本帖最后由 winner245 于 2013-10-28 00:45 编辑

**背景介绍**  
Matlab提供了一系列绘图函数，常见的包括绘制2D曲线的plot函数、绘制2D隐函数曲线的ezplot函数、绘制3D曲面的mesh和surf函数、绘制3D显函数曲面的ezmesh和ezsurf函数。值得注意的是，ez系列的绘图函数里只有ezplot是绘制隐函数曲线的，ezmesh和ezsurf都是画显函数曲面的（不要被ez的名字误解了）。遗憾的是，matlab里并没有提供直接绘制3D隐函数曲面的函数。本帖的目的就是归纳总结几种方便易用的绘制隐函数曲面的办法。

**问题描述**

如何绘制3元方程f(x, y,z) = 0确立的隐函数曲面z = g(x,y)？其中，方程f(x, y,z) = 0无法求解z关于x、y的表达式，即g(x, y)的显式表达式无法获取。

**准备工作——基础函数介绍**

为了解决上述问题，我们需要先对几个重要的图形函数isosurface、patch、isonormals取得初步的了解，如果您已经对这三个函数很熟悉，可以直接跳过这一步。 

l.  **isosurface 等值面函数**

调用格式：**fv = isosurface(X,Y,Z,V,isovalue)**

作用：返回某个等值面（由isovalue指定）的表面（faces）和顶点（vertices）数据，存放在结构体fv中（fv由vertices、faces两个域构成）。如果是画隐函数 v = f(x,y,z) = 0 的三维图形，那么等值面的数值为isovalue = 0。

2.  **patch函数**

调用格式：patch(X,Y,C) 以平面坐标(X, Y)为顶点，构造平面多边形，C是RGB颜色向量

                    patch(X,Y,Z,C)以空间3-D坐标(X, Y,Z)为顶点，构造空间3D曲面，C是RGB颜色向量

**patch(fv)** 通过包含vertices、faces两个域的结构体fv来构造3D曲面，fv可以直接由等值面函数isosurface得到  
例如：patch(isosurface(X,Y,Z,V,0))

3.  **isonormals等值面法线函数**

调用格式：**isonormals(X,Y,Z,V,p)**

实现功能：计算等值面V的顶点法线，将patch曲面p的法线设置为计算得到的法线（p是patch返回得到的句柄）。如果不设置法线的话，得到曲面在过渡地带看起来可能不是很光滑

有了上述三个函数后，我们已经具备间接绘制3D隐函数曲面的能力了。下面以方程  
**f(x,y, z) = x.\*y.\*z.\*log(1+x.^2+y.^2+z.^2)-10 = 0**为例，讲解如何画3D隐函数曲面。

**解决办法一：isosurface + patch+ isonormals**

**实现原理**：先定义3元显函数v =f(x, y, z), 则 v = 0 定义的等值面就是z = g(x,y)的3D曲面。利用isosurface函数获取v= 0 的等值面，将得到的等值面直接输入给patch函数，得出patch句柄p，并画出patch曲面的平面视角图形。对p用isonormals函数设置曲面顶点数据的法线，最后设置颜色、亮度、3D视角，得到3D曲面。

代码如下：

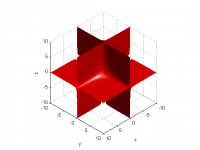
1. f = @(x,y,z) x.\*y.\*z.\*log(1+x.^2+y.^2+z.^2)-10;      % 函数表达式
2. [x,y,z] = meshgrid(-10:.2:10,-10:.2:10,-10:.2:10);       % 画图范围
3. v = f(x,y,z);
4. h = patch(isosurface(x,y,z,v,0));
5. isonormals(x,y,z,v,h)
6. set(h,'FaceColor','r','EdgeColor','none');
7. xlabel('x');ylabel('y');zlabel('z');
8. alpha(1)
9. grid on; view([1,1,1]); axis equal; camlight; lighting gouraud

复制代码

代码说明：

* alpha函数用于设置patch曲面的透明度（可以是0~1任意数值），1 表示不透明，0 表示最大透明度。如果想设置透明度为0.7，可以修改alpha(1)为alpha(0.7)。
* 使用此代码解决特定问题时，只需将第1行的函数表达式替换为特定问题的函数表达式，将第2行数据（x、y、z）范围换成合适的范围，后续代码无需任何变动。

得到图形：

   
[登录/注册后可看大图](http://www.ilovematlab.cn/member.php?mod=logging&action=login)

