

zTi**k**Z 接口文档

Eureka

由于本人时间有限, 目前此宏包的开发暂停.

2025 年 5 月 23 日

总目录

1	基本介绍	3	5.3	cache 库	21
	1.1 项目地址	3	5.4	python 库	23
	1.2 功能概述	3	5.5	wolfram 库	27
	1.3 坐标对齐	4	5.6	l3draw 库	32
	1.4 缓存机制	4	6	附录	36
	1.5 局限	4	6.1	gnuplot Support Functions .	36
2	安装使用	5	6.2	marker style	38
	2.1 兼容情况	5	6.3	测试数据/代码	39
	2.2 环境配置	5	7	TODO	40
	2.2.1 gnuplot	5	8	zTikZ 源码	41
	2.2.2 Python	5	8.1	ztikz.sty	41
	2.2.3 Wolfram	5	8.2	Library	46
	2.2.4 Mathics	7	8.2.1	basic	46
3	宏包选项	8	8.2.2	gnuplot	54
4	杂项	9	8.2.3	cache	58
5	zTikZ 库	10	8.2.4	python	60
	5.1 basic 库	11	8.2.5	wolfram	63
	5.2 gnuplot 库	16	Index		69

1 基本介绍

直到今天为止, 其实已经有很多基于 `tikz` 开发的绘图宏包了, 它们有着不同的用途, 在不同的领域中你都能看到 `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 的痕迹. 部分宏包已经提供了和 `ztikz` 功能差不多接口, 这系列的宏包包括:

- `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 的常见命令封装: `tzplot`;
- 用于 3D 绘图的 `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 宏包: `tikz-3dplot`;
- 基于 `PSTricks` 的 (特殊) 函数绘制宏包: `pst-func`;
- 用于缓存编译结果的宏包: `robust-externalize`;
- ...

如果你觉得 `ztikz` 宏包并不符合你的需求, 不妨试试上面的几个宏包, 或者是直接使用原始的 `tikz` 宏包提供的命令与库进行绘图. 在网络上也有着丰富的 `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 资源; 比如 `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 绘图的网站 – `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` `Example`, 这个网站中有着丰富的绘制样例并且提供了对应的绘图代码.

但是上述的系列宏包提供的接口并不是那么的统一, 自己用着不习惯, 所以我才决定开发 `ztikz` 宏包. `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 的命令格式基本遵守了类似 `Mathematica` 中函数的命名规范.

1.1 项目地址

本宏包在 Github 上的地址如下:

https://github.com/zongpingding/zTeX_bundle

该仓库中包含本宏集的源码, 用户手册以及一些测试用例; 当前宏集的稳定版本于半年之前发布, 最新的开发版请切换到 “dev” 分支; 本手册适用于当前最新的开发版.

1.2 功能概述

`zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 宏包主要用于绘图与计算, 支持调用外部程序, 比如 `Python`, `Mathematica`, `gnuplot`; 同时也提供了调用缓存机制; 虽然 `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 提供了这些软件的调用接口, 但这并不意味着你需要安装以上的所有软件; 在 `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 中每一个软件的调用接口是独立的, 用户仅需在操作系统上安装自己需要功能对应的软件即可. `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 的功能概述如下:

- **绘图:** 二维绘图, 三维绘图;
- **计算:** 浮点数计算, 符号计算.

绘图部分基于: `Ti $\kern-0.1em\textcolor{brown}{k}$ Z` 的 2d 绘图部分,¹ `Python` 的 `matplotlib` 库, `WolframScript` 的绘图功能; 计算部分基于: `LATEX3` 的 `xfp` 模块, `Python` 的 `numpy`, `sympy` 和 `scipy` 库, 以及 `WolframScript` 的计算功能.

虽然这个宏集名字中仅有 “`Ti $\kern-0.1em\textcolor{brown}{k}$ Z`” 字样, 但是 `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 能够完成 (或想要完成) 的功能是不止于此的.

¹ 由于 3d 绘图涉及的几个变换矩阵接口我还没想好怎么在 `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 中声明, 所以目前 `zTi $\kern-0.1em\textcolor{brown}{k}$ Z` 不提供 3d 绘图功能

1.3 坐标对齐

`ztikz` 提供的所有绘图命令可以和 `TikZ` 中的命令配合使用, 即 – 它们可以在同一个 `tikzpicture` 环境中使用. `ztikz` 对函数绘制时的坐标进行了“对齐”: `ztikz` 命令中的坐标, 和 `TikZ` 命令中的坐标, 亦或者是 Geogebra 中的坐标是一致的.

为何要在 `ztikz` 中把坐标“对齐”? 试想这么一个情景: 你在 Geogebra 中找到了两个函数图像的交点为 $P(1,2)$, 首先使用 `TikZ` 自带的 `\filldraw` 命令把 P 点绘制出来了; 然后使用 `ztikz` 中的 `\ShowPoint` 命令再次绘制这个 P 点. 然而结果就是: 这两个 P 点没有重合, 尽管我们指定的坐标都是 $(1,2)$.

所以当你不方便使用 `ztikz` 求解某些特殊的点时, 你可以先在诸如 Geogebra 这样的软件中把对应的 P 点求解出来, 然后直接在 `ztikz` 中使用 `\ShowPoint` 命令绘制此点.

1.4 缓存机制

`ztikz` 除了提供和外部程序交互的接口外, 还内置了一套 cache 系统, `ztikz` 会自动把 \TeX 和外部程序交互产生的结果缓存下来, 并且记录下 \LaTeX 文档中调用部分源代码的 Hash 值.

如果 \LaTeX 文档中的源代码对应的 Hash 值发生了改变, 那么 `ztikz` 就会重新和外部程序交互, 重新产生结果, 然后缓存新的 Hash 值. 如果文档中的源代码的 Hash 值没有改变, 那么 `ztikz` 就会直接调用上一次的缓存结果. cache 系统的优势: 我们不必反复的编译没有变化的内容, 直接引用之前的缓存, 减少文档的编译时间. 在实际测试中, 结果缓存后, 再次编译源文档的时间和直接插入对应数量的图片的时间几乎一致.

`ztikz` 中的 `basic`, `python`, `wolfram`, `gnuplot` 库均已实现缓存机制. `tikzpicture` 环境或者是 `\tikz` 命令生成图片的 cache 机制是依靠 `TikZ` 的 `external` 库实现的; (它的实现是出了名的复杂, 用户如果感兴趣, 也可以去看看.)

因为 `ztikz` 还没有进行完整的测试, 所以可能存在没有发现的 bug; 例如, 用户可能会遇到类似下面的问题:

- 过时的缓存 Hash 值: 如果一个环境最开始的 Hash 值为“A”, 在你修改了这个环境的内容后, 使得此环境中代码的 Hash 值变为“B”. 但是如果你现在再次修改会 Hash 值为“A”时对应的源代码, 此刻的 Hash 值已经缓存在了文件 `ztikz.hash` 中, 所以再次编译时此环境对应的绘制结果并不会改变. 调用的缓存结果仍然是 Hash 值为“B”对应的那个缓存结果.
- 和 `indextool` 宏包冲突: 有可能你在启用缓存库后, 发现编译报错 `missing \begin{document}....`. 这个问题和宏包 `indextool` 的索引功能有关. 可以先注释 `\makeindex`, `\printindex` 命令, 随后在图片缓存结束后, 取消注释, 最后再生成索引.

1.5 局限

`ztikz` 未来也许会提供 3d 绘图相关的接口, 但是如果你的图像需要复杂的计算或布局, 那么还请使用其余的宏包或使用对应的专业绘图软件. `asymptote` 宏包就是一个比较好的选择.

2 安装使用

2.1 兼容情况

目前 ztikz 宏包兼容 Windows/Linux/MacOS 三个平台. 各个平台中不同 T_EXLive 版本的兼容性如下:

Windows : T_EXLive 最低版本 2023

Linux : T_EXLive 最低版本 2022

MacOS : MacT_EX 最低版本 2024

zTi**k**Z 在 Windows 下的表现可能没有在 Linux/MacOS 下的那么好, 建议用户在 Linux/MacOS 下使用本宏包.

2.2 环境配置

如果用户需要使用 zTi**k**Z 提供的调用外部程序的库, 用户不仅需要配置文档的导言区, 还需在系统中安装对应的应用程序; 应用程序安装后需要将其添加到环境变量, 使得该应用可以在命令行被调用. 最后在编译文档时加上 `--shell-escape` 参数, 就像下面这样:

```
pdflatex --shell-escape main.tex
```

在 Windows 下推荐用户使用 **scoop** 这一包管理器安装一系列的软件, 这样可以免去配置环境变量这一烦恼. 以下是不同程序在配置过程中需要注意的事项:

2.2.1 gnuplot

在 Windows 下, 用户使用 GUI 界面安装 gnuplot 时请一定勾选 “Add gnuplot to PATH” 这一选项.

2.2.2 Python

若用户需要使用 python 库提供的功能, 用户需要同时安装 Python 以及 matplotlib, sympy 与 scipy 库; 前者用于绘图, 后者用于计算.

在 Windows 平台, 由于 T_EXLive 的编译配置, 需确保系统环境变量 PATHEXT 中已经删除 “.PY” 后缀.

2.2.3 Wolfram

若用户需要使用 wolfram 库对应的功能, 那么用户需要安装 WolframScript 或 Mathematica 软件. 执行命令时可以选择在云端执行, 这样就避免调用本地 Mathematica 计算内核. 用户需首先在命令行完成 wolfram 账号绑定, 绑定方法如下 (当用户第一次在命令行调用 Wolfram Cloud 上执行时):

```
> wolframscript -cloud -code 2+2
Wolfram ID: <Account>
Password: <Password>
```

上述命令会提示用户输入 Wolfram ID 和密码, 输入对应的 *<Account>* 和 *<Password>* 后即可使用. 但云端执行速度可能比较慢, 用户需自行决定是否采用此方案. 下面介绍在本地安装 wolfram 引擎的方法:

在 Linux 下, 除 wolfram 以外的软件都是很好安装的, 直接使用 Linux 发行版自带的包管理器即可. 这里我提供一个在 WSL 中使用 Windows 下 Mathematica 的方法 (用户也可以不按照此方法配置 WolframScript): 其实就是创建一个从 Linux 到 Windows 的软连接, 命令中 WolframScript 在 Windows 下的路径请根据自己的实际情况更改, 命令如下:

```
sudo ln -sf \
"/mnt/c/Program Files/Wolfram Research/WolframScript/wolframscript.exe" \
/usr/bin/wolframscript
```

请务必确保 WolframScript 在命令行中能被正常调用. 可以使用如下代码测试 WolframScript 是否成功配置:

```
plotFunction[fun_, xlimits_, ylimits_] := ContourPlot[fun,
  xlimits, ylimits,
  ContourStyle->{
    RGBColor["#00C0A3"],
    Thickness[0.004]
  },
  AspectRatio->((xlimits[[2]]//Abs) + (xlimits[[3]]//Abs))/((ylimits[[2]]//Abs) + (ylimits[[3]]//Abs)),
  AxesOrigin->{0,0},
  Axes->True,
  Frame->False,
  AxesStyle->Arrowheads[{0, 0.03}],
  AxesLabel->{"x", "y"},
  PlotRange -> Full
]

xlimits = {x, -3, 6};
ylimits = {y, -4, 5};
fp1 = plotFunction[y==Sin[x], xlimits, ylimits];
fp2 = plotFunction[x^2/4 + y^2/3 == 5, {x, -5, 5}, {y, -5, 5}];

figure = Show[fp2, fp1];
(* 1. 保存的图片格式为:*.wls.pdf; 2. 保存路径在:../ztikz_output/mma_data *)
Export["works_well.pdf", figure];
```

把上述的源码保存为 `test.wls`, 然后在命令行运行如下命令:

```
wolframscript -script test.wls
```

如果配置成功, 那么在当前工作目录下会产生一个名为 `works_well.pdf` 的 PDF 文件; 反之, 则说明你的 WolframScript 没有配置成功, 也就不能够使用本库.

2.2.4 Mathics

用户除了选择 WolframScript 作为计算引擎外, 还可以选择 Mathics 作为计算引擎. Mathics 是什么? An open-source Mathematica Kernel. MathsciScript 为 Mathics 的一个前端, 具有自动命令/变量补全, 语法高亮等功能.

在本地安装 Mathics 的方法请参见: [Installing Mathics3](#). 若用户在 Windows 下已经安装好 Mathics, 不想要在 WSL 中重新安装一次, 那么在 WSL 下创建软连接的方法和上述 WolframScript 的配置方法同理. 如果用户通过命令 “pip install Mathics-omnibus” 安装了 Mathics, 那么创建软连接的命令如下:

```
sudo ln -sf \
"/mnt/c/Users/<name>/AppData/Local/Programs/Python/Python312/Scripts/mathics.exe" \
/usr/bin/mathics
```

上述命令中的 `<name>` 需要替换为你自己的用户名, 同时也需要注意 Python 的版本号.

NOTE: 部分 Mathematica 中的函数 Mathics 也许并没有支持, 具体请参考 Mathics 的文档.

3 宏包选项

<code>ztikz/library</code>	<code>library = <basic gnuplot cache python wolfram l3draw>.....</code> 初始值: 空
----------------------------	--

此选项和命令 `\ztikzloadlib` 等价, 用于指定 `ztikz` 加载的库名列表, 在加载 `ztikz` 宏包时使用, 一个简单的配置样例如下:

```
\usepackage[library={basic, gnuplot}]{ztikz}
```

例 1

<code>ztikz/wolfram/engine</code> <code>ztikz/wolfram/cloud</code>	<code>engine = <wolfram mathics>.....</code> 初始值: wolfram <code>cloud = <true false>.....</code> 初始值: false
---	--

`<engine>` 用于指定 Wolfram 代码的计算引擎, 目前支持 Wolfram 和 Mathics 两种引擎, 前者为商业闭源软件, 后者为开源软件; `<cloud>` 用于指定是否使用 Wolfram Cloud 进行计算; **注意:** Mathics 目前不支持云计算. 一个简单的配置样例如下:

```
\usepackage[
  library = { wolfram },
  wolfram = { engine=wolfram, cloud=true }
]{ztikz}
```

例 2

4 杂项

`\ztikzMkdir` `\ztikzMkdir{path}`

New: 2025-05-15

此命令用于创建目录, `<path>` 可以为任意合法的路径名, 比如 `./A/B`.

5 $\text{\texttt{tikz}}$ 库

$\text{\texttt{tikz}}$ 提供了多种功能的库, 这些库可以通过 `\tikzloadlib` 命令加载. 用户需要使用 `\texloadlib{<library name>}` 加载对应的库, $\text{\texttt{tikz}}$ 中可用的 `<library name>` 列表如下:

- basic
- cache
- gnuplot
- python
- wolfram
- l3draw

上述的所有库均不自动加载, 需用户手动加载. `basic` 库中仅包含了用于绘制点, 直线, 坐标轴和基本多边形等系列命令. 在导言区使用如下命令加载 `\tikz` 的库方法如下, 比如加载 `cache` 库和 `gnuplot` 库:

```
\tikzloadlib{cache, gnuplot}
```

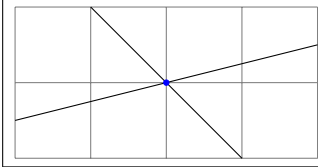
例 3

注意: 只有当用户加载对应的库后, 该库的脚本文件才会被写入项目文件夹下.

<code>\ShowIntersection</code>	<code>\ShowIntersection[⟨key-val⟩]{⟨path-1⟩; ⟨path-2⟩}{⟨number⟩}</code>
New: 2025-05-15	此命令用于求解 $\langle path-1 \rangle$ 和 $\langle path-2 \rangle$ 的交点, 使用 “;” 进行分割; 然后将前 $\langle number \rangle$ 个交点绘制出来. $\langle key-value \rangle$ 对应 <code>\ShowPoint</code> 命令中的 $\langle key-value \rangle$ 选项, 即 <code>\ztikz/point</code> .

```
\begin{tikzpicture}
\draw[gray] (-2, -1) grid (2, 1);
\draw[name path=line1] (-2, -.5) -- (2, .5);
\draw[name path=line2] (-1, 1) -- (1, -1);
\ShowIntersection[color=blue]{line1; line2}{1}
\end{tikzpicture}
```

例 5



<code>\ShowAxis</code>	<code>\ShowAxis[⟨key-value⟩]{⟨start⟩; ⟨end⟩}</code>
New: 2025-05-15	此命令用于绘制坐标轴, $\langle start \rangle$ 和 $\langle end \rangle$ 分别表示坐标轴的起始点和结束点, 使用 “;” 进行分割, 坐标格式为 (x,y) . $\langle key-value \rangle$ 为可选参数, 用于设置坐标轴样式.

