

# 基于python仿真的插值方法分析与实验

沈阳理工大学 徐佳文

本文主要针对基于python仿真中拉格朗日（Lagrange）插值法以及牛顿（Newton）插值法进行内插的分析与实验，对这两种常见的插值方法进行比较，分析其适用性。

随着科学技术的不断发展，在各种复杂工作的研究中仿真是一种不可或缺的手段。然而在仿真过程中，可能会出现数据丢失以及仿真结果不精确的问题。这一问题可通过使用插值方法对仿真结果行优化处理来解决。本文对两种插值算法进行分析实验，得出结论。

## 1 插值法

插值法是通过所有已知的点来构建一个插值函数 $f(x)$ ，对于未知的点 $x_i$ 可以利用插值函数 $f(x)$ 来算出函数的值 $f(x_i)$ ，用 $(x_i, f(x_i))$ 来近似的替代。对于在平面上相异的 $n+1$ 个点，我们必定可以构建一个 $n$ 次多项式。不同的插值法所要做的就是构造这个多项式函数，只是求得多项式的方式有些不同。

### 1.1 拉格朗日插值法

已知函数有 $n+1$ 个互不相同的已知点，其中自变量用 $x_j$ 表示， $x_j$ 对应的函数值用 $y_j$ 表示。拉格朗日插值多项式为：

$$L(x) = \sum_{j=0}^n y_j l_j(x) \tag{1}$$

其中插值基函数表达式为：

$$l_j(x) = \prod_{i=0, i \neq j}^n \frac{x - x_i}{x_j - x_i} = \frac{(x - x_0) \cdots (x - x_{j-1})(x - x_{j+1}) \cdots (x - x_n)}{(x_j - x_0) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_n)} \tag{2}$$

表1 差商表

节点	0 阶差商	1 阶差商	2 阶差商	3 阶差商	4 阶差商	...
$x_0$	$f(x_0)$					
$x_1$	$f(x_1)$	$f[x_0, x_1]$				
$x_2$	$f(x_2)$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$			
$x_3$	$f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$		
$x_4$	$f(x_4)$	$f[x_3, x_4]$	$f[x_2, x_3, x_4]$	$f[x_1, x_2, x_3, x_4]$	$f[x_0, x_1, x_2, x_3, x_4]$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$

表2 插值表

插值点	0. 18	4. 32	8. 03	13. 68	18. 70
拉格朗日插值	1. 5803727641919982	9. 443505807453578	6. 5749698143206230	19. 56254102367684	15. 852004446355743
拉格朗日插值总时间（s）	0. 5000066757202148				
牛顿插值	1. 5803727641919980	9. 443505807453576	6. 5749698143206246	19. 562541023676836	15. 852004446355721
牛顿插值总时间（s）	0. 21875405311584473				

### 1.2 牛顿插值法

牛顿插值法引入了差商的概念。已知函数 $f(x)$ 在 $n+1$ 个互不相同点处的函数取值为 $f(x_i)$ 。差商计算表如表1所示，高阶差商与低阶差商存在迭代关系，三角形指向为迭代运算方向。

$f[x_0, x_1] = \frac{f(x_0) - f(x_1)}{x_0 - x_1}$  为 $f(x)$ 在点 $x_0, x_1$ 处的一阶差商；

$f[x_0, x_1, x_2] = \frac{f[x_0, x_1] - f[x_1, x_2]}{x_0 - x_2}$  为 $f(x)$ 在点 $x_0, x_1, x_2$ 处的二阶差商；

$f[x_0, x_1, x_2, x_3] = \frac{f[x_0, x_1, x_2] - f[x_1, x_2, x_3]}{x_0 - x_3}$  为 $f(x)$ 在点 $x_0, x_1, x_2, x_3$ 处的三阶差商；

$f[x_0, x_1, \dots, x_n] = \frac{f[x_0, x_1, \dots, x_{n-1}] - f[x_1, x_2, \dots, x_n]}{x_0 - x_n}$  为 $f(x)$ 在点 $x_0, x_1, \dots, x_n$ 处的 $n$ 阶差商。

