

多目标0-1规划的混沌优化算法*

杨玲玲, 马良, 张惠珍

(上海理工大学管理学院, 上海 200093)

摘要: 多目标问题在现实中有广泛应用, 如何获得更多非劣解具有重要意义。通过利用混沌运动的遍历性、随机性、规律性等特点, 设计了一种求解多目标0-1规划问题的混沌优化算法, 并在计算机上予以实现。该混沌优化算法计算时间复杂度较小, 在计算效率上有一定优势。计算结果表明, 与已有算法相比, 该算法能求得较多非劣解, 能够有效求解多目标0-1规划问题。

关键词: 多目标; 混沌; 优化

中图分类号: O22 **文献标志码:** A **文章编号:** 1001-3695(2012)12-4486-03

doi: 10.3969/j.issn.1001-3695.2012.12.021

Chaotic optimization algorithm for multi-objective 0-1 programming problem

YANG Ling-ling, MA Liang, ZHANG Hui-zhen

(College of Management, University of Shanghai for Science & Technology, Shanghai 200093, China)

Abstract: As multi-objective problems boast wild application, how to obtain more Pareto solutions plays a significant importance. By employing the specific features of ergodicity, randomness and regularity of chaotic motion, this paper developed a chaotic optimization algorithm with low time complexity and the advantage of high computational efficiency and implemented on computer for solving the multi-objective 0-1 programming problem. Numerical experiments show that this chaotic optimization algorithm can solve the multi-objective 0-1 programming problem effectively, and can get more Pareto solutions than other algorithms proposed before.

Key words: multi-objective; chaotic; optimization

0 引言

在实际科学问题和工程实践中, 大多为多目标优化问题, 要同时考虑多种因素、且存在多个彼此冲突的目标。多目标问题在现实中应用广泛^[1], 如资本预算^[2]、投资决策、项目选择、生产计划、交通、选址^[3]等。与单目标优化问题不同, 大多数情况下, 多目标问题中, 某目标的改善可能导致其他目标性能降低, 而需在各目标之间进行协调权衡和折中处理, 使所有目标函数尽可能达到最优。这是多目标优化的本质。如何求解这些问题的最优解, 一直是学术界和工程界重要的研究课题^[4]。本文主要讨论多目标0-1线性规划问题。

作为一种普遍的非线性现象, 混沌运动具有遍历性、随机性以及规律性等特点, 其中, 遍历性可以作为搜索过程避免陷入局部极小的有效机制。正是利用混沌的这些特点, 本文提出一种基于Logistic映射的混沌优化算法^[5], 运用到多目标0-1规划问题中, 并进行了大量的算例求解测试。实验结果验证了混沌优化算法对于多目标0-1规划问题的可行性和有效性。

1 数学模型

多目标0-1规划是以一维背包问题的0-1整数规划为原型, 进而发展成多个目标、多个约束, 如实际应用中的投资决策等问题, 都可归结为这类模型。

设变量 x_j 为0-1变量, 本文考虑如下形式的多目标0-1规划问题:

$$\max Z = \{Z_1, Z_2, \dots, Z_k\}$$

其中:

$$\begin{aligned} Z_i &= \sum_{j=1}^n c_{ij}^i x_j \\ \text{s. t. } \begin{cases} g_i(x) = \sum_{j=1}^n a_{ij} x_j \leq b_i & i=1, 2, \dots, m \\ x_j \in \{0, 1\} & j=1, 2, \dots, n \end{cases} \end{aligned}$$

其中: $X = \{x_1, x_2, \dots, x_n\}$ 为决策变量, Z 为目标向量。

由于多目标意义下, 使得所有目标都达到最佳化的最优解往往并不存在, 所以一般所要求的都是非劣解(non-dominated solution), 也称有效解或Pareto解。其含义为: 记当前解为 X_0 , 则不存在任何其他解 X , 使得 $Z_i(X) \geq Z_i(X_0)$, $i=1, 2, \dots, k$, 其中, 至少有一个不等式严格成立。Pareto最优解只是一个可以接受的非劣解, 并且大多数Pareto解的个数很多, 甚至是无穷大。因此, 求解多目标优化问题的关键是求出尽可能多的前沿均匀分布的Pareto解^[6]。

在具体求解时, 采用罚函数的方法, 将约束条件转换到目标函数中, 将问题转换为无约束形式。

令罚函数

$$f = \sum_{i=1}^m \max[0, g_i(x) - b_i]$$

故多目标0-1规划的数学模型可写为

收稿日期: 2012-05-10; **修回日期:** 2012-06-30 **基金项目:** 国家自然科学基金资助项目(70871081); 上海市重点学科建设资助项目(S30504)

