基于Matlab平台的插值法技术实现与应用

黄光东 管建和 李 响 武翠霞

摘 要 本文主要讨论插值法中Lagrange插值、Hermite插值、分段低次插值及三次样条插值，并应用Matlab软件中实现了这些插值法。在此基础上，我们用插值法解决了海底地形测量中的海底形状图绘制问题，取得了良好效果。

关键词 拉格朗日插值，分段低次插值，埃尔米特插值，三次样条插值

插值法是一种古老的数学方法，它来自生产实践。早在一千多年前，我国的数学家在研究历法上就应用了线形插值与二次插值，但它的基本理论和结果却是在微积分产生以后才逐步完善的，其应用也日益增多，特别是在计算机广泛使用以后，由于航空、造船、精密机械加工等实际问题的需要，使插值法在实践上和理论上显得更重要，并得到进一步发展，尤其是近几十年发展起来的样条插值，获得了更广泛的应用。本文主要讨论插值法中重要的拉格朗日（Lagrange）插值、埃尔米特（Hermite）插值、分段低次插值及三次样条插值，并且应用Matlab实现这些插值法。在此基础上我们应用插值法解决了在海底地形测量中的绘制海底形状图问题。

一、插值法的数学定义与几何解释

设函数 $y=(x)$ 在区间 $[a,b]$ 上有定义，且已知在点 $a\leq x_0\leq x_1\leq\cdots\leq x_n\leq b$ 上的值 $y_0,y_1,\cdots,y_n$，若存在以简单函数 $P(x)$，使

$$P(x_i)=y_i\ (i=0,1,2,\cdots\cdots,n)$$

成立，就称 $P(x)$ 为 $y=f(x)$ 的插值函数，点 $x_0,x_1,\cdots\cdots,x_n$ 称为插值节点，包含插值节点的区间 $[a,b]$ 称为插值区间，求插值函数 $P(x)$ 的方法称为插值法，若 $P(x)$ 是次数不超过 n 的代数多项式，即

$$P(x)=a_0+a_1x\cdots+a_nx^n$$

其中 a_i 为实数，就称 $P(x)$ 为的插值多项式，相应的插值法称为多项式插值。若 $P(x)$ 是分段多项式，就是分段插值。若 $P(x)$ 是三角多项式，就称三角插值。

从几何上看插值法就是求曲线 $y=P(x)$，使其通过给定的 $n+1$ 个点 (x_i,y_i)，$i=0,1,\cdots\cdots,n$，并用它近似已知曲线 $y=fx$。

二、拉格朗日（Lagrange）插值问题

Lagrange插值的方法是：对给定的n个插值节点 $x_1,x_2,\cdots\cdots,x_n$ 及对应的函数值 $y_1,y_2,\cdots,y_n$，利用n次Lagrange插值多项式，则对插值区间内任意x的函数值y可通过下式求

$$y(x)=\sum_{k=1}^{n}y_k\left(\prod_{\substack{j=1\\j\neq k}}^{n}\frac{x-x_j}{x_k-x_j}\right)$$

Lagrange插值法的Matlab编程实现见附录lagrange.m

在Matlab中调用Lagrange插值法只需输入命令 lagrange(x0,y0,x)下举一例：

例1 已知 sin0.32=0.3146, sin0.34=0.3335, sin0.36=0.3623, 试用Lagrange插值法求sin0.3367。

解：在Matlab命令窗口调用命令 lagrange(x0,y0,x)的程序如下：

```
x=0.32:0.02:0.36; y=[ 0.3146 0.3335 0.3523];
y1=lagrange(x,y,0.3367)
```

运行结果：y1=0.3304。

三、埃尔米特（Hermite）插值问题

不少实际问题中不但要求在节点上函数值相等，而且要求导数值也相等，甚至要求高阶导数值也相等，满足这一要求的插值多项式就是埃尔米特（Hermite）插值多项式。已知n个插值节点 $x_1,x_2,\cdots\cdots,x_n$ 及对应的函数值 $y_1,y_2,\cdots\cdots,y_n$ 和一阶导数值 $y_1,y_2,\cdots\cdots,y_n$。则计算插值区间内任意的函数值的Hermite插值公式为：

$$y(x)=\sum_{i=1}^{n}h_i[(x_i-x)(2a_iy_i-y'_i)+y_i]$$

其中 $h_i=\prod_{\substack{j=1\\ j\neq i}}^{n}(\frac{x-x_j}{x_i-x_j})^2$； $a_i=\sum_{\substack{j=1\\ j\neq i}}^{n}\frac{1}{x_i-x_j}$；

