最优控制的上界法与边际控制法

黄午阳　华兆麟　编著

科学出版社

最优控制的
上界法与边际控制法

黄午阳　华兆麟　编著

科学出版社
1989

内 容 簡 介

本书介绍多级优化问题与快速控制问题的新的求解方法——上界法与边际控制法. 内容包括：概论、一级系统、多级系统、计算机算法、边际控制法，以及上界法和边际控制法的主要结果和应用实例.

本书可供自动化技术、计算技术、经济管理专业的教师和学生以及有关专业的科技人员参考.

最优控制的

上界法与边际控制法

黄午阳 华兆麟 编著

责任编辑 李淑兰 李雪芹

科学出版社 出版

北京东黄城根北街 16 号

中国科学院印刷厂 印刷

新华书店北京发行所发行 各地新华书店经售

*

1989 年 6 月第 一 版 开本：787×1092 1/32

1989 年 6 月第一次印刷 印张：3

印数：0001—2 340 字数：65 000

ISBN 7-03-001060-4/TP·69

定价： 2.60 元

前　　言

本书着重讨论多级优化问题与快速控制问题的新的求解方法——上界法与边际控制法．虽然上述问题也可用动态规划和 Pontryagin 极大值原理求解，但却受到一定的限制．利用动态规划求解多级优化问题将受到计算机存储器的限制，而用 Pontryagin 极大值原理求解多级优化问题，则又要求每一级的状态与目标函数的所有可能值组成的集合是凸集．与 Pontryagin 的接换曲线法相比，边际控制法可解析地求出快速控制问题的最优时间，并且允许目标集随时间变动．

对于上界法与边际控制法，我们不仅给出了主要结果，而且通过实例来说明这两种方法的应用．

本书的编著得到了张钟俊教授和姚允龙教授的帮助和指导，在此表示衷心感谢．对于书中的不足之处，欢迎读者批评指正．

编著者

1987 年 7 月

目　　录

第一篇　上　界　法

第二篇　边际控制法

第一篇 上 界 法

在处理多级系统时，离散时间系统的优化往往是设计过程的主要步骤之一．一般地讲，其目的是寻求合适的参数值以便在某种指定的意义下最优地求解上述问题．为了寻求这些参数值，我们首先构造一个相应的数学模型．这样，寻求好的参数值实际上就起到了优化多级系统的离散时间模型的作用．本篇主要讨论离散时间系统优化的一个新的方法——上界法，应用上界法可研究出许多新的成果．

第一章 概　　论

1.1 优化问题的提出

实际的优化问题是多种多样的，这里仅举一些简单的但具有代表性的例子：

(1) 煤气的多级压缩问题

设煤气从初始压力 p_0 至最终压力 p_N 被等熵地压缩，且压缩按 N 级进行．在每一级上，煤气首先被绝热地压缩，然后等温地冷却到其初始温度．由物理学可知，第 i 级上的能量消耗为

$$E_i = mRT\frac{\nu}{\nu-1}\left[\left(\frac{x_i}{x_{i-1}}\right)^{\frac{\nu-1}{\nu}}-1\right]$$

其中，m 是压缩煤气的摩尔数；R 是普适煤气常数；T 是煤气的初始温度；ν 是固定压力下的煤气比热与固定容积下的煤气比热之比；x_i 是第 i 级压缩结束时的煤气压力．

我们希望确定级间压力以使压缩过程中的总的能量消耗为极小．如果第 i 级的决策变量 u_i 被定义为

$$u_i = \frac{x_{i+1}}{x_i}$$

则便可将我们的问题表达为

$$\min_{u_0,\cdots,u_{N-1}} \sum_{i=0}^{N-1} u_i^{\frac{\nu-1}{\nu}}$$

$$x_{i+1} = x_i u_i$$

$$x_0 = p_0$$

$$x_N = p_N$$

(2) 运输问题

设某种物资从 n 个仓库被运输到 N 个需求点(见图 1.1),且总的供应量等于总的需求量.

