一类非凸优化问题的遗传算法

叶成绪[1],李和成[2,3]
YE Cheng-xu[1],LI He-cheng[2,3]

1.青海师范大学 计算机系,西宁 810008
2.西安电子科技大学 计算机学院,西安 710071
3.青海师范大学 数学与信息科学系,西宁 810008
1.Department of Computer Science,Qinghai Normal University,Xining 810008,China
2.School of Computer Science and Technology,Xidian University,Xi'an 710071,China
3.Department of Mathematics and Information Science,Qinghai Normal University,Xining 810008,China
E-mail:hclwxy@yahoo.com.cn

YE Cheng-xu,LI He-cheng.Genetic algorithm for a class of nonconvex optimization problems.Computer Engineering and Applications,2009,45(24):60-62.

Abstract: Linear bilevel programming problem is a special class of non-convex optimization problems,in order to solve the problems efficiently,a new genetic algorithm based on the simplex method is proposed.At first,a novel encoding scheme is designed based on the lower-level constraints.Then,the solutions to the lower-level problem,as functions of upper-level variables, are got by using the information in the simplex tableau,and a new fitness is given by combining these solution functions with optimality conditions.At last,new genetic operators are designed according to the characteristics of individuals.The numerical results illustrate that the proposed algorithm is feasible and efficient.
Key words: nonconvex optimization problems;linear bilevel programming;genetic algorithm;simplex method;optimal solutions

摘 要: 线性二层规划是一类特殊的非凸优化问题,为了有效求解该问题,提出了一种基于单纯形方法的遗传算法。首先基于下层约束给出了一种新的编码方法;其次利用单纯形表的信息得到了下层问题的解函数,并结合最优性条件给出了适应度函数;最后基于个体编码的特点,设计了新的遗传算子。数值结果表明,所提出的算法是可行有效的。
关键词: 非凸优化问题;线性二层规划;遗传算法;单纯形方法;最优解
DOI: 10.3778/j.issn.1002-8331.2009.24.019 **文章编号:** 1002-8331(2009)24-0060-03 **文献标识码:** A **中图分类号:** TP18

1 引言

具有递阶结构的线性二层规划是一类复杂的系统优化问题,它包含一个上层优化问题和一个或多个下层优化问题[1-2]。在这类问题中,决策变量分为两个集合,即上层变量和下层变量,它们分别由上下层决策者控制。在一个二层规划问题中,上层问题和下层问题都有各自的目标函数和约束条件,上层问题的目标函数和约束条件不仅与上层决策变量有关,而且还依赖于下层变量和下层问题的最优解,同时下层问题的最优解则受上层决策变量的影响。这类问题的决策机制如下:首先上层决策者选定一个决策 x,下层决策者在给定上层决策的前提下,选择一个最有利于自己的决策 y;其次上层再调整自己的决策,使得自己的目标达到最优。二层规划是工程领域的一个重要问题,在经济、工程技术、社会政治和军事指挥等领域均有广泛的应用背景[2]。但由于内在的递阶结构,对二层规划问题的求解一般非常困难[3-4],主要体现在两个方面:一是可行点获得困难。要获得一个二层问题的可行点,必须对给定的上层变量求一次下层优化问题,这往往会加大算法的计算量。二是这种嵌套的结构使得问题成为一个非凸优化问题,这给基于梯度的传统优化方法带来了极大的挑战,一般很难找到问题的全局最优解。目前对二层规划问题的求解大致可归为两种途径:一是利用线性二层规划问题的最优解可以在约束域极点上达到的性质设计算法,代表算法有枚举法和 k 次最好算法(Kth-Best);二是利用 K-K-T 条件将问题化为一个单层问题求解,代表算法有分支定界法和罚函数方法[4]。其中第一类算法适合求解小规模问题,对大规模问题很难穷举所有的极点。而第二类算法尽管对大规模问题的处理较第一类算法好,但基于 K-K-T 条件的变形并不改变问题的非凸性,同时由于引入了 Lagrange 乘子而使变量个数大量增加,这不利于算法搜索全局最优解。为了克服这些不足,遗传算法等智能计算方法被用于求解该类问题[5-9]。其中文[6]将遗传算法与极点枚举法结合起来,克服了纯粹枚举法的不足,但该算法需要对所得的点求解下层问题,因此当下层问题规模较大时,大的计算量不可避免。文[7]基于

