \min \{ \bar{f} | \text{Pos} \{ f(\bar{x}, \xi) \leq \bar{f} \} \geq \beta \}$, 则最小值 \bar{f}_{\min} 应满足下面的不等式,

$$\text{Pos} \{ f(\bar{x}, \xi) \leq \bar{f}_{\min} \} \geq \beta.$$

更进一步, 有 $\mu_{f(\bar{x}, \xi)}(\bar{f}_{\min}) \geq \beta$, 否则, 至少存在一个 $f < \bar{f}_{\min}$ 使得 $\mu_{f(\bar{x}, \xi)}(f) \geq \beta$, 说明 $\text{Pos} \{ f(\bar{x}, \xi) \leq f \} \geq \beta$, 而这与 \bar{f}_{\min} 为满足

$$\text{Pos} \{ f(\bar{x}, \xi) \leq \bar{f} \} \geq \beta$$

的 \bar{f} 的最小值矛盾.

既然已经证明 $\mu_{f(\bar{x}, \xi)}(\bar{f}_{\min}) \geq \beta$ 成立, 并且

$$\bar{f}_{\max} = \max \{ \bar{f} | \text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f} \} \geq \beta \},$$

故有

$$\begin{aligned} \text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f}_{\min} \} &= \sup_f \{ \mu_{f(\bar{x}, \xi)}(f) | f \geq \bar{f}_{\min} \} \\ &\geq \mu_{f(\bar{x}, \xi)}(\bar{f}_{\min}) \geq \beta. \end{aligned}$$

并且

$$\bar{f}_{\max} \geq \bar{f}_{\min} \quad (12.15)$$

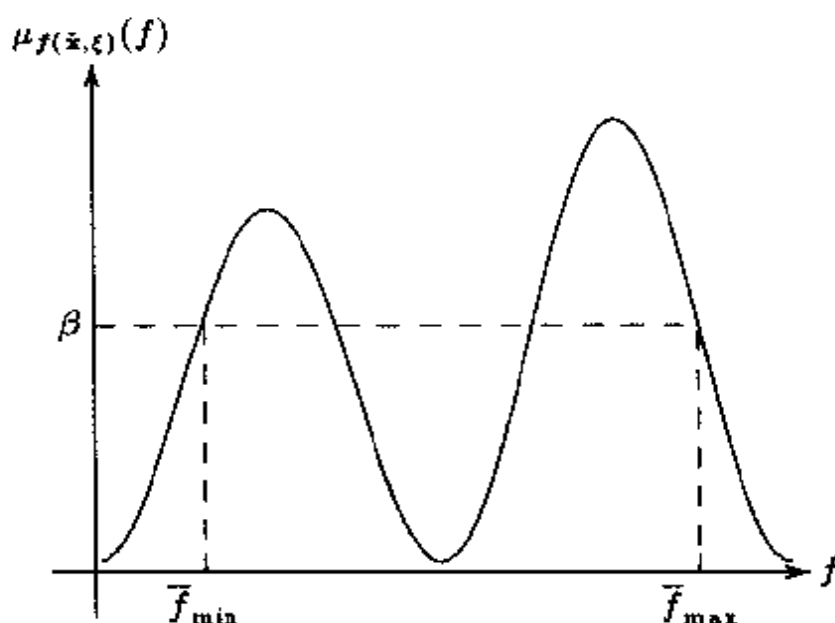
永远成立. 如图 12.1 所示.

更一般地, 既然对每一个固定的模糊决策 \bar{x} , 有 $\bar{f}_{\max} \geq \bar{f}_{\min}$, 因此 Maximax 解永远大于或等于 Minimax 解.

本节内容可总结成如下的定理.

定理12.1 对任意固定的模糊决策 \bar{x} , 有 $\bar{f}_{\max} \geq \bar{f}_{\min}$, 其中 \bar{f}_{\max} 和 \bar{f}_{\min} 由式 (12.14) 确定. 更进一步, 对 Maximax 模型 (12.4) 和 Minimax 模型 (12.10), 有

$$\max_{\bar{x}} \max_f \bar{f} \geq \max_{\bar{x}} \min_f \bar{f}. \quad (12.16)$$

图 12.1 极小值 \bar{f}_{\min} 和极大值 \bar{f}_{\max}

类似地，可以证明下面的定理，

定理12.2 对 Minimax 机会约束规划模型

$$\left\{ \begin{array}{l} \min_{\bar{x}} \max_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\bar{x}, \xi) \geq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha \end{array} \right. \quad (12.17)$$

和 Minimin 模型

$$\left\{ \begin{array}{l} \min_{\bar{x}} \min_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos} \{f(\bar{x}, \xi) \leq \bar{f}\} \geq \beta \\ \text{Pos} \{g_j(\bar{x}, \xi) \leq 0, j = 1, 2, \dots, p\} \geq \alpha, \end{array} \right. \quad (12.18)$$

总有

$$\min_{\tilde{x}} \max_{\tilde{f}} \tilde{f} \geq \min_{\tilde{x}} \min_{\tilde{f}} \tilde{f}. \quad (12.19)$$

12.4 模 糊 模 拟

前面已经介绍了检验模糊系统约束以及处理模糊目标函数和模糊目标约束的模糊模拟技术.

为了求解 Minimax 机会约束规划模型, 本节介绍另外两种类型的模拟技术, 用来寻找使

$$\text{Pos} \{f(\tilde{x}, \xi) \leq \tilde{f}\} \geq \beta$$

成立的 \tilde{f} 的最小值, 以及为模糊目标约束

$$\text{Pos} \{f(\tilde{x}, \xi) + d^- - d^+ = \tilde{b}\} \geq \beta$$

寻找最大的正偏差和负偏差, 其中 \tilde{x} , ξ 和 \tilde{b} 假设是模糊数, 隶属函数分别为 $\nu(\cdot)$, $\mu(\cdot)$ 和 $\mu_{\tilde{b}}(\cdot)$.