作者简介: 杨玲玲(1987-), 女, 硕士研究生, 主要研究方向为智能优化(weenieyang@gmail.com); 马良(1964-), 男, 教授, 博导, 博士, 主要研究方向为智能优化、系统工程; 张惠珍(1979-), 女, 讲师, 博士, 主要研究方向为智能优化、系统工程。

$$\max Z = \{Z_1 - Mf, Z_2 - Mf, \dots, Z_i - Mf, \dots, Z_k - Mf\}$$

其中, M 为惩罚因子。

2 算法设计

Logistic 映射形式为

$$x_{n+1} = ax_n(1 - x_n)$$

其中: $a \in [0, 4]$ 为控制参数, $x_n \in [0, 1]$, $n = 0, 1, 2, \dots$

当 $x_n \in [0, 1]$ 时, 取 $a = 4$, 系统完全处于混沌状态, 利用混沌的初值敏感性, 赋予上式 i 个微小差异的初值即可得到 i 个混沌变量。

基于 Logistic 映射, 本文设计了求解多目标 0-1 规划问题的混沌优化算法, 具体流程如下所述:

a) 确定优化问题的目标函数、约束条件以及混沌优化算法的基本参数:

(a) 对 $x_{n+1} = ax_n(1 - x_n)$, 令 $a = 4$, 设 $v_0 = 0.2027$;

(b) 另设置最大迭代次数 nc , 其中, 最大迭代次数可根据问题的规模进行设定, 问题规模越大, 最大迭代次数也越大。

b) 第 i 次迭代时, 混沌变量根据 $v_i = a * v_{i-1} * (1 - v_{i-1})$ 生成, 可行解 X 按如下方法确定: 若 $v_i < \text{rand}$, 则 $x_j = 1$; 否则 $x_j = 0$ ($j = 1, 2, \dots, n$), 转向步骤 c)。

c) 运用可行解 X 计算目标集函数值, 转向步骤 d)。

d) 与动态记忆库中的每个非劣解比较, 判断新解是否为非劣解, 如果是非劣解, 则将新解放入记忆库中。

判断方法如下:

(a) 如果新解的每个目标函数值比记忆库中的每个值小, 则不添加;

(b) 如果都大于记忆库中的某个解的值, 或者一部分大于, 另一部分等于, 则把该记忆库的解删除。

e) 若迭代次数小于预定最大迭代次数, 则转向步骤 b), 继续进行迭代; 否则, 转向步骤 f)。

f) 输出所得非劣解。

不难估算, 该算法的时间复杂度为 $O(nc \times n)$, 其中, nc 为迭代次数, n 为变量个数。而用于对比的其他方法, 如蚁群算法, 其时间复杂度为 $O(nc \times n^3)$, nc 为迭代次数, n 为蚂蚁个数; 元胞蚁群算法, 其时间复杂度为 $O(nc \times n^2 \times m)$, nc 为迭代次数, m 为蚂蚁个数, n 为顶点个数, 如取 $m \approx n$, 则算法的时间复杂度为 $O(nc \times n^3)^{[7]}$ 。较之于现有方法, 混沌优化算法的时间复杂度更低, 算法效率更高。

3 算例测试

为检验算法效果, 对大量的实例进行了计算测试, 并与已有算法进行比较。实验所用硬件为 Intel Core™ 2 Duo CPU T7100 1.80 GHz, 内存 2 GB, 软件为 Windows 7 操作系统和 Eclipse 开发环境。下面给出五个算例。

算例 1

$$\begin{aligned} \max Z_1 &= 7x_1 + 3x_2 + 71x_3 \\ \max Z_2 &= 4x_1 + 2x_2 + 2x_3 \\ \text{s. t. } &\begin{cases} 5x_1 + 2x_2 + 3x_3 \leq 7 \\ 4x_1 + 2x_2 + 2x_3 \leq 6 \\ x_i \in \{0, 1\}, i = 1, 2, 3 \end{cases} \end{aligned}$$

运用混沌优化算法, 得到非劣解:

$$Z = (10, 6), X = (1, 1, 0)$$

算例 1 结果对照如表 1 所示。

表 1 算例 1 结果对照

	算法		
	HZG ^[8]	元胞蚁群 ^[9]	混沌优化
结果	(10, 6)	(10, 6)	(10, 6)