作者简介: 叶成绪(1970-),男,副教授,研究方向:智能计算方法、移动网络、网络安全等;李和成,男,博士生,副教授,研究方向:进化计算,最优化理论与算法等。
收稿日期: 2008-06-25 **修回日期:** 2008-09-08

K-K-T 条件设计了遗传算法，利用遗传算法的全局收敛性克服了传统优化方法很难找到全局最优解的不足。但无法处理由于 Lagrange 乘子的引入而使变量个数增加的问题。为了克服目前算法的不足，从单纯形表中的信息入手，基于最优性条件设计了一个新的遗传算法。该算法一方面保留了极点枚举的优势，缩小了搜索空间，而另一方面没有引入 Lagrange 乘子，从而保持决策变量数不变。该文的工作主要体现在以下几个方面：首先基于下层约束给出了一种新的编码方法；其次对于给定的一个个体，利用单纯形表的信息给出了下层问题的解函数和最优性条件，并利用这些解函数和最优性条件得到了一个只与上层变量有关的单层问题，该单层问题的最优值作为个体的适应度值；最后基于个体编码的特点，设计了新的遗传算子。数值结果表明，所提出的算法是可行有效的。

2 线性二层规划问题

一般的线性二层规划问题可表示为如下形式：

$$\begin{cases}\min\limits_{x\in X} \boldsymbol{c}_1\boldsymbol{x}+\boldsymbol{d}_1\boldsymbol{y}\\ \text{s.t. } \boldsymbol{A}_1\boldsymbol{x}+\boldsymbol{B}_1\boldsymbol{y}\leqslant\boldsymbol{b}_1,\text{其中 }\boldsymbol{y}\text{ 求解}\\ \min\limits_{y\in Y} \boldsymbol{c}_2\boldsymbol{x}+\boldsymbol{d}_2\boldsymbol{y}\\ \text{s.t. } \boldsymbol{A}_2\boldsymbol{x}+\boldsymbol{B}_2\boldsymbol{y}\leqslant\boldsymbol{b}_2\end{cases}\tag{1}$$

其中：$\boldsymbol{x}\in\boldsymbol{R}^n,\boldsymbol{y}\in\boldsymbol{R}^m;\boldsymbol{c}_i^{\mathrm{T}}\in\boldsymbol{R}^n,\boldsymbol{d}_i^{\mathrm{T}}\in\boldsymbol{R}^m,i=1,2;\boldsymbol{b}_1\in\boldsymbol{R}^p,\boldsymbol{b}_2\in\boldsymbol{R}^q$，且 $\boldsymbol{A}_i$ 和 $\boldsymbol{B}_i$ 为相应阶的实数矩阵。$\boldsymbol{X}$、$\boldsymbol{Y}$ 一般为其他约束，如上下界约束、整数约束等，这里讨论界约束的情况。

一些相关概念定义如下[1]：

（1）搜索空间：$\boldsymbol{\Omega}=\{(\boldsymbol{x},\boldsymbol{y})|\boldsymbol{x}\in\boldsymbol{X},\boldsymbol{y}\in\boldsymbol{Y}\}$；

（2）约束域：$\boldsymbol{S}=\{(\boldsymbol{x},\boldsymbol{y})\in\boldsymbol{\Omega}|\boldsymbol{A}_i\boldsymbol{x}+\boldsymbol{B}_i\boldsymbol{y}\leqslant\boldsymbol{b}_i,i=1,2\}$；

