

# 一种求解随机期望值模型的有效算法

肖宁, 曾建潮

(太原科技大学 系统仿真与计算机应用研究所, 山西 太原 030024)

**摘要:** 随机期望值模型是一类有着广泛应用背景的随机规划问题. 为了寻找更为有效的求解随机期望值模型的算法, 通过采用随机仿真来逼近随机函数, 在微粒群算法中利用随机仿真进行适应值估计和实现为了检验解的可行性, 从而给出了求解随机期望值模型的新的算法. 最后, 通过实例仿真说明了算法的正确性和有效性.

**关键词:** 随机规划; 随机期望值模型; 微粒群算法; 随机仿真

**中图分类号:** 0221.5 **文献标识码:** A **文章编号:** 1673-4785 (2008) 03-0279-04

## An efficient algorithm for solving stochastic expected value models

XIAO Ning, ZENG Jian-chao

(Division of System Simulation & Computer Application, Taiyuan University of Science and Technology, Taiyuan 030024, China)

**Abstract:** The stochastic expected value model is a class of stochastic programming problems with wide applicability. In order to find a more effective algorithm for solving these problems, we used random simulations to approach the stochastic function in particle swarm optimization, obtaining an estimation of the degree of fitness and verifying the feasibility of the solution. Finally, results of simulations show the correctness and effectiveness of this algorithm.

**Keywords:** stochastic programming; stochastic expected value models; particle swarm optimization; random simulation

对于随机规划问题中所出现的随机变量, 由于不同的管理目标和技术要求, 采用的方法也不尽相同, 而最自然的方法就是取随机变量所对应的函数的平均值 (数学期望). 这种在期望值约束下使目标函数的期望值达到最优的随机规划模型称为随机期望值模型 (stochastic expected value models, SEVM).

随机期望值模型问题的提取并不难, 但其求解却很难. 因此, 探索高效的求解随机期望值模型的算法就非常具有研究价值.

求解随机期望值模型的主要方法是利用随机仿真与智能算法相结合来进行, 其中以遗传算法 (genetic algorithm, GA) 最为成功<sup>[1-3]</sup>. 但遗传操作中的选择、交叉、变异过程复杂, 不易掌握, 而且其收敛速度缓慢、精度低. 目前, 国内外相关学者仍在继续探

索求解该类问题的更新和更为有效的算法<sup>[2,4]</sup>.

随着计算机技术的迅速发展, 使得智能技术具有解决大规模、更复杂优化问题的能力. 微粒群算法 (particle swarm optimization, PSO) 是由 Eberhart 与 Kennedy 于 1995 年提出的一种新的智能技术<sup>[5]</sup>, 它与 GA 类似, 采用基于种群的并行全局搜索策略, 但不具有选择、变异等操作, 仅采用简单的速度—位置模型实现对整个空间的寻优操作. 该算法只需调整很少的参数, 具有简单、易于实现、收敛速度快、精度高等优点. 它已在各类问题的求解及应用中展现了它的特点和魅力<sup>[6-8]</sup>. 所以, 尝试将 PSO 算法应用于随机期望值模型是一个很有现实意义的研究方向, 然而, 将 PSO 算法应用于该类问题中的研究至今尚无文献报导, 更无文献给出利用 PSO 算法求解该类问题的统一算法. 本文的工作是把随机仿真与 PSO 算法结合起来求解该类问题, 给出了统一的求解算法并通过仿真实验取得了十分满意的效果, 从而实

收稿日期: 2007-10-11.

基金项目: 国家自然科学基金资助项目 (60674104).

通讯作者: 肖宁. E-mail: xnwyh@126.com.

现了 PSO 算法对这一大类连续空间的随机规划问题的求解.

### 1 随机期望值模型

一般地,随机期望值模型可表示为

$$\begin{cases} \max E[f(x, \omega)], \\ \text{s t } E[g_j(x, \omega)] \leq 0, j = 1, 2, \dots, p. \end{cases}$$

式中:  $x$ 、 $\omega$  分别是决策向量和随机向量,  $f(x, \omega)$  为目标函数,  $g_j(x, \omega)$  为一组随机约束函数,  $j=1, 2, \dots, p$ ;  $E$  是期望值算子.

