

非线性 0-1 规划问题的连续化及其遗传算法解法

隋允康, 贾志超, 杜家政

(北京工业大学 机械工程与应用电子技术学院, 北京 100022)

摘 要: 为了求解非线性 0-1 离散规划问题, 通过非线性等式的“离散性约束”将其转化为 $[0, 1]$ 区间上等价的连续变量非线性规划。对于目标函数非线性、约束线性的 0-1 规划问题, 可以使用乘法法来解决含“离散性约束”的非线性优化问题。对于目标函数和约束函数均为非线性的问题, 可以采用约束松弛法将离散性约束松弛为不等式约束。两种方法处理后均使用遗传算法程序 GENOCOP 求解。乘法法求解得到的结果比较准确, 约束松弛法属于近似方法, 可以求解带非线性不等式约束的问题。用本文的方法对多个非线性 0-1 规划问题的算例进行了计算, 并将计算结果同枚举法的计算结果比较, 结果表明该方法准确、有效。

关键词: 非线性 0-1 规划; 连续化方法; 遗传算法; GENOCOP

中图分类号: O 343

文献标识码: A

文章编号: 0254-0037(2008)08-0785-07

0-1 非线性规划问题是整数非线性规划问题的特例, 在实际中应用很广泛。研究 0-1 非线性规划问题的求解方法有很重要的实际意义。由于设计变量的离散性, 0-1 规划问题求解起来比较困难。当目标为非线性, 特别是高度非线性时, 其求解难度更大。通常 0-1 非线性规划问题的求解方法可以分为离散方法和连续化方法。

离散方法是从设计变量的离散特性出发直接求解整数规划。传统的离散方法大多属于组合算法^[1], 如穷举法、隐枚举法等。这类方法能准确地找到问题的全局最优解, 但随着问题规模的增大, 其计算代价很大。另一种是离散的启发式算法, 如遗传算法^[2]。这种方法的主要缺点是不能很好地处理约束, 而且很容易出现过早收敛问题。大部分文献都是关于求解 0-1 背包问题, 对于非线性问题没有比较完善和通用的计算程序。

连续化方法是将原问题转化为关于连续变量的规划问题求解。Templeman 等对结构离散变量优化中的分段优化方法通过构造力学模型将离散截面转化为连续杆长, 最后要对少数分节杆进行圆整^[3]; 文献[4]在结构拓扑优化中提出的 ICM 方法利用过滤函数用连续变量逼近 0-1 离散变量。这些方法属于模型近似转化的方法; Pardalos 提出通过引进新的设计变量, 把原问题转化为线性互补问题进行求解^[5-6]; Kiwiel 等提出通过求解凸规划的 Bregman 临近点算法来求解大规模 0-1 规划问题^[7]; 李兴斯等利用互补约束代替 0-1 条件, 并利用凝聚函数法进行光滑处理来求解 0-1 规划问题^[8]。这些方法属于模型精确转化的方法。无论是模型近似还是精确转化都一定程度上降低了问题的求解难度, 提高了求解速度。但是这些方法只适用于求解某一类的 0-1 规划问题, 实际应用中有很多的局限性, 对于非线性的 0-1 规划问题, 比较通用的方法和程序更少。

本文以 0-1 规划设计变量的离散特点和目标函数的非线性为出发点, 通过增加一个关于设计变量的非线性等式约束代替设计变量的 0-1 离散限制(以下简称为离散性约束), 将原规划表示成一个 $[0, 1]$ 区间上的等价连续变量规划问题。文中将每个分量的 0-1 离散条件整体化, 把 n 个线性互补约束化成一个, 减少了约束个数。由于新产生的规划问题是一个非凸非线性规划问题, 传统的方法很难得到问题的全局最优解。考虑到遗传算法良好的全局搜索能力, 本文采用 Michalewicz 和 Janikow 设计的遗传算法程序 GENOCOP 求解新的连续变量规划问题^[9-10]。