处理模糊目标函数 现在, 考虑对给定的模糊决策向量 \tilde{x} , 如何搜索使下面的不等式

$$\text{Pos} \{f(\tilde{x}, \xi) \leq \tilde{f}\} \geq \beta \quad (12.20)$$

成立的 \tilde{f} 的最小值. 首先, 置 $\tilde{f} = +\infty$. 然后, 分别由模糊向量 ξ 和 \tilde{x} 随机地生成两个清晰向量 ξ^0 和 \tilde{x}^0 , 使得 $\mu(\xi^0) \wedge \nu(\tilde{x}^0) \geq \beta$, 即分别在模糊向量 ξ 和 \tilde{x} 的 β 水平截集中随机抽取向量. 若模糊向量 ξ 的 β 水平截集过于复杂而难于确定, 可以从包含模糊向量 ξ 的 β 水平截集的超几何体 Ω 中随机抽取一个向量 ξ^0 , 接受与否依赖于 $\mu(\xi^0) \geq \beta$ 是否成立. 为了加快模拟速度, 超几何体 Ω 应设计得尽可能小. 对模糊决策 \tilde{x} , 抽样过程类似于 ξ . 只要

$\bar{f} > f(\bar{x}^0, \xi^0)$, 置 $\bar{f} = f(\bar{x}^0, \xi^0)$. 重复以上过程直到给定的次数 N 完成为止, 所得值 \bar{f} 作为目标函数 $f(\bar{x}, \xi)$ 在可能性至少为 β 时所取的最小值. 以上过程可以总结如下.

步骤 1 置 $\bar{f} = +\infty$;

步骤 2 分别由模糊向量 ξ 和 \bar{x} 的 β 水平截集随机地生成两个清晰向量 ξ^0 和 \bar{x}^0 ;

步骤 3 若 $\bar{f} > f(\bar{x}^0, \xi^0)$, 则置 $\bar{f} = f(\bar{x}^0, \xi^0)$;

步骤 4 重复步骤 2 和步骤 3 共 N 次;

步骤 5 返回 \bar{f} .

处理模糊目标约束 在 Minimax 机会约束目标规划模型中, 存在下面类型的模糊目标约束,

$$\text{Pos} \{f(\bar{x}, \xi) + d^- - d^+ = \bar{b}\} \geq \beta, \quad (12.21)$$

即对给定的模糊决策 \bar{x} , 我们的目的是寻找 d^- 和 d^+ 的最大非负值使得

$$\text{Pos}\{d^- \leq \bar{b} - f(\bar{x}, \xi)\} \geq \beta, \quad \text{Pos}\{d^+ \leq f(\bar{x}, \xi) - \bar{b}\} \geq \beta.$$

首先, 置 $d^- = d^+ = 0$. 然后从模糊向量 ξ , \bar{x} 和 \bar{b} 的 β 水平截集中分别随机生成清晰向量 ξ^0 , \bar{x}^0 和 \bar{b}^0 . 若 $d^- < \bar{b}^0 - f(\bar{x}^0, \xi^0)$, 则置 $d^- = \bar{b}^0 - f(\bar{x}^0, \xi^0)$. 类似地, 若 $d^+ < f(\bar{x}^0, \xi^0) - \bar{b}^0$, 则置 $d^+ = f(\bar{x}^0, \xi^0) - \bar{b}^0$. 通过生成新的清晰向量 ξ^0 , \bar{x}^0 和 \bar{b}^0 重复以上过程 N 次. 则 d^+ 和 d^- 的值分别作为在模糊决策 \bar{x} 处的正偏差和负偏差的估计值. 这个过程总结如下,

步骤 1 置 $d^+ = d^- = 0$;

步骤 2 从模糊向量 ξ , \bar{x} 和 \bar{b} 的 β 水平截集中分别随机地生成清晰向量 ξ^0 , \bar{x}^0 和 \bar{b}^0 ;

步骤 3 若 $d^+ < f(\bar{x}^0, \xi^0) - \bar{b}^0$, 则置 $d^+ = f(\bar{x}^0, \xi^0) - \bar{b}^0$;

步骤 4 若 $d^- < \bar{b}^0 - f(\bar{x}^0, \xi^0)$, 则置 $d^- = \bar{b}^0 - f(\bar{x}^0, \xi^0)$;

步骤 5 重复步骤 2 到步骤 4 共 N 次;

步骤 6 返回 d^+ 和 d^- .

12.5 数值例子

在第 11 章中设计的基于模糊模拟的遗传算法也适用于模糊决策系统中的 Minimax 机会约束规划模型. 对一个模糊决策, 解的代码必须表示成模糊染色体, 其中每一个基因是一个模糊集合而不是一个清晰数字. 这里, 使用模糊向量 $V = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)$ 作为一个染色体表示 Minimax 模型的一个模糊解, 其中 n 是维数, 所有分量 \bar{x}_i 分别取自于给定的模糊集构成的参考类 X_i , $i = 1, 2, \dots, n$. 基于模糊模拟的遗传算法的程序总结如下.

遗传算法程序

步骤 0 输入参数 pop_size 、 P_c 和 P_m ;

步骤 1 初始产生 pop_size 个染色体, 其中可能使用模糊模拟检验可行性;

步骤 2 对染色体进行交叉和变异操作, 其中可能使用模糊模拟检验后代的可行性;

步骤 3 使用模糊模拟计算所有染色体的目标值;

步骤 4 根据目标值, 使用基于序的评价函数计算每一个染色体的适应度;

步骤 5 旋转赌轮选择染色体;

步骤 6 重复步骤 2 到步骤 5 直到给定的次数完成为止;

步骤 7 把所得的最好染色体作为最优解.

这里, 给出两个在 PC 机上实现的数值例子, 所使用的参数为: 种群规模为 30, 交叉概率 $P_c = 0.3$, 变异概率 $P_m = 0.2$, 基于序的评价函数中的参数 $\alpha = 0.05$. 每次模糊模拟执行 2000 次.

例 1 假设模糊决策 $\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3)$ 从给定的 Cartesian 积

$X_1 \otimes X_2 \otimes X_3$ 中生成, 其中参考类 X_1 由所有三角模糊数 $(t-1, t, t+1)$ 构成, 这里 $t \in [1, 5]$, X_2 由所有隶属函数为 $\mu_2(x) = 1/[1 + (x-t)^2]$ 的模糊数构成, 这里 $t \in [3, 6]$, X_3 由所有隶属函数为 $\mu_3(x) = \exp(-|x-t|)$ 的模糊数构成, 这里 $t \in [2, 4]$.