<code>ztikz/axis/tickStart</code>	<code>tickStart</code>	= $\langle \text{浮点数} \rangle$	初始值: -5
<code>ztikz/axis/tickEnd</code>	<code>tickEnd</code>	= $\langle \text{浮点数} \rangle$	初始值: 5
<code>ztikz/axis/axisRotate</code>	<code>axisRotate</code>	= $\langle \text{浮点数} \rangle$	初始值: 0
<code>ztikz/axis/mainStep</code>	<code>mainStep</code>	= $\langle \text{浮点数} \rangle$	初始值: 1
<code>ztikz/axis/subStep</code>	<code>subStep</code>	= $\langle \text{浮点数} \rangle$	初始值: 0.1
<code>ztikz/axis/tickLabelShift</code>	<code>tickLabelShift</code>	= $\langle \text{长度} \rangle$	初始值: 0pt
<code>ztikz/axis/mainTickLength</code>	<code>mainTickLength</code>	= $\langle \text{长度} \rangle$	初始值: 4pt
<code>ztikz/axis/subTickLength</code>	<code>subTickLength</code>	= $\langle \text{长度} \rangle$	初始值: 2pt
<code>ztikz/axis/axisColor</code>	<code>axisColor</code>	= $\langle \text{颜色} \rangle$	初始值: black
<code>ztikz/axis/mainTickColor</code>	<code>mainTickColor</code>	= $\langle \text{颜色} \rangle$	初始值: black
<code>ztikz/axis/subTickColor</code>	<code>subTickColor</code>	= $\langle \text{颜色} \rangle$	初始值: black
<code>ztikz/axis/tickStyle</code>	<code>tickStyle</code>	= $\langle below above cross \rangle$	初始值: 无
<code>ztikz/axis/mainTickLabel</code>	<code>mainTickLabel</code>	= $\langle \text{字符串} \rangle$	初始值: \CurrentFp
<code>ztikz/axis/mainTickLabelColor</code>	<code>mainTickLabelColor</code>	= $\langle \text{颜色} \rangle$	初始值: black
<code>ztikz/axis/mainTickLabelPosition</code>	<code>mainTickLabelPosition</code>	= $\langle below above cross \rangle$	初始值: below

$\langle mainTickLabel \rangle$ 主要用于自定义坐标标签的样式, `\CurrentFp` 表示当前刻度处的浮点数值. $\langle tickStyle \rangle$ 会受到 `tikzpicture` 环境可选参数中的 $\langle rotate \rangle$ 选项的影响.

注意: 在使用 `\ShowAxis` 时若没有指定键 $\langle tickStyle \rangle$ 的值, 那么此时并不会绘制任何的刻度.

<code>\xAxi s</code>	<code>\xAxi s[⟨start⟩][⟨end⟩]</code>
New: 2025-05-15	此命令来自 <code>\ShowAxis</code> , 用于绘制 x 轴; $\langle start \rangle$ 和 $\langle end \rangle$ 均为浮点数, 分别表示坐标轴的起始点和结束点.
<code>\yAxi s</code>	<code>\yAxi s[⟨start⟩][⟨end⟩]</code>
New: 2025-05-15	此命令来自 <code>\ShowAxis</code> , 用于绘制 y 轴; $\langle start \rangle$ 和 $\langle end \rangle$ 均为浮点数, 分别表示坐标轴的起始点和结束点.

```
\begin{tikzpicture}[>=Latex]
\yAxis[-1][1]
\ShowAxis{(-2, 0); (2, 0)}
\draw (-2, -1) grid (2, 1);
\end{tikzpicture}
```

例 6

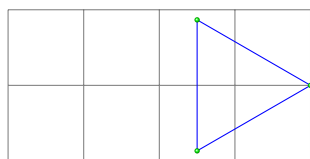
<code>\ShowGrid</code>	<code>\ShowGrid[⟨draw-keyval⟩]{⟨start⟩; ⟨end⟩}</code>
New: 2025-05-15	此命令用于绘制网格线, $\langle start \rangle$ 和 $\langle end \rangle$ 分别表示网格线的左下角和和右上角的坐标, 使用 “;” 进行分割, 坐标的格式为 (x,y) . $\langle key-value \rangle$ 为可选参数, 用于设置网格线的样式;
<code>\Polygon</code>	<code>\Polygon[⟨key-value⟩]{⟨number⟩}</code>
New: 2025-05-15	此命令用于绘制正多边形, $\langle number \rangle$ 表示多边形的边数, 其值必须为大于等于 3 的整数. $\langle key-value \rangle$ 为可选参数, 用于设置多边形的样式;

<code>ztikz/polygon/radius</code>	<code>radius</code>	= $\langle \text{浮点数} \rangle$	初始值: 1
<code>ztikz/polygon/edgeColor</code>	<code>edgeColor</code>	= $\langle \text{颜色} \rangle$	初始值: black
<code>ztikz/polygon/fillColor</code>	<code>fillColor</code>	= $\langle \text{颜色} \rangle$	初始值: 无
<code>ztikz/polygon/fillOpacity</code>	<code>fillOpacity</code>	= $\langle \text{浮点数} \rangle$	初始值: 0
<code>ztikz/polygon/rotate</code>	<code>rotate</code>	= $\langle \text{浮点数} \rangle$	初始值: 0
<code>ztikz/polygon/shift</code>	<code>shift</code>	= $\langle \text{坐标} \rangle$	初始值: (0, 0)
<code>ztikz/polygon/marker</code>	<code>marker</code>	= $\langle \text{key-value} \rangle$	初始值: 无
$\langle radius \rangle$ 表示此正多边形外接圆的半径, 而非 $\langle marker \rangle$ 的半径; $\langle shift \rangle$ 外围的 “()” 不能省略. $\langle marker \rangle$ 对应 <code>ztikz/point</code> . $\langle marker \rangle$ 的设置请参见 图 (3).			

```
\begin{tikzpicture}
\ShowGrid[gray, thin]{(-2, -1); (2, 1)}
\Polygon[
  edgeColor=blue, shift={(1, 0)},
  marker={type=ball, color=green}
]
```

例 7

```
]{3}
\end{tikzpicture}
```



```
\StairsPlot [plot option]; [jump option]] [draw-keyval]
```

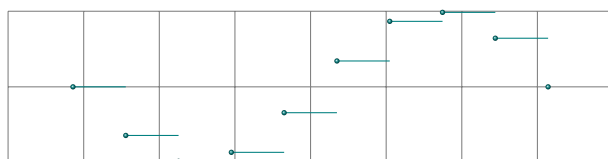
```
[[key-value]]{file}
```

New: 2025-05-15

此命令用于绘制阶梯图, 绘图数据由 *file* 指定; *plot option* 用于设置阶梯图的绘制样式, 可选值有: plot left, plot right, plot mid; *jump option* 用于设置阶梯图的跳跃样式, 可选值有: jump left, jump right, jump mid; *key-value* 对应 *ztikz/point*;

```
\begin{tikzpicture}
\ShowGrid[step=1, color=gray]{(-4, -1); (4, 1)}
\StairsPlot[jump-left][teal][type=ball, color=teal]{./sine.data}
\end{tikzpicture}
```

例 8



```
\StemPlot [direction]] [draw-keyval]
```

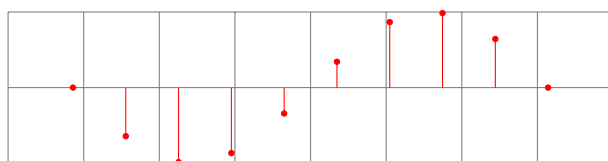
```
[[key-value]]{file}
```

New: 2025-05-15

此命令用于绘制火柴棍图, 绘图数据由 *file* 指定; *direction* 用于指定系列线段的方向, 可选值有: x, y, o, 分别表示垂直 *x* 轴, 垂直 *y* 轴, 以及指向坐标原点; *key-value* 对应 *ztikz/point*.

```
\begin{tikzpicture}
\ShowGrid[step=1, color=gray]{(-4, -1); (4, 1)}
\StemPlot[x][red][type=*, color=red]{./sine.data}
\end{tikzpicture}
```

例 9



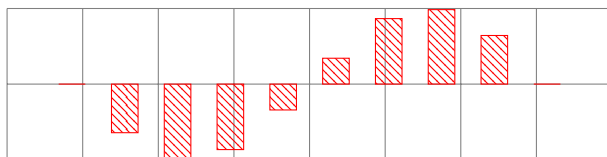
 \backslash BarPlot

New: 2025-05-15

 \backslash BarPlot[\langle position \rangle][\langle draw-keyval \rangle]
[\langle key-value \rangle]{ \langle file \rangle }

此命令用于绘制条形图, 绘图数据由 \langle file \rangle 指定; \langle position \rangle 用于指定每个小矩形的位置以及宽度, 可选值有: x, y, xc, yc; \langle key-value \rangle 对应 \langle ztikz/point \rangle .

```
\begin{tikzpicture}
\ShowGrid[step=1, color=gray]{(-4, -1); (4, 1)}
\BarPlot[x][red, pattern=north west lines, pattern color=red]{./sine.data}
\end{tikzpicture}
```

例 10

5.2 gnuplot 库

需要说明的是: $\text{\textit{tikz}}$ 宏包内部已经提供了直接调用 gnuplot 程序的命令 (需启用 `-shell-escape` 参数), 其调用格式如下:

```
\draw[key-value] plot[id] function{function};
```

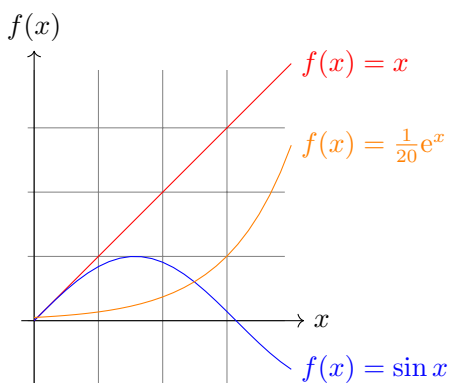
上述命令中 $\langle id \rangle$ 用于区分不同的数据文件, 在 $\langle file \rangle.tex$ 文件 (不妨设文件名为 $\langle file \rangle$) 的根路径下会产生两个文件: 一个是 gnuplot 用于绘图的样式文件 $\langle file \rangle.\langle id \rangle.gnuplot$; 第二个是 gnuplot 产生的数据文件 $\langle file \rangle.\langle id \rangle.table$. 命令中的 $\langle function \rangle$ 可用值请参见: 表 (1).

$\text{\textit{tikz}}$ 的内置命令也支持另外两种格式: “parametric”, “raw gnuplot”: 第一个参数表示绘制参数方程, 第二个参数表示直接在文档中使用 gnuplot 的原始绘图命令 (比如 “set samples 25; plot sin(x)”). 两者的调用格式如下:

```
\draw[key-value] plot [parametric, <id>]{function};
\draw[key-value] plot [raw gnuplot, <id>]{gnuplot code};
```

```
\begin{tikzpicture}[domain=0:4, scale=.85]
\draw[very thin,color=gray] (-0.1,-1.1) grid (3.9,3.9);
\draw[->] (-0.2,0) -- (4.2,0) node[right] {$x$};
\draw[->] (0,-1.2) -- (0,4.2) node[above] {$f(x)$};
\draw[color=red] plot[id=x] function{x} node[right] {$f(x)=x$};
\draw[color=blue] plot[id=sin] function{sin(x)} node[right] {$f(x)=\sin x$};
\draw[color=orange] plot[id=exp] function{0.05*exp(x)} node[right]
{$f(x)=\frac{1}{20}\mathrm{e}^x$};
\end{tikzpicture}
```

例 11



关于 $\text{\textit{tikz}}$ 中这部分原生绘图命令更加详细使用方法请参见 $\text{\textit{tikz}}$ 官方文档中 Section 22: Plots of Functions.

但是为了 gnuplot 这一系列绘图命令的统一, $\text{\textit{tikz}}$ 并没有采用上面的方式, 而是借用 `ztool` 宏包, 然后配合预定义的绘图脚本去完成绘图任务. $\text{\textit{tikz}}$ 中 gnuplot 库的绘图逻辑大致如下:

- 首先通过 ztool 的 `\ztool_replace_file_line:nnn` 函数修改预定义脚本;
- 然后通过命令行的 `-shell-escape` 参数去调用 gnuplot 运行修改后的脚本;
- 最后使用命令 `\draw[⟨key-value⟩] plot file [⟨data⟩]`; 调用上一步生成的数据文件完成绘图.

不熟悉 gnuplot 的用户可阅读这份 7 页的快速入门指南: [gnuplot card](#).

NOTE: 调用此库后, 需在编译时启用 “-shell-escape” 参数.

ztikz/2dplot/domain	domain = ⟨浮点数: 浮点数; 浮点数: 浮点数⟩..... 初始值: (不确定)
ztikz/2dplot/style	style = ⟨draw-keyval⟩..... 初始值: black
ztikz/2dplot/marker	marker = ⟨key-value⟩..... 初始值: 空

⟨maker⟩ 中的 ⟨key-value⟩ 对应 ⟨ztikz/point⟩. ⟨domain⟩ 二者之间使用 “;” 进行分割, 在不同的函数中 ⟨domain⟩ 的意义不同: 在 `\Plot` 中用于设置自变量 x 的范围; 在 `\ParamPlot` 和 `\PolarPlot` 中, 用于设置参数 t 或极坐标系中角度 θ 的范围; 在 `\ContourPlot` 中, “;” 前后两个 ⟨domain⟩ 分别表示 x 和 y 的范围.

<code>\PlotPrecise</code>	<code>\PlotPrecise{⟨type⟩}{⟨number⟩}</code>
New: 2025-05-15	<code>\PlotPrecise*{⟨type⟩}{⟨number⟩}</code>
	此命令用于设置 gnuplot 中一系列二维绘图函数对应的精度, ⟨type⟩ 可选值有: “plot, param, polar, contour”, 分别对应命令 <code>\Plot</code> , <code>\ParamPlot</code> , <code>\PolarPlot</code> 和 <code>\ContourPlot</code> 的绘制精度. 含有 “*” 的命令会应用于对应绘图命令之后的所有实例, 没有 “*” 的命令仅会应用于之后的第一个绘图命令.

<code>\Plot</code>	<code>\Plot[⟨key-value⟩]{⟨function⟩}</code>
New: 2025-05-15	此命令用于绘制函数 $y = y(x)$, ⟨function⟩ 为 gnuplot 中的函数表达式, 自变量为 “x”; ⟨key-value⟩ 用于设置绘图样式, 对应 ⟨ztikz/2dplot⟩. ⟨domain⟩ 默认为 <code>-5:5</code> . 注意: 只需将 ⟨opacity⟩ 置为 0, 即可实现散点图绘制.

<code>\ContourPlot</code>	<code>\ContourPlot[⟨key-value⟩]{⟨equation⟩}</code>
New: 2025-05-15	此命令用于绘制方程 $f(x, y) = c$, ⟨equation⟩ 为 gnuplot 中的方程表达式, 变量为 “x, y”, 且表达式中不需要书写 “=” 符号; ⟨key-value⟩ 用于设置绘图样式, 对应 ⟨ztikz/2dplot⟩. ⟨domain⟩ 默认为 <code>“-5:5;*:*”</code> (即自变量 y 的范围自适应). 注意: 绘制 $x = c$ 这种垂直线段时, 可以使用此函数.

<code>\ParamPlot</code>	<code>\ParamPlot[⟨key-value⟩]{⟨equation⟩}</code>
New: 2025-05-15	此命令用于绘制参数方程 $x = x(t), y = y(t)$, ⟨equation⟩ 为 gnuplot 中的方程表达式, 参数为 “t”; ⟨key-value⟩ 用于设置绘图样式, 对应 ⟨ztikz/2dplot⟩. ⟨domain⟩ 默认为 <code>0:2*pi</code> .

 $\backslash\text{PolarPlot}$ $\backslash\text{PolarPlot}[\langle\text{key-value}\rangle]\{\langle\text{equation}\rangle\}$

New: 2025-05-15

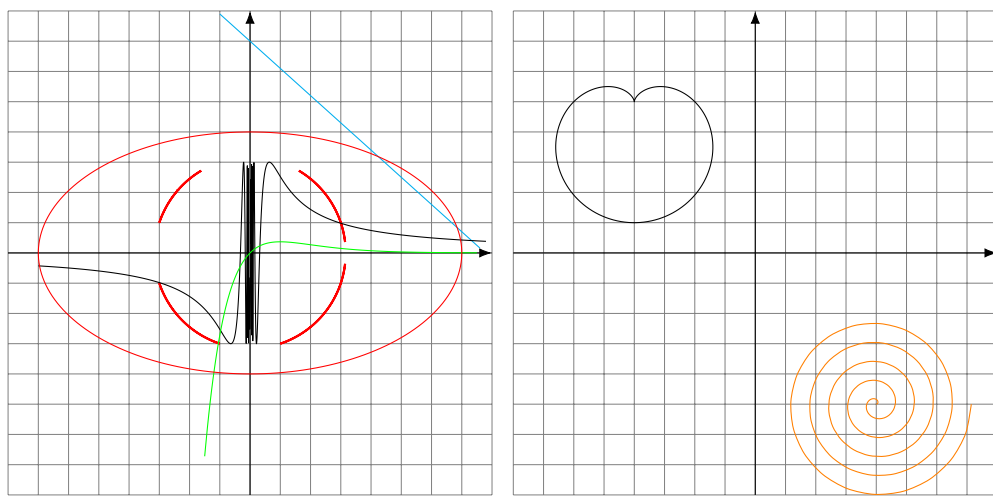
此命令用于绘制极坐标方程 $\rho = \rho(t)$, $\langle\text{equation}\rangle$ 为 gnuplot 中的方程表达式, 参数为 “t”; $\langle\text{key-value}\rangle$ 用于设置绘图样式, 对应 $\langle\text{ztikz}/2\text{dplot}\rangle$. $\langle\text{domain}\rangle$ 默认为 $0:2\pi$.