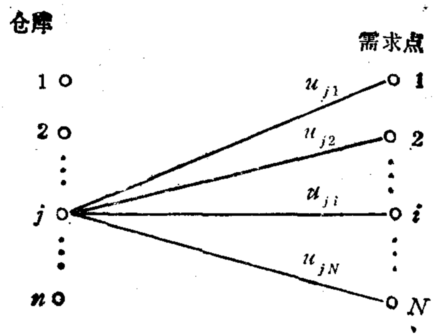

图 1.1　运输问题示意图

当以 u_i^j 表示从第 j 个仓库运到第 i 个需求点的物资数量,而以 $r_i^j(u_i^j)$ 表示相应的运输费用时,问题便成为确定 u_i^j, $= 1,2,\cdots,n; i = 1,2,\cdots,N$,运输物资的总费用为

$$R = \sum_{i=1}^{N}\sum_{j=1}^{n} r_i^j(u_i^j)$$

在约束 $u_i^j \geqslant 0$ 之下达到最小.

显然,第 j 个仓库备有的物资供应量为

$$w^j = \sum_{i=1}^{N} u_i^j,\ j = 1, 2, \cdots, n$$

而第 i 个需求点对物资的需求量为

$$d_i = \sum_{j=1}^{n} u_i^j,\quad i = 1, 2, \cdots, N$$

若将状态变量 x_i^j 定义为从第 j 个仓库运输到前 i 个需

求点的总物资量时，则可列出下列方程

$$x_i^j = x_{i-1}^j + u_i^j$$

$$x_0^j = 0$$

$$x_N^j = w^j$$

$$j = 1,2,\cdots,n-1;\ i = 1,2,\cdots,N$$

应该指出，虽然共有 n 个仓库，但状态变量只需要 $n-1$ 个. 这是因为每个需求点的需求量是预先指定的. 因此，第 n 个仓库对第 i 个需求点的供应量，可以从第 i 个需求点的总需求量 d_i 中减去其余 $n-1$ 个仓库对其的供应量而求得，即

$$u_i^n = d_i - \sum_{j=1}^{n-1} u_i^j$$

引进如下的向量

$$\boldsymbol{u}_i = (u_i^1, u_i^2, \cdots, u_i^{n-1})^T$$

$$\boldsymbol{x}_i = (x_i^1, x_i^2, \cdots, x_i^{n-1})^T$$

$$\boldsymbol{w} = (w^1, w^2, \cdots, w^{n-1})^T$$

可将运输问题表达为

$$\min_{\boldsymbol{u}_1,\cdots,\boldsymbol{u}_N} \sum_{i=1}^{N} \sum_{j=1}^{n} r_i^j(u_i^j)$$

$$\boldsymbol{x}_i = \boldsymbol{x}_{i-1} + \boldsymbol{u}_i$$

$$\boldsymbol{x}_0 = \boldsymbol{0}$$

$$\boldsymbol{x}_N = \boldsymbol{w}$$

$$\boldsymbol{u}_i \geqslant 0, \quad d_i - \sum_{j=1}^{n-1} u_i^j \geqslant 0$$

$$i = 1, 2, \cdots, N$$

(3) 催化剂的更换问题

在催化反应器中，催化效应随着催化剂的老化而逐渐减

弱．这是因为最佳反应条件随时间的推移而发生变化．我们的问题是，寻求某个时间周期内的最佳反应条件和更换催化剂的最佳时间，以获得最大收益．考察图 1.2 所示的系统，它

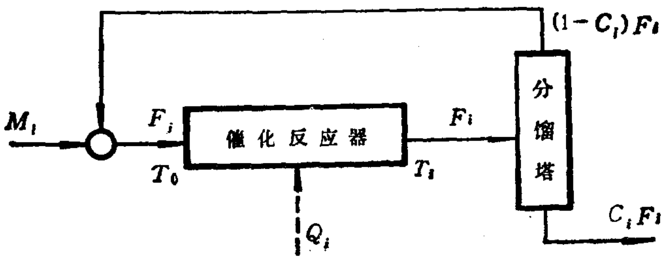

图 1.2　示意流程图

由管式催化反应器与分馏塔所组成．在第 i 个周期中，物质以固定流率 F_i 被注入反应器．在反应器中，化合物 A 被热解成化合物 B 和 G 后进入分馏塔．经分馏后，起反应的物质(最终产品)以速率 C_iF_i 流出，而未起反应的物质则以速率 $(1-C_i)F_i$ 重新循环．重新循环的物质加上以速率 M_i 注入的物质共同进入反应器．根据物质平衡原理可知

