

二阶随机占优约束优化问题的遗传算法求解

张宏伟^{*1}, 于海生¹, 庞丽萍¹, 王金鹤²

(1. 大连理工大学 数学科学学院, 辽宁 大连 116024;

2. 青岛理工大学 计算机系, 山东 青岛 266033)

摘要: 随机占优是经济学和决策论中的基本概念,在投资组合优化中得到了广泛的应用. 遗传算法无须求解目标函数和约束函数的次微分,也不用满足 Slater 约束规范,解决了约束的半无限性和非光滑性等问题. 两个算例表明,遗传算法能很好地解决投资组合优化问题,并且效率得到了很大提高.

关键词: 二阶随机占优;遗传算法;投资组合优化

中图分类号: O224

文献标识码: A

doi: 10.7511/dllgxb201603012

0 引言

2003年, Dentcheva 等^[1]将随机占优作为约束引入优化模型,作为一种新的风险规避模型得到了国内外学者的广泛关注. 由于约束的非光滑性及其半无限性,模型对算法提出了更高的要求,如何求解这类模型成为大家研究的重要课题. 文献[2-4]提出了用割平面算法求解二阶随机占优模型;文献[5]提出了随机拟梯度(SCQ)算法;文献[5-6]提出了水平函数(LF)算法和投影水平函数(PLF)算法,其中提出的算法以惩罚为主题,这必然离不开 Slater 约束规范,限制了模型的应用范围;文献[2]中的算法虽然不要求约束规范,但是对大规模约束的求解略显不足. 目前提出的算法,很难应用于实际当中,它们不是限制太多,就是得到的结果可以进一步优化,而且对大规模的计算无能为力或是需要很长时间.

遗传算法(genetic algorithm)是仿生于自然界中的进化机制和生物遗传机制,由 Holland 等在 1975 年提出的. 其通过生成初始种群来进行第一步运行,初始种群通常表示为字符串,即二进制符号编码,然后通过不断地生成下一代并根据某

种规则来求解问题的最优解. 适应度高的个体通过将自身的部分字符串与其他个体的部分字符串进行交换得到下一代个体. 在遗传过程中个体的字符串也会发生变异. 随着时间的推移,将适应度差的个体淘汰,然后利用适应度高的个体在随机选取的相同的字符串点位进行交换得到新的个体,从而产生下一代种群,这种运行方法非常有效. 遗传算法的人工繁殖计划在 20 世纪 70 年代被首先提出,到 20 世纪 80 年代末逐渐得到大量学者的关注及研究. 遗传算法给出了一种解决复杂优化问题的通用框架,它直接对问题的目标函数进行操作,不要求约束函数可以微分等条件,对问题具有很强的鲁棒性. 由于遗传算法的这些优点,如今它已经被广泛应用于工程设计、组合优化等诸多领域. 本文将遗传算法应用于投资组合优化问题.

1 二阶随机占优模型

二阶随机占优的基本模型为

$$\begin{aligned} \min \quad & \{-E[f(x, \xi)]\} \\ \text{s. t.} \quad & g(x, \xi) \geq_{(2)} g(y, \xi) \\ & x \in Z \end{aligned} \quad (1)$$

式中: $g(\boldsymbol{x}, \boldsymbol{\xi})$ 是关于 $\boldsymbol{x}$ 的连续凹函数, $\boldsymbol{x}$ 是决策向量; $\boldsymbol{\xi}:\boldsymbol{\Omega} \rightarrow \boldsymbol{\Xi}$ 是概率空间 $(\boldsymbol{\Omega}, \mathcal{F}, \boldsymbol{P})$ 上的随机向量,支撑集 $\boldsymbol{\Xi} \subset \boldsymbol{R}^q$. $f:\boldsymbol{R}^n \times \boldsymbol{\Xi}$ 是关于 $\boldsymbol{x}$ 的连续凸函数. $\boldsymbol{Z} \subset \boldsymbol{R}^n$ 是闭凸集, $f, g \in L^1(\boldsymbol{\Omega}, \mathcal{F}, \boldsymbol{P})$. $\boldsymbol{y}$ 是 $\boldsymbol{Z}$ 中某一固定的向量.

随机占优法则是经济、决策论中的基本概念,随机占优的定义如下.