埃尔米特（Hermite）插值法的 Matlab 编程实现见附录 Hermite. m，在 Matlab 调用埃尔米特（Hermite）插值法只需输入命令 hermite(x0, y0, y1, x)下举一例。

例 2 对于给定数据如下表，试构造 Hermite 多项式，并给出 sin0. 34 的近似值：

x	0.31	0.33	0.35
sinx	0.3051	0.3240	0.3429
cosx	0.9523	0.9460	0.9394

解：在 Matlab 命令窗口调用命令 hermite(x0, y0, y1, x)的程序如下：

x0 = [0. 31 0. 33 0. 35];

y0 = [0. 3051 0. 3240 0. 3429]; y1 = [0. 9523 0. 9460 0. 9394];

y = hermite(x0, y0, y1, 0. 34); y1 = sin(0. 34);

运行结果： y = 0. 3335 y1 = 0. 3335。

四、分段低次插值与三次样条插值问题

1. 分段低次插值

在应用 Lagrange 插值多项式是一般人认为次数越高逼近的精度越好。19 世纪 Runge 给出了一个等距节点插值多项式不收敛的例子 $f(x)=\frac{1}{1+x^2}$。为了解决 Runge 问题，引入分段低次插值。分段低次插值就是通过插值点用折段连接起来逼近原曲线。Matlab 自身提供了内部函数 interp1 来实现分段低次插值，该函数应用多项式级数计算插值点，其一般的调用格式为：y_i = interp1(x, yxi, method)，其中 y 为函数值矢量，x 为自变量矢量，x 与 y 长度相同。x_i 为插值点的自变量矢量，method 为插值方法选项。对于一维插值 Matlab 提供了 4 种方法：

①邻近点插值（method = nearest）

②线形插值（method = linear）

③三次样条插值（method = spline）

④立方插值(method = cubic)

2. 三次样条插值

样条函数可以给出光滑的插值曲线，样条曲线在工程实践与科学应用中有广泛的应用。正是因为样条函数的重要性和广泛应用，Matlab 专门提供了样条工具箱和三次样条插值函数 spline。三次样条插值函数 spline 的调用方式为：

y = spline(x0, y0, x)，其中 y0 为函数值矢量，x0 为自变量矢量，x0 与 y0 长度相同。x 为插值点的自变量矢量.

分段低次插值与样条插值在任何一本 Matlab 书上都讲的很清楚，我们再此就不再细说。

五、插值法在海底地形测量的应用实例

例 3 假设有一组海底深度测量数据，采用插值方式绘制海底形状图。

解 (1) 为仿真，用以下模型产生一组分度稀疏的“海底深度测量数据”见图 1。

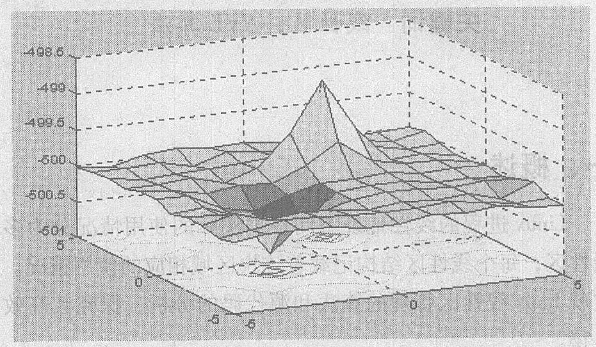

图 1

```
randn( ‘state’, 2)
x = -5: 5; y = -5: 5; [X, Y] = meshgrid(x, y);
zz = 1. 2 * exp( -((X-1). ^ 2 + (Y-2). ^ 2))...
  -0. 7 * exp( -((X+2). ^ 2 + (Y+1). ^ 2));
Z = -500 + zz + randn(size(X)) * 0. 05;
surfc(X, Y, Z); view( -25, 25)
```