$$M_i = C_iF_i \tag{1.1}$$

$$F_i = M_i + (1 - C_i)F_i \tag{1.2}$$

反应过程可被表达为

$$C_i = a_1T_i - a_2F_i - a_3S_i \tag{1.3}$$

其中，T_i 是第 i 个周期中的出口温度，S_i 是通过催化反应器的累加流率，即

$$S_i = \sum_{j=1}^{i} F_j \tag{1.4}$$

S_i 表示系统的状态，它等价于催化剂的使用期限．a_1, a_2, a_3 是已知常数．根据能量平衡原理可知

$$Q_i = F_i c_p(T_i - T_0) + hC_iF_i \tag{1.5}$$

其中，Q_i 是在第 i 个周期中输入反应器的热量，c_p 是正在起反应的混合物的平均热容量，h 是反应热量，T_0 是进入反应器的混合物的温度。

通常 T_0 被认为是常量，而 C_i，T_i，F_i 则满足如下的约束条件

$$C_{\min} \leqslant C_i \leqslant C_{\max}$$
$$T_{\min} \leqslant T_i \leqslant T_{\max}$$
$$F_{\min} \leqslant F_i \leqslant F_{\max}$$

每单位时间所获得的收益被定义为

$$r_i = C_iF_iv_1 - M_iv_2 - Q_iv_3 - (1 - C_i)F_iv_4 - v_5 \tag{1.6}$$

其中；v_1 是产品 B 和 G 的总价值；v_2 是加料的代价；v_3 是加热的代价；v_4 是通过分馏塔重复循环的液流的代价；v_5 是固定费用。

若引进状态变量 $x_i = S_i$ 及决策变量

$$\boldsymbol{u}_i = \begin{bmatrix} u_i^1 \\ u_i^2 \end{bmatrix} = \begin{bmatrix} T_i \\ F_i \end{bmatrix}$$

则根据方程 (1.4) 有

$$x_i = x_{i-1} + u_i^2, \quad i = 1, 2, \cdots, N$$

而由方程(1.1)，(1.3)，(1.5)和(1.6)可得

$$\begin{aligned} r_i(x_i, \boldsymbol{u}_i) = {} & (a_1u_i^1 - a_2u_i^2 - a_3x_i)u_i^2(v_1 - v_2 - hv_3 + v_4) \\ & - c_pu_i^2(u_i^1 - T_0)v_3 - u_i^2v_4 - v_5 \end{aligned}$$

于是催化剂的更换问题可被表达为

$$\max_{\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N, N} \sum_{i=1}^{N} r_i(x_i, \boldsymbol{u}_i)$$

$$x_i = x_{i-1} + u_i^2$$

$$x_0 = 0$$

$$C_{\min} \leqslant a_1 u_i^1 - a_2 u_i^2 - a_3 x_i \leqslant C_{\max}$$
$$T_{\min} \leqslant u_i^1 \leqslant T_{\max}$$
$$F_{\min} \leqslant u_i^2 \leqslant F_{\max}$$

1.2　多级优化问题

在 1.1 节的三个实例中，系统均由状态方程加以描述，且状态变量的初值和(或)终值是已知的．它们都是在某些约束之下求出每一级的决策变量，以使目标函数达到极大或极小．这类问题通常被称为多级优化问题．

本篇仅讨论多级优化问题中的一类子问题．在这类子问题中，总的目标函数是每一级的目标函数 $r_i(\boldsymbol{x}_i, \boldsymbol{u}_i)$ 的和，级数 N 是固定的，状态的初始点 $\boldsymbol{x}_0$ 和终点 $\boldsymbol{x}_N$ 是已知的，关于决策变量的约束在每一级分别加以考虑．这样，我们所讨论的问题便可被表达为

$$\begin{aligned} &\max \sum_{i=0}^{N-1} r_i(\boldsymbol{x}_i, \boldsymbol{u}_i) \\ &\boldsymbol{x}_{i+1} = \boldsymbol{g}_i(\boldsymbol{x}_i, \boldsymbol{u}_i) \\ &\boldsymbol{x}_0 = \bar{\boldsymbol{x}}_0, \quad \boldsymbol{x}_N = \bar{\boldsymbol{x}}_N \\ &\boldsymbol{u}_i \in U_i \end{aligned} \tag{1.7}$$