### 2 随机仿真

随机仿真也叫随机模拟或 Monte Carlo 模拟,主要是依据概率分布来对随机变量进行抽样,从而为系统决策提供依据或对系统决策进行检验.虽然它只给出统计估计而非精确结果,且应用其研究问题需要花费大量的计算时间,然而它的确是处理解析方法行不通的复杂问题的有效工具,该技术已被应用到许多领域中.针对本文的需要,下面给出随机仿真的期望值估计算法.

设  $\omega$  为定义在概率空间  $(\Omega, A, Pr)$  上的  $n$  维随机向量,  $f: R^n \rightarrow R$  为可测函数,则  $f(\omega)$  为随机变量.利用随机仿真计算  $E[f(\omega)]$  的步骤如下:

算法 1:随机仿真算法之期望值估计算法:

- 1)置  $L=0$ ;
- 2)根据概率测度  $Pr$  从  $\Omega$  中产生样本  $\omega_i$ ;
- 3) $L=L+f(\omega_i)$ ;
- 4)重复 2)和 3) 共  $N$  次;
- 5)  $E[f(\omega)] = L/N$ .

### 3 微粒群算法

微粒群算法是最新的群智能算法,它由 Eberhart 和 Kennedy 于 1995 年正式提出<sup>[5]</sup>,其基本思想是受他们早期对鸟类群体行为的研究结果的启发并利用了生物学家 Frank Heppner 的生物模型.它的进化规则与“优胜劣态,适者生存”的 GA 截然不同.它强调的是群体中个体之间信息的社会共享和协同进化.

微粒群中的每一个粒子定义为  $d$  维空间 (待优化问题的解空间) 中的粒子,以一定的速度  $V_i=(V_{i1}, V_{i2}, \dots, V_{id})$  在搜索空间中飞行.算法开始时,初始化一组随机解  $(x_1, x_2, \dots, x_N)$ ,  $N$  为粒子的个数,然后粒子根据自己在解空间中的飞行经验以及群体的飞

行经验动态调整自己位置、速度,并用适应值来评价解的优劣,选出  $P_{best}$  (个体极值) 与  $G_{best}$  (全局极值) 并记录它们的位置,再根据速度、位置更新方程 (1)、(2)、(3) 更新下一代粒子的速度、位置,通过迭代寻找最优值. PSO 算法的数学描述为

$$V_{id} = \omega V_{id} + c_1 \times \text{rand}() \times (P_{id} - X_{id}) + c_2 \times \text{rand}() \times (P_{gd} - X_{id}). \quad (1)$$

$$\begin{cases} V_{id} = V_{\max}, \text{ if } V_{id} > V_{\max}; \\ V_{id} = -V_{\max}, \text{ if } V_{id} < -V_{\max}. \end{cases} \quad (2)$$

$$X_{id} = X_{id} + V_{id}. \quad (3)$$

式中:  $\omega$  为惯性权重,它使微粒保持运动的惯性,使其有能力探索新的区域;  $c_1$ 、 $c_2$  为正的加速度常数,通常取值为 2,它们使每个微粒向  $P_{best}$  和  $G_{best}$  位置加速运动,分别起到了协调“勘探”和“开发”解之间的作用;  $\text{rand}()$  为  $[0, 1]$  上均匀分布的随机数,它们用来模拟自然界中群体行为的轻微扰动;  $P_{id}$ 、 $P_{gd}$  分别为个体极值、全局极值的第  $d$  维分量;在式 (2) 中对微粒的最大速度进行了最大限制:如果当前对微粒的加速将导致它的某维的速度分量  $V_{id}$  超过该维的最大速度限额  $V_{\max}$ , 则该维的速度被限制为  $V_{\max}$ , 它决定了微粒在解空间的搜索精度,如果  $V_{\max}$  过大,粒子容易飞过最优解,反之,粒子容易陷入局部搜索空间而无法进行全局搜索,若问题的搜索空间限制在  $[-X_{\max}, X_{\max}]$  内,则可设定  $V_{\max} = kX_{\max}$ ,  $0 < k < 1$ <sup>[6]</sup>.

### 4 随机仿真与 PSO 算法相结合的随机期望值模型算法

在利用 PSO 算法求解随机期望值模型问题时,其核心是对随机函数进行计算,这显然可以利用随机仿真的方法进行,它主要体现在为了检验解的可行性、估计目标函数的适应值上.

