

宏包制作测试

Eureka

2023 年 6 月 11 日

1 测试调用宏包时传递选项

下面是宏包 `FunctionPlot` 的一个选项（彩蛋）：

Surprise, You Have Found The Hidden Command!

2 测试宏包文本命令

这个其实就是在之前的 `tcolorbox` 宏包中学到的样式，所以我就定义了这些彩色文本框：`Note formal`, `Note tformal`, `Note warning`, 其实样式还是很简单的，目前就只有下面三个：

没有标题的 `formal` 环境，自己定义颜色即可

Title

相比于 `formal` 环境，多了一个标题选项，所以这个环境也叫做 `tformal`

:

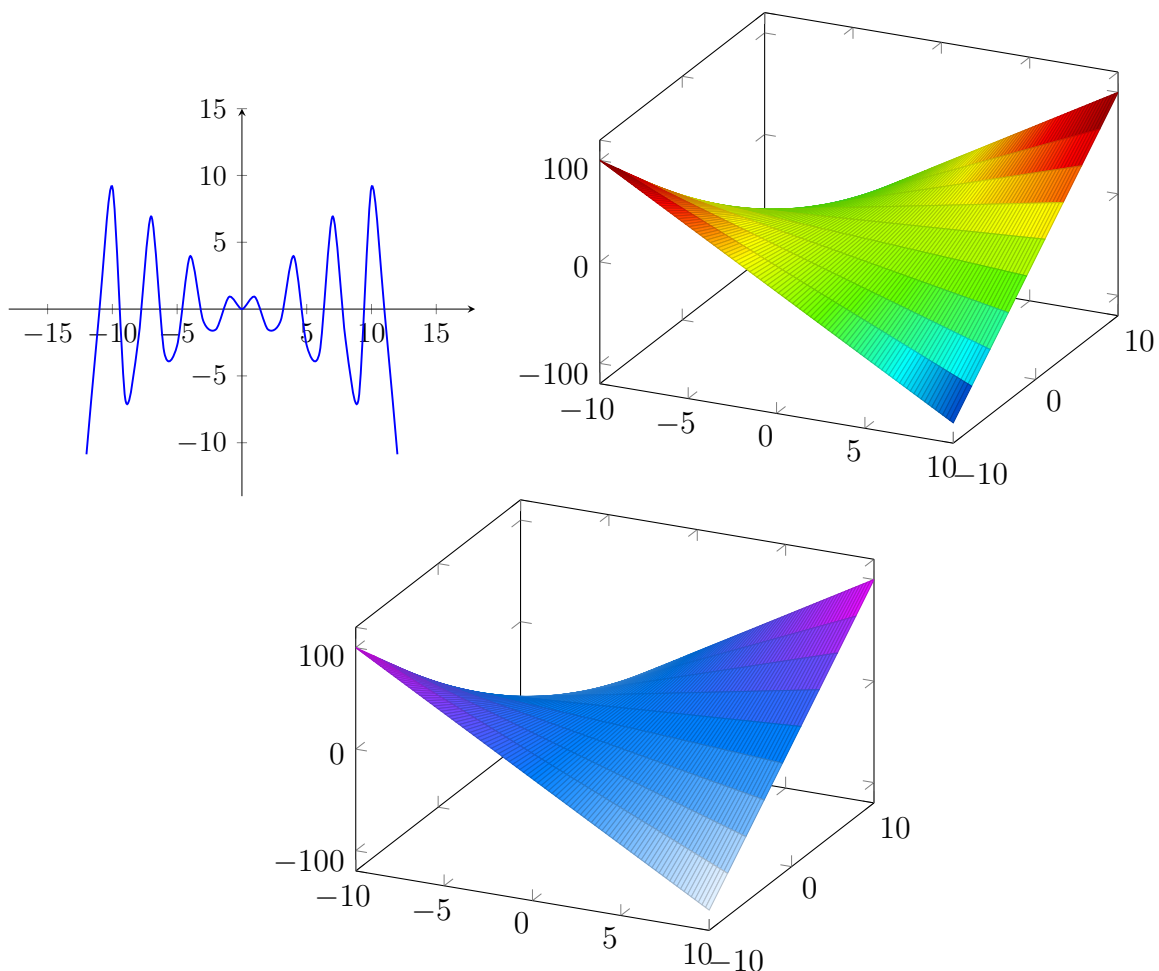
! 这个中间的内容就是警告

3 绘图命令

自己在之前的命令基础上又重新定义了两个使用 GNU Plot 的绘图命令，这样可以使得编译速度进一步加快。

3.1 函数绘图

至于普通的二三维的函数绘图十分简单的，你可以指定一下绘制的颜色，或者说是 `colormap`，定义域之类的 `plot parameters`，尽情发挥。下面我们主要绘制了 $y = x \sin(2x)$, $z = x * y$ 两个函数，效果还是可以的。



