# 基于熵的自适应变异的粒子群优化算法

朱永利[1]，陈英伟[1]，韩　凯[1]，王　磊[2]
(1.华北电力大学 计算机科学与技术学院，河北 保定 071003；
2.华北电力大学 控制科学与工程学院，河北 保定 071003)

**摘　要**：在研究标准粒子群算法原理的基础上，提出了一种基于熵的自适应变异粒子群优化算法。此算法利用熵来评价种群的多样性，并根据种群的多样性自适应地调整变异概率和变异算子，进而利用变异操作丰富种群多样性，扩大搜索空间，避免陷入局部最优。将改进后的算法运用常见的几个测试函数进行了寻优仿真，仿真结果表明了基于熵的自适应变异粒子群优化算法的可行性和有效性。

**关键词**：熵；粒子群算法；变异
**中图分类号**：TP301.5　　**文献标识码**：A

# Particle swarm optimization with adaptive mutation based on entropy

ZHU Yong Li[1]，CHEN Ying Wei[1]，HAN Kai[1]，WANG Lei[2]
(1.School of Computer Science and Technology，North China Electric Power University，Baoding 071003，China；
2.School of Control Science and Engineering，North China Electric Power University，Baoding 071003，China)

**Abstract**：Based on studying of the principles of traditional PSO，a particle swarm optimization with adaptive mutation based on entropy is proposed .This algorithm introduces entropy to evaluate population diversity，and adaptively adjustes mutation rate and mutation operator based on population operator，further uses mutation operation to enrich population diversity，enlarge search space and avoid be wrapped in local optimal solution. The improved algorithm is applied to optimize several functions.The simulation results show its feasibility and validity.

**Key words**：entropy；particle swarm algorithm；emulation

粒子群优化算法 PSO(Partide Swarm Optimization)是由 Eberhart 和 Kennedy 于 1995 年提出的一类基于群智能的随机优化算法，适用于求解大量非线性、不可微和多峰值的复杂优化问题。由于 PSO 算法程序实现起来异常简洁、需要调整的参数也少，因而已应用于多个学科和工程领域。但是与其他全局优化算法(遗传算法)一样，粒子群优化算法同样存在着早熟收敛现象，尤其是在比较复杂的多峰搜索问题中。目前国内外学者已经提出了多种改进算法来克服粒子群算法的早熟收敛问题。这些改进主要集中在 3 个方面。一方面是根据粒子群的多样性程度来自适应调整惯性权重，从而扩大搜索空间，避免陷入局部最优解[1]。另一方面是根据种群多样性和当前最优解的状况，利用变异操作改变全局最优位置或者重新初始化种群的位置和速度，从而改变粒子群的聚集方向，减少局部最优解的吸引[2-3]。最后一方面就是粒子群算法和局部搜索算法的结合，利用局部搜索算法来改善粒子群算法较差的局部搜索能力，从而避免陷入局部最优。但是这些改进算法仍存在一些问题：(1)加大了运算量，粒子结构的破坏导致收敛速度慢；(2)某些问题的全局最优位置不易确定，因此无法知道当前得到的最优解是局部最优解还是全局最优解，这时一些粒子群的改进算法将显得无能为力[4]。

本文针对这些改进算法存在的一些问题，提出了一种基于熵的自适应变异的粒子群算法，来克服早熟收敛问题。随着 Shannon 把热力学中熵的概念引入信息论，信息熵的应用已经渗透到多个学科。基于熵采样的寻优算法借助熵估计搜索空间，对它的勘探更为充分和具目的性，表现在这类算法较易于跳出局部极值点

而求得全局最优值。因此，本文提出的算法基于熵来评价种群多样性，并根据熵来自适应调节变异概率和变异算子，即当种群多样性较丰富时，利用较小的变异概率和变异步长来变异每个粒子的位置和速度；当种群多样性较贫乏时，利用较大的变异概率和变异步长来调整每个粒子的位置和速度。可见，此算法能够有效改善种群多样性的丢失状况，避免陷入局部最优，克服其早熟收敛现象。