随机仿真与 PSO 算法相结合的求解随机期望值模型算法具体步骤描述如下:

- 1)在  $d$  维问题空间上对微粒群进行初始化:设定群体规模为 pop size,在决策向量  $x$  的可行域中产生一随机数,利用随机仿真的期望值估计算法计算  $E[g_j(x, \omega)]$  并检验该随机数的可行性 (即判断  $x$  是否满足  $E[g_j(x, \omega)] \leq 0$ ),重复该过程 pop size 次,从而得到 pop size 个初始可行的微粒:  $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$ ,  $i = 1, 2, \dots, \text{pop size}$ ,然后再对速度等进行初始

化;

2)利用随机仿真期望值估计算法计算每个微粒的适应值 (即  $E[f(x, )]$ );

3)对每个微粒,将其适应值与所经历的最好位置的适应值进行比较,若较好,则将其作为当前最好位置;

4)对每个微粒,将其最好适应值与全局所经历的最好适应值进行比较,若较好,则将其作为当前的全局最好位置;

5)根据进化方程 (1)、(2)、(3)进化;

6)对更新后的粒子再次利用随机仿真的期望值估计算法计算  $E[g_j(x, )]$ 并检验粒子的可行性 (即判断  $x$ 是否满足  $E[g_j(x, )] \leq 0$ ),若可行,则接受,否则保持原位置不变;

7)重复 2)~6)至一个预设的最大迭代次数或一个足够好的适应值;

8)输出最好的微粒及对应的适应值作为最优解及对应的最优值.

### 5 实例仿真

为了测试本文算法的性能并便于比较,特选文献 [2]中的实例来进行.

以下是具有 3 个决策变量和 3 个随机变量的期望值模型:

$$\begin{cases} \min E[\sqrt{(x_1 - 1)^2 + (x_2 - 2)^2 + (x_3 - 3)^2}] ; \\ \text{s t } x_1^2 + x_2^2 + x_3^2 = 10 \end{cases}$$

式中:  $x_1$  服从均匀分布  $U(1, 2)$ ,  $x_2$  服从正态分布  $N(3, 1)$ ,  $x_3$  服从指数分布  $EXP(4)$ .

在实例中取与文献 [2] 相同的以下数据:

模拟次数: 3 000, 迭代次数: 300, 种群规模: 30, 运行次数: 1.

此外, 从 0.9~0.4 线性递减, 微粒的最大速度取为搜索空间最大值的 0.1 倍 (注: 通过实验, 在一定范围内改变这 2 个参数的值发现对求解结果影响不大).

利用 VC++6.0 编程: 运行一次的结果如表 1 所示, 从表中可以看出, 其结果明显优于文献 [2]; 为了更直观地体现迭代过程, 将该实例的迭代过程抽样 15 次 (见图 1). 该算法不仅收敛速度相当快而且精度高, 在 20 代时的最优值就已经超过了文献 [2] 在迭代 300 代时所得到的最优值, 而且从 180

代起就获得了稳定解. 为避免一次运行结果的偶然性, 将程序运行 50 次, 所得的平均值见表 1 最后一行. 显然利用该算法所得平均最优值优于文献 [2] 而且由 50 次运行中的统计结果可知每次所得最优值也均优于文献 [2] (限于篇幅在此未列出每次运行所得的数据).

表 1 运行一次的结果比较及运行 50 次的平均值

Table 1 Results comparison of running one time and average of running 50 times of example

|             | $x_1$    | $x_2$    | $x_3$    | 最优值   |
|-------------|----------|----------|----------|-------|
| GA (文献 [2]) | 1. 103 5 | 2. 169 3 | 2. 019 1 | 3. 56 |
| PSO (本文)    | 1. 195 9 | 2. 346 3 | 1. 739 3 | 3. 15 |
| 平均值         | 1. 181 1 | 2. 244 6 | 1. 823 2 | 3. 16 |