例 12

```

\begin{tikzpicture}[>=Latex, scale=.4]
\ShowGrid{(-8, -8); (8, 8)}\ShowAxis{(0, -8); (0, 8)}\ShowAxis{(-8, 0); (8, 0)}
% draw functions/curves
\Plot[domain=-1:7.6, style=cyan] {-0.9*x+7}
\ContourPlot[
  domain={-3*pi; -3:exp(1)}, style={red, thick}
]{x**2 + y**2 - 10}
% change plot precise
\PlotPrecise{plot}{1500}
\Plot[domain=-7:7.8]{3*sin(1/x)}
\Plot[domain=-1.5:7.5, style=green] {x*exp(-x)}
\ParamPlot[domain=0:2*pi, style=red]{7*sin(t), 4*cos(t)}
\end{tikzpicture}
\hskip.5em
\begin{tikzpicture}[>=Latex, scale=.4]
\ShowGrid{(-8, -8); (8, 8)}\ShowAxis{(0, -8); (0, 8)}\ShowAxis{(-8, 0); (8, 0)}
% draw functions/curves
\begin{scope}[xshift=4cm, yshift=-5cm]
  \PolarPlot[domain=0:10*pi, style=orange]{0.1*t}
\end{scope}
\begin{scope}[xshift=-4cm, yshift=5cm]
  \PolarPlot{2*(1-sin(t))}
\end{scope}
\end{tikzpicture}

```



回顾上面给出的这个简单案例: 这个案例中我们使用了 $\backslash\text{Plot}$, $\backslash\text{ParamPlot}$, $\backslash\text{PolarPlot}$ 和 $\backslash\text{ContourPlot}$ 四个命令; 同时也应用了 $\backslash\text{PlotPrecise}$ 命令, 它更改

`\currentTikzIndex`

New: 2025-05-15

该命令表示当前 `tikzpicture` 环境的索引, 返回值为整数, 从 1 开始.

`\gnudata` ★ `\gnudata{<index>}`

New: 2025-05-22

该命令会用引用当前 `tikzpicture` 环境中产生的绘图数据, 返回一个 (数据) 文件名, 从 1 开始. `<index>` 接受一个整数, 表示当前环境中绘图数据的编号. 每一个已经绘制的函数都会在对应的文件夹下生成一个对应的数据文件, 用户可以使用此数据文件进行后续的绘图操作.

`\gnudata` 的用法补充说明, 为后面区域填充案例做铺垫: 比如 `\gnudata{2}`, 参数中的 “2” 表示此数据是在当前 `tikzpicture` 环境中的第二个函数绘图数据; 所以在第一个 `tikzpicture` 环境中它的返回值可能为 “./ztikz_output/gnuplot_data/gnu_data_1_2.table”.

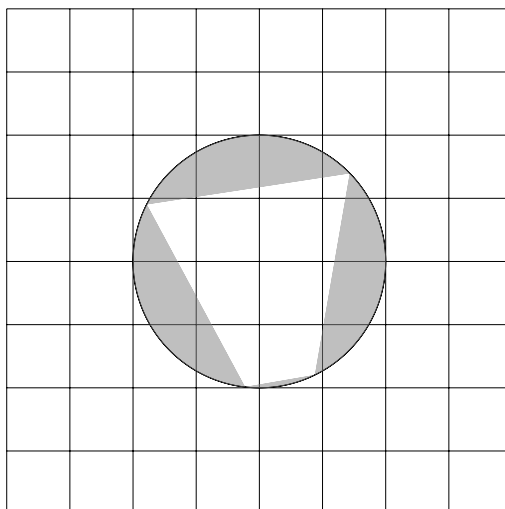


图 1: `\ContourPlot` Fill Issue

注意: 由于技术原因, `\ContourPlot` 命令生成的数据暂时不可用于后续填充操作. 可考虑先将隐函数转化为参数方程形式或极坐标形式, 再导出对应的数据. 如果你强行使用此类型数据, 那么用户可能会得到类似 图 (1) 这样的不良输出.

5.3 cache 库

当用户加载 `cache` 库后, 随后在命令行中编译文档, 不妨设其名称为 $\langle file \rangle$; 那么用户会看到如下的日志输出:

```
\write18 enabled.
entering extended mode
```

编译结束后, 在你的项目文件夹下会生成一个名为 `ztikz_output` 的文件夹, 这个文件夹在你第一次调用 `ztikz` 宏包时便会产生; 这个文件夹用于存放 $\text{\textit{zTikZ}}$ 的缓存文件: 包括 $\text{\textit{TikZ}}$ external 库的缓存结果, Python 脚本的缓存结果, WolframScript 脚本的缓存结果, 以及 `gnuplot` 的一系列缓存结果.

现在我们来谈谈这个文件夹的构成: 比如, 若用户运行了 `\Plot` 命令, 此时会在 `ztikz_output/tikz_data/` 目录下生成了如 图 (2) 中所示的 4 个文件:

```
ztikz_output
├── gnuplot_data..... gnuplot 缓存文件夹
│   └── gnu_data_1_1.table
├── mma_data..... WolframScript 缓存文件夹
├── python_data..... Python 缓存文件夹
├── scripts..... gnuplot 绘图脚本
│   ├── 3d_plot.gp
│   ├── contour_plot.gp
│   ├── param_plot.gp
│   ├── plot_plot.gp
│   └── polar_plot.gp
├── tikz_data..... TikZ 缓存文件夹
│   ├──  $\langle file \rangle$ -figure0.dpth
│   ├──  $\langle file \rangle$ -figure0.log
│   ├──  $\langle file \rangle$ -figure0.md5
│   ├──  $\langle file \rangle$ -figure0.pdf
│   └──  $\langle file \rangle$ -figure0.run.xml
└── ztikz.hash..... Hash 值记录
```

图 2: $\text{\textit{zTikZ}}$ 缓存目录结构示意图

`tikz_data` 中的 $\langle file \rangle$ -figure0.pdf 为 `tikzpicture` 环境缓存的 PDF 文件; 此时在对应的 $\langle file \rangle$.md5 文件中可以看到如下内容:

```
\def \tikzexternallastkey {AE7F2539E81C96848ADCCEE3994993D1}%
```

上述命令保存了此 `tikzpicture` 环境中代码的 Hash 值, 当我们改变 `tikzpicture` 环境中的代码时, 这个 Hash 值就会改变, 从而 $\text{\textit{TikZ}}$ 就会再次运行此环境, 重新生成图片. 这便是 $\text{\textit{TikZ}}$ 的 external 库所提供的缓存功能的大致描述. $\text{\textit{zTikZ}}$ 中的 Cache 机制和此原理是十分类似的.

`\ztikzHashClean`

New: 2025-05-15

此命令不接受任何参数, 用于清除之前缓存的所有 Hash 值.

`\ztikzHashCurrent`

New: 2025-05-15

`\ztikzHashCurrent*`

`\ztikzHashCurrent[⟨separator⟩]`

此命令主要用于调试或与命令 `\ztikzForceToSkip` 配合使用; `\ztikzHashCurrent*` 将输出最近的一次 Hash 值计算结果; `\ztikzHashCurrent[⟨separator⟩]` 用于输出截至当前位置所有缓存的 Hash 值, 以 `⟨separator⟩` 分隔输出到 PDF. `⟨separator⟩` 默认为 “,”.

`\ztikzForceToSkip`

New: 2025-05-15

此命令会强制跳过 (重新) 运行它之后的第一个具有 cache 机制的环境或命令, 即使该环境或命令对应的 Hash 已经改变; 后续的 `\wolframResult` 或 `\wolframOutputFile` 命令对应的引用结果都将受到此命令的影响.

注意: 目前该命令目前很不成熟, 仅对 `python` 库中的 `\sympy` 命令有效; 尽量避免连续使用多个 `\ztikzForceToSkip` 命令, 否则会导致引用的缓存结果出现较大的偏差.

`\ztikzForceToRun`

New: 2025-05-21

此命令会强制运行它之后的第一个具有 cache 机制的环境或命令, 即使该环境或命令对应的 Hash 并没有改变.

5.4 python 库

`python` 库主要用于和 Python 交互, 其使用方法和 `gnuplot` 库类似. `python` 库中主要提供了图片绘制与计算接口, 其中计算接口包含数值计算与符号计算.

除去 $\text{\texttt{tikz}}$ 提供的 Python 绘图功能外, 我们需要着重说明 $\text{\texttt{tikz}}$ 提供的浮点数计算功能: $\text{\texttt{tikz}}$ 在调用此库时默认导入 Python 的 `numpy`, `sympy`, `scipy` 三个包; 此外, 用户在使用 `numpy` 中的函数时不用再加以前缀, 比如求解 `sin(2.345)` 时, 直接使用 `\py{sin(2.345)}` 即可, 不必写为 `\py{np.sin(2.345)}` 之类的格式了. 对于其它 Python 库中的函数, 使用方法同理.

NOTE: 调用此库后, 需在编译时启用 “`-shell-escape`” 参数.

 $\text{\texttt{py}}$

New: 2025-05-15

`\py[$\langle\text{raw}\rangle$ | $\langle\text{str}\rangle$]{ $\langle\text{code}\rangle$ }`

此命令会调用 Python 进行浮点数运算, $\langle\text{code}\rangle$ 为合法的 Python 表达式; 这部分的结果并不会被缓存, 也就是说每次编译此文档时, Python 都会重新计算此部分的结果. 用户可以把 `\py` 命令嵌套到自己定义的宏命令中.

注意: $\langle\text{raw}\rangle$ 会将返回的结果按照 $\text{\texttt{TeX}}$ 原始的 `catcode` 进行 tokenize; $\langle\text{str}\rangle$ 则是将返回的结果处理为 `string`.

```
\newcommand{\pypow}[1]{\py{#1}}
\newcommand{\pyreverse}[1]{\py{'#1'[::-1]}}
\newcommand{\pyuppercase}[1]{\py{'#1'.upper()}}
\begin{itemize}
  \item Power Calculation:  $2^{10} = \text{\texttt{\pypow{2**10}}}$ 
  \item Reverse a string using Python:  $\text{\texttt{\pyreverse{Hello-LaTeX}}}$ 
  \item Uppercase a string:  $\text{\texttt{\pyuppercase{hello-latex}}}$ 
  \item Modulus:  $102 = \text{\texttt{\py{mod(102, 8)}}} \bmod 8$ 
  \item Return string Options:  $\text{\texttt{\py[str]{ '$$'+str(2**10)+' $$' }}}}$ 
\end{itemize}
```

例 14

-
- Power Calculation: $2^{10} = 1024$
 - Reverse a string using Python: XeTaL-olleH
 - Uppercase a string: HELLO-LATEX
 - Modulus: $102 = 6 \bmod 8$
 - Return string Options: \$\$1024\$\$

 $\backslash\text{sympy}$ $\backslash\text{sympy}\{\langle\text{expression}\rangle\}$

New: 2025-05-15

此命令主要用于调用 Python 的 sympy 库进行符号计算, $\langle\text{expression}\rangle$ 为符号表达式. \LaTeX 对此命令提供了 cache 机制. python 库中预定义了一系列的符号变量, 包括: x, y, z, u, v, t , 这些预定义变量无需用户再次声明.

注意: 默认的情况下, 此命令的返回结果中包含: $\wedge, _$ 等特殊字符, 所以请将此命令置于数学环境中.

例 15

```
\[
\int x^8 + \cos(7x) + 6t \, \mathrm{d}x
= \text{sympy}\{\text{integrate}(x**8 + \cos(7*x) + 6*t, x)\}
\]
\[\text{eig}(\begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix})
= \text{sympy}\{\text{Matrix}([[1, 2], [2, 2]]).\text{eigenvals}()\}
\]
```

$$\int x^8 + \cos(7x) + 6t \, dx = 6tx + \frac{x^9}{9} + \frac{\sin(7x)}{7}$$

$$\text{eig}\left(\begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix}\right) = \left\{ \frac{3}{2} - \frac{\sqrt{17}}{2} : 1, \frac{3}{2} + \frac{\sqrt{17}}{2} : 1 \right\}$$

 pyfig

Updated: 2025-05-21

 $\backslash\text{begin}\{\text{pyfig}\}\{\langle\text{output file}\rangle\}$
 $\langle\text{plot code}\rangle$
 $\backslash\text{end}\{\text{pyfig}\}$

此环境用于调用 Python 进行绘图, 不会返回任何结果, 该环境具有缓存机制; $\langle\text{output file}\rangle$ 用于指定代码 $\langle\text{plot code}\rangle$ 的输出文件名, $\langle\text{output file}\rangle$ 中无需给出输出文件路径, 但需指定输出文件的拓展名;

注意: 针对不同的 pyfig 环境建议使用不同的 $\langle\text{output file}\rangle$ 值; 用户不需要在代码末尾添加 `plt.savefig()` 命令, \LaTeX 会自动处理相关的问题. 代码在抄录过程中会保留用户的缩进格式, 从行首开始抄录, 所以请不要添加多余的行首缩进; 请确保 $\langle\text{output file}\rangle$ 与后续环境代码中的输出文件名保持一致, 否则会报错.

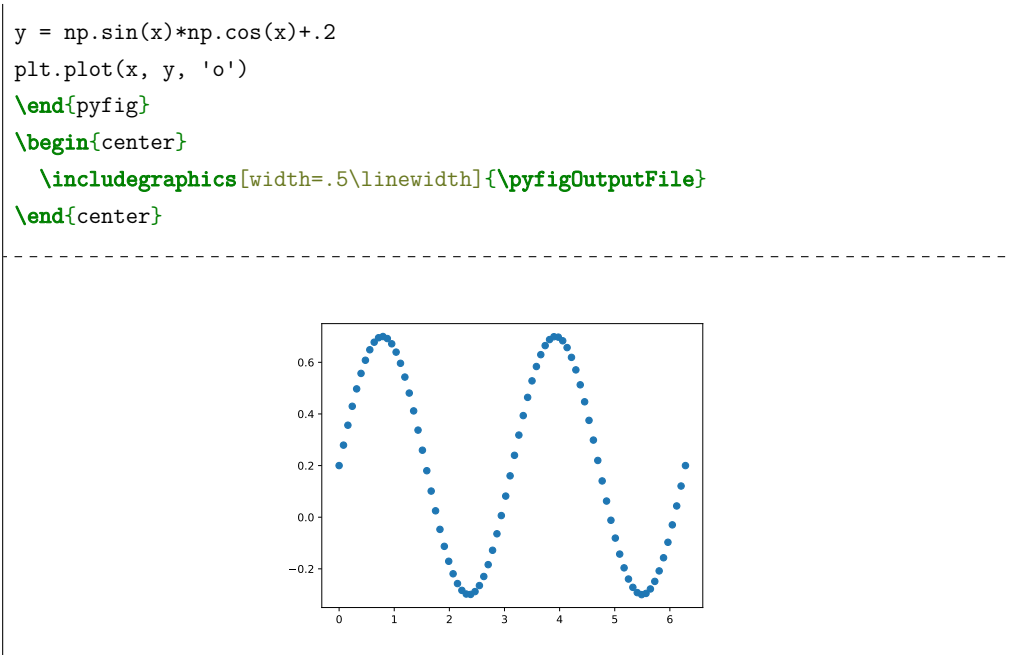
 $\backslash\text{pyfigOutputFile} \star$

New: 2025-04-21

此命令将返回 pyfig 环境运行输出的文件名, 用户可以使用 $\backslash\text{input}$ 或 $\backslash\text{includegraphics}$ 之类的命令导入该文件.

例 16

```
\begin{pyfig}\sin_graph.pdf
import matplotlib
matplotlib.use('Agg')
from matplotlib import pyplot as plt
import numpy as np
x = np.linspace(0, 2*np.pi, num = 80)
```

pycode

New: 2025-05-21

```
\begin{pycode}{\langle output file \rangle}
\langle any python code \rangle
\end{pycode}
```

此环境用于调用 Python 执行环境中的 `\langle any python code \rangle`，不会返回任何结果，该环境具有缓存机制；`\langle output file \rangle` 用于指定该环境代码的输出文件名，`\langle output file \rangle` 中无需给出输出文件路径，但需指定输出文件的拓展名；随后 `zTikZ` 会自动调用 Python 执行该文件，该环境的运行结果保存于文件 `\pycodeOutputFile` 中，用户后续仅需导入该文件即可；

注意：针对不同的 `pycode` 环境建议使用不同的 `\langle output file \rangle` 值；代码在抄录过程中会保留用户的缩进格式，从行首开始抄录，所以不要过度使用缩进；请确保 `\langle output file \rangle` 与后续环境代码中的输出文件名保持一致，否则会报错。

`\pycodeOutputFile` ★

New: 2025-04-21

此命令将返回 `pycode` 环境运行输出的文件名，用户可以使用 `\input` 或 `\includegraphics` 之类的命令导入该文件。

下面是一个关于 `pycode` 环境的简单使用示例，`table.py.txt` 对应的文件内容请参见 [节 \(6.3\)](#)。

```
\input{./table.py.txt}
\begin{center}
\input{\pycodeOutputFile}
\end{center}
```

例 17

	number/function	sin	cos	tan
	1	0.8415	0.5403	1.5574
	2	0.9093	-0.4161	-2.185
	3	0.1411	-0.99	-0.1425
	4	-0.7568	-0.6536	1.1578
	5	-0.9589	0.2837	-3.3805
	6	-0.2794	0.9602	-0.291
	7	0.657	0.7539	0.8714
	8	0.9894	-0.1455	-6.7997
	9	0.4121	-0.9111	-0.4523
	10	-0.544	-0.8391	0.6484
	11	-1.0	0.0044	-225.9508
	12	-0.5366	0.8439	-0.6359
	13	0.4202	0.9074	0.463
	14	0.9906	0.1367	7.2446
	15	0.6503	-0.7597	-0.856

5.5 wolfram 库

$\text{\texttt{tikz}}$ 的 `wolfram` 库可看作是原始宏包 `latexalpha2` 的一个新实现, 可以弥补 `latexalpha2` 宏包的一系列不足. 目前 `wolfram` 库已经实现 `latexalpha2` 中除 `\wolframanimation` 命令外的所有命令, 并且在兼容性, 易用性和可拓展性上相较于原始的 `latexalpha2` 宏包都有了极大的提升. 例如, $\text{\texttt{tikz}}$ 的 `wolfram` 库可以在 Windows/Linux/MacOs 三大平台上使用; `wolfram` 库的环境源码中支持直接键入 “\, #, \$, _, ^, &” 等特殊字符. 下面是使用 `wolfram` 库时的一些注意事项:

- 用户需注意 WolframScript 脚本中注释的写法, 不是 “(* something*)”, 而是 “(* something *)”, 即注释内容不能够紧挨 “*”, 否则可能会造成 WolframScript 的解析错误.
- 由于 WolframScript 的限制, 脚本的后缀只能为: “.wls”, 否则 WolframScript 会无法识别此脚本 (也就不会去执行此脚本了).