同时假设模糊决策 $\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3)$ 的约束条件为

$$\text{Pos} \{ \bar{a}_1 \bar{x}_1 + \bar{a}_2 \bar{x}_2 + \bar{a}_3 \bar{x}_3 \leq 100 \} \geq 0.80,$$

其中系数 \bar{a}_1 , \bar{a}_2 和 \bar{a}_3 分别为三角模糊数 $(8, 9, 10)$, $(6, 7, 8)$ 和 $(16, 17, 18)$.

假设要极大化的目标函数为 $\bar{c}_1 \bar{x}_1 + \bar{c}_2 \bar{x}_2^2 + \bar{c}_3 \bar{x}_3^3$, 其中系数 \bar{c}_1 , \bar{c}_2 和 \bar{c}_3 是隶属函数分别为 $\mu_{\bar{c}_1}(x) = 1/[1 + |x-3|]$, $\mu_{\bar{c}_2}(x) = 1/[1 + |x-2|]$ 和 $\mu_{\bar{c}_3}(x) = 1/[1 + |x-1|]$ 的模糊数. 我们感兴趣的是在可能性至少为 0.90 时的目标值.

由于存在多个满足给定的可能性水平的目标值, 因此, 不但可以建立 Minimax 机会约束规划模型, 而且可以建立 Maximax 机会约束规划模型. 所得到的 Maximax 模型为

$$(\text{Maximax}) \begin{cases} \max_{\bar{x}} \max_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos} \{ \bar{c}_1 \bar{x}_1 + \bar{c}_2 \bar{x}_2^2 + \bar{c}_3 \bar{x}_3^3 \geq \bar{f} \} \geq 0.90 \\ \text{Pos} \{ \bar{a}_1 \bar{x}_1 + \bar{a}_2 \bar{x}_2 + \bar{a}_3 \bar{x}_3 \leq 100 \} \geq 0.80, \end{cases}$$

而 Minimax 模型为

$$(\text{Minimax}) \begin{cases} \max_{\bar{x}} \min_{\bar{f}} \bar{f} \\ \text{s.t.} \\ \text{Pos} \{ \bar{c}_1 \bar{x}_1 + \bar{c}_2 \bar{x}_2^2 + \bar{c}_3 \bar{x}_3^3 \leq \bar{f} \} \geq 0.90 \\ \text{Pos} \{ \bar{a}_1 \bar{x}_1 + \bar{a}_2 \bar{x}_2 + \bar{a}_3 \bar{x}_3 \leq 100 \} \geq 0.80. \end{cases}$$

对 Maximax 模型, 经过 1000 代进化以后, 基于模糊模拟的遗传算法所给出的最优解 $(\tilde{x}_1^*, \tilde{x}_2^*, \tilde{x}_3^*)$ 为

$$\begin{aligned}\tilde{x}_1^* &= (3.878, 4.878, 5.878), \\ \mu_{\tilde{x}_2^*}(x) &= \frac{1}{1 + (x - 4.757)^2}, \\ \mu_{\tilde{x}_3^*}(x) &= \exp(-|x - 2.002|),\end{aligned}$$

以可能性至少为 0.90 时的最大目标值为 $\bar{f}_{\max}^* = 28.57$. 对 Minimax 模型, 经过 1000 代进化以后, 基于模糊模拟的遗传算法所给出的最优解 $(\tilde{x}_1^*, \tilde{x}_2^*, \tilde{x}_3^*)$ 为

$$\begin{aligned}\tilde{x}_1^* &= (3.852, 4.852, 5.852), \\ \mu_{\tilde{x}_2^*}(x) &= \frac{1}{1 + (x - 4.615)^2}, \\ \mu_{\tilde{x}_3^*}(x) &= \exp(-|x - 2.003|),\end{aligned}$$

以可能性至少为 0.90 时的最小目标值为 $\bar{f}_{\min}^* = 23.87$. 在此例中, 有 $\bar{f}_{\max}^* > \bar{f}_{\min}^*$, 这与第 12.3 节中的结果吻合.

例 2 假设存在 4 个模糊决策变量 $\tilde{x}_1, \tilde{x}_2, \tilde{x}_3$ 及 \tilde{x}_4 , 分别取自于由模糊集构成的参考类 X_1, X_2, X_3 和 X_4 中, 其中 X_1 由所有隶属函数为 $\mu_1(x) = 1/[1 + |x - t|]$ 的模糊数构成, 这里 $t \in [3, 5]$, X_2 由所有隶属函数为 $\mu_2(x) = 1/[1 + (x - t)^2]$ 的模糊数构成, 这里 $t \in [2, 4]$, X_3 由所有隶属函数为 $\mu_3(x) = \exp[-|x - t|]$ 的模糊数构成, 这里 $t \in [4, 7]$, X_4 由所有隶属函数为 $\mu_4(x) = \exp[-(x - t)^2]$ 的模糊数构成, 这里 $t \in [5, 6]$. 模糊决策 $\hat{x} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3, \tilde{x}_4)$ 的系统约束为

$$\text{Pos}\{(\tilde{a}_1\tilde{x}_1 + \tilde{a}_2\tilde{x}_2)(\tilde{a}_3\tilde{x}_3 + \tilde{a}_4\tilde{x}_4) \leq 300\} \geq 0.80,$$

其中 $\tilde{a}_1, \tilde{a}_2, \tilde{a}_3$ 及 \tilde{a}_4 分别为三角模糊数 $(0,1,2)$, $(1,2,3)$, $(2,3,4)$ 和 $(3,4,5)$.

在第一优先级中, 存在一个模糊目标值 \tilde{b}_1 , 其隶属函数为 $\mu_{\tilde{b}_1}(x) = 1/[1 + |x - 30|]$. 如果希望乘积 $\tilde{x}_1\tilde{x}_3$ 至少以 0.95 的可能性达到 \tilde{b}_1 , 则有模糊目标约束

$$\text{Pos} \left\{ \tilde{x}_1\tilde{x}_3 + d_1^- - d_1^+ = \tilde{b}_1 \right\} \geq 0.95,$$

其中 d_1^- 将被极小化.