定义 1 设 X, Y 是随机变量, $X, Y \in L^1(\boldsymbol{\Omega}, \mathcal{F}, \boldsymbol{P})$,称随机变量 X 二阶占优随机变量 Y ,若 $F^{(2)}(X; \eta) \leq F^{(2)}(Y; \eta), \forall \eta \in \boldsymbol{R}$ 成立,表示为 $X \geq_{(2)} Y$.其中 $F^{(2)}(X; \eta) = \int_{-\infty}^{\eta} F(X; \alpha) d\alpha, \forall \eta \in \boldsymbol{R}, F(X; \eta)$ 是 X 的累积分布函数.

由文献[7]可知二阶随机占优 $X \geq_{(2)} Y$ 等价于如下形式:

$$E[(\eta - X)_+] \leq E[(\eta - Y)_+], \forall \eta \in \boldsymbol{R}$$

由上述二阶随机占优的等价条件可知模型可以重新改写为

$$\begin{aligned} \min \quad & \{-E[f(\boldsymbol{x}, \boldsymbol{\xi})]\} \\ \text{s. t.} \quad & E[(\eta - g(\boldsymbol{x}, \boldsymbol{\xi}))_+] \leq E[(\eta - g(\boldsymbol{y}, \boldsymbol{\xi}))_+], \\ & \forall \eta \in \boldsymbol{R} \\ & \boldsymbol{x} \in \boldsymbol{Z} \end{aligned} \tag{2}$$

为了避免技术上带来的不便,实际中,通常考虑如下的松弛问题:

$$\begin{aligned} \min \quad & \{-E[f(\boldsymbol{x}, \boldsymbol{\xi})]\} \\ \text{s. t.} \quad & E[(\eta - g(\boldsymbol{x}, \boldsymbol{\xi}))_+] \leq E[(\eta - g(\boldsymbol{y}, \boldsymbol{\xi}))_+], \\ & \forall \eta \in [a, b] \\ & \boldsymbol{x} \in \boldsymbol{Z} \end{aligned} \tag{3}$$

本文研究 $\boldsymbol{\xi}$ 是离散分布的情况,即 $P(\boldsymbol{\xi} = \boldsymbol{\xi}^i) = p_i, i = 1, \dots, m$.则由文献[2]可知模型等价于下面的形式:

$$\begin{aligned} \min \quad & \left\{ -\sum_{i=1}^m p_i f(\boldsymbol{x}, \boldsymbol{\xi}^i) \right\} \\ \text{s. t.} \quad & \sum_{i=1}^m p_i (\eta_j - g(\boldsymbol{x}, \boldsymbol{\xi}^i))_+ \leq \sum_{i=1}^m p_i (\eta_j - g(\boldsymbol{y}, \boldsymbol{\xi}^i))_+, \forall j = 1, \dots, m \\ & \boldsymbol{x} \in \boldsymbol{Z} \end{aligned} \tag{4}$$

其中 $\eta_j = g(\boldsymbol{y}, \boldsymbol{\xi}^j), j = 1, \dots, m$.

由以上易知,此模型的约束是非光滑的,且此

模型不满足 Slater 约束规范,这是因为若取 $\eta_j = \min\{g(\boldsymbol{y}, \boldsymbol{\xi}^j), j = 1, \dots, m\}$,则式(4)中不等式两边均为零.因此求解非光滑约束的算法难以应用于此模型.

2 遗传算法与数值实验

遗传算法通过在种群中利用重组的方式和保持有用的模块来有效地搜索空间.每个种群成员都能找到其所对应的字符串.例如,字符串1001110代表种群空间1####(其中#代表0或1)中的一个个体.这样,在样本空间采样的方式是隐式采样.遗传算法这种内在采样的能力被称为隐并行性.这指的是众多的样本模块和有效的采样模式在种群的每一代中都维持.

遗传算法的基本运算过程如下.

(I)初始化:生成所求问题可行域的初始种群并设定终止准则;

(II)个体评价:给出种群中每个个体的适应度函数值;

(III)选择运算:根据第(II)步中的适应度函数值对种群进行选择运算;

(IV)交叉运算:对种群进行交叉运算;

(V)变异运算:对种群进行变异运算;

(VI)终止条件判断:若达到终止准则,则将适应度函数值最大的个体作为最优解输出,终止计算.