（3）对固定的 $\boldsymbol{x}$，下层问题的可行域为：$\boldsymbol{S}(\boldsymbol{x})=\{\boldsymbol{y}\in\boldsymbol{Y}|\boldsymbol{A}_2\boldsymbol{x}+\boldsymbol{B}_2\boldsymbol{y}\leqslant\boldsymbol{b}_2\}$；

（4）$\boldsymbol{S}$ 在上层决策空间的投影：$\boldsymbol{S}(\boldsymbol{X})=\{\boldsymbol{x}\in\boldsymbol{X}|\exists\boldsymbol{y}\in\boldsymbol{Y},(\boldsymbol{x},\boldsymbol{y})\in\boldsymbol{S}\}$；

（5）对每一个 $\boldsymbol{x}\in\boldsymbol{X}$，下层合理反应集：$\boldsymbol{P}(\boldsymbol{x})=\{\boldsymbol{y}\in\boldsymbol{Y}|\boldsymbol{y}\in\arg\min[f(\boldsymbol{x},\boldsymbol{v}):\boldsymbol{v}\in\boldsymbol{S}(\boldsymbol{x})]\}$；

（6）诱导域 $IR=\{(\boldsymbol{x},\boldsymbol{y})|(\boldsymbol{x},\boldsymbol{y})\in\boldsymbol{S},\boldsymbol{y}\in\boldsymbol{P}(\boldsymbol{x})\}$。

为了使（1）有意义，通常假设 $\boldsymbol{S}$ 为非空紧集，且对于上层的决策 $\boldsymbol{x}$，有 $\boldsymbol{P}(\boldsymbol{x})\neq\phi$。诱导域中的点即为双层规划问题（1）的可行点。事实上，问题（1）也可以写为：

$$\min\{\boldsymbol{c}_1\boldsymbol{x}+\boldsymbol{d}_1\boldsymbol{y}|(\boldsymbol{x},\boldsymbol{y})\in\boldsymbol{IR}\}$$

为了讨论方便，进行如下处理：

①取 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 为非负性约束集；

②在问题（1）的下层中加入松弛变量 $\boldsymbol{y}_0$，使得下层约束变为等式。不失一般性，下层变量仍表示为 $\boldsymbol{y}\in\boldsymbol{R}^m$，且相关系数仍然用原符号表示；

③求解下层问题时，上层变量 $\boldsymbol{x}$ 是常数，因此可省去下层目标中的 $\boldsymbol{c}_2\boldsymbol{x}$。

则问题（1）变为：

$$\begin{cases}\min\limits_{x\geqslant 0} \boldsymbol{c}_1\boldsymbol{x}+\boldsymbol{d}_1\boldsymbol{y}\\ \text{s.t. } \boldsymbol{A}_1\boldsymbol{x}+\boldsymbol{B}_1\boldsymbol{y}\leqslant\boldsymbol{b}_1,\text{其中 }\boldsymbol{y}\text{ 求解}\\ \min\limits_{y\geqslant 0} \boldsymbol{d}_2\boldsymbol{y}\\ \text{s.t. } \boldsymbol{A}_2\boldsymbol{x}+\boldsymbol{B}_2\boldsymbol{y}=\boldsymbol{b}_2\end{cases}\tag{2}$$

值得指出的是，①并不改变问题（1）的一般性，因为对于一个满足界约束的变量，总可以通过简单的变量替换使之满足非负性约束。由此求解问题（2）并不失一般性。另外，为了讨论简便，进一步假设：对于每一个 $\boldsymbol{x}\in\boldsymbol{S}(\boldsymbol{X})$，$\boldsymbol{S}(\boldsymbol{x})$ 有界，下层问题有唯一的最优解，且 $\boldsymbol{B}_2$ 行满秩。