对于一般的多级优化问题来说，上述表达是有一定局限性的．例如，1.1 节中的催化剂更换问题与问题 (1.7) 有所不同：

首先，催化剂更换问题不仅对决策变量施加约束，而且对状态变量也施加约束．因为我们总可以通过

$$\boldsymbol{x}_i = \boldsymbol{g}_{i-1}(\boldsymbol{x}_{i-1}, \boldsymbol{u}_{i-1})$$
$$\boldsymbol{x}_{i-1} = \boldsymbol{g}_{i-2}(\boldsymbol{x}_{i-2}, \boldsymbol{u}_{i-2})$$
$$\cdots\cdots$$

$$x_1 = g_0(x_0, u_0)$$

将状态变量 x_i 表达为决策变量的函数，故对状态变量具有约束的问题总可被转化为对决策变量具有约束的问题. 然而，它们不一定具有问题(1.7)中那样的约束形式，因为可能会遇到多于一个决策变量的函数的某些约束. 此时，利用罚函数可给出更好的结果.

第二，催化剂更换问题中的终点是不固定的. 本篇虽不考虑这种问题，但在所介绍的体系之下，处理这种问题是相当容易的.

第三，催化剂更换问题中的级数N是不固定的，但它受极大化的支配. 这种问题可通过对不同的N多次求解固定级数问题而加以解决.

本篇所介绍的方法虽有一定的局限性，但进行适当的推广，扩充其应用领域是完全有可能的. 下面概述求解多级优化问题的发展历史.

1.3 多级优化问题求解方法的发展

处理多级优化问题的许多不同技巧已被提出. 在特殊情况下，可试用某些直观推断法或简单的直接计算法. 在其他情况下，可应用经典的微分学或变分学，也可应用线性或非线性规划来处理问题. 50 年代末，发明了解决这类问题的两个最一般的方法——动态规划与极大值原理法.

动态规划方法主要是由 Bellman 发现和发展的. 这一方法基于如下的优化原理："无论初始决策如何，剩余决策对由初始决策引起的结果必然是最优的."

动态规划方法虽然也可被推广到连续时间系统，但它主要用于离散时间系统的优化问题或如前面所述的多级优化问

题．在处理这些问题时，这一方法非常有效，但其应用却受到在某些情况下需要扩展计算机存储器的限制．

极大值原理由 Pontryagin 提出，由他和他的助手加以发展．极大值原理的主要思想是构造一个取决于控制和状态的特殊函数——Hamiltonian 函数．当已知状态的最优值时，则在每一级由该级的最优决策使 Hamiltonian 函数最优化．起初，这方面的工作仅限于连续时间系统．正确理解极大值原理对于优化离散时间系统的作用，是由 Propoi[1,2] 和 Halkin[3,4] 完成的．他们指出，当每一级的状态与目标函数的所有可能值组成的集合是凸集时，极大值原理是有效的．1966 年 Holtzman 将上述假设减弱为有方向的凸性，即减弱为对于唯一方向(目标值增加方向)的凸性．

在近期的工作中，也有人强调使用非线性规划方法，详见 Canon 等[5]及 Propoi[6] 的著作．由于应用极大值原理可减少计算每一级的变量个数，所以极大值原理比非线性规划更能引起人们的兴趣．但极大值原理也有缺点，其主要缺点是，要求被优化的函数必须具有连续性与可微性，且在许多实际情况下还必须具有凸性或线性．为了克服极大值原理的缺点，人们又先后提出了弱极大值原理、局部极大值原理以及准极大值原理．除此之外，推广某种意义上的Hamiltonian 函数以保持其在最优决策特征上的极大值，也是这方面的一项重要工作．Ravn 和 Vidal 通过引入广义的 Hamiltonian 函数建立了更为广泛的理论体系．1980 年，Ravn 以简明的假设原理的形式概述了上述理论，并命名为上界法．