算例 2

$$\max Z_1 = 10x_1 + 14x_2 + 21x_3 + 42x_4$$

$$\max Z_2 = 30x_1 + 15x_2 + 20x_3 + 18x_4$$

$$\text{s. t. } \begin{cases} 8x_1 + 11x_2 + 9x_3 + 18x_4 \leq 34 \\ 2x_1 + 2x_2 + 7x_3 + 14x_4 \leq 11 \\ 9x_1 + 6x_2 + 3x_3 + 6x_4 \leq 14 \\ x_i \in \{0, 1\}, i = 1, 2, 3, 4 \end{cases}$$

运行混沌优化算法, 得到非劣解分别为:

a) $Z = (35, 35)$, $X = (0, 1, 1, 0)$;

b) $Z = (31, 50)$, $X = (1, 0, 1, 0)$ 。

算例 2 结果对照如表 2 所示。

表 2 算例 2 结果对照

	算法			
	遗传 ^[10]	蚂蚁 ^[11]	自适应粒子群 ^[9]	蜂群 ^[13]
结果	(31, 50)	(31, 50)	(31, 50)	(31, 50)
	(35, 35)	(35, 35)	(35, 35)	(35, 35)

算例 3

$$\min Z_1 = 4x_1 + 5x_2 + 3.5x_3 + 4x_4$$

$$\max Z_2 = 3x_1 + 4x_2 + 2.5x_3 + 3.5x_4$$

$$\text{s. t. } \begin{cases} 4x_1 + 5x_2 + 3.5x_3 + 4x_4 \leq 10 \\ x_i \in \{0, 1\}, i = 1, 2, 3, 4 \end{cases}$$

运行混沌优化算法, 得到非劣解分别为:

a) $Z = (0, 0)$, $X = (0, 0, 0, 0)$;

b) $Z = (4, 3.5)$, $X = (0, 0, 0, 1)$;

c) $Z = (5, 4)$, $X = (0, 1, 0, 0)$;

d) $Z = (7.5, 6)$, $X = (0, 0, 1, 1)$;

e) $Z = (8, 6.5)$, $X = (1, 0, 0, 1)$;

f) $Z = (9, 7.5)$, $X = (0, 1, 0, 1)$ 。

算例 3 结果对照表如表 3 所示。

表 3 算例 3 结果对照表

算法	分目标乘除法 ^[14]	功效系数法 ^[14]	评价函数法 ^[14]	元胞蚁群算法 ^[9]	混沌优化算法
结果	(9, 7.5)	(8, 6.5)	(9, 7.5)	(0, 0)	(0, 0)
				(4, 3.5)	(4, 3.5)
				(5, 4)	(5, 4)
				(7.5, 6)	(7.5, 6)
				(8, 6.5)	(8, 6.5)
				(9, 7.5)	(9, 7.5)

算例 4

$$\max Z_1 = 4x_3 + 5x_4 + 7x_7 + 6x_8 + 5x_9 + 6x_{10}$$

$$\max Z_2 = 9x_1 + 4x_2 + 6x_5 + 9x_6 + 6x_7 + 5x_8 + 6x_9 + 3x_{10}$$

$$\max Z_3 = 6x_1 + 3x_2 + 2x_3 + 8x_4 + 7x_5 + 2x_8$$

约束条件:

$$g_1(x) = 3x_2 + 4x_3 + 8x_4 + 4x_5 + 7x_6 + 6x_7 + 5x_8 + 2x_9 + 6x_{10} \leq 37$$

$$g_2(x) = 5x_1 + 3x_2 + 6x_4 + 8x_6 + 5x_7 + 6x_8 + 3x_9 + 5x_{10} \leq 24$$

$$g_3(x) = 2x_1 + 2x_2 + 7x_3 + 4x_4 + 8x_5 + 9x_6 + 9x_7 + 3x_8 + 7x_9 + 5x_{10} \leq 43$$

$$g_4(x) = 2x_1 + 3x_2 + 3x_3 + 6x_4 + 3x_5 + 8x_7 + 4x_8 + 6x_9 \leq 31$$

$$x_i \in \{0, 1\}, i = 1, 2, 3, 4, \dots, 10$$

运行混沌优化算法, 得到非劣解分别为:

