

GA 算法原理

一、算法原理

遗传算法(Genetic Algorithm, GA)是由美国 J. Holland 教授于 1975 年在专著《自然界和人工系统的适应性》中首先提出。它参考了达尔文的进化理论：“物竞天择，适者生存”，是一种模拟了自然选择和遗传中发生的复制、交叉和变异等现象，从任一初始种群出发，通过随机选择、交叉和变异操作，产生一群更适合环境的个体，使群体进化到搜索空间中越来越好的区域，这样不断繁衍进化，最后收敛到一群最适应环境的个体，从而求得问题的优质解的随机化搜索算法。其基本思想是模拟自然界遗传机制和生物进化论而形成的一种过程搜索全局最优解的算法。

二、算法步骤

Step1)创建初始种群

随机生成一个包含多个个体的初始种群。每个个体都代表一个潜在的解。

Step2)计算适应度

针对每个个体分别计算它们在目标函数空间中的适应度。对于初始种群，此操作将执行一次，若执行完 **Step3)**后仍没有达到终止条件，则对新种群再进行一次 **Step2)**。

Step3)选择、交叉和变异

将选择、交叉和突变的遗传算子应用到种群中，产生新一代种群。新一代将会产生在当前具有优势的个体中。

选择：操作负责当前种群中选择有优势的个体。

交叉：操作从选定的个体创建后代。

变异：操作可以将每个新创建个体的一个或多个染色体值随机进行变化。突变通常以非常低的概率发生。

Step4)算法终止条件

在确定算法是否可以停止时，可能有多种条件可以用于检查。两种最常用的停止条件是：

①已达到最大迭代数。

②在过去的几代中，个体没有明显的改进。这可以通过存储每一代获得的最

佳适应度值，然后将当前的最佳值与预定的几代之前获得的最佳值进行比较来实现。如果差异小于某个阈值，则算法可以停止。

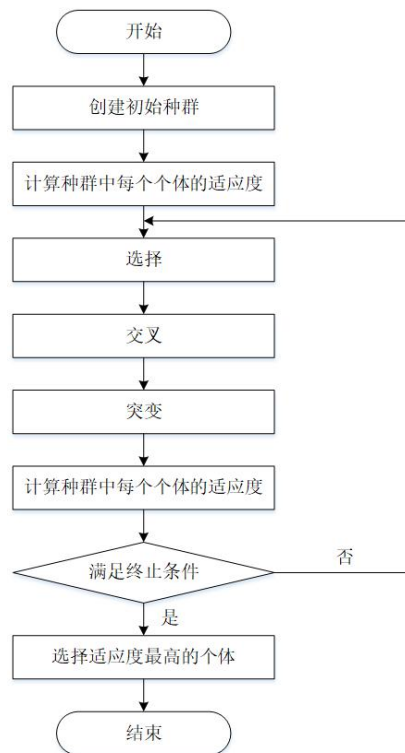


图 1 遗传算法(GA)流程图

三、问题求解

目标函数：

$$\begin{aligned} \max f(x) &= 2x_1 + 3x_1^2 + 3x_2 + x_2^2 + x_3 \\ s.t. \begin{cases} x_1 + 2x_1^2 + x_2 + 2x_2^2 + x_3 \leq 10 \\ x_1 + x_1^2 + x_2 + x_2^2 - x_3 \leq 50 \\ 2x_1 + x_1^2 + 2x_2 + x_3 \leq 40 \\ x_1^2 + x_3 = 2 \\ x_1 + 2x_2 \geq 1 \\ x_1 \geq 0 \end{cases} \end{aligned}$$

用 MATLAB 求解得：

$fval$ 的值为-16.0030， x_best 的值为 $[2.3353, -0.6664, -3.4454]$ 。

因此当 $x_1 = 2.3353, x_2 = -0.6664, x_3 = -3.4454$ 时，目标函数取到最大值-16.0030。