收稿日期: 2007-05-17。

基金项目: 国家自然科学基金资助项目(10472003); 高校博士点基金资助项目(2006005010)。

作者简介: 隋允康(1943-), 男, 辽宁大连人, 教授, 博士生导师。

1 非线性 0-1 规划问题的等价连续化

对于非线性 0-1 规划问题

$$\begin{cases} \text{求} & t_i = 0 \text{ 或 } 1 (i = 1, \dots, n) \\ \text{使} & W = W(t) \rightarrow \min \\ \text{s. t.} & G = G_j(t) \leq 0 (j = 1, \dots, m) \end{cases} \quad (1)$$

可以等价地表述为连续变量优化问题

$$\begin{cases} \text{求} & t \in E^n \\ \text{使} & W = W(t) \rightarrow \min \\ \text{s. t.} & G = G(t) \leq 0 \\ & T = \sum_{i=1}^n \alpha_i (t_i - t_i^2) = 0 \quad (\text{其中 } \alpha_i > 0, \text{ 且已给定}) \\ & 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \end{cases} \quad (2)$$

证明这 2 个问题的等价性, 只需要证明

$$t_i = 0 \text{ 或 } 1 (i = 1, \dots, n) \Leftrightarrow \begin{cases} \sum_{i=1}^n \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \end{cases}$$

证明: 首先证明

$$\begin{cases} \sum_{i=1}^n \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \end{cases} \Leftrightarrow \begin{cases} \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \end{cases} \quad (i = 1, \dots, n)$$

只要对右边第一式求和就可由右式推出左式, 下面用数学归纳法由左式推出右式.

1) 当 $n = 1$ 时, 左边和右边都是

$$\begin{cases} \alpha_1 (t_1 - t_1^2) = 0 \\ 0 \leq t_1 \leq 1 \end{cases}$$

2) 假设当 $n = k$ 时, 有

$$\begin{cases} \sum_{i=1}^k \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \quad (i = 1, \dots, k) \end{cases} \Rightarrow \begin{cases} \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \end{cases} \quad (i = 1, \dots, k)$$

假设 $n = k + 1$ 时, $\alpha_{k+1}(t_{k+1} - t_{k+1}^2) \neq 0 (0 \leq t_{k+1} \leq 1)$, 那么由

$$\begin{cases} \alpha_{k+1}(t_{k+1} - t_{k+1}^2) \neq 0 \\ 0 \leq t_{k+1} \leq 1 \\ \alpha_{k+1} > 0 \end{cases} \Rightarrow \alpha_{k+1}(t_{k+1} - t_{k+1}^2) > 0$$

又考虑到

$$\begin{cases} \sum_{i=1}^k \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 (i = 1, \dots, k) \end{cases}$$

得

$$\begin{cases} \sum_{i=1}^{k+1} \alpha_i (t_i - t_i^2) > 0 \\ 0 \leq t_i \leq 1 (i = 1, \dots, k + 1) \end{cases}$$

与

$$\begin{cases} \sum_{i=1}^{k+1} \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \quad (i = 1, \dots, k+1) \end{cases}$$

相矛盾,所以 $\alpha_{k+1}(t_{k+1} - t_{k+1}^2) = 0 (0 \leq t_{k+1} \leq 1)$ 成立.

3) 综合上述证明,得到

$$\begin{cases} \sum_{i=1}^n \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \end{cases} \Leftrightarrow \begin{cases} \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \end{cases} \quad (i = 1, \dots, n)$$

再证明

$$\begin{cases} \alpha_i (t_i - t_i^2) = 0 \\ 0 \leq t_i \leq 1 \end{cases} \quad (i = 1, \dots, n) \Leftrightarrow t_i = 0 \text{ 或 } 1 \quad (i = 1, \dots, n)$$

由左式推出右式,由左式得

$$\begin{cases} \alpha_i t_i (1 - t_i) = 0 \\ 0 \leq t_i \leq 1 \end{cases} \quad (i = 1, \dots, n)$$

因为 $\alpha_i > 0$, 故推出右式.

