

# 智能算法

孔德瑞

2021 年 9 月 8 日

# 1 粒子群算法

粒子群算法是一种智能算法，最早由Kennedy and Eberhart (1995) 提出，其通过模拟鸟群觅食的行为寻找函数的最优解。具体来说，粒子群算法需要在函数定义域内某一具体范围内（通常依经验或问题具体情景确定）随机分配一定数量的“粒子”及其初速度，并允许“粒子”根据其自己经历的最优解信息和群体的最优解信息调整移动速度，从而接近并找到最优解。

下面以生产者成本最小化问题为例进行说明：

$$\begin{aligned} \min_{\mathbf{x}} \quad & c(\mathbf{x}) = \mathbf{w}^T \mathbf{x} \\ \text{s.t.} \quad & f(\mathbf{x}) \geq q \end{aligned} \quad (1)$$

其中  $\mathbf{x}_{n \times 1}$  代表各生产要素的投入量向量， $\mathbf{w}_{n \times 1}$  代表各要素的价格向量， $f(\cdot)$  代表生产函数， $q$  代表目标产量。

首先确定适应度函数为  $F(\mathbf{x}) = c(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ 。根据经验值确定不同要素的投入量的上下界，在此范围内生成符合约束的一定规模随机分布的粒子群，其中粒子  $n$  初始位置为  $\mathbf{x}_0^n$ ，并为每个粒子赋予特定的速度使得其不断移动以趋近最优解。定义一期内的粒子  $n$  的移动速度为：

$$\mathbf{v}_t^n = \mathbf{x}_t^n - \mathbf{x}_{t-1}^n \quad (2)$$

其中，初速度是  $\mathbf{v}_1^n$  是随机分布的，为了使得粒子群向最优解处移动， $t \geq 2$  时我们依照下式确定各期速度：

$$\mathbf{v}_t^n = \omega^n \mathbf{v}_{t-1}^n + c_1^n r_{1,t}^n (\mathbf{x}_{t-1}^{best(n)} - \mathbf{x}_{t-1}^n) + c_2^n r_{2,t}^n (\mathbf{x}_{t-1}^{globalbest} - \mathbf{x}_{t-1}^n) \quad (3)$$

其中  $\omega^n \mathbf{v}_{t-1}^n$  反映了本期粒子移动速度受上一期移动速度的影响，或者说受到“惯性”的影响，其存在能够使得粒子保持寻找全局最优的能力和充分利用本个体和群体信息能力的平衡；其中系数  $\omega^n$  代表了惯性权重，由外生给定。 $c_1^n r_{1,t}^n (\mathbf{x}_{t-1}^{best(n)} - \mathbf{x}_{t-1}^n)$  代表粒子移动时能够利用其自身经过的所有点的信息，向着自己经历的最优点方向移动； $c_2^n r_{2,t}^n (\mathbf{x}_{t-1}^{globalbest} - \mathbf{x}_{t-1}^n)$  代表粒子移动时能够利用其群体经过的所有点的信息，向着群体经历的最优点方向移动。其中  $c_1^n$  和  $c_2^n$  分别代表  $n$  粒子对于自身信息和群体信息的利用程度，一般为外生给定； $r_{1,t}^n$  和  $r_{2,t}^n$  分别代表  $n$  粒子在  $t$  期内受群体信息影响的程度，一般是随机数； $\mathbf{x}_{t-1}^{best(n)}$  和  $\mathbf{x}_{t-1}^{globalbest}$  分别代表粒子  $n$  和所有粒子经历过的所有点中的最优点。

这样，依据式3就能确定各粒子在不同时期的移动速度，在根据式2可知：

$$\mathbf{x}_t^n = \mathbf{x}_{t-1}^n + \mathbf{v}_t^n \quad (4)$$

所以便可以根据上一期粒子位置和本期粒子移动速度便可以确定本期粒子位置。通过确定合适的参数并迭代后，粒子群便能逐渐向最优解点收敛，因此我们便能得到最优点。

（下面根据网上整理的，没看包的程序可能不对）

由于生产者成本最小化问题是一个受约束的最优化问题，面对约束有两种可能的处理方式：

一是在确定迭代粒子位置时对于其新位置判定其是否满足约束条件，若不满足则重新确定其移动速度并生成新位置；

二是确定设定一个“惩罚项”将约束条件纳入目标函数，从而将问题转化为无约束的最优化问题。以成本最小化问题为例，我们需要设定惩罚项

$$e = \max\{0, q - f(\mathbf{x})\} \quad (5)$$

并设定目标函数  $F(\mathbf{x}, \delta) = c(\mathbf{x}) + \delta e$ ，其中  $\delta > 0$ ，这样我们只需求解：

$$\min_{\mathbf{x}} F(\mathbf{x}, \delta) = c(\mathbf{x}) + \delta e = \mathbf{w}^T \mathbf{x} + \delta \max\{0, q - f(\mathbf{x})\} \quad (6)$$