(2) 通过插值画更细致的海底图，见图 2。

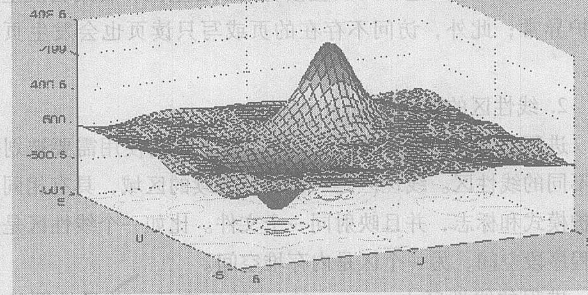

图 2

```
xi = linspace( -5, 5, 50);
yi = linspace( -5, 5, 50); [XI, YI] = meshgrid(xi, yi);
ZI = interp2(X, Y, Z, XI, YI, ‘ * cubic’);
surfc(XI, YI, ZI), view( -25, 25)
```

总结

插值法在工程和科学应用中有广泛的应用，几乎各个领

(下转第 36 页)

```
    node->vm_avl_right = noderightleft->vm_avl_left;
    noderightleft->vm_avl_right = noderight;
    noderightleft->vm_avl_left = node;
    noderight->vm_avl_height =  node->vm_avl_height =
heightrightleft;
    noderightleft->vm_avl_height = heightright;
    * nodeplace = noderightleft;
//根节点为右子树的左子树的根所替代
            }
        }
        else {    //如果插入新节点后以 node 为根的树平
        //衡没有被打破
        int height = (heightleft<heightright ? heightright :
heightleft) + 1;
/* 计算的当前 node 的新高度，如果高度没有变化，退出
循环，否则还要继续检查上一层节点的平衡因子 */
if (height == node->vm_avl_height)
            break;
        node->vm_avl_height = height;
/* 通过左右子树的高度得到 node 的新高度 */
        }
    }
}
```

线性区节点插入 AVL 树的过程就是搜索插入位置的过程，从根节点沿搜索路径逐层向下最终到达插入位置沿途经过的节点指针按顺序保存在 stack[ ]数组中；而节点插入后 AVL 树平衡化的过程就是从插入节点沿搜索路径到达根节点(从 stack_ptr 到 stack[0])的反向检测过程，如果某节点为根的子树的高度发生了变化，就要检查其父亲节点的平衡是否被打破，必要时采用 RL, RR, RLR, RRL 进行调整，调整后如果高度与插入节点前没有变化，就终止检查，此时树已平衡，否则还要继续检查更上一层节点。

三、结束语

进程的线性区操作是非常频繁的，因此其执行效率对系统性能影响很大，通过采用线性链表和 AVL 树两种方式来管理线性区，兼顾了搜索效率和调整开销，从而优化了性能。

参考文献

1. Linux 内核 2.4.0 源代码

2. Understangding The Linux Kernel

3. 数据结构(用面向对象方法与 C++描述)

(收稿日期：2005 年 11 月 20 日)

(上接第 31 页)

域都用插值法来对数据进行处理。使数据能够应用各学科的分析工具来分析数据所反映的对象的性质。

附录

```
Lagrange.m
function y=lagrange(x0, y0, x)
ii=1:length(x0);
y=zeros(size(x));
for i=ii
  ij=find(ii~=i); y1=1;
  for j=1:length(ij)
    y1=y1.*(x-x0(ij(j)));
  end
  y=y+y1*y0(i)/prod(x0(i)-x0(ij));
end
hermite.m
function y=hermite(x0, y0, y1, x)
n=length(x0);
m=length(x);
for k=1:m
  yy=0.0;
  for i=1:n
    h=1.0;
    a=0.0;
    for j=1:n
      if j~=i
          h=h*((x(k)-x0(j))/(x0(i)-x0(j)))^2;
          a=1/(x0(i)-x0(j))+a;
        end
      end
      yy=yy+h*((x0(i)-x(k))*(2*a*y0(i)-y1(i))
+y0(i));
    end
    y(k)=yy;
end
```

参考文献

1. 李庆扬等．数值分析．华中科技大学出版社

2. 张志涌等．精通 Matlab 6.5 版．北京航空航天大学出版社

3. 王沫然．Matlab 与科学计算（第二版）．电子工业出版社

4. 李海涛等．Matlab 程序设计教程．高等教育出版社

5. 苏金明等．Matlab 工具箱应用．电子工业出版社

6. 薛定宇等．高等应用数学问题的 Matlab 求解．清华大学出版社

7. 孙华．关于 Matlab 符号工具箱的若干问题．计算机应用 第二十卷第二期 2000.5

(收稿日期：2006 年 1 月 10 日)