## 1 基本粒子群优化算法

基本 PSO 算法通过个体间的协作和竞争实现对问题最优解的搜索。算法首先随机生成初始种群，其中每个粒子都为优化问题的一个候选解，并根据目标函数为之确定一个适应值。每个粒子在解空间中运动，并由一个速度矢量决定其运动方向和距离。通常，每个粒子追随当前自身最优位置和种群的最优位置而动。算法经过逐次搜索得到问题的最优解。

记第 $t$ 代第 $i$ 个粒子位置为 $x_i(t)=(x_{i1}(t),x_{i2}(t),...,x_{in}(t))$；第 $t$ 代第 $i$ 个粒子速度为 $v_i(t)=(v_{i1}(t),v_{i2}(t),\cdots,v_{in}(t))$；第 $t$ 代个体 $i$ 的最优位置为 $p_i(t)$ - $(p_{i1}(t),p_{i2}(t),\ldots,p_{in}(t))$；第 $t$ 代种群的全局最优位置为 $p_g(t)=(p_{g1}(t),p_{g2}(t),\cdots,p_{gn}(t))$，粒子 $i$ 将按照公式(1)和(2)改变其速度和位置：

$$v_{id}(t+1)=wv_{id}(t)+c_1\times r_1\times[p_{id}(t)-x_{id}(t)]+c_2\times r_2\times[p_{gd}(t)-x_{id}(t)] \quad (1)$$

$$x_{id}(t+1)=x_{id}(t)+v_{id}(t+1) \quad (2)$$

其中 $d=1,2,\ldots,n$；$i=1,2,\cdots,N$；$N$ 为种群规模；$r_1$ 和 $r_2$ 为均匀分布于[0，1]的随机数；$c_1$ 和 $c_2$ 是学习因子，通常取值为 2；$w$ 为加权系数，用来控制历史速度对当前速度的影响程度，用来平衡粒子的勘探和开发能力，一般在[0.1，0.9]之间取值。但若 $w$ 能随迭代的进行而线性减少，将显著改善算法的收敛性能；令 $w_{\max}$ 为最大加权系数，$w_{\min}$ 为最小加权系数，$iter$ 为当前迭代次数，$iter_{\max}$ 为算法总迭代次数，则有：

$$w=w_{\max}-iter\times\frac{w_{\max}-w_{\min}}{iter_{\max}} \quad (3)$$

一般 $w_{\max}$ 取值为 0.9，$w_{\min}$ 取值为 0.4。更新过程中，粒子每一维的速度限制在 $v_{\max}$，粒子每一维的坐标也被限制在允许范围之内。同时，个体极值 $p_i$ 和全局极值 $p_g$ 在迭代过程中不断更新，最后输出的 $p_g$ 就是算法找到的最优解[5]。

## 2 基于熵的自适应变异的粒子群算法

粒子群算法一般采用实数编码，由于没有选择、交叉与变异操作，算法结构相对简单，运行速度很快。但是，算法在运行过程中，如果全局最优粒子位置 $p_g$ 在较长时间内未发生变化，那么所有粒子就会迅速向其靠拢。当粒子群很接近 $p_g$ 时，其速度更新将主要由 $w\times v_{id}(t)$ 来决定。由于 $w$<1，则速度将会越来越小，因此粒子群表现出强烈的趋同性。当粒子数较少时，表现在优化性能上就是收敛速度快，但易于陷入局部最优。故与其他优化算法一样，粒子群算法同样存在着早熟收敛现象，尤其在比较复杂的多峰函数搜索问题中。

粒子群算法根据当前个体最优位置和全局最优位置来指导粒子下一步的位置和速度，故所有粒子都聚集到当前全局最优位置，导致种群多样性的严重丢失，不易于扩大搜索空间，很难在此区域内搜索到更多有价值的解。因此，本文引入变异算子到粒子群算法中，并根据种群多样性自适应调整变异概率和变异算子。当种群多样性较丰富时，则采用较小的变异概率和变异步长；当种群多样性较贫乏时，则采用较大的变异概率和变异步长。同时，引入熵这一概念来评价种群多样性。