遗传算法适用于高维的含有许多非线性和不连续性目标函数或是约束函数的随机问题.这些问题具有可靠性设计问题的特点:求解问题的部分解决方案或是一个状态受另一个状态影响. Coit 等^[8]成功地证明遗传算法可以应用于确定性的组合可靠性问题.遗传算法是一种启发式的算法,并不保证得到的解必是全局最优.

遗传算法在对可行解空间进行搜索的过程中,并不需要来自外部的信息,只通过内部的选择、交叉和变异即可以完成搜索过程.但是搜索过程却是依据种群中个体适应度函数值的大小来决定个体遗传到下一代的概率,并且要用适应度函数值来评价解的好坏程度.因此,适应度函数值必须是非负的,在实际应用中,适应度函数通常满足

下面的条件:(1)单值、连续、非负、最大化;(2)合理、一致性;(3)计算量小;(4)通用性强.

在实际应用中,适应度函数的给出需要依据实际问题来进行.适应度函数的选择会对遗传算法的结果产生重要的影响.适应度函数选择不当解有可能收敛于局部最优,而不能达到全局最优,因此需要将模型中的目标函数改写为 $10 - \sum_{i=1}^m p_i f(\mathbf{x}, \xi^i)$,这样便可以保证适应度函数大于零.

在此部分,由于所取数据为样本数据,令 $p_i = 1/m$,则模型化为

$$\begin{aligned} \min \quad & \left\{ 10 - \frac{1}{m} \sum_{i=1}^m f(\mathbf{x}, \xi^i) \right\} \\ \text{s. t.} \quad & \sum_{i=1}^m (\eta_j - g(\mathbf{x}, \xi^i))_+ \leq \sum_{i=1}^m (\eta_j - g(\mathbf{y}, \xi^i))_+, \\ & \forall j = 1, \dots, m \\ & \mathbf{x} \in \mathbf{Z} \end{aligned}$$

(5)

例 1 以文献[5]中的例题 1 为例用遗传算法求解,并对所得结果进行比较.考虑 10 个月份的月收益率及 5 种资产,具体如表 1 所示.

表 1 5 种资产 10 个月的月收益率

月份	收益率				
	资产一	资产二	资产三	资产四	资产五
1	1.2	1.3	0.9	1.1	0.80
2	1.3	1.0	1.1	1.1	0.75
3	1.4	0.8	1.0	1.2	0.65
4	1.5	0.9	1.1	1.3	0.75
5	1.1	1.4	1.1	1.2	0.80
6	1.2	1.3	1.3	1.2	0.90
7	1.1	1.2	1.2	1.1	1.00
8	1.0	1.1	1.1	1.0	1.10
9	1.0	1.2	1.0	1.1	1.10
10	1.1	1.1	1.1	1.2	1.20

设置遗传算法的初始种群数分别为 100、1 000 和 10 000,表示为 GA(100)、GA(1 000)、GA(10 000),应用遗传算法求解,得到的结果如图 1~3 所示.