NOTE: 调用此库后, 需在编译时启用 “-shell-escape” 参数.

 $\text{\texttt{wolframResult}}$

New: 2025-05-15

 $\text{\texttt{wolframResult}}[\langle separator \rangle]$
 $\text{\texttt{wolframResult}}*[\langle index \rangle]$

此命令用于引用前一次 WolframScript 的计算结果, $\text{\texttt{wolframResult}}[\langle separator \rangle]$ 表示使用 $\langle separator \rangle$ 进行分隔, 然后引用全部计算结果; $\text{\texttt{wolframResult}}*[\langle index \rangle]$ 仅引用部分计算结果, $\langle index \rangle$ 为整数或整数表达式, 默认为 1.

 $\text{\texttt{wolframOutputFile}}$ ★

New: 2025-05-15

此命令会返回 WolframScript 上次运行结果对应的文件名; 此命令在引用一些图片结果时是十分方便的. 此命令比之 $\text{\texttt{wolframResult}}$ 更加的灵活, 前者调用上一次的文本文件, 后者仅返回上次 WolframScript 调用产生的文件名.

 $\text{\texttt{wolfram}}$

New: 2025-05-15

 $\text{\texttt{wolfram}}\{\langle code \rangle\}$
 $\text{\texttt{wolfram}}*\{\langle code \rangle\}$

此命令用于调用 WolframScript 中的进行计算, 具有缓存机制; $\langle code \rangle$ 为合法的 WolframScript 代码; 默认计算结果为 L^AT_EX 代码, 含有 “*” 的命令计算结果为普通的字符串 (catcode 并没有改变).

```
\wolfram{LaplaceTransform[t^4 Sin[3*t], t, s]}
\[
  \mathcal{L}(t^4 \sin(3t)) = \text{\texttt{wolframResult}}
\]
```

例 18

$$\mathcal{L}(t^4 \sin(3t)) = \frac{72(5s^4 - 90s^2 + 81)}{(s^2 + 9)^5}$$

`\wolframTex` `\wolframTex{<Tex code>}`

New: 2025-05-18

此命令和上述的 `\wolfram` 命令类似, 不同的是, 此命令会将 `<Tex code>` 中的所有内容转化为对应的 Mathematica 代码, 返回的结果为 L^AT_EX 代码.

NOTE: 由于此命令的实现原理较为复杂与特殊, 所以 `<Tex code>` 中不能包含 “\$” 符号, 否则会出现解析错误.

```
\wolframTex{\int_a^b \sin(x)\,dx}
\[
  \int_a^b \sin(x)\,dx = \wolframResult
\]
```

例 19

$$\int_a^b \sin(x) dx = \cos(a) - \cos(b)$$

`\wolframTable` `\wolframTable{<code>}`

New: 2025-05-18

`\wolframTable*[[<key-value>]]{<code>}`

此命令用于调用 Wolfram 引擎生成表格, 具有缓存机制; 前者 (不带有 “*” 的命令) 不会在 PDF 中输出对应的表格, 此命令设置了 `\wolframTablePData`, `\wolframTableFData` 两个临时变量, 其中保存了表格数据; 后者 (带有 “*” 的命令) 会在 PDF 中输出对应的表格, 表格的样式可以通过 `<key-value>` 进行指定; `<code>` 为合法的 WolframScript/Mathics 代码;

<code>ztikz/wolfram/table/format</code>	<code>format</code>	= <code><列格式></code>	初始值: <code>*{t2}{t1}</code>
<code>ztikz/wolfram/table/header</code>	<code>header</code>	= <code><表头></code>	初始值: 空
<code>ztikz/wolfram/table/hdbt-rule</code>	<code>hdbt-rule</code>	= <code><false true></code>	初始值: <code>false</code>
<code>ztikz/wolfram/table/cell-cmd</code>	<code>cell-cmd</code>	= <code><单元格命令></code>	初始值: <code>#1</code>

`<format>` 用于设置表格的列格式; `<header>` 用于设置表格的表头, 该参数会在表格的第一行输出; `<hdbt-rule>` (header bottom rule) 用于设置是否显示表头与后续表格内容之间的横线, 默认不显示; 该选项仅对 “`\wolframTable*`” 命令有效, 即仅在需要排版表格时有效; `<cell-cmd>` 接受一个参数, 其将应用于每一个 `\wolframTablePData` 中的单元格 (不包括表头), 使用 “`#1`” 表示当前单元格内容;

`\wolframTablePData` *

`\wolframTableFData` *

New: 2025-05-18

这两个命令表示最近一次 `\wolframTable` 命令的运算结果, 不含有 `\hline`, `\midrule` 等命令, 即二者仅含有纯表格数据; `\wolframTablePData` (Part Data) 返回的数据不包括表格的表头, `\wolframTableFData` (Full Data) 返回的数据包括表格的表头; 此二命令可以传入 `tabularray` 等表格排版宏包的数据输入. **注意:** 命令 `\wolframTable` 中的 `<cell-cmd>` 不建议与 `tabularray` 中的 `<cells/cmd>` 混用.

```
\wolframTable*[
  format=cccc, hdbt-rule,
```

例 20


```

& \wolframResult*[3-1]
\end{align}
\wolframSolve[
  var={x, y}, domain=Integers
]{x^2 + 2 y^3 == 3681 && x > 0 && y > 0}
\begin{align}
  \wolframResult
\end{align}

```

$$x = \frac{9}{a+b}, y = -\frac{a-8b}{a+b} \quad (5.1)$$

$$x = \frac{9}{a+b} || y = -\frac{a-8b}{a+b} \quad (5.2)$$

$$x = \frac{9}{a+b} \quad (5.3)$$

$$y = -\frac{a-8b}{a+b} \quad (5.4)$$

$$x = 15, y = 12, x = 41, y = 10, x = 57, y = 6 \quad (5.5)$$

`\wolframDSolve` `\wolframDSolve[⟨key-value⟩]{⟨equation⟩}`

New: 2025-05-15

`\wolframDSolve*{⟨full code⟩}`

此命令用于调用 WolframScript 中的进行微分方程的求解, 具有缓存机制; `⟨equation⟩` 表示方程的表达式; `⟨key-value⟩` 用于设置求解的自变量与定义域; `⟨full code⟩` 为完整的微分方程表达式, 包含自变量, 因变量;

<code>ztikz/wolfram/dsolve/depend</code>	<code>depend</code>	= ⟨因变量⟩..... 初始值: <code>y[x]</code>
<code>ztikz/wolfram/dsolve/independ</code>	<code>independ</code>	= ⟨自变量⟩..... 初始值: <code>x</code>

`⟨depend⟩` 用于指定该微分方程的因变量, 比如 `⟨depend⟩=y[x]` 表示 y 是 x 的函数;
`⟨independ⟩` 用于指定该微分方程的自变量, 比如 `⟨independ⟩=x` 表示 x 是自变量;

```

\wolframDSolve{y'[x] + y[x] == a*Sin[x], y[0] == 1}
\begin{align}
  &\wolframResult
\end{align}
\wolframDSolve[depend={y[x], z[x]},{y'[x] == Exp[z[x]] + 1, z'[x] == y[x] - x}
\begin{align}\left\{\begin{aligned}
  &\wolframResult[\\&
\end{aligned}\right\}
\end{aligned}\right. \end{align}

```

例 22

$$y(x) = -\frac{1}{2}e^{-x}(-ae^x \sin(x) + ae^x \cos(x) - a - 2) \quad (5.6)$$

$$\begin{cases} z(x) = \log \left(c_1 \tan^2 \left(\frac{1}{2} \left(\sqrt{2} \sqrt{c_1} x + 2 \sqrt{2} \sqrt{c_1} c_2 \right) \right) + c_1 \right) \\ y(x) = x + \sqrt{2} \sqrt{c_1} \tan \left(\frac{1}{2} \left(\sqrt{2} \sqrt{c_1} x + 2 \sqrt{2} \sqrt{c_1} c_2 \right) \right) \end{cases} \quad (5.7)$$

wolframGraphics `\begin{wolframGraphics}[\langle spec \rangle]`

`\plot code`

New: 2025-05-15

`\end{wolframGraphics}`

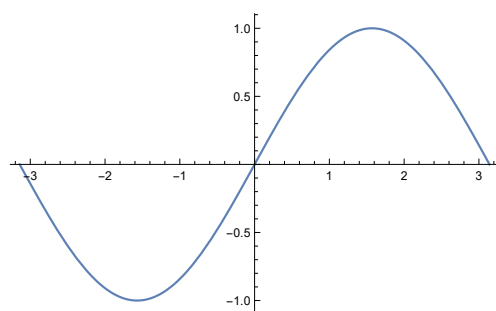
此环境用于调用 WolframScript 进行绘图, 具有缓存机制; $\langle spec \rangle$ 用于设置图片的排版参数, 默认为空, 此时该环境不会返回任何的结果, 可以通过 `\wolframOutputFile` 调用其产生的文件; $\langle spec \rangle$ 可以设置值, 对应图片的排版参数, 比如 `width=10em`; 若 $\langle spec \rangle$ 非空, 则该环境会返回: `\includegraphics[\langle spec \rangle]{\langle path \rangle/\langle HASH \rangle.pdf}`, 其中 $\langle HASH \rangle$ 为当前 wolframGraphics 环境中代码的 Hash 值, $\langle path \rangle$ 为 WolframScript 缓存文件夹对应的目录.

NOTE: $\langle plot code \rangle$ 中最后得到的图片名称必须为 “FIGURE”, 否则会报错.

```
\begin{wolframGraphics}
  FIGURE=Plot[Sin[x], {x, -Pi, Pi}]
\end{wolframGraphics}

\includegraphics[width=.5\linewidth]{\wolframOutputFile}
```

例 23



5.6 l3draw 库

zTikZ 基于 l3draw 宏包封装了一个 l3draw 库, 此库主要用于完成一些比较简单的绘图需求. 在普通用户层面: l3draw 库提供了 \zrule 和 \zplot 两个命令, 前者用于绘制渐变矩形, 后者用于绘制函数, 同样也支持渐变; zTikZ 也对 l3draw 提供的绘图环境与命令进行了简单的封装, 目前不是很完善, 且不稳定, 不推荐普通用户使用.


\zdrawSetUnit	\zdrawSetUnit[⟨unit⟩]
New: 2025-05-15	此命令用于设置当前绘图的单位, 例如 ⟨unit⟩ 可以取值为 “cm”.

\zdrawSetPathWidth	\zdrawSetPathWidth[⟨width⟩]
New: 2025-05-15	此命令用于设置当前绘图的线宽, 例如 ⟨width⟩ 可以取值为 “0.5pt”; l3draw 中默认的线径为 0.4pt.

\zrule	\zrule[⟨key-value⟩]
New: 2025-05-15	此命令用于绘制渐变矩形, ⟨key-value⟩ 用于设置渐变矩形的属性.

ztikz/zdraw/zrule/width	width	= ⟨浮点数⟩	初始值:	1
ztikz/zdraw/zrule/height	height	= ⟨浮点数⟩	初始值:	1
ztikz/zdraw/zrule/startColor	startColor	= ⟨颜色⟩	初始值:	red
ztikz/zdraw/zrule/endColor	endColor	= ⟨颜色⟩	初始值:	blue
ztikz/zdraw/zrule/step	step	= ⟨浮点数⟩	初始值:	0.25
⟨width⟩ 和 ⟨height⟩ 用于设置渐变矩形的宽度和高度; ⟨startColor⟩ 和 ⟨endColor⟩ 用于设置渐变矩形的起始颜色和结束颜色; ⟨step⟩ 用于控制渐变精度.					

```
\zrule[width=10, startColor=red, step=1]
```



例 24

\zplot	\zplot[⟨key-value⟩]{⟨function⟩}
New: 2025-05-15	此命令用于绘制函数, 水平方向和垂直方向的渐变, ⟨key-value⟩ 用于设置函数的属性; ⟨function⟩ 为合法的函数表达式.

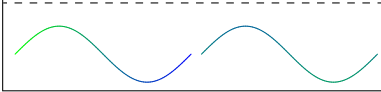
NOTE: 目前 \zplot 命令不太稳定, 在部分情况下可能会报错, 用户应该谨慎使用该命令.

ztikz/zdraw/zplot/action	action = $\langle \text{draw stroke fill clip shade} \rangle$初始值: draw
ztikz/zdraw/zplot/domain	domain = $\langle \text{浮点数, 浮点数, 浮点数} \rangle$初始值: -5,0.1,5
ztikz/zdraw/zplot/range	range = $\langle \text{浮点数, 浮点数} \rangle$初始值: -5, 5
ztikz/zdraw/zplot/startColor	startColor = $\langle \text{颜色} \rangle$初始值: black
ztikz/zdraw/zplot/endColor	endColor = $\langle \text{颜色} \rangle$初始值: white
ztikz/zdraw/zplot/axis	axis = $\langle \text{x y} \rangle$初始值: y

$\langle \text{action} \rangle$ 用于控制绘制的行为; $\langle \text{domain} \rangle$ 用于设置函数的自变量范围, 其中第一个浮点数为起始值, 第二个浮点数为步长, 第三个浮点数为结束值; $\langle \text{range} \rangle$ 用于设置 y 轴范围, 在 $\langle \text{action} \rangle = \text{shade}$ 时比较有用; $\langle \text{startColor} \rangle$ 和 $\langle \text{endColor} \rangle$ 用于设置函数的起始颜色和结束颜色; $\langle \text{axis} \rangle$ 用于设置渐变方式, ‘x’ 对应水平渐变, ‘y’ 对应垂直渐变.

```
\def\PI{3.1415926}
\zplot[
  domain={0, 0.02*\PI, 2*\PI},
  action=shade, startColor=blue,
  endColor=green, axis=x]{sin(x)}
\zplot[
  domain={0, 0.02*\PI, 2*\PI},
  action=shade, startColor=blue,
  endColor=green, axis=y]{sin(x)}
```

例 25



Zdraw	\begin{zdraw} $\langle \text{l3draw code} \rangle$ \end{zdraw}
New: 2025-05-15	此环境为 \draw_begin: 和 \draw_end: 的封装.
Zgroup	\begin{zgroup} $\langle \text{l3draw code} \rangle$ \end{zgroup}
New: 2025-05-15	此环境为 \draw_path_scope_begin: 和 \draw_path_scope_end: 的封装.
\zmoveto	\zmoveto $\langle \text{coordinate} \rangle$
\zlineto	\zlineto $\langle \text{coordinate} \rangle$
New: 2025-05-15	这两个命令用于移动当前画笔的坐标, $\langle \text{coordinate} \rangle$ 为 l3draw 中合法的坐标表达式. 比如 “1mm, 2cm+3em”.
\zscolor	\zmoveto $\langle \text{l3color} \rangle$
\zfcolor	\zlineto $\langle \text{l3color} \rangle$
New: 2025-05-15	这两个命令用于移动当前画笔的坐标, $\langle \text{l3color} \rangle$ 为 l3draw 中合法的颜色表达式; zTikZ 对常见的颜色预定义了其对应的 l3color 变量, 一些常见的颜色, 用户可以直接使用.