在第二优先级中, 有模糊目标约束

$$\text{Pos} \left\{ \tilde{x}_2 + d_2^- - d_2^+ = \tilde{b}_2 \right\} \geq 0.90,$$

其中 \tilde{b}_2 是隶属函数为 $\mu_{\tilde{b}_2}(x) = 1/[1 + |x - 3.5|]$ 的模糊数, 并且 d_2^- 将被极小化.

在第三优先级中, 有模糊目标约束

$$\text{Pos} \left\{ \tilde{x}_4 + d_3^- - d_3^+ = \tilde{b}_3 \right\} \geq 0.90,$$

其中 \tilde{b}_3 是隶属函数为 $\mu_{\tilde{b}_3}(x) = 1/[1 + |x - 7|]$ 的模糊数, 并且 d_3^- 将被极小化.

这样, 有下面的机会约束目标规划模型,

$$\left\{ \begin{array}{l} \text{lexmin} \{d_1^-, d_2^-, d_3^-\} \\ \text{s.t.} \\ \text{Pos} \left\{ \tilde{x}_1\tilde{x}_3 + d_1^- - d_1^+ = \tilde{b}_1 \right\} \geq 0.95 \\ \text{Pos} \left\{ \tilde{x}_2 + d_2^- - d_2^+ = \tilde{b}_2 \right\} \geq 0.90 \\ \text{Pos} \left\{ \tilde{x}_4 + d_3^- - d_3^+ = \tilde{b}_3 \right\} \geq 0.90 \\ \text{Pos} \{(\tilde{a}_1\tilde{x}_1 + \tilde{a}_2\tilde{x}_2)(\tilde{a}_3\tilde{x}_3 + \tilde{a}_4\tilde{x}_4) \leq 300\} \geq 0.80 \\ d_i^-, d_i^+ \geq 0, \quad i = 1, 2, 3. \end{array} \right.$$

当正偏差和负偏差分别理解为 d_i^+ 和 d_i^- 所取的最小值时, 机会约束规划是一个 Minimin 模型; 当正偏差和负偏差分别理解为 d_i^+ 和 d_i^- 所取的最大值时, 机会约束规划是一个 Minimax 模型.

对 Minimin 模型, 经过 1000 代进化以后, 基于随机模拟的遗传算法给出的最优模糊解 $(\hat{x}_1^*, \hat{x}_2^*, \hat{x}_3^*, \hat{x}_4^*)$ 为

$$\mu_{\hat{x}_1^*}(x) = \frac{1}{1 + |x - 4.8974|}, \quad \mu_{\hat{x}_2^*}(x) = \frac{1}{1 + (x - 3.0664)^2},$$

$$\mu_{\hat{x}_3^*}(x) = \exp[-|x - 6.0061|], \quad \mu_{\hat{x}_4^*}(x) = \exp[-(x - 5.6800)^2],$$

它满足前两个目标, 但第三个目标是 0.8843. 对 Minimax 模型, 经过 1000 代进化以后, 基于模糊模拟的遗传算法给出的最优模糊解 $(\hat{x}_1^*, \hat{x}_2^*, \hat{x}_3^*, \hat{x}_4^*)$ 为

$$\mu_{\hat{x}_1^*}(x) = \frac{1}{1 + |x - 4.8883|}, \quad \mu_{\hat{x}_2^*}(x) = \frac{1}{1 + (x - 3.3181)^2},$$

$$\mu_{\hat{x}_3^*}(x) = \exp[-|x - 6.2672|], \quad \mu_{\hat{x}_4^*}(x) = \exp[-(x - 5.0072)^2],$$

它满足第一个目标, 但第二个和第三个分别为 0.6263 和 2.4285.

这些数值例子的计算时间包括模糊模拟所花费的时间和遗传算法所花费的时间, 其中模糊模拟的计算时间与抽样次数成比例. 应注意的是模糊模拟的计算时间是必须的, 因为已经假定没有可取代它的直接方法. 如果有直接方法去估计一些特殊约束的可能性的话, 则应该使用这些直接方法去代替模糊模拟以加快进化速度.

12.6 注

除了再次讨论 Maximax 机会约束规划模型, 本章还介绍了一系列 Minimax 模型. 在 Maximax 模型中, 极大化的函数是

$$\bar{f}_{\max} = \max \{ \bar{f} \mid \text{Pos} \{ f(\bar{x}, \xi) \geq \bar{f} \} \geq \beta \},$$

而在 Minimax 模型中, 极大化的函数是

$$\bar{f}_{\min} = \min \{ \bar{f} \mid \text{Pos} \{ f(\tilde{x}, \xi) \leq \bar{f} \} \geq \beta \}.$$

显然, Minimax 和 Maximax 模型是两个极端的情况. 实际上, 也可以把机会约束规划模型定义为

$$\begin{cases} \max \lambda \bar{f}_{\min} + (1 - \lambda) \bar{f}_{\max} \\ \text{s.t.} \\ \text{Pos} \{ g_j(\tilde{x}, \xi) \leq 0, j = 1, 2, \dots, p \} \geq \alpha, \end{cases} \quad (12.22)$$

其中 λ 是 0 和 1 之间一个给定的数, 当 $\lambda = 0$ 时, 它是一个 Maximax 模型; 当 $\alpha = 1$ 时, 它是一个 Minimax 模型.