### 2.1 种群多样性的度量

熵是状态的多样性、丰富程度(或者说混乱程度、复杂程度)的定量计量标尺，因此，本文利用熵来作为种群内部进化状态的量测工具，用于评价种群的多样性[6-7]。

设粒子群算法待搜索问题的解空间可以分为 $M$ 个不同区域 $A_1,A_2,\cdots,A_M$，进行如下随机试验：从粒子中任意抽出一个个体，考察它属于解空间的哪一个区域。设该个体属于区域 $A_i$ 的概率为 $p_i,i=1,2,\cdots,M$；易知 $p_i\geqslant 0$ 及 $\sum_{i=1}^{M}p_i=1$。这样，每一个体属于哪个区域都将具有一定的不确定性。

种群熵是种群多样性的测度，由公式(4)进行定义。

$$\Delta\tau_{ij}(u)=\sum_{k\in A_u}\Delta\tau_{ij}^k \quad (4)$$

系统熵是不能预先知道的。实际上，知道一个系统的熵就等同于解决了对应的问题。因此，本文使用如下方法估计种群 $p(t)$的熵 $E$。

(1)估计解空间 $S$ 及划分区域数 $M$。解空间 $S$ 用区间[$A$，$B$]来度量，其长度为 $\lambda_0$。设算法运行过程中迄今为止所发现的最小和最大适应值分别为 $f_{\min}$ 和 $f_{\max}$，则令 $A=(1-\delta)f_{\min},B=(1+\delta)f_{\max},\lambda_0=A-B$。这里，$\delta$是一个很小的正数(如 0.05)。设种群规模为 $N$，则 $M$=$N$。

(2)把区间[$A$，$B$]分为相等的 $M$ 个小区间，统计适应值在每一个区间内个体的数目 $M_i$。

(3)按照 $p_i=M_i/M$ 计算个体出现在第 $i$ 区间的概率 $p_i$。

(4)把 $p_i$ 代入公式(4)计算种群熵。

由定义可知，当种群中所有粒子的适应值都相同，种群多样性最贫乏时，熵取最小值 $E$=0。当种群中所有粒子的适应值平均位于不同的区间，种群多样性最丰富时，熵取最大值即 $E=\lg N$。可见，熵随着种群多样性的增加而增加。

### 2.2 基于种群熵的变异操作

当种群多样性较丰富时，粒子就不需要在全局搜索最优解而只在此局部区域搜索最优解。当种群多样性较贫乏时，粒子需要扩大搜索空间，丰富种群多样性，避免陷入局部最优解。因此，变异概率如公式(5)所示[4]。

$$p_m = P_M \times \exp\left[-\frac{g \times E}{G \times H}\right] \tag{5}$$

其中 $P_M$ 为常值，取值为 0.4。$G$ 为最大迭代次数，$g$ 为当前进化代数，$E$ 为当前种群熵，$H$ 为种群最大熵。可见变异概率 $p_m$ 随着种群多样性(种群熵 $E$)的下降而增大，扩大搜索空间，避免陷入局部最优解。

同时，在种群多样性较丰富时，应该采用较小的变异步长，防止错过最优解；当种群多样性较贫乏时，应该采用较大的变异步长，防止陷入局部最优。因此，本文以概率 $r$ 选择如下 2 个变异算子：

若 $r>0.5$ 时，

$$x'_{ij}(t) = x_{ij}(t) + (u[j] - l[j]) \times \exp(-\frac{g}{G} \times \frac{E}{H}) \tag{6}$$

$$v'_{ij}(t) = v_{ij}(t) + 2v_{\max} \times \exp(-\frac{g}{G} \times \frac{E}{H}) \tag{7}$$

若 $r \leqslant 0.5$ 时，

$$x'_{ij}(t) = x_{ij}(t) - (u[j] - l[j]) \times \exp(-\frac{g}{G} \times \frac{E}{H}) \tag{8}$$

$$v'_{ij}(t) = v_{ij}(t) - 2v_{\max} \times \exp(-\frac{g}{G} \times \frac{E}{H}) \tag{9}$$