<code>\zxvec</code>	<code>\zxvec{<coordinate>}</code>
<code>\zyvec</code>	<code>\zyvec{<coordinate>}</code>
New: 2025-05-15	这两个命令用于设置当前坐标系的 x 轴和 y 轴的单位向量, $\langle coordinate \rangle$ 为合法的坐标表达式; 比如 “1mm, 2cm+3em”.
<code>\zpolar</code>	<code>\zpolar{<radius>}{<angle>}</code>
<code>\zcoor</code>	<code>\zcoor{<x-scale>}{<y-scale>}</code>
New: 2025-05-15	<code>\zpolar</code> 命令按照极坐标的方式获取点的坐标: $\langle radius \rangle$ 为合法的长度, 如 “2em”; $\langle angle \rangle$ 为浮点数; <code>\zcoor</code> 命令按照直角坐标的方式获取点的坐标: $\langle x-scale \rangle$ 为浮点数, $\langle y-scale \rangle$ 为浮点数; 此命令获取的最终坐标还取决于 x 和 y 方向两个基向量的影响, $(\langle x-scale \rangle, \langle y-scale \rangle)$ 也就是所谓的在基 $\{\backslash svec, \backslash yvec\}$ 下的坐标.
<code>\zrect</code>	<code>\zrect{<coordinate>}{<coordinate>}</code>
<code>\zcirc</code>	<code>\zcirc{<center>}{<radius>}</code>
New: 2025-05-15	前者用于绘制矩形, 两个坐标点分别为矩形的左下角和右上角; 后者用于绘制圆形, $\langle center \rangle$ 为圆心坐标, $\langle radius \rangle$ 为半径; $\langle coordinate \rangle$ 和 $\langle center \rangle$ 均为合法的坐标表达式, 比如 “1mm, 2cm+3em”.
<code>\znewtext</code>	<code>\znewtext{<coffin>}</code>
<code>\zsethtext</code>	<code>\zsethtext{<coffin>}{<content>}</code>
<code>\zsetvtext</code>	<code>\zsetvtext{<coffin>}{<width>}{<content>}</code>
<code>\zscaletext</code>	<code>\zscaletext{<coffin>}{<x-scale>}{<y-scale>}</code>
<code>\zputtext</code>	<code>\zputtext{<coffin>}{<hpole>}{<vpole>}{<point>}</code>
New: 2025-05-15	这系列命令用于在 <code>l3draw</code> 中创建, 变换与放置文本.
<code>\zbg</code>	这两个命令为 <code>\draw_path_scope_begin:</code> 和 <code>\draw_path_scope_end:</code> 的封装.
<code>\zeg</code>	
New: 2025-05-15	
<code>\zcapbutt</code>	这系列命令用于设置线段之间的连接方式.
<code>\zcaproun</code>	
<code>\zcaprect</code>	
<code>\zclosepath</code>	
New: 2025-05-15	

<code>\zshift</code>	<code>\zshift{⟨vector⟩}</code>
<code>\zxscale</code>	<code>\zxscale{⟨x-scale⟩}</code>
<code>\zyscale</code>	<code>\zyscale{⟨y-scale⟩}</code>
<code>\ztrans</code>	<code>\ztrans{⟨a⟩}{⟨b⟩}{⟨b⟩}{⟨d⟩}</code>

New: 2025-05-15

这一系列的命令用于对坐标轴进行仿射变换, `\ztrans` 对应的仿射变换矩阵为:

$$\begin{bmatrix} a & c \\ b & d \end{bmatrix}$$

<code>\usepath</code>	<code>\usepath{⟨style⟩}</code>
-----------------------	--------------------------------

New: 2025-05-15

此命令用于显示最终的路径, `⟨style⟩` 的可选值有: “draw, stroke, fill, clip”.

6 附录

6.1 gnuplot Support Functions

我们在这里补充说明 gnuplot 中内建的函数: Arguments to math functions in gnuplot can be integer, real, or complex unless otherwise noted. Functions that accept or return angles (e.g. $\sin(x)$) treat angle values as radians, but this may be changed to degrees using the command `set angles`. (摘录自: [gnuplot support functions](#))

表 1: gnuplot math library functions

Function	Arguments	Returns
$\text{abs}(x)$	any	$ x $, absolute value of x ; same type
$\text{abs}(x)$	complex	length of x , $\sqrt{\text{Re}(x)^2 + \text{Im}(x)^2}$
$\text{acos}(x)$	any	$\cos^{-1} x$ (inverse cosine)
$\text{acosh}(x)$	any	$\cosh^{-1} x$ (inverse hyperbolic cosine) in radians
$\text{airy}(x)$	any	Airy function $\text{Ai}(x)$
$\text{arg}(x)$	complex	the phase of x
$\text{asin}(x)$	any	$\sin^{-1} x$ (inverse sine)
$\text{asinh}(x)$	any	$\sinh^{-1} x$ (inverse hyperbolic sine) in radians
$\text{atan}(x)$	any	$\tan^{-1} x$ (inverse tangent)
$\text{atan2}(y, x)$	int or real	$\tan^{-1}(y/x)$ (inverse tangent)
$\text{atanh}(x)$	any	$\tanh^{-1} x$ (inverse hyperbolic tangent) in radians
$\text{EllipticK}(k)$	real k in $(-1 : 1)$	$K(k)$ complete elliptic integral of the first kind
$\text{EllipticE}(k)$	real k in $[-1 : 1]$	$E(k)$ complete elliptic integral of the second kind
$\text{EllipticPi}(n, k)$	real $n, k < 1$	$\Pi(n, k)$ complete elliptic integral of the third kind
$\text{besj0}(x)$	int or real	J_0 Bessel function of x , in radians
$\text{besj1}(x)$	int or real	J_1 Bessel function of x , in radians
$\text{besy0}(x)$	int or real	Y_0 Bessel function of x , in radians
$\text{besy1}(x)$	int or real	Y_1 Bessel function of x , in radians
$\text{ceil}(x)$	any	$\lceil x \rceil$, smallest integer not less than x (real part)
$\text{cos}(x)$	radians	$\cos x$, cosine of x
$\text{cosh}(x)$	any	$\cosh x$, hyperbolic cosine of x in radians
$\text{erf}(x)$	any	$\text{erf}(\text{Re}(x))$, error function of $\text{Re}(x)$
$\text{erfc}(x)$	any	$\text{erfc}(\text{Re}(x))$, 1.0– error function of $\text{Re}(x)$
$\text{exp}(x)$	any	e^x , exponential function of x
$\text{expint}(n, x)$	any	$E_n(x)$, exponential integral function of x
$\text{floor}(x)$	any	$\lfloor x \rfloor$, largest integer not greater than x (real part)

$\text{gamma}(x)$	any	$\Gamma(\text{Re}(x))$, gamma function of $\text{Re}(x)$
$\text{ibeta}(p, q, x)$	any	$\text{ibeta}(\text{Re}(p, q, x))$, ibeta function of $\text{Re}(p, q, x)$
$\text{inverf}(x)$	any	inverse error function $\text{Re}(x)$
$\text{igamma}(a, x)$	any	$\text{igamma}(\text{Re}(a, x))$, igamma function of $\text{Re}(a, x)$
$\text{imag}(x)$	complex	$\text{Im}(x)$, imaginary part of x as a real number
$\text{invnorm}(x)$	any	inverse normal distribution function $\text{Re}(x)$
$\text{int}(x)$	real	integer part of x , truncated toward zero
$\text{lambertw}(x)$	real	Lambert W function
$\text{lgamma}(x)$	any	$\text{lgamma}(\text{Re}(x))$, lgamma function of $\text{Re}(x)$
$\text{log}(x)$	any	$\ln x$, natural logarithm (base e) of x
$\text{log10}(x)$	any	$\log_{10} x$, logarithm (base 10) of x
$\text{norm}(x)$	any	$\text{norm}(x)$, normal distribution function of $\text{Re}(x)$
$\text{rand}(x)$	int	pseudo random number in the interval $(0 : 1)$
$\text{real}(x)$	any	$\text{Re}(x)$, real part of x
$\text{sgn}(x)$	any	1 if $x > 0$, -1 if $x < 0$, 0 if $x = 0$. $\Im(x)$ ignored
$\text{sin}(x)$	any	$\sin x$, sine of x
$\text{sinh}(x)$	any	$\sinh x$, hyperbolic sine of x in radians
$\text{sqrt}(x)$	any	\sqrt{x} , square root of x
$\text{tan}(x)$	any	$\tan x$, tangent of x
$\text{tanh}(x)$	any	$\tanh x$, hyperbolic tangent of x in radians
$\text{voigt}(x, y)$	real	convolution of Gaussian and Lorentzian
$\text{cerf}(z)$	complex	complex error function
$\text{cdawson}(z)$	complex	complex Dawson's integral
$\text{faddeeva}(z)$	complex	$w(z) = \exp(-z^2) \times \text{erfc}(-iz)$
$\text{erfi}(x)$	real	imaginary error function $\text{erfi}(x) = -i \times \text{erf}(ix)$
$\text{VP}(x, \sigma, \gamma)$	real	Voigt profile

■ 注记 6.1 $\text{faddeeva}(z)$: rescaled complex error function

6.2 marker style

TikZ 中的可以使用的 Marker 样式表如下:

```

\pgfuseplotmark{-}
\pgfuseplotmark{|}
\pgfuseplotmark{o}
\pgfuseplotmark{asterisk}
\pgfuseplotmark{star}
\pgfuseplotmark{10-pointed star}
\pgfuseplotmark{oplus}
\pgfuseplotmark{oplus*}
\pgfuseplotmark{otimes}
\pgfuseplotmark{otimes*}
\pgfuseplotmark{square}
\pgfuseplotmark{square*}
\pgfuseplotmark{triangle}
\pgfuseplotmark{triangle*}
\pgfuseplotmark{diamond}
\pgfuseplotmark{diamond*}
\pgfuseplotmark{halfdiamond*}
\pgfuseplotmark{halfsquare*}
\pgfuseplotmark{halfsquare right*}
\pgfuseplotmark{halfsquare left*}
\pgfuseplotmark{pentagon}
\pgfuseplotmark{pentagon*}
\pgfuseplotmark{Mercedes star}
\pgfuseplotmark{Mercedes star flipped}
\pgfuseplotmark{halfcircle}
\pgfuseplotmark{halfcircle*}
\pgfuseplotmark{heart}
\pgfuseplotmark{text}

```

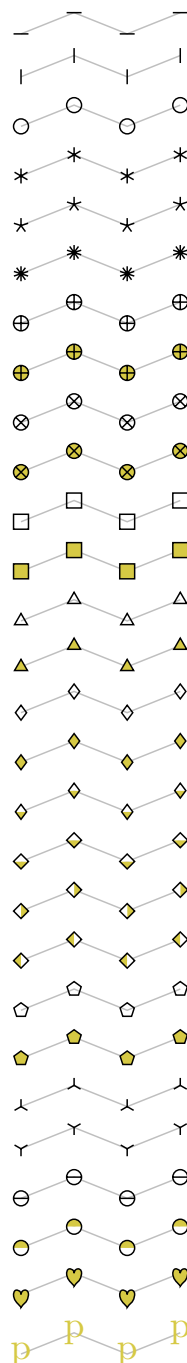


图 3: TikZ Marker Style

6.3 测试数据/代码

```
# Curve 0 of 1, 10 points
# Curve title: "f(x)"
# x y type
-3.14159 -0.00000 i
-2.44346 -0.64279 i
-1.74533 -0.98481 i
-1.04720 -0.86603 i
-0.34907 -0.34202 i
0.34907 0.34202 i
1.04720 0.86603 i
1.74533 0.98481 i
2.44346 0.64279 i
3.14159 0.00000 i
```

sine.data

```
\begin{pycode}{pycode_table.txt}
import numpy as np

# write file
with open ('pycode_table.txt', 'w') as file:
    file.write("\\begin{tabular}{p{3cm}ccc}\\n")
    file.write("\\hline\\n")
    file.write("number/function & $\\sin$ & $\\cos$ & $\\tan$\\n")
    file.write("\\hline\\n")
    for i in range(1, 16):
        file.write(
            f"${i}$ & ${np.around(np.sin(i), decimals=4)}$ &
            ${np.around(np.cos(i), decimals=4)}$ & ${np.around(np.tan(i),
            decimals=4)}$\\n"
        )

    file.write("\\hline\\n")
    file.write("\\end{tabular}\\n")
\\end{pycode}
```

table.py.txt

✓
✓

7 TODO

$\text{\texttt{\textit{tikz}}}$ 的开发暂且告一段落了, 这里列出部分将来可能会增加的功能 (☐ – 未完成; ☒ – 已完成; ☐ – 不考虑该功能):

- ☐ 实现类似 `tikz-3dplot` 的接口, 使用 $\text{\texttt{L\text{A}T\text{E}X3}}$ 对其进行重写.
- ☐ 增加 Matlab 脚本的调用接口, 或者直接使用其开源替代 `GNU Octave` ?
- ☐ 实现 `wolframAny` 环境, 该环境实现的功能类似 `pycode`.
- ☐ 重写缓存机制对应的函数 `\ztikz_hash_if_change:nn`, 目前不够灵活 (或许直接使用 `robust-externalize` 宏包).
- ☐ 针对 `cache` 库, 需要清除多余的 Hash 值: 例如某个环境/命令产生的原 Hash 值为 “A”, 对应环境/命令中的参数改变后, 其 Hash 值变为了 “B”, 那么此时需要清除原始的 “A”.
- ☐ 实现 `\ztikzForceToSkip` 命令: 如何处理 `\xsim_file_write_start:nn` 环境? 如何处理多个 `\ztikzForceToRun` ? 也许使用 `\str_mdfive_hash:n` 会有帮助?

8.1 ztikz.sty

```

1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 1
2  %% ztikz.sty % 2
3  %% Copyright 2024, 2025 Zongping Ding. % 3
4  % % 4
5  % This work may be distributed and/or modified under the conditions of the % 5
6  % LaTeX Project Public License, either version 1.3 of this license or any % 6
7  % later version. % 7
8  % The latest version of this license is in % 8
9  % http://www.latex-project.org/lppl.txt % 9
10 % and version 1.3 or later is part of all distributions of LaTeX % 10
11 % version 2005/12/01 or later. % 11
12 % % 12
13 % This work has the LPPL maintenance status `maintained'. % 13
14 % % 14
15 % The Current Maintainer of this work is Zongping Ding. % 15
16 % % 16
17 % This work consists of the files ztikz.sty, % 17
18 % the libraries: ztikz.library.gnuplot.tex, % 18
19 % ztikz.library.cache.tex, % 19
20 % ztikz.library.python.tex, % 20
21 % ztikz.library.wolfram.tex, % 21
22 % ztikz.library.l3draw.tex, % 22
23 % and the script files: ztikz.library.gnuscript.tex, % 23
24 % ztikz.library.pyscript.tex. % 24
25 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 25
26 \NeedsTeXFormat{LaTeX2e} 26
27 \ProvidesExplPackage{ztikz}{2025/05/20}{1.0.0}{A~pre-release-diagram~plot~package} 27
28 28
29 29
30 % ----- 30
31 % basic tools 31
32 % ----- 32
33 \RequirePackage{ztool} 33
34 \RequirePackage{graphicx} 34
35 \__ztool_load_library:n {shell-escape, file-io} 35
36 \cs_new_protected:Npn \ztikz_hook_preamble_last:n #1 36
37 { \AddToHook{env/document/before}{#1} } 37
38 \cs_new_protected:Npn \ztikz_label_hook_preamble_last:nn #1#2 38
39 { \AddToHook{env/document/before}{#1}{#2} } 39
40 \cs_new_protected:Npn \ztikz_hook_doc_begin:n #1 40
41 { \AddToHook{begindocument}{#1} } 41
42 \cs_new_protected:Npn \ztikz_hook_doc_end:n #1 42
43 { \AddToHook{enddocument}{#1} } 43
44 44
45 45
46 46
47 % ----- 47
48 % load module interface 48

```

```

49 % -----
50 \clist_new:N \g__ztikz_library_loaded_clist
51 \clist_gclear:N \g__ztikz_library_loaded_clist
52 \cs_new_nopar:Npn \__ztikz_load_library:n #1
53 {
54     \clist_map_inline:nn {#1} {
55         \clist_if_in:NnTF \g__ztikz_library_loaded_clist {##1} {
56             \msg_set:nnn {ztikz} {library-loaded}
57             {
58                 ztikz-library~"##1"~already~loaded,ignored~loading.
59                 \msg_line_context:
60             }
61             \msg_warning:nnn {ztikz} {library-loaded} {##1}
62         }{
63             \file_if_exist:nTF {library/ztikz.library.##1.tex}{
64                 \clist_gput_right:Nn \g__ztikz_library_loaded_clist {##1}
65                 \makeatletter\file_input:n {library/ztikz.library.##1.tex}
66             }{
67                 \msg_set:nnn {ztikz} {library-not-found} {ztikz-library~`##1'~not~found.}
68                 \msg_error:nnn {ztikz} {library-not-found} {##1}
69             }
70         }
71     }
72 }
73 \NewDocumentCommand\ztikzloadlib{m}
74 {
75     \__ztikz_load_library:n {#1}
76     \ExplSyntaxOff
77 }
78
79
80
81 % -----
82 %                               Init ztikz's Environment
83 % -----
84 \ztool_shell_mkdir:n {ztikz_output/}
85 \tl_const:Nn \g__ztikz_scripts_path_tl {ztikz_output/scripts}
86 \NewDocumentCommand\ztikzMkdir{m}{ \ztool_shell_mkdir:n {#1} }
87 \cs_new_protected:Npn \ztikz_term_info:n #1
88 {
89     \iow_now:Nn \c_term_iow {#1}
90 }
91 \cs_new_protected:Npn \ztikz_term_info:e #1
92 {
93     \iow_now:Ne \c_term_iow {#1}
94 }
95
96
97
98 % -----
99 %                               ztikz's options
100 % -----

```

```

101 \cs_new_protected:Npn \ztikz_keys_define:n #1
102 { \keys_define:nn { ztikz }{ #1 } }
103 \cs_new_protected:Npn \ztikz_keys_define:nn #1#2
104 { \keys_define:nn { ztikz / #1 }{ #2 } }
105 \cs_new_protected:Npn \ztikz_keys_set:nn #1#2
106 { \keys_set:nn { ztikz / #1 }{ #2 } }
107 \ztikz_keys_define:n
108 {
109     library .multichoice:,
110     library / basic .code:n = { \__ztikz_load_library:n {basic} },
111     library / gnuplot .code:n = { \__ztikz_load_library:n {gnuplot} },
112     library / cache .code:n = { \__ztikz_load_library:n {cache} },
113     library / python .code:n = { \__ztikz_load_library:n {python} },
114     library / wolfram .code:n = { \__ztikz_load_library:n {wolfram} },
115     library / l3draw .code:n = { \__ztikz_load_library:n {l3draw} },
116     library / unknown .code:n = {
117         \msg_set:nnn {ztikz} {library-not-found}
118         {ztikz~library~`#1'~not~found,~valid~libraries~are~'basic',
119         ~'gnuplot',~'cache',~'python',~'wolfram'~and~'l3draw'}.}
120         \msg_error:nn {ztikz} {library-not-found}
121     },
122     wolfram .meta:nn = { ztikz/wolfram }{ #1 },
123 }
124
125 \tl_new:N \g__ztikz_wolfram_engine_tl
126 \bool_new:N \g__ztikz_wolfram_cloud_bool
127 \ztikz_keys_define:nn { wolfram }
128 {
129     engine .choice:,
130     engine / wolfram .code:n =
131     {
132         \tl_gset:Nn \g__ztikz_wolfram_engine_tl { wolframscript }
133     },
134     engine / mathics .code:n =
135     {
136         \tl_gset:Nn \g__ztikz_wolfram_engine_tl { mathics }
137     },
138     engine / unknown .code:n =
139     {
140         \msg_set:nnn {ztikz} {wolfram-engine-not-found}
141         {ztikz's~'wolfram'~engine~option~`#1'~invalid,
142         ~valid~engine~is~'wolframscript'~or~'mathics'}.}
143         \msg_error:nn {ztikz} {wolfram-engine-not-found}
144     },
145     engine .initial:n = { wolfram },
146     cloud .choices:nn = { true, false }
147     {
148         \use:c { bool_gset_ \l_keys_choice_tl :N }
149         \g__ztikz_wolfram_cloud_bool
150     },
151     cloud .initial:n = { false },
152     cloud .default:n = { true },

