**Matlab中使用Plot函数动态画图方法总结**

本文来自: [MATLAB爱好者论坛(MATLABFAN)](http://www.matlabfan.com/) 作者: admin 日期: 2009-8-17 11:23 阅读: 550 人 [打印](http://www.matlabfan.com/viewthread.php?action=printable&tid=736) [收藏](javascript:bookmark()) 大 中 小

[Plot](http://www.matlabfan.com/tag-Plot.html), [Matlab](http://www.matlabfan.com/tag-Matlab.html), [函数](http://www.matlabfan.com/tag-%BA%AF%CA%FD.html), [动态](http://www.matlabfan.com/tag-%B6%AF%CC%AC.html), [画图](http://www.matlabfan.com/tag-%BB%AD%CD%BC.html)

|  |
| --- |
| Matlab除了强大的矩阵运算，仿真分析外，绘图功能也是相当的强大，静态画图没什么问题，由于Matlab本身的多线程编程缺陷，想要动态的画图，并且能够很好的在GUI中得到控制，还不是一件很容易的事情，下面总结几种方法。    **一. AXIS 移动坐标系**     这种方法是最简单的一种方法，适合于数据已经全部生成的场合，先画图，然后移动坐标轴。实例代码如下：   1. %% 2. %先画好，然后更改坐标系 3. %在命令行中 使用 Ctrl+C 结束 4. t=0:0.1:100\*pi; 5. m=sin(t); 6. plot(t,m); 7. x=-2\*pi; 8. axis([x,x+4\*pi,-2,2]); 9. grid on 10. while 1 11. if x>max(t) 12. break; 13. end 14. x=x+0.1; 15. axis([x,x+4\*pi,-2,2]); %移动坐标系 16. pause(0.1); 17. end   复制代码  **二. Hold On 模式**              此种方法比较原始，适合于即时数据，原理是先画上一帧，接着保留原始图像，追加下一幀图像，此种方式比较繁琐，涉及画图细节，并且没有完整并连续的Line对象数据。     例如：   1. %% 2. % Hold On 法 3. % 此种方法只能点，或者分段划线 4. hold off 5. t=0; 6. m=0; 7. t1=[0 0.1]; %要构成序列 8. m1=[sin(t1);cos(t1)]; 9. p = plot(t,m,'\*',t1,m1(1,:),'-r',t1,m1(2,:),'-b','MarkerSize',5); 10. x=-1.5\*pi; 11. axis([x x+2\*pi -1.5 1.5]); 12. grid on; 13. for i=1:100 14. hold on 15. t=0.1\*i; %下一个点 16. m=t-floor(t); 17. t1=t1+0.1; %下一段线(组) 18. m1=[sin(t1);cos(t1)]; 19. p = plot(t,m,'\*',t1,m1(1,:),'-r',t1,m1(2,:),'-b','MarkerSize',5); 20. x=x+0.1; 21. axis([x x+2\*pi -1.5 1.5]); 22. pause(0.01); 23. end   复制代码  **三. Plot 背景擦除模式**        这种模式比较适合画动画，效率比较高，刷新闪烁小，适合即时数据，最终的Line结构数据完整。        了解此方法之前要搞清楚 Plot函数的原型是什么： Plot函数，输入为 X-Y (-X)坐标元组、以及“属性”-“值对，输出为一个列向量（每条曲线岁对应的Line结构 Handle，每一行代表一个 线条的handles）, 每一线条都有 XData，YData 向量。如果你画了2条线，那么会返回 2×1的向量。 重新画图不需要 重新书写 Plot，只需要 刷新图像即可，使用drawnow函数。  完整实例如下：  1. 画一个点的动画：   1. %% 2. %采用背景擦除的方法，动态的划点，并且动态改变坐标系 3. % t,m 均为一行 ，并且不能为多行 4. t=0; 5. m=0; 6. p = plot(t,m,'\*',... 7. 'EraseMode','background','MarkerSize',5); 8. x=-1.5\*pi; 9. axis([x x+2\*pi -1.5 1.5]); 10. grid on; 11. for i=1:1000 12. t=0.1\*i;       %两个变量均不追加 13. m=sin(0.1\*i); 14. set(p,'XData',t,'YData',m) 15. x=x+0.1; 16. drawnow 17. axis([x x+2\*pi -1.5 1.5]); 18. pause(0.1); 19. end   复制代码  2. 动态多条曲线(即时数据)   1. %% 2. %采用背景擦除的方法，动态的划线，并且动态改变坐标系 3. % 多行划线 4. t=[0] 5. m=[sin(t);cos(t)] 6. p = plot(t,m,... 7. 'EraseMode','background','MarkerSize',5); 8. x=-1.5\*pi; 9. axis([x x+2\*pi -1.5 1.5]); 10. grid on; 11. for i=1:1000 12. t=[t 0.1\*i];                   %Matrix 1\*(i+1) 13. m=[m [sin(0.1\*i);cos(0.1\*i)]]; %Matrix 2\*(i+1) 14. set(p(1),'XData',t,'YData',m(1,:)) 15. set(p(2),'XData',t,'YData',m(2,:)) 16. drawnow 17. x=x+0.1; 18. axis([x x+2\*pi -1.5 1.5]); 19. pause(0.5); 20. end   复制代码  上面的这几个画图方式的示例只是简单的for循环，是单线程的，如果是涉及到GUI的编程，那么请使用Timer来完成这件事情，Timer是我在Matlab中实现多线程唯一方法(没有找到别的方法). |