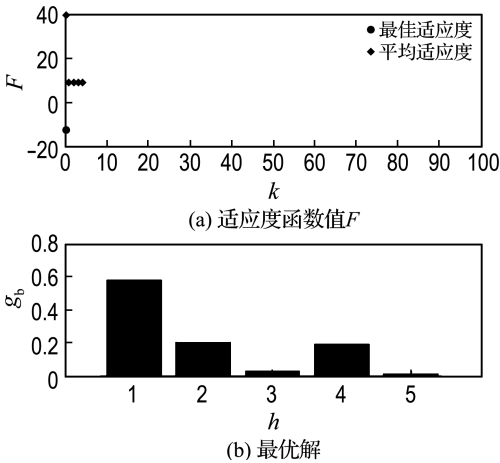


图 1 初始种群为 100 时的计算结果

Fig. 1 The result with initial population 100

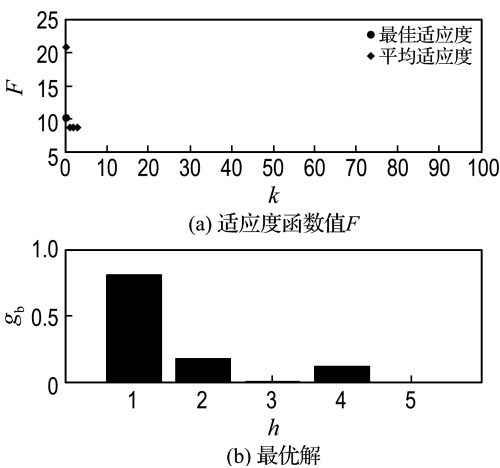


图 2 初始种群为 1 000 时的计算结果

Fig. 2 The result with initial population 1 000

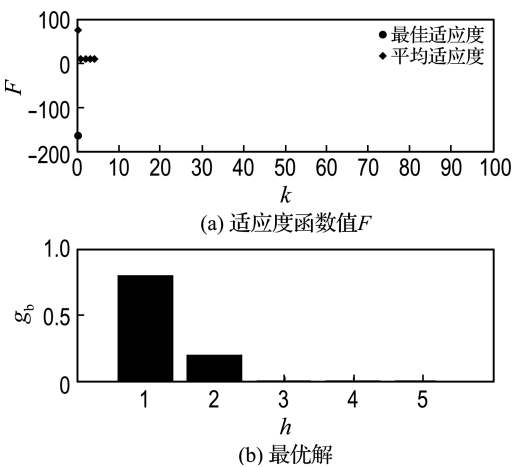


图 3 初始种群为 10 000 时的计算结果

Fig. 3 The result with initial population 10 000

表 2 记录了由遗传算法得出的数值结果,并给

出了投影水平函数(PLF)算法的结果. 从中能够得出:当遗传算法初始种群数为 100 时, $E[g(x,\xi)]$ 的计算结果明显好于 PLF 算法的结果,而且运行时间大大快于 PLF 算法的运行时间. 当初始种群数为 1 000 时,遗传算法运行时间和 PLF 算法运行时间相当,但 $E[g(x,\xi)]$ 的计算结果明显好于 PLF 算法. 从表 2 中还可以看出, $E[g(x,\xi)]$ 的计算结果随着初始种群数量的增加而改善,但是运行时间相应增加. 初始种群数从 1 000 增加到 10 000,结果改善不明显,但是计算时间却大大增加,故初始种群数为 1 000 时,算法效果最好.

表 2 不同最优投资组合策略之间的比较

Tab. 2 Comparison of different optimal portfolio strategies

算法	迭代 步数	时间/ min	最优解	$E[g(x,\xi)]$
GA(100)	4	0.132 2	(0.576,0.203, 0.027,0.188, 0.006)	1.166
GA(1 000)	3	0.653 6	(0.750,0.175,0, 0.074,0)	1.176
GA(10 000)	4	8.565 0	(0.799,0.201, 0,0,0)	1.178
PLF	5	0.635 5	(0.325,0.231, 0.177,0.266,0)	1.148

例 2 为了检验遗传算法对含有大规模约束的随机占优模型的计算能力,随机选取我国上市公司中的 75 只股票资产在 2000~2012 年的月收益率来构建随机占优模型, $E[g(y,\xi)]=1.007\ 1$,当初始种群数为 1 000 时,用遗传算法得出的结果如图 4 所示.

利用上述根据遗传算法得出的最优结果求得 $E[g(x,\xi)]=1.008\ 3$, 优于原来的投资组合策略. 程序运行时间为 6.134 min,相同投资方案组合情况下,大大快于文献[6]中算法的运行时间,而且从图 5 中可以看出,用遗传算法求出的投资组合结果整体优于原来的投资组合策略.