3 算法设计

遗传算法能有效求解各类复杂的非凸优化问题，根据问题（2）的特点，给出了个体的编码和适应度函数，并根据编码方式设计了杂交和变异算子。

3.1 个体编码

因为 $\boldsymbol{B}_2$ 行满秩，令 $Rank(\boldsymbol{B}_2)=q$。从 $\boldsymbol{B}_2$ 中抽取 q 列，组成一个子矩阵 $\boldsymbol{B}$，同时这 q 列的列序号记为 $i_1,i_2,\cdots,i_q$（满足 $1\leqslant i_1<i_2<\cdots<i_q\leqslant m$）；取 $(i_1,i_2,\cdots,i_q)$ 作为一个种群个体，即以可能构成下层线性规划基的列序号表示种群的个体。

3.2 适应度函数

对于个体 $(i_1,i_2,\cdots,i_q)$，令对应列构成的矩阵为 $\boldsymbol{B}$。若矩阵 $\boldsymbol{B}$ 不可逆，则记该个体的适应度为充分大的一个正数 M；若矩阵 $\boldsymbol{B}$ 可逆，由单纯形算法可知，若存在 $\boldsymbol{x}\in\boldsymbol{S}(\boldsymbol{X})$，使得：

$$\begin{cases}\boldsymbol{B}^{-1}\boldsymbol{b}(\boldsymbol{x})\geqslant 0\\ \boldsymbol{d}_2-\boldsymbol{d}_2^{B}\boldsymbol{B}^{-1}\boldsymbol{B}_2\geqslant 0\end{cases}\tag{3}$$

其中 $\boldsymbol{b}(\boldsymbol{x})=\boldsymbol{b}_2-\boldsymbol{A}_2x$，$\boldsymbol{d}_2^B$ 为 $\boldsymbol{d}_2$ 中基变量对应的分量。则称 $\boldsymbol{B}$ 为关于 $\boldsymbol{x}$ 的下层最优可行基。此时对于 $\boldsymbol{x}\in\boldsymbol{S}(\boldsymbol{X})$，下层最优解 $\boldsymbol{y}(\boldsymbol{x})$ 的基变量值为 $\boldsymbol{B}^{-1}\boldsymbol{b}(\boldsymbol{x})$，非基变量值为 0；显然 $\boldsymbol{y}(\boldsymbol{x})$ 是 $\boldsymbol{x}$ 的线性函数。另外，注意到（3）的第二个不等式不含 $\boldsymbol{x}$，因此可提前验证该式。若不成立，则取该个体的适应度为 M；若成立，则进行如下处理：将 $\boldsymbol{y}=\boldsymbol{y}(\boldsymbol{x})$ 代入问题（2）的上层函数中，消去 $\boldsymbol{y}$；并用（3）的第一个不等式替换下层问题，得到如下单层规划问题：

$$\begin{cases}\min\limits_{x\geqslant 0} \boldsymbol{c}_1\boldsymbol{x}+\boldsymbol{d}_1\boldsymbol{y}(\boldsymbol{x})\\ \text{s.t. } \boldsymbol{A}_1\boldsymbol{x}+\boldsymbol{B}_1\boldsymbol{y}(\boldsymbol{x})\leqslant\boldsymbol{b}_1\\ \boldsymbol{B}^{-1}\boldsymbol{b}(\boldsymbol{x})\geqslant 0\end{cases}\tag{4}$$

将（4）的目标值作为个体的适应度值。

3.3 杂交算子

设 $l=(i_1,i_2,\cdots,i_q)$ 是一个待杂交的个体。令 $Q=\{1,2,\cdots,m\}$，$Q_l=\{i_1,i_2,\cdots,i_q\}$，且 $\bar{Q}_l=Q\backslash Q_l$。从 Q_l 中任意删除一个元素，并在 $\bar{Q}_l$ 中任意取一个元素补充到 Q_l 中，重新从小到大排序得到一个新的个体 l'，记为 l 的杂交后代。