参考文献

- [1] Angelov P. A generalized approach to fuzzy optimization. *International Journal of Intelligent Systems*, 1994, 9: 261-268.
- [2] Bäck T. *Evolutionary Algorithms in Theory and Practice*. New York: Oxford University Press, 1995
- [3] Balas E. An additive algorithm for solving linear programs with zero-one variables. *Operations Research*, 1965, 13: 517-546
- [4] Bard JF. An algorithm for solving the general bilevel programming problem. *Mathematics of Operations Research*, 1983, 8: 260-272
- [5] Bard JF, Moore JT. A branch and bound algorithm for the bilevel programming problem. *SIAM J. Sci. Statist. Comput.*, 1990, 11: 281-292
- [6] Bard JF. Optimality conditions for the bilevel programming problem. *Naval Research Logistics Quarterly*, 1984, 31: 13-26
- [7] Bazaraa MS, Jarvis JJ. *Linear Programming and Network Flows*. Wiley, 1977
- [8] Bellman R. *Dynamic Programming*. New Jersey: Princeton University Press, 1957
- [9] Bellman R, Zadeh LA. Decision making in a fuzzy environment. *Management Science*, 1970, 17: 141-164
- [10] Ben-Ayed O, Blair CE. Computational difficulties of bilevel linear programming. *Operations Research*, 1990, 38: 556-560
- [11] Ben-Ayed O. Bilevel linear programming. *Computers and Operations Research*, 1993, 20(5): 485-501
- [12] Bertsekas DP, Tsitsiklis JN. *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1989

- [13] Bialas WF, Karwan MH. Two-level linear programming. *Management Science*, 1984, 30: 1004-1020
- [14] Bit AK, Biswal MP, Alam SS. Fuzzy programming approach to multicriteria decision making transportation problem. *Fuzzy Sets and Systems*, 1992, 50: 135-141
- [15] Bouchon-Meunier B, Kreinovich V, Lokshin A, Nguyen HT. On the formulation of optimization under elastic constraints (with control in mind). *Fuzzy Sets and Systems*, 1996, 81: 5-29
- [16] Buckley JJ, Hayashi Y. Fuzzy genetic algorithm and applications. *Fuzzy Sets and Systems*, 1994, 61: 129-136
- [17] Candler W, Townsley R. A linear two-level programming problem. *Computers and Operations Research*, 1982, 9: 59-76
- [18] Changchit C, Terrell MP. CCGP model for multiobjective reservoir systems. *Journal of Water Resources Planning and Management*, 1989, 115(5): 658-670
- [19] Chankong V, Haims YY. *Multiobjective Decision Making: Theory and Methodology*. Amsterdam: North-Holland, 1983
- [20] Charnes A, Cooper WW. Chance-constrained programming. *Management Science*, 1959, 6(1): 73-79
- [21] Charnes A, Cooper WW. *Management Models and Industrial Applications of Linear Programming*. New York: John Wiley & Sons, Inc., 1961
- [22] Charnes A, Cooper WW. Chance constraints and normal deviates. *Journal of the American Statistical Association*, 1962, 57: 134-148
- [23] Charnes A, Cooper WW. Deterministic equivalents for optimizing and satisficing under chance-constraints. *Operations Research*, 1963, 11(1): 18-39
- [24] Charnes A, Cooper WW. Goal programming and multiple objective optimizations: Part I. *European Journal of Operational Research*, 1977, 1(1): 39-54
- [25] Chanas S. Fuzzy programming in multiobjective linear programming - a parametric approach. *Fuzzy Sets and Systems*, 1989, 29: 303-313

- [26] Clayton E, Weber W, Taylor III B. A goal programming approach to the optimization of multiresponse simulation models. *IEE Trans.*, 1982, 14: 282-287
- [27] Dakin R.J. A tree search algorithm for mixed integer programming problems. *Computer Journal*, 1965, 8: 250-255
- [28] Dantzig GB. *Linear Programming and Extensions*. Princeton University Press, 1963
- [29] De PK, Acharya D, Sahu KC. A chance-constrained goal programming model for capital budgeting. *Journal of the Operational Research Society*, 1982, 33: 635-638
- [30] Delgado M, Verdegay JL, Vila MA. A general model for fuzzy linear programming. *Fuzzy Sets and Systems*, 1989, 29: 21-29
- [31] Dempster MAH(ed). *Stochastic Programming*. London: Academic Press, 1980
- [32] Dharmadhikari S, Joag-dev K. *Unimodality, Convexity, and Applications*. Academic Press, Inc., 1988
- [33] Dubois D, Prade H. *Fuzzy Sets and Systems, Theory and Applications*. New York: Academic Press, 1980
- [34] Dubois D, Prade H. *Possibility Theory*. New York: Plenum, 1988
- [35] Dubois D, Prade H. Fuzzy numbers: An overview. In: Bezdek JC, Ed. *Analysis of Fuzzy Information*, Vol.2, CRC Press, Boca Raton, 1988, 3-39
- [36] Dyer J. Interactive goal programming. *Management Science*, 1972, 19: 62-70
- [37] Elmaghraby S. *Activity Networks: Project Planning and Control by Network Models*. New York: John Wiley & Sons, 1977
- [38] Ermoliev Y, Wets RJ-B(eds). *Numerical Techniques for Stochastic Optimization*. Berlin: Springer-Verlag, 1988
- [39] Esogbue AO, Bellman RE. Fuzzy dynamic programming and its extensions. *TIMS/Studies in the Management Sciences*, 1984, 20: 147-167
- [40] Filer D. Fuzzy modeling of complex systems. *International Journal of Approximate Reasoning*, 1991, 5: 281-290

- [41] Flachs J, Pollatschek MA. Further results on fuzzy mathematical programming. *Information and Control*, 1978, 38: 241-257
- [42] Fogel DB. An introduction to simulated evolutionary optimization. *IEEE Transactions on Neural Networks*, 1994, 5: 3-14
- [43] Fogel DB. *Evolution Computation: Toward a New Philosophy of Machine Intelligence*. Piscataway, NJ: IEEE Press, 1995
- [44] Foulds LR. *Optimization Techniques: An Introduction*. New York: Springer-Verlag, 1981
- [45] Garfinkel RS, Nemhauser GL. *Integer Programming*. Wiley, 1972
- [46] Gen M, Liu B. Evolution program for production plan problem. *Engineering Design and Automation*, 1995, 1(3): 199-204
- [47] Gen M, Liu B, Ida K. Evolution program for deterministic and stochastic optimizations. *European Journal of Operational Research*, 1996, 94(3): 618-625
- [48] Gen M, Liu B. Evolution program for optimal capacity expansion. *Journal of Operations Research Society of Japan*, 1997, 40(1): 1-9
- [49] Gen M, Liu B. A genetic algorithm for nonlinear goal programming. to appear in *Evolutionary Optimization*, 1998, 1(1)
- [50] Goldberg DE. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989
- [51] Gomory RE. Outline of an algorithm for integer solutions to linear programs. *Bull. Amer. Math. Soc.*, 1958, 64: 275-278
- [52] Greenberg N. *Integer Programming*. Academic Press, 1971
- [53] Hammersley IM, Handscomb DC. *Monte Carlo Methods*. New York: Wiley, 1964
- [54] Hillier FS. Chance-constrained programming with 0-1 or bounded continuous decision variables. *Management Science*, 1967, 14(1): 34-56
- [55] Holland JH. *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975
- [56] Hwang C, Masud A. *Multiple Objective Decision Making Methods and Applications*. New York: Springer-Verlag, 1979