由右式推出左式,由右式得 $t_i(1 - t_i) = 0$, 进而得到左式.

据以上证明可以得式(1)和式(2)是等价的.

2 连续化后 0-1 规划问题的遗传算法求解

2.1 乘子法

对于式(1)的等价问题式(2),使用乘子法得

$$\begin{cases} \text{求 } t \in E^n, \lambda \in E^1 \\ \text{使 } W = W(t) + \lambda \sum_{i=1}^n \alpha_i (t_i - t_i^2) \rightarrow \min \\ \text{s. t. } G = G(t) \leq 0 \\ \quad 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \\ \quad \underline{\lambda} \leq \lambda \leq \bar{\lambda} \end{cases} \quad (3)$$

式中, λ 的上下限 $\bar{\lambda}$ 和 $\underline{\lambda}$ 由数值经验确定; α_i 的取法对于优化收敛的稳定性有影响, 本文最初取 $\alpha_i = 1 (i = 1, \dots, n)$, λ 的上下限须根据不同的问题而变化. 为了使 λ 的上下限相对不变, 必须取适当的 α_i 使原目标和非线性约束 T 的量级接近. 为了使 α_i 的确定简单, 易操作, 将所有的 α_i 取相同的值 k . 当式(2)的目标函数为多项式形式时, 可取 k 为所有系数的绝对值中最大的一个. 这样式(3)转化为

$$\begin{cases} \text{求 } t \in E^n, \lambda \in E^1 \\ \text{使 } W = W(t) + \lambda k \sum_{i=1}^n (t_i - t_i^2) \rightarrow \min \\ \text{s. t. } G = G(t) \leq 0 \\ \quad 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \\ \quad 5 \leq \lambda \leq 20 \end{cases} \quad (4)$$

2.2 离散约束松弛法

将等式离散性约束 $\sum_{i=1}^n \alpha_i (t_i - t_i^2) = 0$ 转换成 2 个不等式约束而直接利用遗传算法求解. 考虑到

$$\sum_{i=1}^n a_i(t_i - t_i^2) = 0 \Leftrightarrow \begin{cases} \sum_{i=1}^n a_i(t_i - t_i^2) \geq 0 \\ \sum_{i=1}^n a_i(t_i - t_i^2) \leq 0 \end{cases} \quad (5)$$

因为 $0 \leq t_i \leq 1 (i = 1, \dots, n)$, 所以式(5)中第1个不等式自然满足, 只需要满足

$$\sum_{i=1}^n a_i(t_i - t_i^2) \leq 0 \quad (6)$$

式(6)是个很强的约束, 很难找到满足式(6)的可行解, 在遗传过程中不利于新个体的产生, 所以对式(6)进行约束松弛得到

$$\sum_{i=1}^n a_i(t_i - t_i^2) \leq \delta \quad (7)$$

其中 δ 为确定的正数.

式(2)按离散性约束松弛法最终求解时, 取 $a_i = 1 (i = 1, \dots, n)$, 所用的模型为

$$\begin{cases} \text{求} & t \in E^n \\ \text{使} & W = W(t) \rightarrow \min \\ \text{s. t.} & G = G(t) \leq 0 \\ & \sum_{i=1}^n (t_i - t_i^2) \leq \delta \\ & 0 \leq t_i \leq 1 \quad (i = 1, \dots, n) \end{cases} \quad (8)$$

其中 δ 为确定的正数, 理论上 δ 应该比较小, 但是为了遗传过程的稳定, δ 又不能取得太小. 数值试验表明, 由于误差累积的原因, δ 的大小与问题规模有关系, 当问题的规模比较小可以取 $\delta = 0.01$. 当问题的规模比较大时可以取 $\delta = 0.1$.