从杂交过程不难看出，当个体所对应的矩阵满足（3）时，近似相当于单纯形迭代，即在下层可行域的相邻顶点上产生下一代个体，这有助于局部搜索。

3.4 变异算子

设 $l=(i_1,i_2,\cdots,i_q)$ 是一个待变异的个体，记 $\Delta l_j=i_{j+1}-i_j,j=0,1,2,\cdots,q$，其中 $i_0=0,i_{q+1}=m$；$r\in[0,1]$ 是一个随机数，任取非负整数 $\delta_j\in[0,\Delta l_j),j=0,1,\cdots,q-1,\delta_q\in[0,\Delta l_q]$。按如下方式产生变异后代 $\tilde{l}$：

（1）若 $r\leqslant 0.5$，则 $\tilde{l}=(i_1-\delta_0,i_1+\delta_1,\cdots,i_{q-1}+\delta_{q-1})$，称之为前向变异；

（2）若 $r>0.5$，则 $\tilde{l}=(i_1+\delta_1,i_2+\delta_2,\cdots,i_q+\delta_q)$，称之为后向变异；

例如，当 $m=15,q=6$ 时，l：$\boxed{1}\boxed{3}\boxed{4}\boxed{6}\boxed{7}\boxed{9}$

当 $r\leqslant 0.5$ 时，一个变异后代为：$\tilde{l}$：$\boxed{1}\boxed{2}\boxed{3}\boxed{5}\boxed{6}\boxed{8}$

当 r>0.5 时,一个变异后代为:$\tilde{l}$: | 2 | 3 | 5 | 6 | 8 | 13 |

从变异结果可以看出,变异后代与原个体的差异较大,从而有助于全局搜索。

4 提出的算法

基于单纯形方法的遗传算法(Simplex Method Based Genetic Algorithm,SMGA):

步骤 1 初始种群:随机生成 N 个均匀分布的个体 $l_i,i=1,2,\cdots,N$,构成规模为 N 的初始种群 $pop(0)$。计算每个个体的适应度值。令 k=0;

步骤 2 杂交:按杂交概率为 p_c 从 $pop(k)$中抽取杂交个体 l_i,用杂交算子作用,产生杂交后代 l_i';

步骤 3 变异:对 $pop(k)$中每个个体 l_i,按变异概率 p_m 进行变异,得到变异后代 $\tilde{l}_i$;

步骤 4 选择:从当前种群和杂交变异后代集中选择最好的 N_1 个个体,并在剩下的个体中随机选择 $N-N_1$ 个个体组成下一代种群 $pop(k+1)$;

步骤 5 循环:如果终止条件成立,则停止;否则,令 $k=k+1$,转步骤 2。

5 数值实例

从文[3-4,8-9]中选择了 5 个广泛被采用的测试函数,分别记为 T1~T5。将这几个测试问题写成二层规划问题(2)的形式后,利用 SMGA 求解。

算法所采用的参数取值如下:$N=5,p_c=0.8,p_m=0.2,N_1=2,M=10\ 000$;最大代数为 10 代;数据中,绝对值小于 10^{-10} 的计为 0;在 Pentium IV-2.66 GHz 的 PC 机上对每个测试问题独立运行 30 次,记录所得的最好和最差目标函数值,并计算 30 次运算所得的目标函数值的平均值、中值以及算法运行 10 代所需的平均 CPU 时间(简记为 CPU)。所有数据见表 1、表 2,其中表 1 列出了算法所需的平均 CPU 时间,以及算法所得上层目标函数的最好、最差值和中值,并计算平均值和标准差。表中最后一列是文献的最好计算结果。表 2 给出了最优解的位置,并与文献结果进行了比较。表中,序号带"*"表示极大化模型,Ref 表示相应文献中的算法。