第二章 一级系统

本章将研究受等式约束的静态系统的优化问题。我们从引进某些基本概念着手，然后导出由四个定理组成的主要结果。它们是处理 Hamiltonian 函数极大化以及原始问题的最优解与极大化 Hamiltonian 函数的解之间关系的定理。这里对于被优化的函数既不要求可微也不要求连续。可以认为，这部份中的鞍点定理介绍了一个新的尚未发表过的结果。在证明了这一主要定理之后，我们透彻地讨论了线性支集与二次型支集的某些重要情形。

2.1 基本概念

本节通过一级优化问题引进可行解、最优解、上界、支集以及广义 Hamiltonian 函数等基本概念。

设我们所考察的问题为

$$\begin{aligned} &\max r(\boldsymbol{u}) \\ &\boldsymbol{f}(\boldsymbol{u}) = \bar{\boldsymbol{x}} \\ &\boldsymbol{u} \in U \end{aligned} \tag{2.1}$$

其中，$r(\boldsymbol{u})$ 是实函数；$\boldsymbol{f}(\boldsymbol{u})$ 是 n 维列向量函数；$\boldsymbol{u}$ 是 m 维列向量 $(m > n)$；U 是 m 维空间 R^m 中的给定集合；$\bar{\boldsymbol{x}}$ 是 n 维空间 R^n 中的给定向量。

对于给定的 $\boldsymbol{u}$，$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{u})$ 称为状态，而

$$\hat{\boldsymbol{x}} = \begin{bmatrix} r(\boldsymbol{u}) \\ \boldsymbol{f}(\boldsymbol{u}) \end{bmatrix}$$

称为扩展状态。

当 $\boldsymbol{u}$ 在 U 中取值时，我们便得到一个 $\boldsymbol{x}$ 值的集合，该集合被称为可达状态集，记作 X。同样，$\boldsymbol{u}$ 在 U 中取值时，$\hat{\boldsymbol{x}}$ 将在被称为可达扩展状态集的集合中取值，可达扩展状态集记作 $\hat{X}$。

对于问题 (2.1)，如 $\boldsymbol{u}\in U$ 且 $\boldsymbol{f}(\boldsymbol{u})=\bar{\boldsymbol{x}}$，则称向量 $\boldsymbol{u}$ 为它的可行解。如 $\boldsymbol{u}^*$ 是问题 (2.1) 的可行解，且对于 $\boldsymbol{u}\in U$ 有

$$r(\boldsymbol{u}^*)\geqslant r(\boldsymbol{u})$$

则称向量 $\boldsymbol{u}^*$ 为最优解。

对于给定的 $\boldsymbol{x}$，满足 $\boldsymbol{u}\in U$，$\boldsymbol{f}(\boldsymbol{u})=\boldsymbol{x}$ 的所有 $r(\boldsymbol{u})$ 的上确界定义为 $\mathrm{ub}(\boldsymbol{x})$，即

$$\mathrm{ub}(\boldsymbol{x})=\sup_{\substack{\boldsymbol{u}\in U\\ \boldsymbol{f}(\boldsymbol{u})=\boldsymbol{x}}} r(\boldsymbol{u})$$

对所有 $\boldsymbol{x}\in X$，$\mathrm{ub}(\boldsymbol{x})$ 可看作一个函数，我们称它为上界函数。有时也常把它看作 $n+1$ 维空间中的一个点集，并称之为 $\hat{X}$ 的上界，记作 $\{\mathrm{ub}\}$。

为了讨论方便起见，我们假设对所有的 $\boldsymbol{x}\in X$，$\mathrm{ub}(\boldsymbol{x})<\infty$ 且有

$$\begin{bmatrix}\mathrm{ub}(\boldsymbol{x})\\ \boldsymbol{x}\end{bmatrix}\in\hat{X}$$