```

```

153     }
154     \ProcessKeyOptions [ ztikz ]
155     % only 'wolfram' support 'cloud'
156     \bool_if:NT \g__ztikz_wolfram_cloud_bool
157     {
158         \tl_if_eq:VnT \g__ztikz_wolfram_engine_tl { mathics }
159         {
160             \msg_set:nnn {ztikz}{mathics-cloud}
161             {mathics~'cloud'~is~not~supported.}
162             \msg_error:nn {ztikz}{mathics-cloud}
163         }
164     }
165
166
167
168 % -----
169 %                               cache, sed file and ztikz logo
170 % -----
171 \cs_generate_variant:Nn \ior_open:Nn { Ne }
172 \cs_generate_variant:Nn \iow_open:Nn { Ne }
173 \cs_generate_variant:Nn \tl_map_function:nN { eN }
174 \ztikz_hook_preamble_last:n
175 {
176     \clist_if_in:NnF \g__ztikz_library_loaded_clist {cache}
177     {
178         \prg_new_conditional:Npnn \ztikz_hash_if_change:nn #1#2 {p, T, F, TF}
179         {
180             \bool_gset_true:N \g__hash_change_bool
181             \prg_return_true:
182         }
183         \prg_generate_conditional_variant:Nnn \ztikz_hash_if_change:nn
184         { ne } { p, T, F, TF }
185     }
186 }
187 % sed script
188 \cs_new:Npn \__ztikz_addto_script:nn #1#2
189 {% #1: file; #2: line; #3:new
190     \bool_if:NF \g__ztikz_hashchg_norun_bool
191     {
192         \ztool_append_to_file:nn
193         {#1}{#2}
194     }
195 }
196 \cs_generate_variant:Nn \__ztikz_addto_script:nn { ne, en, ee }
197 \ztikz_keys_define:nn { sed-script }
198 {
199     file .tl_set:N = \l__ztikz_sed_script_file_tl,
200     line .int_set:N = \l__ztikz_sed_script_line_int,
201     ori .tl_set:N = \l__ztikz_sed_script_ori_tl,
202     new .tl_set:N = \l__ztikz_sed_script_new_tl,
203 }
204 \cs_new:Npn \__ztikz_sed_script:nnn #1#2#3

```

205	{% #1: file; #2: line; #3:new	205
206	\bool_if:NF \g__ztikz_hashchg_norun_bool	206
207	{	207
208	\ztool_replace_file_line:nnn	208
209	{\g__ztikz_scripts_path_tl/#1}	209
210	{#2}{#3}	210
211	}	211
212	}	212
213	\cs_generate_variant:Nn __ztikz_sed_script:nnn {eee}	213
214	\cs_generate_variant:Nn __ztikz_sed_script:nnn {nne}	214
215	% hash change norun	215
216	\bool_new:N \g__ztikz_hashchg_norun_bool	216
217	\bool_gset_false:N \g__ztikz_hashchg_norun_bool	217
218	% ztik logo	218
219	\NewDocumentCommand\TikZ{}{Ti\textcolor{orange}{\textit{k}}Z}	219
220	\NewDocumentCommand\zTikZ{}	220
221	{	221
222	\ztool_scale_to_wd_and_ht:nnn {.9ex}{1.3ex}{	222
223	\ztool_rotate:nn {89}{(\aleph)}	223
224	}\kern-0.3423ex\hbox{\TikZ}	224
225	}	225
226	\let\ztikz\zTikZ	226

8.2 Library

8.2.1 basic

```
1 \ProvidesExplFile{ztikz.library.basic.tex}{2024/12/17}{1.0.0}{basic~library~for~  ✓ 1
  ztikz}
2
3
4
5 % -----
6 %                               basic packages
7 % -----
8 \RequirePackage{tikz}
9 \RequirePackage{etoolbox}
10 \patchcmd{\pgfutil@InputIfFileExists}{\input #1}{%
11   \@pushfilename
12   \xdef\@currname{#1}
13   \input #1
14   \@popfilename
15 }{}{}
16 \usetikzlibrary{arrows.meta}
17 \usetikzlibrary{intersections}
18 \usetikzlibrary{patterns}
19 \usetikzlibrary{plotmarks}
20 \usetikzlibrary{positioning}
21 \usetikzlibrary{shapes.geometric}
22 \usetikzlibrary{decorations.markings}
23 \usetikzlibrary{fadings}
24
25
26
27 % ==> coordinate basic components
28 \ztikz_keys_define:nn { point }
29 {
30   type      .str_set:N = \l__point_type_str,
31   type      .initial:n = { * },
32   radius    .dim_set:N = \l__point_radius_dim,
33   radius    .initial:n = { 1pt },
34   color     .tl_set:N = \l__point_color_tl,
35   color     .initial:n = { black },
36   opacity   .tl_set:N = \l__point_opacity_tl,
37   opacity   .initial:n = { 1 },
38   rotate    .fp_set:N = \l__point_rotate_angle,
39   rotate    .initial:n = { 0 },
40 }
41 \NewDocumentCommand\ShowPoint{ O{}mO{}O{} }
42 {
43   \group_begin:
44   \exp_args:Nne \ztikz_keys_set:nn { point } { #1 }
45   \seq_set_split:Nnn \l__point_list_seq { ; }{#2}
46   \seq_set_split:Nnn \l__point_label_seq { ; }{#3}
47   \int_step_inline:nnnn {1}{1}{\seq_count:N \l__point_list_seq}{
48     \draw plot [
```

```

49         only~ marks,
50         mark = \str_use:N \l__point_type_str,
51         mark~ size = \dim_use:N \l__point_radius_dim,
52         mark~ options = {
53             rotate = \fp_use:N \l__point_rotate_angle,
54             opacity = \tl_use:N \l__point_opacity_tl,
55             color = \tl_use:N \l__point_color_tl,
56             ball~ color = \tl_use:N \l__point_color_tl,
57         }
58     ] coordinates{\seq_item:Nn \l__point_list_seq{##1}}
59     node[#4] {\seq_item:Nn \l__point_label_seq{##1}};
60 }
61 \group_end:
62 }
63 \NewDocumentCommand\ShowGrid{ 0{color=gray, very~ thin, step=1}m }
64 {
65     \seq_set_split:Nnn \l__grid_param_ii_seq { ; }{#2}
66     \draw[#1] \seq_item:Nn \l__grid_param_ii_seq{1} grid \seq_item:Nn
        \l__grid_param_ii_seq{2};
67 }
68 % intersection
69 \NewDocumentCommand\ShowIntersection{ omm }
70 {
71     \seq_set_split:Nnn \l__intersection_num_seq { ; }{#2}
72     \path[name~ intersections={
73         of=\seq_item:Nn \l__intersection_num_seq{1}~
74         and~ \seq_item:Nn \l__intersection_num_seq{2}
75     }];
76     \int_step_inline:nnnn {1}{1}{#3}{
77         \ShowPoint[#1]{(intersection-##1)}
78     }
79 }
80 % polygon plot
81 \ztikz_keys_define:nn { polygon }
82 {
83     radius .fp_set:N = \l__polygon_radius_fp,
84     radius .initial:n = { 1 },
85     edgeColor .tl_set:N = \l__polygon_edge_color_tl,
86     edgeColor .initial:n = { black },
87     fillColor .tl_set:N = \l__polygon_fill_color_tl,
88     fillColor .initial:n = { },
89     fillOpacity .fp_set:N = \l__polygon_fill_opacity_fp,
90     fillOpacity .initial:n = { 0 },
91     rotate .fp_set:N = \l__polygon_rotate_angle,
92     rotate .initial:n = { 0 },
93     shift .tl_set:N = \l__polygon_shift_tl,
94     shift .initial:n = { (0,0) },
95     marker .tl_set:N = \l__polygon_marker_option_tl,
96     marker .initial:n = { },
97 }
98 \tl_new:N \l__ztikz_basic_poly_path_tl
99 \NewDocumentCommand\Polygon{ 0{m} }

```

```

100 {
101     \group_begin:
102     \ztikz_keys_set:nn { polygon } { #1 }
103     % strip '(' and ')'
104     \tl_replace_once:Nnn \l__polygon_shift_tl{({}{}}
105     \tl_replace_once:Nnn \l__polygon_shift_tl{)}}{}
106     \coordinate (mv) at (\tl_use:N \l__polygon_shift_tl);
107     % create polygon
108     \begin{scope}[shift=(mv), rotate=\fp_use:N \l__polygon_rotate_angle]
109     % arg require: #2 ≥ 3
110     \int_step_inline:nnn {1}{#2}{
111         % draw edges
112         \fp_set:Nn \l_angle_fp {360/#2*##1*\c_one_degree_fp}
113         \fp_set:Nn \l_angle_next_fp {360/#2*(##1+1)*\c_one_degree_fp}
114         \draw [\tl_use:N \l__polygon_edge_color_tl]
115             ( \fp_eval:n {\l__polygon_radius_fp*cos(\l_angle_fp)},
116               \fp_eval:n {\l__polygon_radius_fp*sin(\l_angle_fp)})
117             -- (
118               \fp_eval:n {\l__polygon_radius_fp*cos(\l_angle_next_fp)},
119               \fp_eval:n {\l__polygon_radius_fp*sin(\l_angle_next_fp)})
120             );
121         % fill polygon path
122         \int_compare:nNnTF {##1}<{#2}
123         {
124             \tl_put_right:Nn \l__ztikz_basic_poly_path_tl {(p##1)--}
125             }{
126             \tl_put_right:Nn \l__ztikz_basic_poly_path_tl {(p##1)--cycle}
127             }
128         % mark coordinates
129         \coordinate (p##1) at (
130             \fp_eval:n {\l__polygon_radius_fp*cos(\l_angle_fp)},
131             \fp_eval:n {\l__polygon_radius_fp*sin(\l_angle_fp)})
132         );
133     }
134     % fill polygon (none-color -> opacity=1; or opacity=.75)
135     \tl_if_empty:NTF \l__polygon_fill_color_tl {
136         \fp_set:Nn \l__polygon_fill_opacity_fp {0}
137     }{
138         \fp_set:Nn \l__polygon_fill_opacity_fp {.75}
139     }
140     \fill [\tl_use:N \l__polygon_fill_color_tl, fill~opacity=\fp_use:N
141           \l__polygon_fill_opacity_fp] \l__ztikz_basic_poly_path_tl;
142     % show markers
143     \int_step_inline:nnn {1}{#2}{
144         \ShowPoint[\l__polygon_marker_option_tl]{(p##1)}
145     }
146     \end{scope}
147     \group_end:
148 }
149
150 % ==> axis

```



```

151 \ztikz_keys_define:nn { axis }
152 {
153     % basic tick args
154     tickStart      .fp_set:N = \l__start_fp,
155     tickStart      .initial:n = { -5 },
156     tickEnd        .fp_set:N = \l__end_fp,
157     tickEnd        .initial:n = { 5 },
158     axisRotate     .fp_set:N = \l__axis_rotate_angle,
159     axisRotate     .initial:n = { 0 },
160     % tick dimension spec
161     mainStep       .fp_set:N = \l__main_step_fp,
162     mainStep       .initial:n = { 1.0 },
163     subStep        .fp_set:N = \l__sub_step_fp,
164     subStep        .initial:n = { 0.1 },
165     mainTickLabel  .tl_set:N = \l__main_tick_label_tl,
166     mainTickLabel  .initial:n = { \fp_use:N {\CurrentFp} },
167     tickLabelShift .dim_set:N = \l__tick_label_shift_dim,
168     tickLabelShift .initial:n = { 0pt },
169     mainTickLength .dim_set:N = \l__main_tick_length_dim,
170     mainTickLength .initial:n = { 4pt },
171     subTickLength  .dim_set:N = \l__sub_tick_length_dim,
172     subTickLength  .initial:n = { 2pt },
173     mainTickLabelPosition .tl_set:N = \l__main_tick_label_position_tl,
174     mainTickLabelPosition .initial:n = { below },
175     % color spec
176     axisColor      .tl_set:N = \l__axis_color_tl,
177     axisColor      .initial:n = { black },
178     mainTickColor  .tl_set:N = \l__main_tick_color_tl,
179     mainTickColor  .initial:n = { black },
180     subTickColor   .tl_set:N = \l__sub_tick_color_tl,
181     subTickColor   .initial:n = { black },
182     mainTickLabelColor .tl_set:N = \l__main_tick_label_color_tl,
183     mainTickLabelColor .initial:n = { black },
184     % tick cross type spec
185     tickStyle      .choice:,
186     tickStyle/cross .code:n = \tl_set:Nn \l__tick_spec_tl { cross },
187     tickStyle/above .code:n = \tl_set:Nn \l__tick_spec_tl { above },
188     tickStyle/below .code:n = \tl_set:Nn \l__tick_spec_tl { below },
189 }
190 % ticks style
191 \tl_new:N \l__tick_type_tl % `main' or `sub'
192 \tl_new:N \l__tick_spec_tl % `cross', `above' or `below'
193 \tl_new:N \l__tick_color_tl
194 \dim_new:N \l__tick_length_dim
195 \tl_new:N \l__node_text_tl
196 % draw ticks (main or sub)
197 \cs_new_protected:Npn \ztikz_draw_axis_ticks_cs:n #1
198 {
199     \str_case:NnT \l__tick_type_tl {
200         {main}{
201             \dim_set_eq:NN \l__tick_length_dim \l__main_tick_length_dim
202             \tl_set:NV \l__tick_color_tl \l__main_tick_color_tl

```

```

203 \tl_set:Nn \l__node_text_tl {\tl_use:N \l__main_tick_label_tl} 203
204 } 204
205 {sub}{ 205
206 \dim_set_eq:NN \l__tick_length_dim\l__sub_tick_length_dim 206
207 \tl_set:NV \l__tick_color_tl \l__sub_tick_color_tl 207
208 \tl_set:Nn \l__node_text_tl {} 208
209 } 209
210 }{} 210
211 \str_case:VnT \l__tick_spec_tl { 211
212 {cross}{ 212
213 \draw[\tl_use:N \l__tick_color_tl] 213
214 (#1, 0)++(0, \dim_eval:n {\l__tick_length_dim/2}) 214
215 -- ++(0, \dim_eval:n {-\l__tick_length_dim}) 215
216 node[\tl_use:N \l__main_tick_label_position_tl] 216
217 { 217
218 \textcolor{\tl_use:N \l__main_tick_label_color_tl} 218
219 {\tl_use:N \l__node_text_tl} 219
220 }; 220
221 } 221
222 {above}{ 222
223 \draw[\tl_use:N \l__tick_color_tl] (#1, 0) 223
224 -- ++(0, \dim_eval:n {\l__tick_length_dim/2}) 224
225 node[\tl_use:N \l__main_tick_label_position_tl] 225
226 { 226
227 \textcolor{\tl_use:N \l__main_tick_label_color_tl} 227
228 {\tl_use:N \l__node_text_tl} 228
229 }; 229
230 } 230
231 {below}{ 231
232 \draw[\tl_use:N \l__tick_color_tl] (#1, 0) 232
233 -- ++(0, \dim_eval:n {-\l__tick_length_dim/2}) 233
234 node[\tl_use:N \l__main_tick_label_position_tl=\dim_use:N 234
235 \l__tick_label_shift_dim] ✓
236 { 235
237 \textcolor{\tl_use:N \l__main_tick_label_color_tl} 236
238 {\tl_use:N \l__node_text_tl} 237
239 }; 238
240 } 239
241 }{} 240
242 } 241
242 % draw axis 242
243 \fp_new:N \CurrentFp 243
244 \int_new:N \l__substep_num_int 244
245 \NewDocumentCommand\ShowAxis{0{}m} 245
246 { 246
247 \group_begin: 247
248 \ztikz_keys_set:nn { axis } { #1 } 248
249 \seq_set_split:Nnn \l__points_seq { ; }{#2} 249
250 \begin{scope}[rotate=\fp_use:N \l__axis_rotate_angle] 250
251 \draw[->, \tl_use:N \l__axis_color_tl] \seq_item:Nn \l__points_seq{1} 251
252 -- \seq_item:Nn \l__points_seq{2}; 252
253 % draw ticks 253