**解决办法二：Mupad**

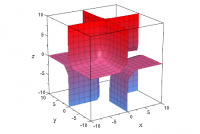
Mupad符号引擎里提供了现成的三维隐函数画图函数：Implicit3d

在matlab里开启Mupad的方法是：在commandwindow 里输入mupad 来启动一个notebook。在启动的notebook里再输入如下代码：

1. plot(plot::Implicit3d(x\*y\*z\*ln(1+x^2+y^2+z^2)-10,
2. x = -10..10, y = -10..10, z = -10..10), Scaling = Constrained)

复制代码

回车后得到如下图形：

   
[登录/注册后可看大图](http://www.ilovematlab.cn/member.php?mod=logging&action=login)

**解决办法三：第三方工具包ezimplot3**

在matlab central 的 file exchange 上有一个非常优秀的绘制3维隐函数的绘图函数，叫ezimplot3。感兴趣的可以在如下链接下载：

<http://www.mathworks.com/matlabcentral/fileexchange/23623-ezimplot3-implicit-3d-functions-plotter>

也可以直接从本帖下载： http://www.ilovematlab.cn/static/image/filetype/zip.gif [ezimplot3.zip](http://www.ilovematlab.cn/forum.php?mod=attachment&aid=MTE2MDk1fDk3YWU3OTgyfDE1MjMxNzU2MDR8MHwyNjQ0NzE%3D) (3.71 KB, 下载次数: 25501) 

ezimplot3一共有三种参数**调用方式**：

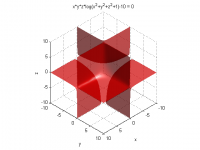
1. **ezimplot3(f)** 画函数f(X,Y,Z)= 0 在-2\*pi< X < 2\* pi, -2\* pi < Y < 2\* pi, -2\* pi < Z < 2\* pi上的图形
2. **ezimplot3(f, [A,B])**画函数f(X,Y,Z)= 0 在A< X < B, A < Y < B, A < Z < B上的图形
3. **ezimplot3(f, [XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX])**画函数f(X,Y,Z)= 0 在XMIN< X < XMAX, YMIN < Y < YMAX, ZMIN < Z < ZMAX上的图形

**ezimplot3使用方法**：解压ezimplot3.zip，将解压得到的ezimplot3.m 添加到matlab当前搜索路径后就可以使用了。然后，可以直接在command window 输入代码：

1. f = @(x,y,z) x\*y\*z\*log(1+x^2+y^2+z^2)-10;
2. ezimplot3(f,[-10,10]);  % [-10, 10] 表示图形范围x、y、z都在区间[-10, 10]

复制代码

即得到如下图形：

   
[登录/注册后可看大图](http://www.ilovematlab.cn/member.php?mod=logging&action=login)

若干说明：

1. **ezimplot3和方法一本质上完全相同**。即ezimplot3实际上也是基于isosurface+ patch + isonormals的实现
2. ezimplot3与方法一的图形视觉效果相同，唯一的区别是，ezimplot3的使用了0.7的透明度：alpha(0.7)
3. ezimplot3在方法一基础上增加了一些外包功能，如：允许函数句柄f是非向量化的函数（即函数定义无需.\*  ./  .^），这在ezimplot3内部会自动调用vectorize实现函数向量化。另外，ezimplot3可以在调用的时候方便的设定坐标范围。

**常见问题和解决办法：**

* **常见问题**：很多人在使用以上方法后，经常出现的问题是代码没有任何错误，程序可以运行，就是出来的图形只有一个空坐标轴，看不到图形。
* **问题分析**：出现这种问题的原因是图形的显示区域没设对。比如，我们上述三种方法都是在x为-10到10的范围内，如果你设的范围内本身就没有图形，那当然就看不到图形了。
* **解决办法**：把图形显示范围重新设置对即可，如果不知道图形的大致范围，就手工多改几次，直到看到图形为止

1. 方法一，图形范围是在第2句的meshgrid函数决定的，meshgrid里给出的x、y、z范围就是最终画图范围，修改meshgrid语句即可。
2. 方法二（Mupad），x =-10..10, y = -10..10, z = -10..10是表示显示范围，修改这里即可。
3. 方法三，用ezimplot3(f,[A, B]) ezimplot3(f, [XMIN,XMAX,YMIN,YMAX,ZMIN,ZMAX])两种方式控制图形显示范围。

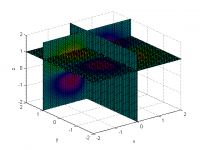
**后记：slice切片函数**

matlab还提供一种画切片图形的函数slice，slice做出的图是在切片上用颜色表示v的值。有时，我们画切片图形也有助于我们理解一个4维图形。以  v= f(x,y,z) = x\*y\*z\*exp(-(x^2+y^2+z^2))  为例，假设我们希望看 v =f(x,y,z) 在 x =0, y = 1, z = 1 这些平面切片的图形，我们可以用以下代码：