牛顿插值法的基本形式为：

$$N_n(x) = f(x_0) + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \cdots + f[x_0, x_1, \dots, x_n](x - x_0)(x - x_1) \cdots (x - x_{n-1}) \tag{3}$$

## 2 基于python的仿真方法设计

拉格朗日插值法与牛顿插值法的设计流程分别如图1、图2所示。

## 3 仿真结果与分析

分别对拉格朗日插值法和牛顿插值法进行实验，得到如图3、

图4所示的结果。图中三角形点代表真实值，圆点代表插值，曲线为插值拟合曲线。

随机取5个插值点，插值结果如表2所示。

由上述实验结果可看出两种插值方法结果精度相差较小，运算时间牛顿插值法较短。由于本

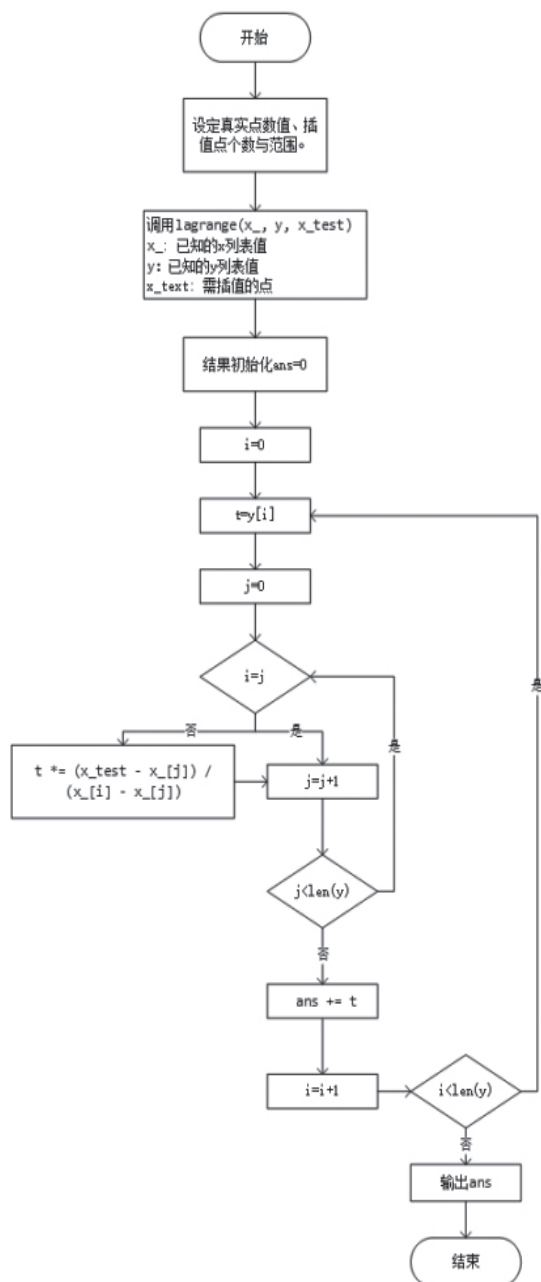