表 1 SMGA 运算 30 次的结果与文献结果的比较

序号	CPU/s	SMGA-$F(x^*,y^*)$					Ref-$F(x^*,y^*)$
		最好值	平均值	中值	最差值	标准差	
T1[3]	0.61	-8	-8	-8	-8	0	-8
T2[4]	0.31	32	32	32	32	0	**31.999 8**
T3[8]	0.47	**-29.200 0**	**-29.200 0**	**-29.200 0**	**-29.200 0**	0	-29.198 0
T4[8]	0.21	-3.111 1	-3.111 1	-3.111 1	-3.111 1	0	-3.111 1
T5[9]*	0.35	**85.090 9**	**85.090 9**	**85.090 9**	**85.090 9**	0	85.085 5

表 2 SMGA 找到的最好解与文献结果比较

序号	最好解	
	SMGA	Ref
T1[3]	(14,8)	(14,8)
T2[4]	(2,6)	(2.000 2,5.999 9)
T3[8]	(0,0.9,0,0.6,0.4)	(0,0.9,0,0.6,0.399 9)
T4[8]	(0.888 9,2.222 2)	(0.888 9,2.222 2)
T5[9]*	(17.454 5,10.909 1)	(17.450 0,10.908 0)

从表 1 可以看出,除 T2 外,SMGA 对其他实例的结果等于或优于文献中给出的结果。对于 T5,SMGA 所得的最差结果也明显好于相应文献的结果。说明文献中的算法并没有找到这个问题的全局最优解;对于 T2,文献中的算法给出的近似最好值似乎好于 SMGA,但从最好解的位置可以看到,两者仅仅是由计算误差的不同而引起的差别。值得指出的是,SMGA 在 30 次的运算结果基本一致,最大偏差均小于 10^{-10} 数量级,说明 SMGA 对这些问题的求解是鲁棒的。

另外,表 1 给出了 SMGA 所需的平均 CPU 时间。从所列的值不难看出,SMGA 所需的 CPU 时间很少,这表明 SMGA 能快速地搜索到最优解。

6 结束语

利用线性二层规划问题的特点,提出了基于单纯形方法的遗传算法(SMGA)。该算法的优点在于将极点枚举法和最优性条件结合起来,在缩小搜索空间的前提下,没有引入新的变量,这极大地简少了计算量。另外,为了提高算法求解问题的效率,设计了新的遗传算子,提高了算法的局部和全局搜索能力。数据实验结果表明,提出的遗传算法是可行有效的。

参考文献:

[1] Bard J F.Practical bilevel optimization[M].The Netherlands:Kluwer Academic Publishers,1998.

[2] Colson B,Marcotte P,Savard G.Bilevel programming:A survey[J].A Quarterly Journal of Operations Research(4OR),2005,3(2):87-107.

[3] 赵茂先,高自友.求解线性双层规划的割平面算法[J].北方交通大学学报,2005,29(3):65-69.

[4] Zhu Dao-li,Xu Qing,Lin Zheng-hua.A homotopy method for solving bilevel programming problem[J].Nonlinear Analysis,2004,57(7/8):917-928.

[5] Wang Guang-min.Genetic algorithms for solving linear bilevel programming[C]//Shen Hong,Nakano K.Proceedings of the Sixth International Conference on Parallel and Distributed Computing,Applications and Technologies(PDCAT'05).Los Alamitos,CA:IEEE Computer Society Press,2005:920-924.

[6] Calvete H I,Gale C,Mateo P M.A new approach for solving linear bilevel problems using genetic algorithm[J].European Journal of Operational Research,2008,188(1):14-28.

[7] Hejazi S R,Memariani A,Jahanshahloo G,et al.Linear bilevel programming solution by genetic algorithm[J].Computers and Operations Research,2002,29(13):1913-1925.

[8] Zhu Xiao-bo,Yu Qian,Wang Xian-jia.A hybrid differential evolution algorithm for solving nonlinear bilevel programming with linear constraints[C]//Proc 5th IEEE Int Conf on Cognitive Informatics(ICCI'06),2006:126-131.

[9] Lan Kuen-Ming,Wen Ue-Pyng,Shih Hsu-Shih,et al.A hybrid neural network approach to bilevel programming problems[J].Applied Mathematics Letters,2007,20(8):880-884.