其中 $r = rand(0,1)$，$u[j]$ 为第 $j$ 个变量取值上限，$l[j]$ 为第 $j$ 个变量取值下限。分析上式，可以看出变异步长随着种群多样性的下降而增加，来扩大搜索空间，增强全局搜索能力。

### 2.3 算法步骤

基于熵的自适应变异粒子群算法在每次迭代时根据种群多样性按照一定的变异概率对解空间中的每个粒子执行变异操作，进而丰富种群多样性，避免陷入局部最优。

基于熵的自适应变异粒子群算法的步骤为：

(1)初始化粒子群，确定种群规模 $N$，随机产生每个粒子的位置和速度；给定算法的最大加权系数 $w_{\max}$ 和最小加权系数 $w_{\min}$；设定参数 $v_{\max}$；设定算法的最大迭代次数 $G$；当前迭代次数 $g$=1；将每个粒子的 $p_i$ 设置为当前位置；$p_g$ 设置为 $f(x_i)$ 最小的那个粒子的位置。

(2)$g>G$(结束准则)，则结束算法并输出最优位置 $p_g$；否则继续执行。

(3)计算每个粒子的适应值 $f(x_i(g))$。若 $f(x_i(g)) < p_i$，则置 $p_i = x_i(g)$；同时，若 $\min f(x_i(g)) < p_g$，则 $p_g = \min f(x_i(g))$。

(4)按照公式计算种群的熵 $E$ 和变异概率 $p_m$。保留全局最优位置 $p_g$，对于每个粒子的位置和速度按照变异概率 $p_m$ 和公式(6)~(9)给定的变异算子进行变异操作，转步骤(2)。

## 3 仿真试验

为了验证改进算法的寻优效果，利用本文提出的基于熵的自适应变异的粒子群优化算法和标准 PSO 算法分别对下面经典的函数进行测试：

第一个测试函数：

$$f_1(x) = \sum_{i=1}^{n} x_i^2, \qquad (-100 \leqslant x_i \leqslant 100) \tag{10}$$

$f_1(x)$ 为单峰二次函数，理论最优值为 0。

第二个测试函数：

$$f_2(x) = \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10] \quad (-100 \leqslant x_i \leqslant 100) \tag{11}$$

$f_2(x)$ 是具有大量局部最优点的多峰函数，理论最优值为 0。

第三个测试函数：

$$f_3(x) = \sum_{i=1}^{n-1} [100 \times (x_{i+1} - x_i^2) + (x_i - 1)^2] \quad (-100 \leqslant x_i \leqslant 100) \tag{12}$$

$f_3(x)$ 为病态二次函数，很难极小化，理论最优值为 0。

两种优化算法的仿真实验设置的主要参数如下：群体规模 $N$=45，最大迭代次数 $G$=500，最大加权系数 $w_{\max}$=0.9，最小加权系数 $w_{\min}$=0.4，$v_{\max}$=5，$n$=10，每个粒子的速度 $v$=0。

仿真结果 1：利用粒子群算法和改进的粒子群算法分别对 3 个函数进行测试。针对每个函数，分别进行 10 次寻优。计算其 10 次寻优的平均最优值。

从表 1 可以看出，基于熵的自适应变异的粒子群优化算法的优化性能要优于标准粒子群优化算法。

表1 3个函数10次寻优的平均最优值

| 测试函数 | 理论最优值 | 标准PSO | 改进PSO |
|---|---|---|---|
| $f_1(x)$ | 0 | 0 | 0.000 021 423 57 |
| $f_2(x)$ | 0 | 2.585 662 195 4 | 0.151 454 737 3 |
| $f_3(x)$ | 0 | 0.308 139 824 5 | 0.045 918 551 0 |

仿真结果 2：为方便比较，在运用基于熵的自适应变异的粒子群算法和标准粒子群优化算法进行仿真试验时，分别记录算法每一次迭代的全局最优值。依次对 3 个函数的仿真实验进行统计分析，结果分别如图 1、图 2、图 3 所示。