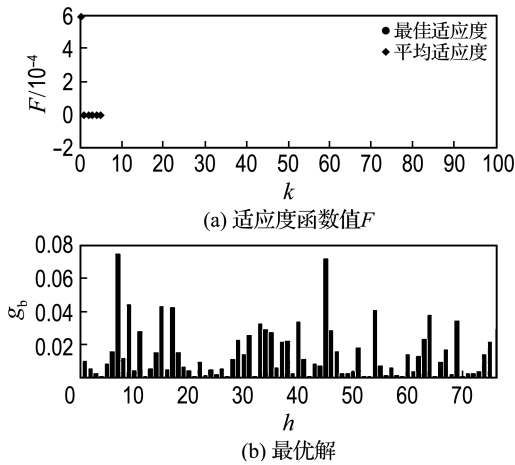


图 4 遗传算法得出的最优结果
Fig. 4 Optimal result by GA

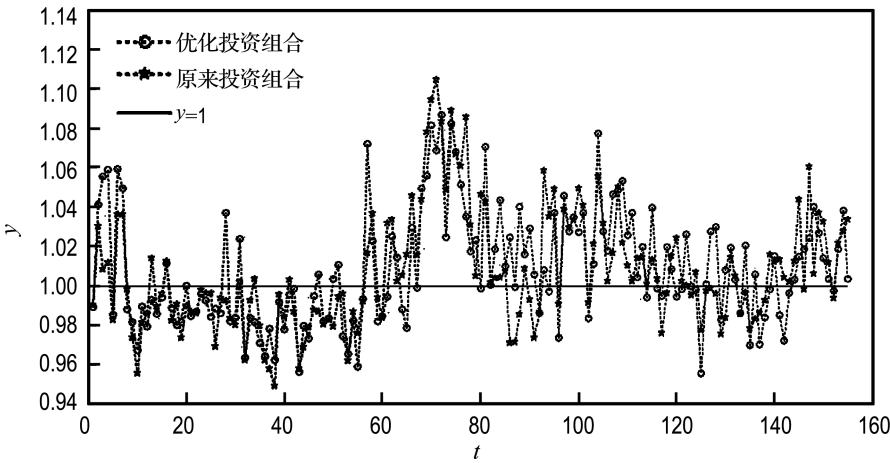


图 5 不同投资组合的样本月收益率比较

Fig. 5 The sample month rate of return comparison of the different portfolios

3 结 语

通过以上结果,可以看出遗传算法在解决二阶随机占优模型中的优势. 遗传算法不需要求解目标函数和约束函数的次微分,也不需要模型满足 Slater 约束规范,大大提高了问题的可解决性,而且结果优于其他算法的结果. 由于遗传算法在解决复杂、困难的优化问题方面的优势,在信息量如此巨大的今天,相信遗传算法会在金融、运筹领域得到广泛的应用.

参考文献：

[1] Dentcheva D, Ruszczyński A. Optimization with stochastic dominance constraints [J]. **SIAM Journal on Optimization**, 2003, **14**(2):548-566.

[2] Fábian C I, Mitra G, Roman D. Processing second order stochastic dominance models using cutting-plane representations [J]. **Mathematical Programming**, 2011, **130**(1):33-57.

[3] Homem-de-Mello T, Mehrotra S. A cutting surface method for uncertain linear programs with polyhedral stochastic dominance constraints [J]. **SIAM Journal on Optimization**, 2009, **20**(3):1250-1273.

[4] Rudolf G, Ruszczyński A. Optimization problems

with second order stochastic dominance constraints: Duality, compact formulations, and cut generation methods [J]. **SIAM Journal on Optimization**, 2008, **19**(3):1326-1343.

[5] Meskarian R, XU Hui-fu, Fliege J. Numerical methods for stochastic programs with second order dominance constraints with applications to portfolio optimization [J]. **European Journal of Operational Research**, 2012, **216**(2):376-385.

[6] SUN Hai-lin, XU Hui-fu, Meskarian R, *et al.* Exact penalization, level function method and modified cutting-plane method for stochastic programs with second order stochastic dominance constraints [J]. **SIAM Journal on Optimization**, 2013, **23**(1):602-631.

[7] Ogryczak W, Ruszczyński A. On consistency of stochastic dominance and mean-semideviation models [J]. **Mathematical Programming**, 2001, **89**(2):217-232.

[8] Coit D W, Smith A E. Solving the redundancy allocation problem using a combined neural network/genetic algorithm approach [J]. **Computers & Operations Research**, 1996, **23**(6):515-526.

Solution to constrained optimization problem of second-order stochastic dominance by genetic algorithm

ZHANG Hong-wei^{*1}, YU Hai-sheng¹, PANG Li-ping¹, WANG Jin-he²

(1. School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China;
2. Department of Computer, Qingdao University of Technology, Qingdao 266033, China)

Abstract: Stochastic dominance is fundamental concept in economics and decision-making theory, and is widely applied to portfolio optimization in recent years. Genetic algorithm has the advantages, which doesn't need to solve the subdifferential of object function and constrained function or to satisfy Slater constrained rules, so it can solve the constrained semi-infinite and non-smooth problem. Two examples show that the genetic algorithm can well solve the portfolio optimization problem and the efficiency is greatly improved.

Key words: second-order stochastic dominance; genetic algorithm; portfolio optimization