这等价于对所有的 $\boldsymbol{x}\in X$，总存在 $\boldsymbol{u}\in U$，使得 $\boldsymbol{f}(\boldsymbol{u})=\boldsymbol{x}$，$r(\boldsymbol{u})=\mathrm{ub}(\boldsymbol{x})$。此外，我们还假设 $\bar{\boldsymbol{x}}\in X$，并以 $\boldsymbol{u}^*$ 表示最优解，其相应的扩展状态记为 $\hat{\boldsymbol{x}}^*$。

由可行解与 $\hat{X}$ 的定义可知，问题(2.1)具有可行解 $\boldsymbol{u}$ 的充要条件是 $\bar{\boldsymbol{x}}\in X$ 及

$$\begin{bmatrix}\mathrm{ub}(\bar{\boldsymbol{x}})\\ \bar{\boldsymbol{x}}\end{bmatrix}\in\hat{X}$$

对于问题 (2.1) 的最优解有如下定理：

定理 1 如果 $\boldsymbol{u}^*$ 是问题(2.1)的最优解，则

$$\hat{\boldsymbol{x}}^* = \begin{bmatrix} r(\boldsymbol{u}^*) \\ \boldsymbol{f}(\boldsymbol{u}^*) \end{bmatrix}$$

位于 $\hat{X}$ 的上界 {ub}.

证明 设 $\tilde{\boldsymbol{u}}$ 是一个最优解且

$$\hat{\hat{\boldsymbol{x}}} = \begin{bmatrix} r(\tilde{\boldsymbol{u}}) \\ \boldsymbol{f}(\tilde{\boldsymbol{u}}) \end{bmatrix}$$

不位于 {ub}，则由{ub}的定义及上述充要条件可知，必存在一个 $\boldsymbol{u}^*$，使得 $\boldsymbol{f}(\boldsymbol{u}^*) = \boldsymbol{f}(\tilde{\boldsymbol{u}})$ 且

$$\hat{\boldsymbol{x}}^* = \begin{bmatrix} r(\boldsymbol{u}^*) \\ \boldsymbol{f}(\boldsymbol{u}^*) \end{bmatrix}$$

位于 {ub}。这表明 $r(\boldsymbol{u}^*) > r(\tilde{\boldsymbol{u}})$，从而与 $\tilde{\boldsymbol{u}}$ 的最优性相矛盾，故定理得证。

下面将引进支集和广义 Hamiltonian 函数的概念。

设有一个定义在集合 X 上的 n 元函数

$$\pi^0(\boldsymbol{x}) = \sum_{i=1}^{n} \pi_i^0(x_i)$$

如存在一个实数 k，使得

$$\pi^0(\boldsymbol{x}^0) + k = \mathrm{ub}(\boldsymbol{x}^0) \tag{2.2}$$

$$\pi^0(\boldsymbol{x}) + k \geqslant \mathrm{ub}(\boldsymbol{x}), \quad \boldsymbol{x} \in X \tag{2.3}$$

则称 $\pi^0(\boldsymbol{x})$ 为 $\boldsymbol{x}^0$ 点的支集。

当 $\pi^0(\boldsymbol{x}) + k = \mathrm{ub}(\boldsymbol{x})$ 时，我们就称 $\pi^0(\boldsymbol{x})$ 在 $\boldsymbol{x}$ 点支承 $\mathrm{ub}(\boldsymbol{x})$。根据 $\boldsymbol{x}^0$ 点的支集定义可知，$\pi^0(\boldsymbol{x})$ 在 $\boldsymbol{x}^0$ 点显然是支承 $\mathrm{ub}(\boldsymbol{x})$ 的，但它也有可能在其他点支承 $\mathrm{ub}(\boldsymbol{x})$。我们将 $\bar{\boldsymbol{x}}$ 点的支集记作 $\pi^*(\boldsymbol{x})$，其特点是：存在一个实数 k，使得

$$\pi^*(\bar{\boldsymbol{x}}) + k = \mathrm{ub}(\bar{\boldsymbol{x}})$$

$$\pi^*(\boldsymbol{x}) + k \geqslant \mathrm{ub}(\boldsymbol{x}), \quad \boldsymbol{x} \in X$$

如果(2.3)用下述条件