- a) $Z = (12, 40, 60)$, $X = (1, 1, 0, 0, 1, 1, 1, 0, 1, 0)$;
 b) $Z = (15, 37, 18)$, $X = (1, 1, 1, 0, 1, 1, 0, 0, 1, 1)$;
 c) $Z = (20, 30, 28)$, $X = (1, 1, 1, 1, 1, 0, 0, 1, 1, 0)$;
 d) $Z = (21, 31, 26)$, $X = (1, 1, 1, 0, 1, 0, 1, 1, 1, 0)$;
 e) $Z = (22, 28, 26)$, $X = (1, 1, 1, 1, 1, 0, 1, 0, 0, 1)$;
 f) $Z = (22, 36, 20)$, $X = (1, 1, 1, 0, 1, 0, 1, 1, 1, 0)$;
 g) $Z = (23, 33, 20)$, $X = (1, 1, 1, 0, 1, 0, 1, 1, 0, 1)$;
 h) $Z = (27, 30, 23)$, $X = (1, 0, 1, 1, 1, 0, 1, 0, 1, 1)$;
 i) $Z = (28, 20, 19)$, $X = (0, 0, 1, 1, 1, 0, 1, 1, 0, 1)$;
 j) $Z = (28, 35, 17)$, $X = (1, 0, 1, 0, 1, 0, 1, 1, 1, 1)$ 。

算例 4 结果对照表如表 4 所示。

表 4 算例 4 结果对照表

	算法			
	遗传 ^[10]	蚂蚁 ^[13]	元胞蚁群 ^[9]	混沌优化
结果	(20, 30, 28)	(12, 40, 60)	(12, 40, 60)	(12, 40, 60)
	(21, 23, 25)	(20, 30, 28)	(20, 38, 28)	(15, 37, 18)
	(21, 29, 17)	(21, 23, 25)	(21, 31, 26)	(20, 38, 28)
	(22, 21, 22)	(21, 29, 17)	(22, 28, 26)	(21, 31, 26)
		(22, 21, 22)	(22, 36, 20)	(22, 28, 26)
		(28, 20, 19)	(23, 33, 20)	(22, 36, 20)
		(28, 27, 17)	(28, 20, 19)	(23, 33, 20)
		(28, 30, 14)	(28, 35, 17)	(27, 30, 23)
				(28, 20, 19)
				(28, 35, 17)

算例 5

$$\begin{aligned} \max Z_1 &= 89x_1 + x_2 + 62x_3 + 43x_4 + 6x_5 + 34x_6 + \\ &\quad 28x_7 + 29x_8 + 38x_9 + 40x_{10} + 29x_{11} + 44x_{12} \\ \max Z_2 &= 59x_1 + 48x_2 + 30x_3 + 33x_4 + 88x_5 + 73x_6 + \\ &\quad 71x_7 + 11x_8 + 59x_9 + 86x_{10} + 17x_{11} + 90x_{12} \\ \max Z_3 &= 19x_1 + 75x_2 + x_3 + 9x_4 + 90x_5 + 36x_6 + \\ &\quad 68x_7 + 49x_8 + 32x_9 + 53x_{10} + 17x_{11} + 24x_{12} \end{aligned}$$

约束条件:

$$\begin{aligned} g_1(x) &= 63x_1 + 70x_2 + 28x_3 + 99x_4 + 67x_5 + 41x_6 + \\ &\quad 75x_7 + 3x_8 + 28x_9 + 96x_{10} + 8x_{11} + 30x_{12} \leq 304 \\ g_2(x) &= 91x_1 + 77x_2 + 74x_3 + 41x_4 + 57x_5 + 95x_6 + \\ &\quad 41x_7 + 47x_8 + 83x_9 + 37x_{10} + 6x_{11} + 86x_{12} \leq 367.5 \\ g_3(x) &= 23x_1 + 89x_2 + 83x_3 + 43x_4 + 24x_5 + 66x_6 + \\ &\quad 60x_7 + 33x_8 + 59x_9 + 45x_{10} + 83x_{11} + 71x_{12} \leq 339.5 \\ x_i &\in \{0, 1\}, i = 1, 2, 3, 4, \dots, 11, 12 \end{aligned}$$

运用混沌优化算法共求得 28 个非劣解, 如表 5 所示。

通过以上测试, 表明本文算法的有效性, 可以搜索出更多 Pareto 最优解, 且由此方法求得的 Pareto 最优解优于遗传算法和蚁群算法所求得的 Pareto 最优解。并且, 由于算法的时间复杂度较低, 就测试的情况而言, 算法的运行时间基本可以忽略不计。

4 结束语

多目标规划在现实生活中有着非常重要的意义, 其中多目标 0-1 规划更是投资决策等问题中不可缺少的原始模型。离散型多目标 0-1 规划与连续型多目标函数优化不同, 尽管有时这些优化问题的解向量表现为一条连续的曲面, 但实际上它们只是离散的, 有时还会是孤立的解点, 这些都会增加算法完成任务的难度。本文所给出的混沌优化算法对于多目标 0-1 线性规划, 可以是一种简单而又快速的获得问题有效解的途径。