图1 拉格朗日插值法设计流程图

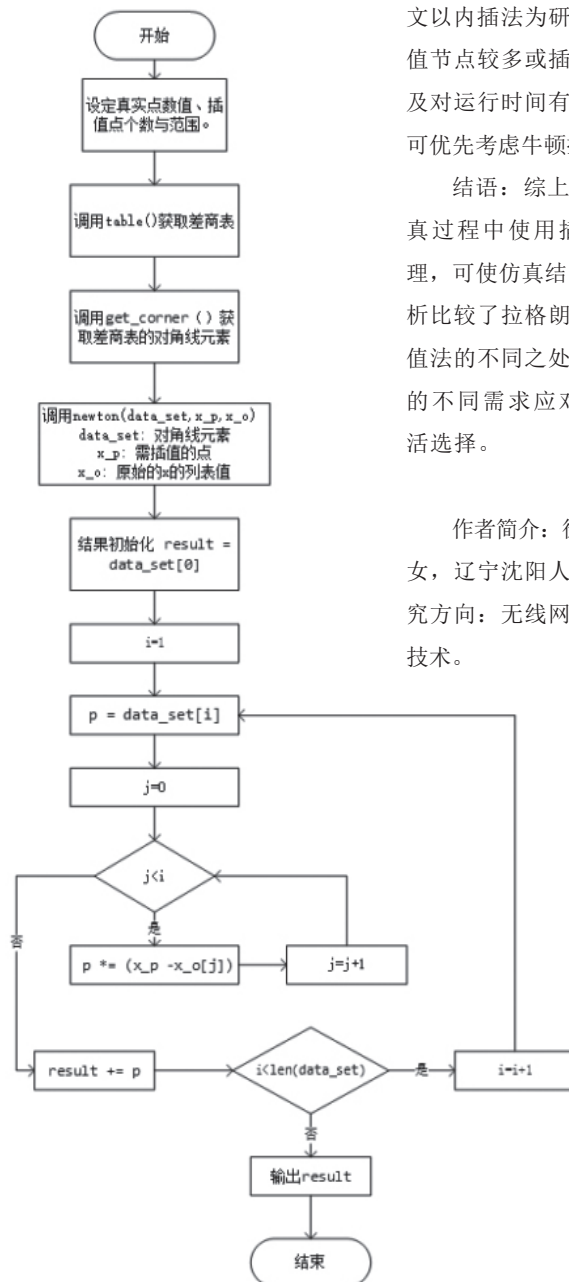


图2 牛顿插值法设计流程图

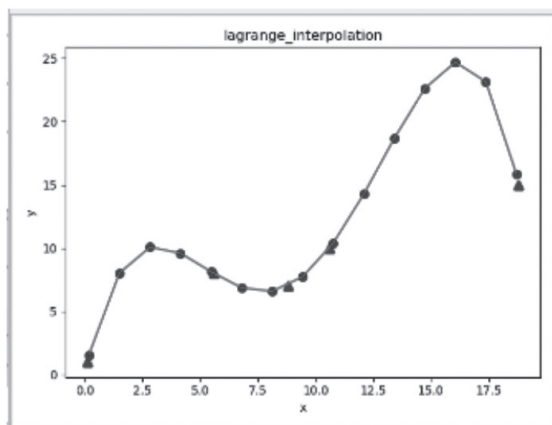


图3 拉格朗日插值法运行结果图

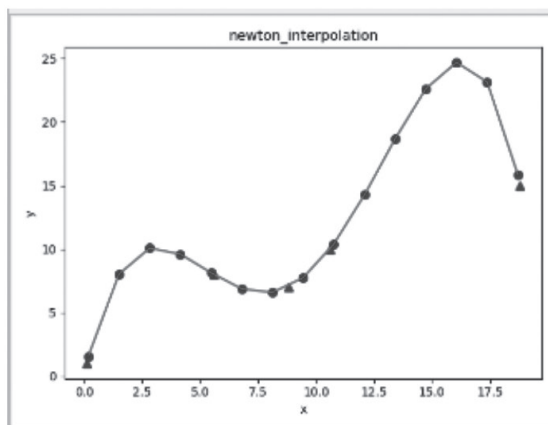


图4 牛顿插值法运行结果图

文以内插法为研究基础,因此当插值节点较多或插值节点增减频繁以及对运行时间有所要求的情况下,可优先考虑牛顿插值法。

结语:综上所述,在python仿真过程中使用插值法进行优化处理,可使仿真结果更精确。本文分析比较了拉格朗日插值法和牛顿插值法的不同之处,对于仿真过程中的不同需求应对插值方法进行灵活选择。

作者简介:徐佳文(1996—),女,辽宁沈阳人,硕士研究生,研究方向:无线网络信息传输与控制技术。