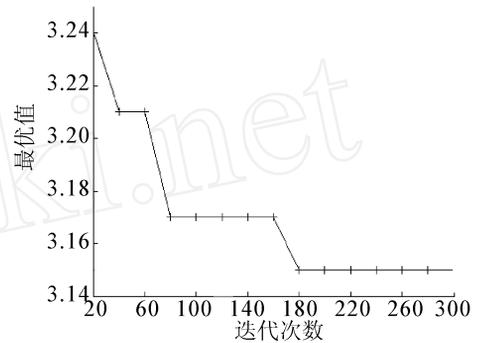


图 1 SEVM 实例的抽样图

Fig 1 Sampled data of example on SEVM

该算法之所以优于文献 [2], 是因为 PSO 算法没有 GA 那样复杂的遗传操作, 而仅仅是利用个体在解空间中的随机速度来改变个体, 表现出更强的随机性, 使其计算复杂度比 GA 低; 在 GA 中染色体互相共享信息, 所以整个种群的移动是比较均匀地向最优区域移动, 而在 PSO 算法中信息的提供仅是全局最优值, 这是信息的单向流动, 整个种群的搜索更新过程是跟随当前的全局最优解的过程, 与 GA 相比, 在大多数情况下, 所有的粒子有可能更快地收敛于最优解; 此外, 微粒所具有的“记忆”特性, 使得它们通过“自我学习”和向“同伴学习”, 使下一代能从上一代继承更多的信息, 从而可在较短时间内找到最优解.

### 6 结束语

本文将随机仿真与 PSO 算法相结合的混合算法应用于随机期望值模型, 给出了一种求解随机期

望模型的统一算法,通过仿真实验与文献中的 GA 结果进行了比较,其优化性能明显优于 GA,显示了本文算法的高效性,体现了它在随机期望值模型问题求解中的优势,从而为连续空间随机期望值模型这一大类问题的求解提供了新的途径,同时也拓展了 PSO 算法研究的应用领域.

值得一提的是,本文作者将随机仿真与 PSO 算法相结合的混合算法应用于随机机会约束规划、随机相关机会规划也取得了与本文相同的效果. 进一步提高该算法的求解速度、将 PSO 算法应用于离散空间随机规划问题是下一步要研究的内容.

### 参考文献:

[1] 陈淑燕,王 炜,郑长江. 基于模拟退火求解的一种新的随机存储规划模型 [J]. 公路交通科技, 2005, 22 (4): 144-147.  
CHEN Shuyan, WANG Wei, ZHENG Changjiang Apply simulated annealing to solve a novel random storage programming model[J]. Journal of Highway and Transportation Research and Development, 2005, 22 (4): 144-147.

[2] 刘宝碇,赵瑞清,王 纲. 不确定规划及应用 [M]. 北京:清华大学出版社, 2003: 32-137.

[3] ZHOU J. Uncapacitated facility layout problem with stochastic demands [C]// Operations Research Society of China Academic Exchange Conference Proceedings HongKong: HongKong GbbalH ink Press, 2000: 904-911.

[4] 彭 锦,刘宝碇. 不确定规划的研究现状及其发展前景 [J]. 运筹与管理, 2002, 11 (2): 1-10.  
PENG Jin, L U Baoding Uncertain programming: current

status and future prospects [J]. Operations Research and Management Science, 2002, 11 (2): 1-10

[5] KENNEDY J, EBERHART R. Particle swarm optimization [C]//Proc International Conference on Neural Networks Perth, Australia, 1995: 1942-1948

[6] 曾建潮,介 婧,崔志华. 微粒群算法 [M]. 北京:科学出版社, 2004: 1-100.

[7] KOAY C A, SRINIVASAN D. Particle swarm optimization based approach for generator maintenancescheduling [C]// Proceedings of the 2003 IEEE Swarm Intelligence Symposium. Indianapolis, USA, 2003: 167-173.

[8] SUN Q, SHI Y H, BAUSON W A. Utilizing particle swarm optimization to label a structured beam matrix [C]// Proceedings of the 2003 IEEE Swarm Intelligence Symposium. Indianapolis, USA, 2003: 118-123.

### 作者简介:



肖 宁,女,1975年生,硕士研究生,主要研究方向为智能计算,发表学术论文 3 篇,1 篇被 EI 收录.



曾建潮,男,1963年生,教授,博士生导师,主要研究方向为智能控制、进化计算、系统建模与仿真等. 主持与参加完成国家“863 项目、国家自然科学基金项目、国家“九五”攻关项目、原机械部跨世纪学科带头人项目、省自然科学基金项目、省青年基金项目、省攻关项目以及企业单位委托项目 30 余项,在国内外发表学术论文 180 余篇,其中被 SCI EI ISTP 等收录 40 余篇.