- [57] Hwang CL, Paidy SR, Yoon K, Masud ASM. Mathematical programming with multiple objectives: A tutorial. *Computers and Operations Research*, 1980, 7: 5-31
- [58] Hu TC. *Integer Programming and Network Flows*. Addison-Wesley, 1969
- [59] Ignizio J. *Goal Programming and Extensions*. Heath, Lexington, Mass, 1976
- [60] Iwamura K, Liu B. A genetic algorithm for chance constrained programming. *Journal of Information & Optimization Sciences*, 1996, 17(2): 40-47
- [61] Iwamura K, Liu B. Chance constrained integer programming models for capital budgeting in fuzzy environments. to appear in *Journal of the Operational Research Society*
- [62] Iwamura K, Liu B. Stochastic operation models for open inventory networks. Technical Report, 1996
- [63] Iwamura K, Liu B. Dependent-chance integer programming applied to capital budgeting. Technical Report, 1996
- [64] Kall P. *Stochastic Linear Programming*. Berlin: Springer-Verlag, 1976
- [65] Kall P, Wallace SW. *Stochastic Programming*. John Wiley & Sons, 1994
- [66] Kaufman A. *Introduction to the Theory of Fuzzy Subsets*. New York: Academic Press, 1975
- [67] Kaufman A, Gupta MM. *Introduction to Fuzzy Arithmetic: Theory and Applications*. New York: Van Nostrand Reinhold, 1985
- [68] Kaufman A, Gupta MM. *Fuzzy Mathematical Models in Engineering and Management Science*, 2nd ed. North-Holland, 1991
- [69] Keeney RL, Raiffa H. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley & Sons, Inc., 1976
- [70] Keown AJ, Martin JD. A chance constrained goal programming model for working capital management. *Engng Econ.*, 1977, 22: 153-174
- [71] Keown AJ. A chance-constrained goal programming model for bank liquidity management. *Decision Sciences*, 1978, 9: 93-106

- [72] Keown AJ, Taylor BW. A chance-constrained integer goal programming model for capital budgeting in the production area. *Journal of the Operational Research Society*, 1980, 31(7): 579-589
- [73] Klein G, Moskowitz H, Ravindran A. Interactive multiobjective optimization under uncertainty. *Management Science*, 1990, 36(1): 58-75
- [74] Klir GJ, Folger TA. *Fuzzy Sets, Uncertainty, and Information*. Englewood Cliffs, NJ: Prentice-Hall, 1980
- [75] Kolbin VV. *Stochastic Programming*. D.Reidel, Dordrecht, 1977
- [76] Koza JR. *Genetic Programming*. Cambridge, MA: MIT Press, 1992
- [77] Koza JR. *Genetic Programming, II*. Cambridge, MA: MIT Press, 1994
- [78] Lai Y-J, Hwang C-L. *Fuzzy Multiple Objective Decision Making: Methods and Applications*. Springer-Verlag, 1994
- [79] Law AM, Kelton WD. *Simulation Modelling & Analysis*, 2nd edition. New York: McGraw-Hill, Inc., 1991
- [80] Lee SM. *Goal Programming for Decision Analysis*. Auerbach, Philadelphia, PA, 1972
- [81] Lee SM, Olson DL. A gradient algorithm for chance constrained nonlinear goal programming. *European Journal of Operational Research*, 1985, 22: 359-369
- [82] Liu B, Ku C. Dependent-chance goal programming and an application. *Journal of Systems Engineering & Electronics*, 1993, 4(2): 40-47
- [83] Liu B. Dependent-chance goal programming and its genetic algorithm based approach. *Mathematical and Computer Modelling*, 1996, 24(7): 43-52
- [84] Liu B, Esogbue AO. Fuzzy criterion set and fuzzy criterion dynamic programming. *Journal of Mathematical Analysis and Applications*, 1996, 199(1): 293-311
- [85] Lin B. Dependent-chance programming: A class of stochastic programming. *Computers & Mathematics with Applications*, 1997, 34(12): 89-104

- [86] Liu B, Iwamura K. Modelling stochastic decision systems using dependent-chance programming. *European Journal of Operational Research*, 1997, 101(1): 193-203
- [87] Liu B, Iwamura K. Chance constrained programming with fuzzy parameters. *Fuzzy Sets and Systems*, 1998, 94(2): 227-237
- [88] Liu B, Iwamura K. A note on chance constrained programming with fuzzy coefficients. to appear in *Fuzzy Sets and Systems*
- [89] Liu B. Dependent-chance programming in fuzzy environments. to appear in *Fuzzy Sets and Systems*
- [90] Liu B. Minimax chance constrained programming models for fuzzy decision systems. to appear in *Information Sciences*
- [91] Liu B. Stackelberg-Nash equilibrium for multilevel programming with multiple followers using genetic algorithms. to appear in *Computers & Mathematics with Applications*
- [92] Liu B, Esogbue AO. Fuzzy criterion dynamic programming for open inventory networks. Technical Report, 1996
- [93] Liu B, Iwamura K. Fuzzy programming with fuzzy decisions and fuzzy simulation based evolutionary algorithm. Technical Report, 1997
- [94] Liu B. Dependent-chance programming with fuzzy decisions. Technical Report, 1997
- [95] Liu B. Uncertain programming: Modelling, evolutionary computation and applications. In *Proceedings of Fourth Joint Conference on Information Sciences*, North Carolina, USA, October 23-28, 1998
- [96] Luhandjula MK. Linear programming under randomness and fuzziness. *Fuzzy Sets and Systems*, 1983, 10: 45-55
- [97] Luhandjula MK. On possibilistic linear programming. *Fuzzy Sets and Systems*, 1986, 18: 15-30
- [98] Luhandjula MK. Fuzzy optimization: An appraisal. *Fuzzy Sets and Systems*, 1989, 30: 257-282
- [99] Luhandjula MK. Fuzziness and randomness in an optimization framework. *Fuzzy Sets and Systems*, 1996, 77: 291-297
- [100] Mares M. *Computation Over Fuzzy Quantities*. CRC Press, Boca Raton, 1994