```

```

254 \fp_step_inline:nnnn
255 {\fp_eval:n {\l__start_fp}}
256 {\fp_use:N \l__main_step_fp}
257 {\fp_use:N \l__end_fp}
258 {
259   % main ticks
260   \tl_set:Nn \l__tick_type_tl {main}
261   \fp_gset:Nn \CurrentFp {##1}
262   \ztikz_draw_axis_ticks_cs:n {##1}
263   % sub ticks
264   \tl_set:Nn \l__tick_type_tl {sub}
265   \int_set:Nn \l__substep_num_int
266     {\fp_eval:n {floor(\l__main_step_fp/\l__sub_step_fp)}}
267   \fp_compare:nNnTF {##1}<{\fp_eval:n {floor(\l__end_fp)}}{
268     \fp_step_function:nnnN
269       {\fp_eval:n {##1+\l__sub_step_fp}}
270       {\fp_use:N \l__sub_step_fp}
271       {\fp_eval:n {##1+\l__substep_num_int*\l__sub_step_fp}}
272       \ztikz_draw_axis_ticks_cs:n
273     }{}
274   }
275 \end{scope}
276 \group_end:
277 }
278 \NewDocumentCommand{\xAxis}{0{-2}0{8}}
279 {
280   \ShowAxis[
281     tickStart=\fp_eval:n {#1+1},
282     tickEnd=\fp_eval:n {#2-0.75},
283     mainTickLabelPosition=below,
284     mainStep=1,          subStep=.25,
285     axisRotate=0,        axisColor=black,
286     mainTickColor=black, subTickColor=black,
287     mainTickLength=10pt, subTickLength=5pt,
288     tickLabelShift=0pt,  tickStyle=below,
289   ]{(#1, 0); (#2, 0)}
290 }
291 \NewDocumentCommand{\yAxis}{0{-2}0{8}}
292 {
293   \ShowAxis[
294     tickStart=\fp_eval:n {#1+1},
295     tickEnd=\fp_eval:n {#2-0.75},
296     mainStep=1,          subStep=.25,
297     axisRotate=90,        axisColor=black,
298     mainTickColor=black, subTickColor=black,
299     mainTickLength=10pt, subTickLength=5pt,
300     tickLabelShift=0pt,  tickStyle=above,
301     mainTickLabelPosition=left
302   ]{(#1, 0); (#2, 0)}
303 }
304
305

```

```

306 % ==> statistic plot function
307 \cs_new_protected:Npn \ztikz_statistic_plot_cs:nnnn #1#2#3#4
308 {% #1:starts option; #2:draw-keyval; #3:point-keyval; #4:filename
309 \tl_if_empty:nTF {#3}{\draw[#2] plot[#1] file {#4};}
310 {
311 \group_begin:
312 \keys_set:nn { ztikz / point } { #3 }
313 \draw[#2] plot [
314 % stairs options
315 #1,
316 % marker options
317 mark = \str_use:N \l__point_type_str,
318 mark~ size = \dim_use:N \l__point_radius_dim,
319 mark~ options = {
320 rotate = \fp_use:N \l__point_rotate_angle,
321 opacity = \tl_use:N \l__point_opacity_tl,
322 color = \tl_use:N \l__point_color_tl,
323 ball~ color = \tl_use:N \l__point_color_tl,
324 }
325 ] file {#4};
326 \group_end:
327 }
328 }
329 \cs_generate_variant:Nn \ztikz_statistic_plot_cs:nnnn {ennn}
330
331 % stairs plot
332 \seq_new:N \l__statistic_option_tl
333 \NewDocumentCommand\StairsPlot{ 0{plot-left;jump-left}0{color=black}0{}m }
334 {
335 \seq_set_split:Nnn \l__statistic_option_tl { ; }{#1}
336 \str_case:enF {\seq_item:Nn \l__statistic_option_tl{1}}{
337 {plot-left}{\tl_set:Nn \l__tmpa_tl {const~plot~mark~left}}
338 {plot-right}{\tl_set:Nn \l__tmpa_tl {const~plot~mark~right}}
339 {plot-mid}{\tl_set:Nn \l__tmpa_tl {const~plot~mark~mid}}
340 {}{\tl_set:Nn \l__tmpa_tl {}}
341 }{
342 \msg_new:nnn
343 {ztikz}{ztikz-stairs-plot}{current~stairs~plot~type~is:~'#1'~,~ invalide}
344 \msg_error:nn {ztikz}{ztikz-stairs-plot}
345 \str_case:enF {\seq_item:Nn \l__statistic_option_tl{2}}{
346 {jump-left}{\tl_set:Nn \l__tmpb_tl {jump~mark~left}}
347 {jump-right}{\tl_set:Nn \l__tmpb_tl {jump~mark~right}}
348 {jump-mid}{\tl_set:Nn \l__tmpb_tl {jump~mark~mid}}
349 {}{\tl_set:Nn \l__tmpb_tl {}}
350 }{
351 \msg_new:nnn
352 {ztikz}{ztikz-stairs-plot}{current~stairs~jump~type~is:~'#1'~,~ invalide}
353 \msg_error:nn {ztikz}{ztikz-stairs-plot}
354 }
355 \ztikz_statistic_plot_cs:ennn {\l__tmpa_tl,\l__tmpb_tl}{#2}{#3}{#4}
356 }

```

```

356 % stem plot 356
357 \NewDocumentCommand\StemPlot{ 0{x}0{color=black}0{m } 357
358 { 358
359 \str_case:enF {#1}{ 359
360 {x}{\tl_set:Nn \l__tmpa_tl {ycomb}} 360
361 {y}{\tl_set:Nn \l__tmpa_tl {xcomb}} 361
362 {o}{\tl_set:Nn \l__tmpa_tl {polar~ comb}} 362
363 {}{\tl_set:Nn \l__tmpa_tl {ycomb}} 363
364 }{ 364
365 \msg_new:nnn {ztikz}{ztikz-stem-plot}{current~stem~plot~type~is::~'#1'~,~ 365 ✓
invalide}
366 \msg_error:nn {ztikz}{ztikz-stem-plot} 366
367 } 367
368 \ztikz_statistic_plot_cs:ennn {\l__tmpa_tl}{#2}{#3}{#4} 368
369 } 369
370 % bar plot 370
371 \NewDocumentCommand\BarPlot{ 0{ybar}0{color=black}0{m } 371
372 { 372
373 \str_case:enF {#1}{ 373
374 {x}{\tl_set:Nn \l__tmpa_tl {ybar}} 374
375 {y}{\tl_set:Nn \l__tmpa_tl {xbar}} 375
376 {xc}{\tl_set:Nn \l__tmpa_tl {ybar~ interval}} 376
377 {yc}{\tl_set:Nn \l__tmpa_tl {xbar~ interval}} 377
378 {}{\tl_set:Nn \l__tmpa_tl {ybar}} 378
379 }{ 379
380 \msg_new:nnn {ztikz}{ztikz-bar-plot}{current~bar~plot~type~is::~'#1'~,~ 380 ✓
invalide}
381 \msg_error:nn {ztikz}{ztikz-bar-plot} 381
382 } 382
383 \ztikz_statistic_plot_cs:ennn {\l__tmpa_tl}{#2}{#3}{#4} 383
384 } 384

```

8.2.2 gnuplot

```

1  \ProvidesExplFile{ztikz.library.gnuplot.tex}{2024/12/17}{1.0.0}{gnuplot~library~  ✓ 1
   for~ztikz}
2
3
4
5  % ==> init variables
6  \ztool_shell_mkdir:n {ztikz_output/gnuplot_data/}
7  \__ztikz_load_library:n {gnuscript}
8  \int_new:N \g__tikz_env_index_int
9  \int_new:N \g__gnu_data_index_int
10 \int_new:N \g__gnu_plotz_index_int
11 \int_gadd:Nn \g__gnu_plotz_index_int {1}
12 \tl_const:Nn \g__ztikz_gnu_path_tl {ztikz_output/gnuplot_data}
13 \pretocmd{\tikzpicture}{
14   \int_gincr:N \g__tikz_env_index_int
15   \int_gset:Nn \g__gnu_data_index_int {0}
16 }{}{}
17
18 % plot args
19 \tl_new:N \l__pairs_x_domain_tl
20 \tl_new:N \l__pairs_y_domain_tl
21 \ztikz_keys_define:nn { 2dplot }
22 {
23   domain .tl_set:N = \l__ztikz_plot_domain_tl,
24   style .tl_set:N = \l__ztikz_plot_style_tl,
25   marker .tl_set:N = \l__ztikz_plot_marker_tl,
26 }
27 \ztikz_keys_define:nn { 3dplot }
28 {
29   domain .tl_set:N = \l__ztikz_plotz_domain_tl,
30   domain .initial:n = {-5:5; -5:5},
31   palette .tl_set:N = \l__ztikz_plotz_palette_tl,
32   palette .initial:n = {rgbformulae~ 22,13,-31},
33   pm3d .bool_set:N = \l__ztikz_plotz_pm_bool,
34   pm3d .initial:n = {true},
35   pm3d .default:n = {true},
36   width .dim_set:N = \l__ztikz_plotz_width_dim,
37   width .initial:n = {0.75\linewidth},
38 }
39 \cs_new:Npn \__ztikz_pairs_domain_parse:w #1; #2\q_stop
40 {
41   \tl_set:Nn \l__pairs_x_domain_tl {#1}
42   \tl_if_empty:nTF {#2}
43     { \tl_set:Nn \l__pairs_y_domain_tl {*:~} }
44     { \tl_set:Nn \l__pairs_y_domain_tl {#2} }
45 }
46
47
48 % ==> plot precise
49 \bool_new:N \g__plot_precise_restore_bool
50 \bool_new:N \g__contour_precise_restore_bool

```

```

51 \bool_new:N \g__param_precise_restore_bool
52 \bool_new:N \g__polar_precise_restore_bool
53 \NewDocumentCommand\PlotPrecise{ smm }
54 {
55     \tl_if_in:nnF {plot, param, polar, contour}{#2}
56     {
57         \msg_new:nnn {ztikz}{ztikz-plot-type}
58         {Valid-plot~type~are:~'plot',~'param',~'polar'~and~'contour'}
59         \msg_error:nn {ztikz}{ztikz-plot-type}
60     }
61     \IfBooleanF{#1}{ \bool_gset_true:c {g__#2_precise_restore_bool} }
62     \tl_if_eq:nnTF {contour}{#2}{
63         \__ztikz_sed_script:nnn {contour_plot.gp}{2}{set~samples~#3}
64         \__ztikz_sed_script:nnn {contour_plot.gp}{3}{set~isosamples~#3,#3}
65     }{
66         \__ztikz_sed_script:nnn {#2_plot.gp}{3}{set~samples~#3}
67     }
68 }
69 % data plot, precise, style
70 \cs_new_protected:Npn \__ztikz_gnu_data_plot:nnn #1#2#3
71 {
72     \ztool_shell_escape:e {gnuplot~ \g__ztikz_scripts_path_tl/#1_plot.gp}
73     \int_gadd:Nn \g__gnu_data_index_int {1}
74     \tl_set:Ne \l__gnu_data_new_name_tl
75     {
76         gnu_data \int_use:N \g__tikz_env_index_int
77         _\int_use:N \g__gnu_data_index_int.table
78     }
79     \tl_set:Ne \l__gnu_data_full_path_tl
80     { \g__ztikz_gnu_path_tl/\l__gnu_data_new_name_tl }
81     \ztool_shell_mv:ee
82     { \g__ztikz_gnu_path_tl/gnu_data.table }
83     { \l__gnu_data_full_path_tl }
84     \tl_if_empty:nTF {#3}
85     {
86         \draw[#2] plot[smooth] file { \l__gnu_data_full_path_tl };
87     }{
88         \group_begin:
89         \ztikz_keys_set:nn { point } { #3 }
90         \draw[#2] plot [
91             mark = \str_use:N \l__point_type_str,
92             mark~ size = \dim_use:N \l__point_radius_dim,
93             mark~ options = {
94                 rotate = \fp_use:N \l__point_rotate_angle,
95                 opacity = \tl_use:N \l__point_opacity_tl,
96                 color = \tl_use:N \l__point_color_tl,
97                 ball~ color = \tl_use:N \l__point_color_tl,
98             }
99             ] file { \l__gnu_data_full_path_tl };
100         \group_end:
101     }
102     \bool_if:cT {g__#1_precise_restore_bool}

```

```

103 { \PlotPrecise{#1}{100} } 103
104 } 104
105 \cs_generate_variant:Nn \__ztikz_gnu_data_plot:nnn {nee} 105
106 106
107 107
108 % ==> simple 2d function plot 108
109 \NewDocumentCommand\Plot{ 0{domain=-5:5, style={color=black}, marker=}m } 109
110 { 110
111 \group_begin: 111
112 \ztikz_keys_set:nn { 2dplot } { #1 } 112
113 \__ztikz_sed_script:nne {plot_plot.gp}{8}{set~xr~[\l__ztikz_plot_domain_tl]} 113
114 \__ztikz_sed_script:nnn {plot_plot.gp}{7}{f(x)~==#2} 114
115 \__ztikz_gnu_data_plot:nee 115 ✓
116 {plot}{\l__ztikz_plot_style_tl}{\l__ztikz_plot_marker_tl}
117 \group_end: 116
118 } 117
119 \NewDocumentCommand\ContourPlot{ 0{domain={-5:5;}, style={color=black}, marker=}m } 118
120 { 119
121 \group_begin: 120
122 \ztikz_keys_set:nn { 2dplot } { #1 } 121
123 \exp_last_unbraced:Nf \__ztikz_pairs_domain_parse:w 122 ✓
124 \l__ztikz_plot_domain_tl\q_stop
125 \__ztikz_sed_script:nne {contour_plot.gp}{11}{set~xr~[\l__pairs_x_domain_tl]} 123
126 \__ztikz_sed_script:nne {contour_plot.gp}{12}{set~yr~[\l__pairs_y_domain_tl]} 124
127 \__ztikz_sed_script:nne {contour_plot.gp}{14}{f(x,~y)~==#2} 125
128 \__ztikz_gnu_data_plot:nee 126 ✓
129 {contour}{\l__ztikz_plot_style_tl}{\l__ztikz_plot_marker_tl}
130 \group_end: 127
131 } 128
132 \NewDocumentCommand\ParamPlot{ 0{domain=0:2*pi, style=black, marker=}m } 129
133 { 130
134 \group_begin: 131
135 \ztikz_keys_set:nn { 2dplot } { #1 } 132
136 \__ztikz_sed_script:nne 133 ✓
137 {param_plot.gp}{8}{set~trange~[\l__ztikz_plot_domain_tl]}
138 \__ztikz_sed_script:nnn {param_plot.gp}{9}{plot~#2} 134
139 \__ztikz_gnu_data_plot:nee 135 ✓
140 {param}{\l__ztikz_plot_style_tl}{\l__ztikz_plot_marker_tl}
141 \group_end: 136
142 } 137
143 \NewDocumentCommand\PolarPlot{ 0{domain=0:2*pi, style=black, marker=}m } 138
144 { 139
145 \group_begin: 140
146 \ztikz_keys_set:nn { 2dplot } { #1 } 141
147 \__ztikz_sed_script:nne 142 ✓
148 {polar_plot.gp}{8}{set~trange~[\l__ztikz_plot_domain_tl]}
149 \__ztikz_sed_script:nnn {polar_plot.gp}{9}{plot~#2} 143
150 \__ztikz_gnu_data_plot:nee 144 ✓
151 {polar}{\l__ztikz_plot_style_tl}{\l__ztikz_plot_marker_tl}
152 \group_end: 145
153 } 146
154 \NewDocumentCommand\Plotz{ 0{m } 147

```


148	{	148
149	\group_begin:	149
150	\ztikz_keys_set:nn { 3dplot } { #1 }	150
151	\bool_if:NTF \l__ztikz_plotz_pm_bool {\def\plotz@pm{with~pm3d}}{\def\plotz@pm{}}	151
152	\exp_last_unbraced:Nf __ztikz_pairs_domain_parse:w	152 ✓
	\l__ztikz_plotz_domain_tl\q_stop	
153	__ztikz_sed_script:nne {3d_plot.gp}{18}{set~palette~\l__ztikz_plotz_palette_tl}	153
154	__ztikz_sed_script:nne {3d_plot.gp}{23}{set~xr~[\l__pairs_x_domain_tl]}	154
155	__ztikz_sed_script:nne {3d_plot.gp}{24}{set~yr~[\l__pairs_y_domain_tl]}	155
156	__ztikz_sed_script:nne {3d_plot.gp}{25}{splot~#2~\plotz@pm}	156
157	\ztool_shell_escape:e {gnuplot~ ./ztikz_output/scripts/3d_plot.gp}	157
158	\tl_set:Ne \l_tmpa_tl {./ztikz_output/gnuplot_data/plot_3d_\int_use:N	158 ✓
	\g__gnu_plotz_index_int.pdf}	
159	\ztool_shell_mv:ne {./ztikz_output/gnuplot_data/plot_3d.pdf}{\l_tmpa_tl}	159
160	\includegraphics[width=\dim_use:N \l__ztikz_plotz_width_dim]{\l_tmpa_tl}	160
161	\int_gadd:Nn \g__gnu_plotz_index_int {1}	161
162	\group_end:	162
163	}	163
164		164
165		165
166	% ==> users' interface	166
167	\NewDocumentCommand{\currentTikzIndex}{}{	167
168	{	168
169	\int_use:N \g__tikz_env_index_int	169
170	}	170
171	\def\gnudata#1	171
172	{	172
173	\tl_use:N \g__ztikz_gnu_path_tl/gnu_data_	173
174	\int_use:N \g__tikz_env_index_int _#1.table	174
175	}	175