3 数值算例及比较

算例 1

$$\begin{cases} \text{使} & 50x_1^2 - 42x_1 + 50x_2^2 - 44x_2 + 50x_3^2 - 45x_3 + 50x_4^2 - 47x_4 + 50x_5^2 - 47x_5 - \\ & 30x_1x_4x_5 - 50x_2x_3 \rightarrow \min \\ \text{s. t.} & 20x_1 + 12x_2 + 11x_3 + 7x_4 + 4x_5 \leq 40 \\ & x_i \in \{0, 1\} \quad (i = 1, \dots, 5) \end{cases}$$

该问题为目标非线性、约束线性的 0-1 规划问题, 使用乘子法和约束松弛法均可求解. 其求解结果和使用枚举法的求解结果如表 1 所示, 目标函数的遗传历史如图 1 所示.

表 1 算例 1 的计算结果

Table 1 Result of example 1

模型处理方法	x_1	x_2	x_3	x_4	x_5	$\lambda(\delta)$
本文乘子法	0.000 0	1.000 0	1.000 0	0.000 0	0.000 0	12.345 9
本文约束松弛法	0.000 6	0.999 9	0.999 9	0.000 7	0.008 7	0.01
枚举法	0	1	1	0	0	-39(目标)

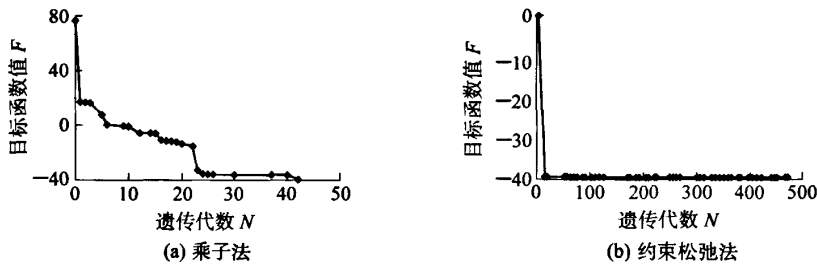


图 1 遗传历史
Fig.1 Genetic history

算例 2

$$\begin{cases} \text{使} & 3x_1^5 + 4x_1x_3 + 2x_2^3 - 7x_4x_5x_6 + 2x_4^2 + 5x_5^3 + 8x_6^2 - 4x_1x_2 - 3x_1x_3 \rightarrow \min \\ \text{s. t.} & 4x_1 + 3x_2 + 2x_3 + 3x_4 + 4x_5 + 6x_6 \geq 14 \\ & 3x_1 + 2x_2 + 3x_3 + 4x_4 + 2x_5 + 3x_6 \geq 10 \\ & x_i \in \{0, 1\} \quad (i = 1, \dots, 6) \end{cases}$$

该问题为目标非线性、约束线性的 0-1 规划问题,使用乘子法和约束松弛法均可以求解。其求解结果和使用枚举法的求解结果如表 2 所示,目标函数的遗传历史如图 2 所示。

表 2 算例 2 的计算结果
Table 2 Result of example 2

模型处理方法	x_1	x_2	x_3	x_4	x_5	x_6	$\lambda(\delta)$
本文乘子法	1.000 0	1.000 0	0.000 0	1.000 0	1.000 0	0.000 0	10.385 9
本文约束松弛法	0.999 9	1.000 0	0.000 0	1.000 0	0.993 9	0.004 0	0.001 0
枚举法	1	1	0	1	1	0	8(目标)