- [101] Mareschal B. Stochastic multicriteria decision making and uncertainty. *European Journal of Operational Research*, 1986, 26(1): 58-64
- [102] Martel A, Price W. Stochastic programming applied to human resource planning. *Journal of the Operational Research Society*, 1981, 32: 187-196
- [103] Masud A, Hwang C. Interactive sequential goal programming. *Journal of Operational Research Society*, 1981, 32: 391-400
- [104] Michalewicz Z. *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd ed. Springer-Verlag, 1996
- [105] Mital KV. *Optimization Methods*. Eastern: Wiley, 1976
- [106] Mitchell M. *An Introduction to Genetic Algorithms*. The MIT Press, 1996
- [107] Morgan B. *Elements of Simulation*. London: Chapman & Hall, 1984
- [108] Morgan DR, Eheart JW, Valocchi AJ. Aquifer remediation design under uncertainty using a new chance constrained programming technique. *Water Resources Research*, 1993, 29(3): 551-561
- [109] Mukherjee SP. Mixed strategies in chance-constrained programming. *Journal of the Operational Research Society*, 1980, 31: 1045-1047
- [110] Ostasiewicz W. A new approach to fuzzy programming. *Fuzzy Sets and Systems*, 1982, 7: 139-152
- [111] Negoita CV, Ralescu D. On fuzzy optimization. *Kybernetes*, 1977, 6: 193-195
- [112] Puri ML, Ralescu D. Fuzzy random variables. *Journal of Mathematical Analysis and Applications*, 1986, 114: 409-422
- [113] Ramer A. Conditional possibility measures. *International Journal of Cybernetics and Systems*, 1989, 20: 233-247
- [114] Rockafellar RT. *Network Flows and Monotropic Optimization*. New York: John Wiley & Sons, 1984
- [115] Rubinstein RY. *Simulation and the Monte Carlo Method*. John Wiley & Sons, Inc., 1981

- [116] Saade JJ. Maximization of a function over a fuzzy domain. *Fuzzy Sets and Systems*, 1994, 62: 55-70
- [117] Saber HM, Ravindran A. Nonlinear goal programming theory and practice: A survey. *Computers and Operations Research*, 1993, 20: 275-291
- [118] Salkin HM. *Integer Programming*. Addison-Wesley, 1974
- [119] Savard G, Gauvin J. The steepest descent direction for nonlinear bilevel programming problem. *Operations Research Letters*, 1994, 15: 265-272
- [120] Schneider M, Kandel A. Properties of the fuzzy expected value and the fuzzy expected interval in fuzzy environment. *Fuzzy Sets and Systems*, 1988, 26: 373-385
- [121] Sengupta JK. *Stochastic Programming: Methods and Applications*. Amsterdam: North-Holland, 1972
- [122] Shin WS, Ravindran A. Interactive multiple objective optimization: Survey I—continuous case. *Computers and Operations Research*, 1991, 18(1): 97-114
- [123] Simmons DM. *Nonlinear Programming for Operations Research*. Prentice-Hall, 1975
- [124] Sivazlian BD, Stanfel LE. *Optimization Techniques in Operations Research*. Prentice-Hall, 1975
- [125] Slowinski R, Teghem Jr J. Fuzzy versus stochastic approaches to multicriteria linear programming under uncertainty. *Naval Research Logistics*, 1988, 35: 673-695
- [126] Smith J. *Computer Simulation Methods*. New York: Hafner, 1968
- [127] Sommer G, Pollatschek MA. A fuzzy programming approach to an air pollution regulation problem. *European Journal of Operational Research*, 1978, 10: 303-313
- [128] Steuer RE. *Multiple Criteria Optimization: Theory, Computation and Application*. John Wiley & Sons, Inc., 1986
- [129] Taha HA. *Integer Programming*. Macmillan, 1978
- [130] Tanaka H, Asai K. Fuzzy linear programming problems with fuzzy numbers. *Fuzzy Sets and Systems*, 1984, 13: 1-10

- [131] Teghem Jr J, DuFrane D, Kunsch P. STRANGE: An interactive method for multiobjective linear programming under uncertainty. *European Journal Operational Research*, 1986, 26(1): 65-82
- [132] Vajda S. *Probabilistic Programming*. New York: Academic Press, 1972
- [133] Van de Panne C, Popp W. Minimum cost cattle feed under probabilistic protein constraints. *Management Science*, 1963, 9: 405-430
- [134] Wagner BJ, Gorelick SM. Optimal ground water quality management under parameter uncertainty. *Water Resources Research*, 1987, 23(7), 1162-1174
- [135] Weistroffer H. An interactive goal programming method for non-linear multiple-criteria decision-making problems. *Computers and Operations Research*, 1983, 10(4): 311-320
- [136] Whalen T. Decision making under uncertainty with various assumptions about available information. *IEEE Transactions on Systems, Man and Cybernetics*, 1984, 14: 888-900
- [137] Williams HP. *Model Building in Mathematical Programming*. John Wiley & Sons, Ltd., 1978
- [138] Yager RR. Mathematical programming with fuzzy constraints and a preference on the objective. *Kybernetes*, 1979, 9: 285-291
- [139] Yager RR. On ordered weighted averaging aggregation operators in multicriteria decision making. *IEEE Transactions on Systems, Man and Cybernetics*, 1988, 18: 183-190
- [140] Yager RR. On the specificity of a possibility distribution. *Fuzzy Sets and Systems*, 1992, 50: 279-292
- [141] Yazenin AV. Fuzzy and stochastic programming. *Fuzzy Sets and Systems*, 1987, 22: 171-180
- [142] Yazenin AV. On the problem of possibilistic optimization. *Fuzzy Sets and Systems*, 1996, 81: 133-140
- [143] Zadeh LA. Fuzzy sets. *Information and Control*, 1965, 8: 338-353
- [144] Zadeh LA. Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man and Cybernetics*, 1973, 3: 28-44

