演化策略概述

姓名：马芮　　学号：21721196

May 10, 2018

目录

1 演化策略简介

1.1 概念与背景

演化策略是由德国的Rechenberg和Schwefel在20世纪60年代初期提出来的，目前已形成演化计算的一个重要分支。演化策略主要用于连续参数优化问题。

在下面的讨论中，假定问题是求解下列连续优化问题：

$$minf(x), x \in R^n$$

在更一般的情形，问题是求解下列约束优化问题：

$$minf(x), x \in D \subseteq R^n$$

其中$D = \{x \in R^n | G_j(x) \geq 0, j = 1, 2, ..., m, G_j(x)$为约束函数$\}$

1.2 演化策略的一般结构

早期的演化策略种群中只包含一个个体，称之为父体。在演化过程中，仅有一种遗传算子：变异。在每一演化代，通过将变异算子应用到父体上得到一个后代，然后将后代与父体进行比较，若后代比父体好且满足所有的约束条件，则后代成为下一代种群中的父体，否则父体保持不变。这种演化策略称为(1+1)演化策略。

1.2.1 (1+1)-演化策略

Algorithm 1: Procedure(1+1)Evolutionary Strategy

begin
$t \leftarrow 0$
initialize $(P(t) = \{x\}$ such that $G_j(x) \geq 0 (j = 1, 2, ..., m)$;
while *not termination condition* **do**
  $y \leftarrow x + N(0, \sigma)$;
  if $f(y) < f(x) and G_j(x) \geq 0 (j = 1, 2, ..., k)$ **then**
    $P(t+1) \leftarrow y$;
    else
      $P(t+1) \leftarrow x$;
    end
  end
  $t = t + 1$;
end
return the best solution;

下面用一个例子来说明(1+1)-演化策略的一步迭代.

考虑下列优化问题：

$maxf(x_1, x_2) = 21.5 + x_1 sin(4\pi x_1) + x_2 sin(20\pi x_2)$

其中$-3.0 \leq x_1 \leq 12.1, 4.1 \leq x_2 \leq 5.8$

假定第t代种群中的父体为$x^t = (x_1^t, x_2^t) = (5.3, 4.9)$, $\sigma = 1$

那么有$y^t = (y_1^t, y_2^t)$

$$y_1^t = x_1^t + N(0,1) = 5.3 + 0.4 = 5.7$$
$$y_2^t = x_2^t + N(0,1) = 4.9 - 0.3 = 4.6$$

由于$f(x^t) = f(5.3, 4.6) = 18.383705 < 24.849532 = f(5.7, 4.6) = f(y^t)$

且有$-3.0 \leq y_1^t \leq 12.1$,4,$1 \leq y_2^t \leq 5.8$,故$y^t$将替换$x^t$成为下一代种群中的父体，即有$x^{t+1} = y^t$

在高斯变异算子中，σ是算法的参数，它的值确定了x^t后代的变化范围，所以σ称为变异步长。

(1+1)演化策略没有体现种群的作用，本质上是一种局部搜索策略，具有明显的局限性。

随后，Rechenberg又提出了$(\mu+1)$-演化策略。后来Schwefel又提出了$(\mu+\lambda)$演化策略和(μ,λ)演化策略。

1.2.2 (μ, λ),$(\mu+\lambda)$-演化策略

Algorithm 2: Procedure (μ,λ),$(\mu+\lambda)$ Evolutionary Strategy

begin
$t \leftarrow 0$
initialize $(P(t))$with μ individuals;
evaluate$(P(t))$;
while *not termination condition* **do**
  for $i \leftarrow 1$ *to* λ **do**
    choose $\rho \geq 2$ parents $p_1, ...p_\rho$ from $P(t)$ at random;
    $o_i \leftarrow$recombine$(p_1, ...p_\rho)$;
    $o_i^{\sim} \leftarrow$mutate(o_i);
    evaluate$(o_i^{\sim})$;
  end
  $P(t+1) \leftarrow$ survivorSelect$(o_1^{\sim}, ..., o_\lambda^{\sim})$ or survivorSelect$(P(t) \bigcup \{o_1^{\sim}, ..., o_\lambda^{\sim}\})$;
  $t = t + 1$;
end
return the best solution;