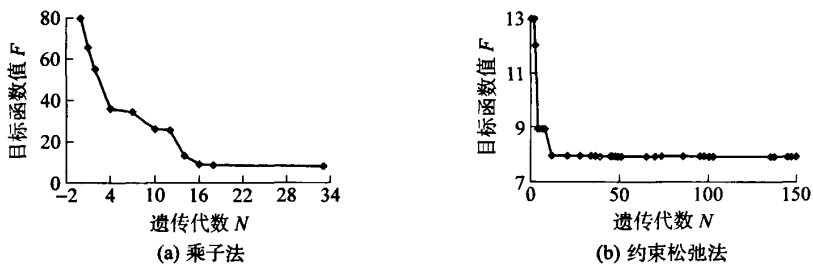


图 2 遗传历史
Fig.2 Genetic history

算例 3

$$\begin{cases} \text{使} & f(x) = -e^{x_1} - \sin x_2 + 1/x_3 - \arctan x_4 \rightarrow \min \\ \text{s. t.} & x_1 + x_2 + x_3 + x_4 \leq 2 \\ & 2x_1 + 3x_2 \leq 4 \\ & x_2 + 2x_3 + x_4 \leq 2 \\ & x_i \in \{0, 1\} \end{cases}$$

该问题为目标非线性约束线性的 0-1 规划问题,使用乘子法和约束松弛法均可以求解。其求解结果和使用枚举法的求解结果如表 3 所示,目标函数的遗传历史如图 3 所示。

表 3 算例 3 的计算结果
Table 3 Result of example 3

模型处理方法	x_1	x_2	x_3	x_4	$\lambda(\delta)$
本文乘法	1.000 0	0.000 0	1.000 0	0.000 0	5.437 2
本文约束松弛法	1.000 0	0.000 1	0.999 8	0.000 0	0.001 0
枚举法	1	0	1	0	-1.718 28(目标)

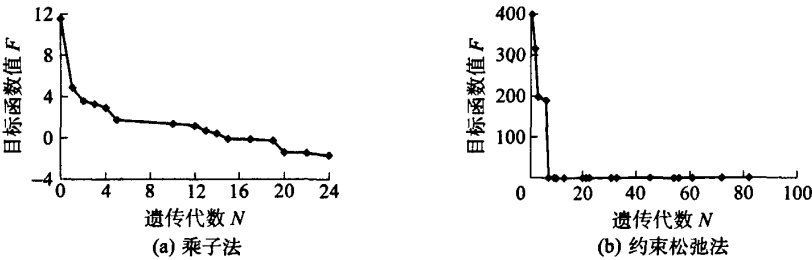


图 3 遗传历史
Fig.3 Genetic history

算例 4

$$\begin{cases} \text{使} & f(x) = x_1^2 + x_2^2 + 2x_3^2 + 2x_4^2 - 5x_1 - 5x_2 - 21x_3 + 7x_4 \rightarrow \min \\ \text{s.t.} & 8 - x_1 + x_2 - x_3 + x_4 - x_1^2 - x_2^2 - x_3^2 - x_4^2 \geq 0 \\ & 10 + x_1 + x_4 - x_1^2 - 2x_2^2 - x_3^2 - 2x_4^2 \geq 0 \\ & x_i \in \{0, 1\} \end{cases}$$

该问题为目标非线性约束线性的 0-1 规划问题,使用约束松弛法求解. 其求解结果和使用枚举法的求解结果如表 4 所示,目标函数的遗传历史如图 4 所示.

表 4 算例 4 的计算结果
Table 4 Result of example 4

模型处理方法	x_1	x_2	x_3	x_4	δ
本文约束松弛法	0.999 9	0.999 9	1.000 0	0.000 0	0.001 0
枚举法	1	1	1	0	-27(目标)

4 讨论

本文通过非线性等式约束将 0-1 离散变量优化问题等价表示成 $[0, 1]$ 区间上的连续变量优化问题,并通过遗传算法对所形成的非凸优化问题进行求解. 数值算例表明该方法是可行的. 该方法的主要特点是可以处理一般的约束,得到的解完全满足约束,避免了罚函数法惩罚因子难以确定的困难. 因为遗传算法的全局搜索能力,该方法在求解 0-1 非线性规划问题的时候可以找到问题的全局最优解. 数值试验结果表明当问题的规模比较小时,乘法法和约束松弛法都能很好地得到问题的全局最优解. 当问题规模比较大时,由于约束松弛法是一种近似方法,不能保证

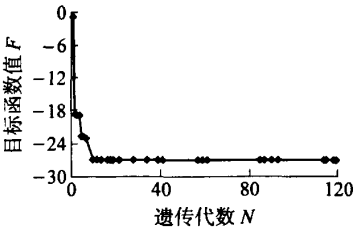


图 4 约束松弛法遗传历史
Fig.4 Genetic history of constraint relaxing

对所有问题都得到问题的全局最优解. 乘子求解方法稳定、有效, 可以应用于目标非线性约束线性的0-1规划问题的计算.