- [145] Zadeh LA. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1978, 1: 3-28
- [146] Zangwill WI. *Nonlinear Programming* Prentice-Hall, 1969
- [147] Zhao R, Iwamura K, Liu B. Chance constrained integer programming and stochastic simulation based genetic algorithms. *Journal of Systems Science and Systems Engineering*, 1998, 7(1): 96-102
- [148] Zimmermann H-J. Description and optimization of fuzzy systems. *International Journal of General Systems*, 1976, 2: 209-215
- [149] Zimmermann H-J. Fuzzy programming and linear programming with several objective functions. *Fuzzy Sets and Systems*, 1978, 3: 45-55
- [150] Zimmermann H-J. Fuzzy mathematical programming. *Computers and Operations Research*, 1983, 10: 291-298
- [151] Zimmermann H-J, Zadeh LA, Gaines BR(eds). *Fuzzy Sets and Decision Analysis*. Amsterdam, New York, Oxford, 1984
- [152] Zimmermann H-J. Applications of fuzzy set theory to mathematical programming. *Information Science*, 1985, 36: 29-58
- [153] Zimmermann H-J. *Fuzzy Set Theory and its Applications*. Boston: Kluwer Nijhof, 1985
- [154] Zimmermann H-J. Applications of fuzzy set theory to mathematical programming. *Information Sciences*, 1985, 36: 29-58

一些常用的符号

x, y	决策变量
\tilde{x}, \tilde{y}	模糊决策变量
\mathbf{x}, \mathbf{y}	决策向量
$\tilde{\mathbf{x}}, \tilde{\mathbf{y}}$	模糊决策向量
$\tilde{a}, \tilde{b}, \tilde{c}$	模糊参数
ξ, η, τ	随机参向量或模糊参向量
ξ, η, τ	随机参数或模糊参数
μ	隶属函数
ϕ	概率密度函数
Φ	概率分布函数
f, f_i	目标函数
g, g_j	约束函数
Pr	概率测度
Pos	可能性测度
α, β	置信水平
d_i^+, d_i^-	正偏差和负偏差
S	可行集
\Re	实数集
\Re^n	n 维欧几里德空间
\vee	最大算子
\wedge	量小算子
lexmin	字典序极小化
eval	遗传算法中的评价函数

索引

α 水平截集, 51
 β 分布, 41
 F 分布, 43
 Γ 分布, 40
Hessian 法, 6
Kuhn-Tucker 条件, 5
maximax 模型, 223
minimax 模型, 227
Monte Carlo 模拟, 37
Pareto 解, 7
 t 分布, 43
0-1 规划, 10

二画

二项分布, 44
二次规划, 4
几何分布, 45
几何规划, 4

三画

三角模糊数, 54
三角分布, 44
上升法, 6, 25

四画

分枝定界法, 11
分枝变量, 10
分枝方向, 11
开放存储网络, 101, 154
风险, 73
不确定环境, 119, 124, 184, 202
不确定规划, ix, 12
无约束规划, 4
韦布分布, 42

五画

卡方分布, 43
可行区域简化, 7
可行集, 2
可行解, 2
对数正态分布, 41
正态分布, 41
目标规划, 1, 8
目标函数, 1
目标, 9
可能性理论, 52
可分离规划, 4
生产过程, 95, 152
加权向量空间削减法, 7

六画

机会约束目标规划, 77, 167,
201, 228
机会约束多目标规划, 77, 166,
201, 228
机会约束规划, x , 75, 165,
227
机会函数, 119, 125, 185, 203
凸组合, 3
凸规划, 4
凸集, 2
交叉操作, 23
决策分量, 4, 124
决策向量, 4, 124
动态规划, 1
多层规划, 1
多维正态分布, 46
多目标规划, 1, 7
负二项分布, 45
伪随机数, 39
约束集, 4
后代, 16
有效前沿面, 7

七画

妥协模型, 7
妥协解, 7
有效解, 7

补偿模型, 70

进化规划, 16

进化策略, 16

极点, 3

初始化过程, 19

报童问题, 64

均匀分布, 38, 46

评价函数, 20

极大流问题, 104

八画

直接法, 6

枚举法, 10

饲料混合问题, 97

线性适应度定标方案, 21

线性规划, 1

线性搜索法, 7

线性同余法, 39

变异操作, 24

非劣解, 7

非线性规划, 1

表示结构, 18

单纯形法, 3

松弛变量, 2

欧几里得空间, 52

泊松分布, 45

事件, 125, 185, 203

九画

染色体, 16
相关机会目标规划, 133, 188, 206
相关机会多目标规划, 130, 186, 204
相关机会规划, x , 117, 121, 128, 184, 202
指数分布, 40
柯西分布, 42
指数适应度, 21
诱导约束, 128, 186, 204
种群规模, 16
种子, 39
选择过程, 22
结构变量, 2
适应度, 16
逆变换法, 38

十画

资金预算, 112, 159, 179
准则锥简化法, 7

十一画

清晰等价类, 168
偏差, 8, 78, 168, 202
 正偏差, 8
 负偏差, 8
梯度法, 47

梯形模糊数, 54
隐含约束, 19
偏好函数, 7
随机搜索算法, 48
基于序的评价函数, 20
随机约束, 74
随机环境, 119
随机积分, 47
随机规划, ix , 1
随机关系, 125
随机资源分配, 98, 144
随机集合, 121
随机模拟, 37, 47, 88, 137
累积分布, 38
接受拒绝法, 46

十二画

割平面法, 11
确定等价类, 79
距离函数, 7
期望值模型, x , 64
遗传算法, 15, 70, 90, 138, 175, 192, 212
最优操作问题, 104
最优解, 4
量小费用流问题, 104
最短路问题, 104
赌轮, 22

十三画

置信水平, 74

数学规划, 1

十四画

模糊染色体, 212

模糊决策, 198

模糊环境, 184

模糊数, 51

模糊规划, ix, 1, 164

模糊关系, 187

模糊集, 50

模糊模拟, 37, 173, 191, 208,
232

十五画以上

整数规划, 1, 10