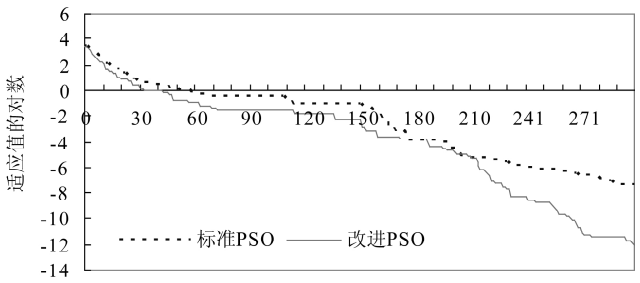


图 1 $f_1(x)$ 的变化趋势

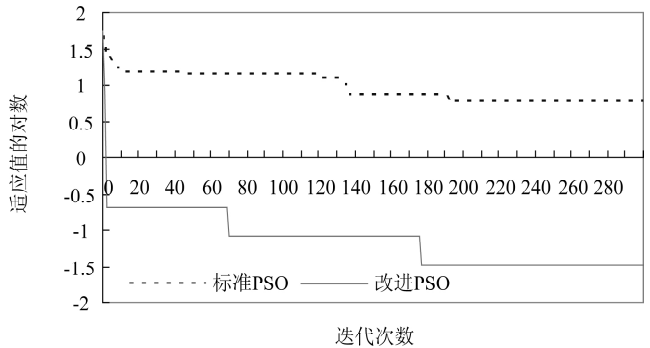


图 2 $f_2(x)$的变化趋势

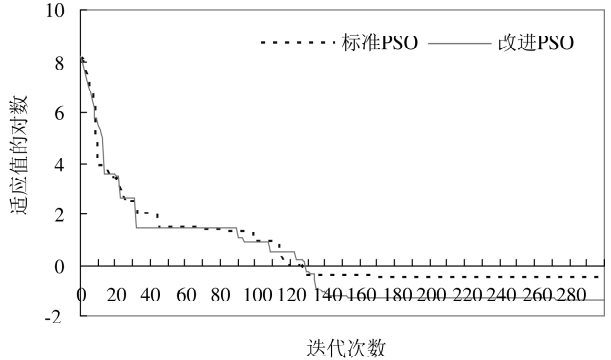


图 3 $f_3(x)$的变化趋势

从以上 3 个函数的变化曲线可以看出，标准的 PSO 算法虽然收敛速度快，但易于陷入局部最优。基于熵的自适应变异的粒子群算法虽然收敛速度稍慢，但是不易陷入局部最优。

粒子群算法在求解优化问题时，需要克服的难点之一就是如何克服早熟收敛。因此，本文提出了一种基于熵的自适应变异的粒子群优化算法。此算法引入变异算子到粒子群算法中，并根据种群熵这一反映种群多样性的指标自适应地调节变异概率和变异步长。仿真结果证明了基于熵的自适应变异的粒子群算法的可行性和有效性。

**参考文献**

[1] 张选平,杜玉平,秦国强.一种动态改变惯性权的自适应粒子群算法[J].西安交通大学学报,2005,39(10).

[2] 高海昌,冯博琴,候芸.自适应变异的混合粒子群优化策略及其应用[J].西安交通大学学报,2006,40(6):663-666.

[3] 吕振肃,候志荣.自适应变异的粒子群优化算法[J].电子学报,2004,32(3).

[4] 李宁,刘飞,孙德宝.基于带变异算子粒子群优化算法的约束布局优化研究[J].计算机学报,2004,27(7).

[5] 熊伟丽,徐保国,吴晓鹏.带变异算子的改进粒子群算法研究[J].计算机工程与应用,2006,26:1-3.

[6] 段晓东,高红霞,刘向东.一种基于种群熵的自适应变异的粒子群算法.计算机工程,2007,33(18):222-248.

[7] 李亚芳,李章维.一种基于信息熵的自适应遗传算法[J].浙江工业大学学报,2007,35(18).

(收稿日期：2009-01-12)