参考文献:

- [1] LENSTRA H W. Integer programming with a fixed number of variables[J]. Mathematics of Operations Research, 1983, 8(4): 538-549.
- [2] 李晓萌, 戴光明, 石红玉. 解决多维0/1背包问题的遗传算法综述[J]. 电脑开发与应用, 2006, 19(1): 4-8.
LI Xiao-meng, DAI Guang-ming, SHI Hong-yu. An introduction to mixed genetic algorithm for solving multi-dimension package problem[J]. Computer Development & Applilations, 2006, 19(1): 4-8. (in Chinese)
- [3] TEMPLEMAN A B, YATES D F. A segmental method for the discrete optimum design of structures[J]. Eng. Opt, 1983(6): 145-155.
- [4] 隋允康. 建模·变换·优化结构综合方法新进展[M]. 大连: 大连理工大学出版社, 1996.
- [5] PARDALOS P M. Continuous approaches to discrete optimization problems[C]//Nonlinear Optimization and Applications. New York: Plenum Publishing, 1996.
- [6] PARDALOS P M, ROMEIJN H E, TUY H. Recent developments and trends in global optimization[J]. Journal of Computational and Applied Mathematics, 2000, 124(1-2): 209-228.
- [7] KIWIEL K C, LINDBERG P O, NOU A. Bregman proximal relaxation of large-scale 0-1 problems[J]. Computational Optimization and Applications, 2000, 15(1): 33-44.
- [8] 李兴斯, 谭涛. 求解二进制二次规划问题的一种连续化方法[J]. 工程数学学报, 2006, 23(3): 499-504.
LI Xing-si, TAN Tao. A continuous approach to binary quadratic problems[J]. Chinese Journal of Engineering Mathematics, 2006, 23(3): 499-504. (in Chinese)
- [9] MICHALEWICZ JANIKOW. GENOCOP: A genetic algorithm for numerical optimization problem with linear constrains[J]. Communications of the ACM, 1992.
- [10] MICHALEWICZ Z, NAZHIYATH G. GENOCOP III: A co-evolutionary algorithm for numerical optimization problems with nonlinear constrains[C]//Proceedings of 1995 IEEE International Conference on Evolutionary Computation. New York: IEEE Neural Network Council, IEEE, 1995: 674-651.

A Continuous Approach to 0-1 Nonlinear Problem and Its Solution With Genetic Algorithm

SUI Yun-kang, JIA Zhi-chao, DU Jia-zheng

(College of Mechanical Engineering and Applied Electronics Technology,
Beijing University of Technology, Beijing 100022, China)

Abstract: To solve the nonlinear 0-1 programming problem, it is transformed into an equivalent continuous nonlinear programming on the domain of $[0, 1]$ by a nonlinear equality discrete constraint. For the 0-1 programming problem with nonlinear object and lineav constraint, the problem with disorete constraint canbe solved with multiplier method. For the problem with conhinean object and nonlinear constraint, the discrete constraint can be relaxed to a non-equality constraint with the constraint relexing method. Both are solved by the Genetic Algorithm GENOCOP. The result of multiplier method is more accurate. Constraint relaxing method is an approximate method, can be used to solve the 0-1 programming problem with nonlinear constraint. Examples are presented and compared with enumerating method, and the results show that the approach is accurate and effective.

Key words: nonlinear 0-1 programming; continuous approach; Genetic Algorithm; GENOCOP

(责任编辑 张士瑛)