1. [x,y,z] = meshgrid(linspace(-2,2));
2. v = x.\*y.\*z.\*exp(-(x.^2+y.^2+z.^2));
3. xslice = 0; yslice = 1; zslice = 1;
4. slice(x,y,z,v,xslice,yslice,zslice)
5. xlabel('x'); ylabel('y'); zlabel('z');
6. colormap hsv

复制代码

得到图形为：

   
[登录/注册后可看大图](http://www.ilovematlab.cn/member.php?mod=logging&action=login)

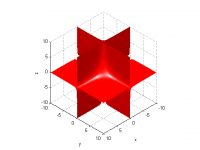
经常听有人说想画**“4D图形”**，前3维数据[x,y,z]表示空间位置，第4维数据v表示颜色（温度等），这类图形可以方便地通过slice切片实现： slice(x,y,z,v,xslice,yslice,zslice)，这里就是在指定的切片上在空间坐标[x,y,z]处，用v值指定颜色画图。关于这类 “4D图形”的画法的一个典型例子：<http://www.ilovematlab.cn/thread-265517-1-1.html>  
  
另外，我在 23 楼提供了一个slice 函数应用的生动例子：[**slice 3D 动画图形**](http://www.ilovematlab.cn/thread-264471-3-1.html)。感兴趣的朋友可以看看  
  
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
  
欢迎大家踊跃讨论，给出更多更好的办法

ezimplot3的方法实质是跟方法1是一样的。  
补充一个：MC(Marching Cube)方法，这是计算机图像学中很有名的方法。  
[http://www.mathworks.com/matlabc ... 2506-marching-cubes](http://www.mathworks.com/matlabcentral/fileexchange/32506-marching-cubes)

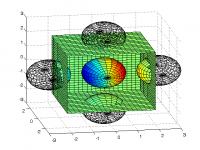
感谢2楼，下面补充第四种办法。  
  
**解决办法四：第三方工具包MarchingCubes.m**  
  
下载地址：[http://www.mathworks.com/matlabc ... ent/MarchingCubes.m](http://www.mathworks.com/matlabcentral/fileexchange/32506-marching-cubes/content/MarchingCubes.m)  
或者直接下载： http://www.ilovematlab.cn/static/image/filetype/zip.gif [MarchingCubes.zip](http://www.ilovematlab.cn/forum.php?mod=attachment&aid=MTE2MTA4fDVkNjE2MWY3fDE1MjMxNzU2MDR8MHwyNjQ0NzE%3D) (6.39 KB, 下载次数: 14277)   
  
代码如下：

1. f = @(x,y,z) x.\*y.\*z.\*log(1+x.^2+y.^2+z.^2)-10;      % 函数表达式
2. [x,y,z] = meshgrid(-10:.2:10,-10:.2:10,-10:.2:10);   % 画图范围
3. v = f(x,y,z);
4. [F,V] = MarchingCubes(x,y,z,v,0);
5. xlabel('x');ylabel('y');zlabel('z');
6. grid on; view([1,1,1]); axis equal on; camlight; lighting gouraud

复制代码



提供一个好玩的 slice 3D动画图形，希望大家对 slice 切片作图能有所了解。下面是动画瞬间的完整截图：

   
[登录/注册后可看大图](http://www.ilovematlab.cn/member.php?mod=logging&action=login)

运行下面的程序，可以得到动画效果，其中 nums 控制动画次数，pause(0.1)控制动画播放速度：

1. nums = 3;    % 动画次数
2. [x,y,z] = meshgrid(-2:.2:2,-2:.25:2,-2:.16:2);
3. v = x.\*exp(-x.^2-y.^2-z.^2);
4. [xsp,ysp,zsp] = sphere;
5. slice(x,y,z,v,[-2,2],2,[-2,2])
6. l = 0;
7. while l<nums
8. for i = -3:.2:3
9. if rem(l,2)==0
10. hsp = surface(xsp+i,ysp,zsp);
11. else
12. hsp = surface(xsp,ysp+i,zsp);
13. end
14. rotate(hsp,[1 0 0],90)
15. xd = get(hsp,'XData');
16. yd = get(hsp,'YData');
17. zd = get(hsp,'ZData');
18. delete(hsp)
19. hold on
20. hslicer = slice(x,y,z,v,xd,yd,zd);
21. axis tight
22. axis([-3 3 -3 3 -3 3])
23. view(-10,20)
24. drawnow
25. pause(0.1)
26. delete(hslicer)
27. hold off
28. end
29. l = l + 1;
30. end

复制代码