表 5 算例 5 计算结果

	目标集(Z_1, Z_2, Z_3)	非劣解 X
1	(182, 294, 318)	(1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 1, 0)
2	(185, 405, 316)	(0, 0, 0, 0, 1, 0, 1, 1, 1, 1, 0, 1)
3	(203, 345, 293)	(0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0)
4	(209, 376, 285)	(0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1)
5	(215, 319, 279)	(1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 1, 0)
6	(219, 305, 275)	(1, 0, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0)
7	(225, 336, 267)	(1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1)
8	(227, 334, 264)	(1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 0)
9	(231, 320, 260)	(1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0)
10	(234, 384, 250)	(1, 0, 0, 0, 1, 0, 1, 0, 1, 0, 1, 1)
11	(237, 351, 252)	(1, 0, 0, 0, 1, 0, 0, 1, 0, 1, 1, 1)
12	(243, 276, 244)	(1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 1, 0)
13	(246, 399, 235)	(1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1, 1)
14	(248, 338, 231)	(1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0)
15	(249, 317, 242)	(1, 0, 0, 0, 0, 1, 1, 1, 0, 1, 1, 0)
16	(253, 303, 238)	(1, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0)
17	(259, 334, 230)	(1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 1, 1)
18	(260, 353, 216)	(1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 0)
19	(264, 339, 212)	(1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 1, 0)
20	(270, 370, 204)	(1, 0, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1)
21	(271, 329, 169)	(1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1)
22	(273, 317, 160)	(1, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, 1)
23	(274, 296, 171)	(1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 1, 1)
24	(277, 274, 207)	(1, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 0)
25	(283, 276, 175)	(1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0)
26	(287, 262, 171)	(1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0)
27	(293, 293, 163)	(1, 0, 1, 0, 0, 0, 0, 1, 0, 1, 1, 1)
28	(296, 240, 119)	(1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1)

参考文献:

- [1] CHENG Zhen, HUANG Yu-fang, CHEN Zhi-hua. Solving the 0-1 multi-objective knapsack problem using self-assembly of DNA tiles [J]. *Bio-inspired Computing*, 2009, 16(17): 137-145.
- [2] ROSENBLATT M J, SINUANY-STERN Z. Generating the discrete efficient frontier to the capital budgeting problem [J]. *Operations Research*, 1989, 37(3): 384-394.
- [3] YANO H, SAKAWA M, SHIBANO T. Fuzzy multiobjective 0-1 programming through revised genetic algorithms [J]. *Knowledge-Based Intelligent Electronic Systems*, 1998, 21(23): 101-108.
- [4] 雷德明, 严新平. 多目标智能优化算法及其应用 [M]. 北京: 科学出版社, 2009.
- [5] 李兵, 蒋慰孙. 混沌优化方法及其应用 [J]. *控制理论与应用*, 1997, 14(4): 613-615.
- [6] 熊小华, 宁爱兵, 马良. 多目标 0-1 背包问题的元胞竞争决策算法 [J]. *计算机应用研究*, 2010, 27(10): 3680-3682.
- [7] 马良, 朱刚, 宁爱兵. 蚁群优化算法 [M]. 北京: 科学出版社, 2008.
- [8] 李毅华, 凌凌. HZG 算法——大型 0-1 启发式目标规划 [J]. *系统工程理论与实践*, 1986, 9(3): 72-74.
- [9] 刘勇, 马良, 许秋艳. 多目标 0-1 规划问题的元胞蚁群优化算法 [J]. *系统工程*, 2009, 27(2): 119-122.
- [10] 马良. 多目标投资决策模型的进化算法 [J]. *上海理工大学学报*, 1998, 20(1): 56-59.
- [11] 崔雪丽, 马良. 多目标 0-1 规划的蚂蚁优化算法 [J]. *计算机应用与软件*, 2007, 24(7): 23-24.
- [12] 孙滢, 高岳林. 一种求解多目标 0-1 规划问题的自适应粒子群算法 [J]. *计算机应用与软件*, 2009, 26(12): 71-72.
- [13] 樊小毛, 马良. 多目标 0-1 规划问题的蜂群算法 [J]. *微计算机信息*, 2010, 26(10): 207-209.
- [14] 冯爱芬, 杨森. 一类农业生产问题的多目标规划类型及其解法 [J]. *黄冈师范学院学报*, 2004, 24(3): 32-34, 74.