2 演化策略的实现技术

2.1 表示

(1)二元表示
$(x,\sigma) = (x_1, x_2, ..., x_n, \sigma_1, \sigma_2, ..., \sigma_{n_\sigma})$ $n_\sigma = 1$或n
(2)三元表示
$(x,\sigma,\alpha) = (x_1, x_2, ..., x_n, \sigma_1, \sigma_2, ..., \sigma_n, \alpha_1, \alpha_2, ...\alpha_s)$

2.2 变异

下面讨论二元表示变异算子的实现方式.
(1) 当$n_\sigma = 1$时,变异算子下式实现

$$\begin{cases} \sigma' = \sigma * exp(\tau * N(0,1)) \\ x_i' = x_i + \sigma' * N_i(0,1) \end{cases} \tag{1}$$

其中τ是一个常数，由用户给定。通常有$\tau \propto 1/\sqrt{n}$经变异后所得到的后代为$(x_1', x_2', ..., x_n', \sigma')$
(2)当$n_\sigma = n$时，变异算子下式实现

$$\begin{cases} \sigma_i' = \sigma_i * exp(\tau * N(0,1) + \tau' * N'(0,1)) \\ x_i' = x_i + \sigma_i' * N_i(0,1) \end{cases} \tag{2}$$

其中$\tau' \propto 1/\sqrt{2n}$, $\tau \propto 1/\sqrt{2\sqrt{n}}$
经变异后所得到的后代为$(x_1', x_2', ..., x_n', \sigma_1', \sigma_2', ..., \sigma_n')$

2.3 重组

演化策略中的重组算子由两个或多个父体得到一个后代。为了得到λ个后代，需要运用重组算子λ次。

(1)离散重组假设两个父体分别为$x = (x_1, x_2, ..., x_n)$,$y = (y_1, y_2, ..., y_n)$,则由这两个父体重组得到的后代为$z = (z_1, z_2, ..., z_n)$,其中

$$Z_i = \begin{bmatrix} x_i,\text{若Random(2)=0} \\ y_i,\text{若Random(2)=1} \end{bmatrix} \tag{3}$$

(2)中值重组由两个父体重组得到的后代为$z = (z_1, z_2, ..., z_n)$

$$Z_i = \frac{x_i + y_i}{2}(i = 1, 2, ..., n) \tag{4}$$

上面两种重组可以以下列方式进行推广：为了得到后代$z=(z_1,z_2,...,z_n)$,对每个(i=1,2,...,n)，从种群中随机地抽取两个父体x和y，然后对x和y的第i个分量进行重组。

这种重组方式称为全局重组。若对每个分量进行离散重组，则称为全局离散重组。若每个分量进行中值重组，则称为全局中值重组。

2.4 父体选择

演化策略中的父体选择与个体的适应值无关。每当重组算子需要一个父体时，则从μ个个体所组成的种群中随机地选取。

2.5 存活选择

演化策略的存活选择策略有两种:(μ,λ)选择和$(\mu+\lambda)$选择。这两种选择策略都是确定性的。

在从μ个父代个体产生λ个后代并计算其适应值后,(μ,λ)选择从$\lambda(\lambda>\mu)$个后代中择优选择μ个个体作为下一代种群，而$(\mu+\lambda)$选择从μ个父代个体和λ个后代中择优选择μ个个体作为下一代种群。

3 演化策略的应用实例

3.1 Ackley函数极小化问题

$$minf(x_1,...,x_{30})=-20*exp\{-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30}x_i^2}\}-exp\{\frac{1}{30}\sum_{i=1}^{30}cos(2\pi*x_i)\}+20+e \quad (5)$$

$$-30\le x_i\le 30, i=1,2,...,30; e=2.71282$$

3.2 演化策略设计

(1)表示

$$(x_1,x_2,...,x_{30},\sigma_1,\sigma_2,...,\sigma_{30})$$

(2)适应函数：适应函数取为目标函数。

(3)重组算子：对变量部分使用离散重组，对策略参数部分使用全局中值重组。

(4)存活选择：使用(μ,λ)选择，其中$\mu=30$,$\lambda=200$。

(5)终止准则：当进行200000次函数值计算或发现最优解后终止算法。

(6)种群初始化：初始种群中每个个体的变量部分随机地产生，每个分量均匀地分布在区间[-30,30]内。每个个体的变异步长都相同，设为$\sigma=3$。

运行上述算法10次，每次找到的最好解都位于全局最优峰上。最后一代最好解的平均函数值为$7.48*10^{-8}$。