此时求得的最优解一定也是原成本最小化问题的最优解。

## 2 遗传算法

遗传算法同样是一种智能算法，由 John Holland 提出，最早见于 Holland (1975) 和 De Jong (1975)。其模仿了生物进化过程中遗传变异过程，包括其中的基因重组和基因突变的过程，以及进化中优胜劣汰的过程。具体为：设定一定规模的生物种群，每个个体有不同的基因型并因此由不同的环境适应度，之后种群内的个体配对经过基因的突变和重组产生后代并进行迭代。其中在选择父代时我们需要根据其适应度进行选择，来达到“优胜劣汰”的效果。这样，通过选择和迭代的过程，种群向最适应度最高的方向进化，从而能够产生最优解；重组保证种群进化的稳定性，而突变则保证了种群的多样性，避免种群向局部最优进化。

我们仍然以生产者成本最小化问题（式1）为例进行说明。

首先确定种群内各个体的基因型  $\mathbf{x}_0^n$ ，其在经验范围内随机分布。为了便于基因重组和交换的操作，我们需要对基因型进行编码，在这里我们采用二进制编码，即将向量  $\mathbf{x}$  中的各元素的

取值以二进制表示并按顺序排列。

之后进行选择。由于我们面临的是最小化问题，所以我们不妨规定  $F(\mathbf{x}) = -c(\mathbf{x}) = -\mathbf{w}^T \mathbf{x}$  作为适应度函数，这样就将问题转化为求适应度函数最大的问题。然后，根据适应度为各个体指定其被选择为父代的概率，保证适应度越高的个体被选择为父代的概率越大。之后依概率选择两个父代个体。

之后根据父代生成子代，即依次进行基因重组和突变的操作。基因重组，即两个父代的基因型中随机指定某一段对应位置进行交换；基因突变，即对交换后个体的基因型，每一个点的数值以某一（较小）的概率发生变化（由 0 变为 1 或由 1 变为 0）。通过基因重组和突变就能产生两个新的子代个体。遵循相同的方法，便可以生成和父代数量相同的子代种群。

这样不断进行迭代和进化，种群中个体便会向着最适应环境的方向进化，也就是逐渐趋近于最优解。

### 3 差分进化算法

差分遗传算法是智能算法的一种，最早由Storn and Price (1997) 基于遗传算法提出。其过程仍然模拟的是自然界中生物进化优胜劣汰的过程。但是与遗传算法不同的是，差分算法中变异（确切地说是突变）的过程与种群中个体差异相关。

我们仍然以生产者成本最小化问题（式1）为例进行说明。

首先确定种群内各个体的基因型  $\mathbf{x}_0^n$ ，其在经验范围内随机分布。之后进行变异的操作。对于任意一代的第  $n$  个个体  $\mathbf{x}_t^n$ ，选定种群中任意的三个个体  $\mathbf{x}_t^{p1}$ 、 $\mathbf{x}_t^{p2}$  和  $\mathbf{x}_t^{p3}$ ，以此生成一个备选的父代个体  $\mathbf{h}_t^n$ ，生成的规则如下：

$$\mathbf{h}_t^n = \mathbf{x}_t^{p1} + r(\mathbf{x}_t^{p2} - \mathbf{x}_t^{p3}) \quad (7)$$

其中  $r$  代表缩放因子，代表差分项  $\mathbf{x}_t^{p2} - \mathbf{x}_t^{p3}$  对于备选父代基因型的影响程度。

之后进行交叉操作。以备选的父代个体  $\mathbf{h}_t^n$  和第  $n$  个个体自身  $\mathbf{x}_t^n$  为父代，进行交叉生成一个备选的子代个体  $\mathbf{v}_t^n$ 。尽管基因型没有进行二进制编码，但是其代表的是一个向量，因此我们以固定的概率选择子代基因型中每一个元素  $v_{i,t}^n$  等于原个体  $\mathbf{x}_0^n$  中的元素或者备选的父代个体

$\mathbf{h}_t^n$  中的元素，即：

$$v_{i,t}^n = \begin{cases} x_{i,t+1}^n & \text{if } rand(0, 1) \leq p_c \\ h_{i,t+1}^n & \text{if } rand(0, 1) > p_c \end{cases} \quad (8)$$

其中  $p_c$  代表交叉的概率，是一个外生给定常数。

之后进行选择，根据适应度函数确定原父代个体  $\mathbf{x}_t^n$  或者备选的子代个体  $\mathbf{v}_t^n$  进入下一代种群。如此便可以确定新一代的种群各个体  $\mathbf{x}_{t+1}^n$ 。

这样通过变异和交叉的过程，我们能够保证种群个体多样性，而选择和迭代的过程保证了种群向适应环境的方向进化，也就是逐渐接近最优解。

## 参考文献

- De Jong, K. A. (1975). *Analysis of the behavior of a class of genetic adaptive systems* (Thesis).
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: University of Michigan Press.
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of icnn'95 - international conference on neural networks* (Vol. 4, 1942–1948 vol.4). doi:[10.1109/ICNN.1995.488968](https://doi.org/10.1109/ICNN.1995.488968)
- Storn, R., & Price, K. (1997). Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4), 341–359. doi:[10.1023/A:1008202821328](https://doi.org/10.1023/A:1008202821328)