3.2 参数方程绘图

主要是在 vscode 中定义了下面 5 个绘制命令的 trigger

```

1 trigger    --> 展开式
2 plot2d     --> \Gplot[scale]{color}{f(x)}
3 plot3d     --> \Gplotz[scale]{colormap style}{f(x, y)}
4 polarplot  --> \polarplot[scale][plot parameters]{f(\t)}
5 paraplot2d --> \paraplot[scale][plot parameters]{{x(t)},{y(t)}}
6 paraplot3d --> \paraplotz[scale][plot parameters]{{x(t)},{y(t)}}

```

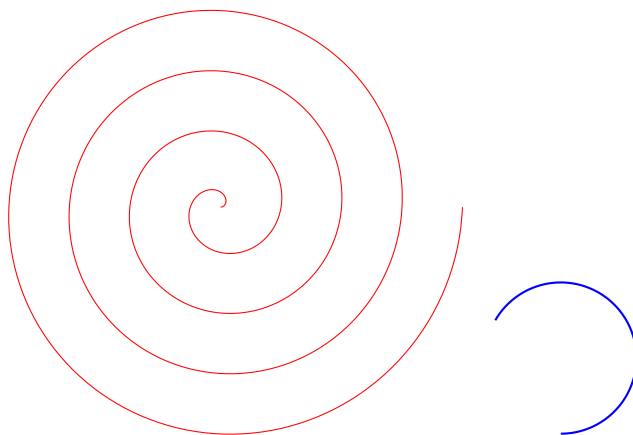
其实参数方程作图主要就是 Note 极坐标，Note 二维参数方程，Note 三维参数方程 这三种常见的情形。

那么我们就首先绘制极坐标的图形，下面我们绘制图形对应的方程分别为：

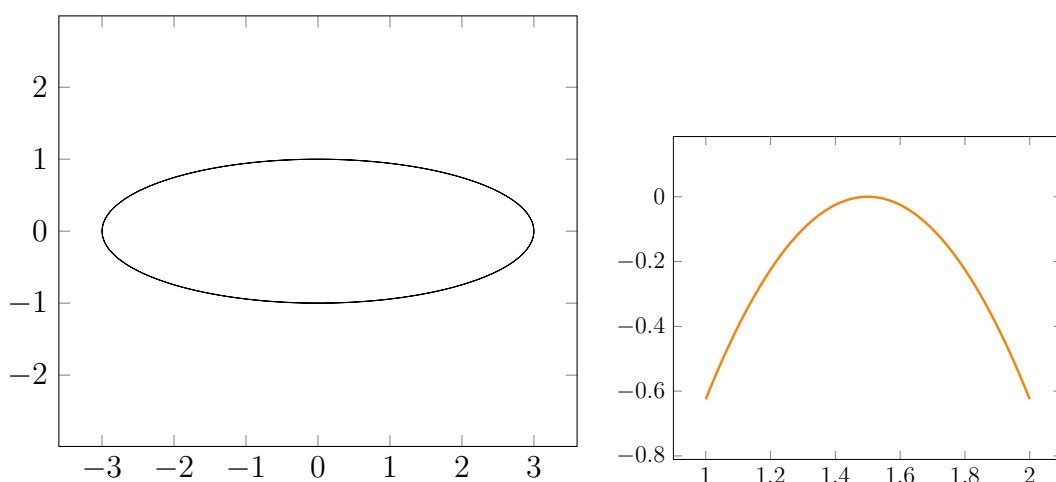
$$\rho = \frac{0.01}{\pi\theta} \quad (1)$$

$$\rho = \sin(\theta) \quad (2)$$

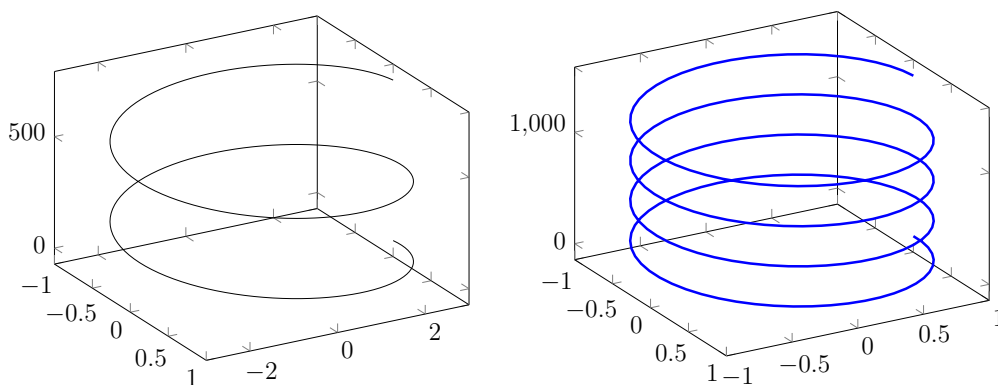
只需要注意一下，就是 polarplot,paraplot 默认使用的是角度，而非弧度



既然极坐标我们能够画出来了，那么接下来就是参数方程了，



最后就是我们的三维参数方程了，气质也是和上面的二维方程一样的，因为我们默认 $z(t) = t$ ，演示效果如下：下面我们绘制了螺旋线的方程 $x = \sin(t), y = 3 * \cos(t)(y = \cos(t)), z = t$



4 图片插入

其中有一个事情需要注意，由于 svg 的支持并不是很好，所以我没有把 svg 集成到宏包中，想要调用 svg 矢量图，你需要安装了 Inkscape，并且在导言区导入 svg 宏包。



图 1: 主人公 Doge 图示

5 测试 MMA

同理，对应的 MMA 模块我也归纳到了 FunctionPlot 宏包中，用于调用 MMA 生成对应的 pdf 矢量图片

5.1 计算

$$1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} + O(x^6) \quad (3)$$

5.2 图片插入

MMA 图片测试，下面的这个绘图还是比较复杂的，于是我们使用 MMA 绘制。

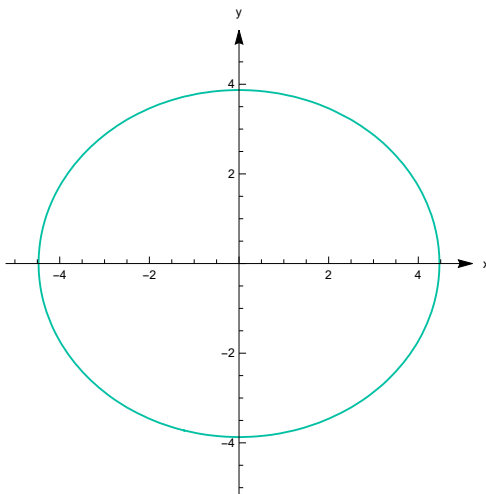


图 2: MMA 二维图形

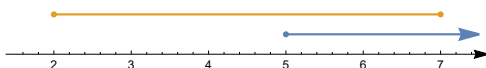


图 3: MMA 二维图形 2

二维的一些图形绘制了之后，自然要去绘制一些 MMA 中的 3 维对象了。下面就是一些例子：

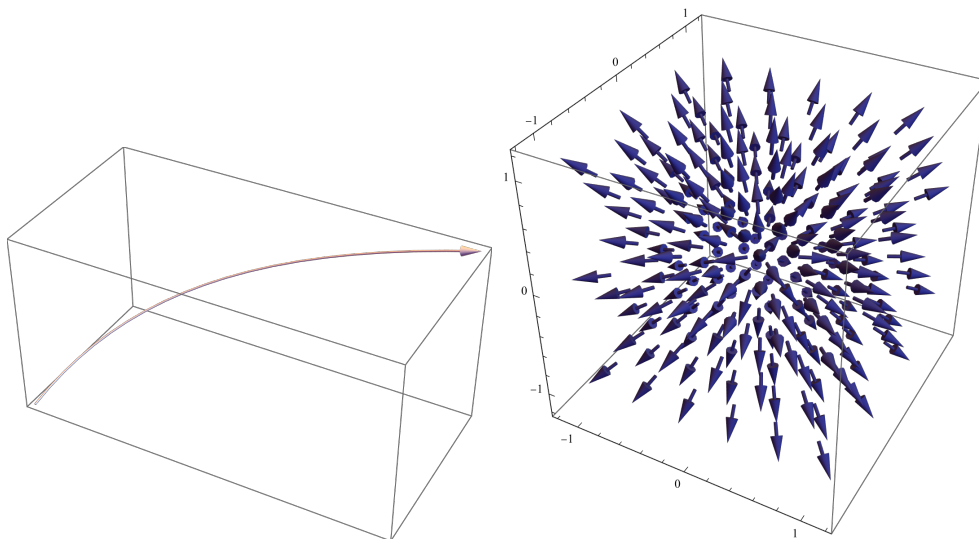


图 4: MMA 三维图形

5.3 表格功能

MMA 还能够解方程，微分方程，求根公式，输出表格等等.