8.2.3 cache

```

1  \ProvidesExplFile{ztikz.library.cache.tex}{2025/05/22}{1.0.0}{cache~library~for~  ✓ 1
   ztikz}
2
3
4
5  % ==> init cache 5
6  \clist_if_in:NnT \g__ztikz_library_loaded_clist {basic} 6
7  { 7
8    \ztool_shell_mkdir:n {ztikz_output/tikz_data/} 8
9    \usetikzlibrary{external} 9
10   \tikzexternalize[prefix=ztikz_output/tikz_data/] 10
11   } 11
12  \ztool_shell_escape:n {touch~ ztikz_output/ztikz.hash} 12
13
14
15  % ==> cache function 15
16  \ior_new:N \g__ztikz_file_ior 16
17  \tl_new:N \l__ztikz_current_hash_tl 17
18  \tl_new:N \g__ztikz_current_hash_tl 18
19  \seq_new:N \g__ztikz_file_hash_seq 19
20  \cs_generate_variant:Nn \ztikz_file_read_lines:n { e } 20
21  \bool_new:N \g__ztikz_hash_nochg_run_bool 21
22  \bool_gset_false:N \g__ztikz_hash_nochg_run_bool 22
23  \prg_set_conditional:Npnn \ztikz_hash_if_change:nn #1#2 { p, T, F, TF } 23
24   {% #1: true/false; #2: True for file, False for hash str 24
25   \bool_if:NTF {#1} 25
26     { \file_get_md5five_hash:nN {#2} \l__ztikz_current_hash_tl } 26
27     { \tl_set:Nn \l__ztikz_current_hash_tl {#2} } 27
28   \tl_set_rescan:Nne \l__ztikz_current_hash_tl 28
29     { \cctab_select:N \c_initex_cctab } 29
30     { \l__ztikz_current_hash_tl } 30
31   \tl_gset_eq:NN \g__ztikz_current_hash_tl \l__ztikz_current_hash_tl 31
32   \ztool_gread_file_as_seq:nnc {\c_false_bool} 32
33     { ztikz_output/ztikz.hash } 33
34     { g__ztikz_file_hash_seq } 34
35   \seq_gremove_duplicates:N \g__ztikz_file_hash_seq 35
36   \ztikz_term_info:e 36
37   { 37
38     \iow_newline: CURRENT~FILE's~HASH:\l__ztikz_current_hash_tl 38
39   } 39
40   \bool_if:NTF \g__ztikz_hash_nochg_run_bool 40
41   { 41
42     \ztikz_term_info:n {FORCE~TO~RUN~AGAIN~...} 42
43     \bool_gset_false:N \g__ztikz_hash_nochg_run_bool 43
44     \prg_return_true: 44
45   }{ 45
46     \bool_if:NTF \g__ztikz_hashchg_norun_bool 46
47     { 47
48       \ztikz_term_info:n {FORCE~TO~SKIP~RUN~AGAIN~...} 48
49       \bool_gset_false:N \g__ztikz_hashchg_norun_bool 49
50       \prg_return_false: 50

```

51	}{	51
52	\seq_if_in:NVTF \g__ztikz_file_hash_seq \l__ztikz_current_hash_tl	52
53	{	53
54	\ztikz_term_info:n {CURRENT~HASH~ALREADY~EXISTS}	54
55	\prg_return_false:	55
56	}{	56
57	\ztikz_term_info:n {CURRENT~HASH~IS~UNIQUE:RECORDING...}	57
58	\ztool_append_to_file:nn	58 ✓
	{ztikz_output/ztikz.hash}{\l__ztikz_current_hash_tl}	
59	\prg_return_true:	59
60	}	60
61	}	61
62	}	62
63	}	63
64	\prg_generate_conditional_variant:Nnn \ztikz_hash_if_change:nn { ne } { p, T, F, TF }	64 ✓
65		65
66		66
67	<i>% ==> clear cache hash</i>	67
68	\cs_new_protected:Npn \ztikz_clear_hash:	68
69	{	69
70	\iow_open:Nn \g__ztikz_file_ior {./ztikz_output/ztikz.hash}	70
71	\ior_close:N \g__ztikz_file_ior	71
72	}	72
73	\NewDocumentCommand{\ztikzHashClean}{s}{	73
74	\ztikz_clear_hash:	74
75	\ztikz_term_info:n {CLEAN~ALL~ztikz~HASH~SUCCESSFULLY...}	75
76	}	76
77	\NewDocumentCommand{\ztikzHashCurrent}{s+0{,}}{	77
78	\IfBooleanTF{#1}	78
79	{ \tl_use:N \l__ztikz_current_hash_tl }	79
80	{ \seq_use:Nn \g__ztikz_file_hash_seq {#2} }	80
81	}	81
82		82
83		83
84	<i>% ==> override the cache mechanism</i>	84
85	\NewDocumentCommand{\ztikzForceToSkip}{}{	85
86	{	86
87	\bool_gset_false:N \g__ztikz_hash_nochg_run_bool	87
88	\bool_gset_true:N \g__ztikz_hashchg_norun_bool	88
89	}	89
90	\NewDocumentCommand{\ztikzForceToRun}{}{	90
91	{	91
92	\bool_gset_false:N \g__ztikz_hashchg_norun_bool	92
93	\bool_gset_true:N \g__ztikz_hash_nochg_run_bool	93
94	}	94

8.2.4 python

```

1  \ProvidesExplFile{ztikz.library.python.tex}{2025/05/22}{1.0.0}{python~library~fo  ✓ 1
   r~ztikz}
2
3
4
5  % ==> writing scripts
6  \RequirePackage{xsimverb}
7  \__ztikz_load_library:n {pyscript}
8  \ztool_shell_mkdir:n {ztikz_output/python_data/}
9  \tl_const:Nn \g__ztikz_python_path_tl {ztikz_output/python_data}
10 \ior_new:N \g__file_read_ior
11 \tl_new:N \g__file_content_tl
12
13
14 % ==> core functions
15 \cs_new_protected:Npn \zlatex_Readlines_cs:nn #1#2
16 {
17   \ior_open:Nn \g__file_read_ior {#2}
18   \str_case:nnF {#1}{
19     {raw}{
20       \ior_get:NN \g__file_read_ior \g__file_content_tl
21     }
22     {str}{
23       \ior_str_get:NN \g__file_read_ior \g__file_content_tl
24     }
25   }{}
26   \tl_use:N \g__file_content_tl
27 }
28 \cs_generate_variant:Nn \zlatex_Readlines_cs:nn {ee}
29 \cs_generate_variant:Nn \xsim_file_write_start:nn {ne}
30
31
32 % ==> users' interface
33 % python-matplotlib
34 \NewDocumentEnvironment{pyfig}{ m }
35 {
36   \xsim_file_write_start:ne {\c_true_bool}{\g__ztikz_python_path_tl/t@mp.py}
37 }{
38   \xsim_file_write_stop:
39   \ztikz_hash_if_change:neTF {\c_true_bool}{\g__ztikz_python_path_tl/t@mp.py}
40   {
41     \__ztikz_addto_script:nn {\g__ztikz_python_path_tl/t@mp.py}
42     { plt.savefig('#1') }
43     \ztool_shell_escape:e {python~\g__ztikz_python_path_tl/t@mp.py}
44     \ztool_shell_mv:ee
45     { \g__ztikz_python_path_tl/t@mp.py }
46     { \g__ztikz_python_path_tl/pyfig_\l__ztikz_current_hash_tl.py }
47     \ztool_shell_mv:ee
48     { #1 }
49     { \g__ztikz_python_path_tl/\l__ztikz_current_hash_tl _#1 }

```

```

50      \ztikz_term_info:e {writing~source~to~file:'\g__ztikz_python_path_tl/pyf  ✓ 50
      ig_\l__ztikz_current_hash_tl.py'}
51    }{ 51
52      \ztikz_term_info:e {skip~recompile~by~python,~using~the~cache~picture:'(  ✓ 52
      \l__ztikz_current_hash_tl)_#1'}
53    } 53
54    \xdef\pyfigOutputFile{ \g__ztikz_python_path_tl/\l__ztikz_current_hash_tl _#1 } 54
55  } 55
56  % inline python command 56
57  \NewDocumentCommand\py{0{raw}m} 57
58  { 58
59    \__ztikz_sed_script:nne {python_script.py}{6}{Float_res~~~\tl_to_str:n {#2}} 59
60    \ztikz_term_info:e {using~python~float~module~calculating...} 60
61    \ztool_shell_escape:e {python~ \g__ztikz_scripts_path_tl/python_script.py} 61
62    \zlatex_Readlines_cs:ee {#1}{\g__ztikz_python_path_tl/PyFloat.out} 62
63    % ---> cause bug that can't write ToC to file 63
64    % \iow_close:N \g__file_read_iior leads to bug ?? 64
65  } 65
66  % python-sympy 66
67  \NewDocumentCommand\sympy{m} 67
68  { 68
69    \__ztikz_sed_script:nne {sympy_script.py}{8}{F_res~~~\tl_to_str:n {#1}} 69
70    \ztikz_hash_if_change:neTF 70 ✓
    {\c_true_bool}{\g__ztikz_scripts_path_tl/sympy_script.py}
71    { 71
72      \ztool_shell_escape:e {python~ \g__ztikz_scripts_path_tl/sympy_script.py} 72
73      \ztool_shell_mv:ee 73
74      {\g__ztikz_python_path_tl/sympy.out} 74
75      {\g__ztikz_python_path_tl/sympy_\l__ztikz_current_hash_tl.out} 75
76      \ztikz_term_info:e {using~python~sympy~calculating~question~...} 76
77      \exp_args:Ne 77 ✓
      \input{\g__ztikz_python_path_tl/sympy_\l__ztikz_current_hash_tl.out}
78    }{ 78
79      \exp_args:Ne 79 ✓
      \input{\g__ztikz_python_path_tl/sympy_\l__ztikz_current_hash_tl.out}
80      \ztikz_term_info:e {skip~recompile,~using~the~cache~sympy~result::~\l__zt  ✓ 80
      ikz_current_hash_tl}
81    } 81
82  } 82
83  % python-code-env 83
84  \NewDocumentEnvironment{pycode}{m } 84
85    {% #1:true to input result; #2:output file name (with ext) 85
86      \xsim_file_write_start:ne {\c_true_bool}{\g__ztikz_python_path_tl/t@mp.py} 86
87    }{ 87
88      \xsim_file_write_stop: 88
89      \ztikz_hash_if_change:neTF {\c_true_bool}{\g__ztikz_python_path_tl/t@mp.py} 89
90      { 90
91        \ztool_shell_escape:e {python~\g__ztikz_python_path_tl/t@mp.py} 91
92        \ztool_shell_mv:ee 92
93        { \g__ztikz_python_path_tl/t@mp.py } 93
94        { \g__ztikz_python_path_tl/pycode_\l__ztikz_current_hash_tl.py } 94
95        \ztool_shell_mv:ee 95

```

96	{ #1 }	96
97	{ \g__ztikz_python_path_tl/\l__ztikz_current_hash_tl _#1 }	97
98	\ztikz_term_info:e {writing~source~to~file:'\g__ztikz_python_path_tl/pyc ode_\l__ztikz_current_hash_tl.py'}	98 ✓
99	}{	99
100	\ztikz_term_info:e {skip~recompile~by~python,use~the~cache~result:'(\l__ ztikz_current_hash_tl)_#1'}	100 ✓
101	}	101
102	\xdef\pycodeOutputFile{\g__ztikz_python_path_tl/\l__ztikz_current_hash_tl _#1}	102
103	}	103

8.2.5 wolfram

```

1  \ProvidesExplFile{ztikz.library.wolfram.tex}{2025/05/22}{1.0.0}{wolfram-library-  ✓ 1
   for~ztikz}
2
3
4
5  % ==> init variables 5
6  \RequirePackage{xsimverb} 6
7  \ztool_shell_mkdir:n {ztikz_output/mma_data/} 7
8  \tl_const:Nn \g__ztikz_wolfram_path_tl {ztikz_output/mma_data} 8
9  \tl_new:N \l__ztikz_wolfram_tmp_arg_tl 9
10 \tl_new:N \l__ztikz_wolfram_tmp_res_tl 10
11 \seq_new:N \l__ztikz_wolfram_tmp_res_seq 11
12 \ior_new:N \g__ztikz_wolfram_ior 12
13 \iow_new:N \g__ztikz_wolfram_iow 13
14 \cs_generate_variant:Nn \xsim_file_write_start:nn {ne} 14
15
16 % writing wolfram script to file 16
17 \file_if_exist:nF { \g__ztikz_scripts_path_tl/wolfram_script.wls } 17
18 { 18
19     \iow_open:Ne \g__ztikz_wolfram_iow {  ✓ 19
        \g__ztikz_scripts_path_tl/wolfram_script.wls }
20     \iow_now:Ne \g__ztikz_wolfram_iow 20
21     { 21
22         TeXResult=ToString[TeXForm[1+1]]; \iow_newline: 22
23         Export["ztikz_output/mma_data/hash.txt",TeXResult] 23
24     } 24
25     \iow_close:N \g__ztikz_wolfram_iow 25
26 } 26
27
28
29 % ==> core function 29
30 \msg_set:nnn {ztikz}{wolfram-arg_empty} 30
31 { wolfram~library~error:calcualting~argument~is~empty. } 31
32 \cs_new:Npn \__ztikz_wolfram_tmp_file_handle:n #1 32
33 { 33
34     \edef\@wolfram@tmp@file{\g__ztikz_wolfram_path_tl/t@mp} 34
35     \tl_if_eq:enT {#1}{TeXResult=ToString[TeXForm[]];} 35
36     { 36
37         \msg_error:nn {ztikz}{wolfram-arg_empty} 37
38     } 38
39     \__ztikz_addto_script:en { \@wolfram@tmp@file.wls }{ #1 } 39
40     \file_get_mdfive_hash:nN { \@wolfram@tmp@file.wls }\l__ztikz_current_hash_tl 40
41     \xdef\wolfram@tmp@file{\g__ztikz_wolfram_path_tl/\l__ztikz_current_hash_tl} 41
42     \ztool_shell_mv:ee {\@wolfram@tmp@file.wls}{\wolfram@tmp@file.wls} 42
43 } 43
44 \cs_new_protected:Npn \__ztikz_wolfram_excute:nnn #1#2#3 44
45 {% #1:contents(empty->not add); #2:extension; #3: output object 45
46     \__ztikz_wolfram_tmp_file_handle:n {#1} 46
47     \__ztikz_addto_script:ee  ✓ 47
        {\wolfram@tmp@file.wls}{Export["\wolfram@tmp@file.#2", #3]}

```

```

48 \ztikz_hash_if_change:neTF { \c_false_bool }{ \l__ztikz_current_hash_tl }
49 {
50 \edef\wolfram@cmd
51 {
52 \g__ztikz_wolfram_engine_tl\space
53 \bool_if:NT \g__ztikz_wolfram_cloud_bool {-cloud\space}
54 -script\space
55 \wolfram@tmp@file.wls
56 }
57 \ztikz_term_info:e
58 {
59 \g__ztikz_wolfram_engine_tl\space~is~running~on:'\wolfram@tmp@file.wls'
60 ... \iow_newline:
61 }
62 \ztool_shell_escape:e { \wolfram@cmd }
63 }{
64 \ztikz_term_info:e {Use~cache~result:'\wolfram@tmp@file.#2'\iow_newline:}
65 }
66 \xdef\wolframOutputFile{\wolfram@tmp@file.#2}
67 }
68 \cs_generate_variant:Nn \__ztikz_wolfram_excute:nnn { e, o }
69
70
71 % ==> user interface
72 % --> load result
73 \NewDocumentCommand\wolframResult{so}
74 {
75 \ior_open:Ne \g__ztikz_wolfram_ior {\wolfram@tmp@file.txt}
76 \ior_get:NN \g__ztikz_wolfram_ior \l__ztikz_wolfram_tmp_res_tl
77 \seq_set_split:NnV \l__ztikz_wolfram_tmp_res_seq {@}
78 \l__ztikz_wolfram_tmp_res_tl
79 \IfBooleanTF{#1}
80 {% must be integer expression, or it will raise bug.
81 \seq_item:Ne \l__ztikz_wolfram_tmp_res_seq
82 {\IfValueTF {#2}{\fp_eval:n {#2}}{1}}
83 }
84 { \seq_use:Nn \l__ztikz_wolfram_tmp_res_seq {\IfValueTF {#2}{#2}{,}}
85 \ior_close:N \g__ztikz_wolfram_ior
86 }
87
88 % --> wolfram graphicx
89 \NewDocumentEnvironment{wolframGraphics}{O{}}
90 {
91 \gdef\graphics@dimen{#1}
92 \xsim_file_write_start:ne {\c_false_bool}{\g__ztikz_wolfram_path_tl/t@mp.wls}
93 }{
94 \xsim_file_write_stop:
95 \__ztikz_wolfram_excute:nnn {}{pdf}{FIGURE}
96 \tl_if_empty:eF {\graphics@dimen}
97 {
98 \exp_after:wN \includegraphics \exp_after:wN
99 [\graphics@dimen]{\wolfram@tmp@file.pdf}

```



```

99     }
100 }
101
102 % --> wolfram simple code
103 \NewDocumentCommand\wolfram{sm}
104 {
105     \__ztikz_wolfram_excute:enn
106     {
107         \IfBooleanTF{#1}
108         { TeXResult = ToString[#2]; }
109         { TeXResult = ToString[TeXForm[#2]]; }
110         }{txt}{TeXResult}
111     }
112
113 % --> wolfram tex code(expandable token replace)
114 \group_begin:
115     \char_set_catcode_escape:n { 36 }
116     \char_set_catcode_letter:n { 92 }
117     $cs_gset:Nn $__double_backslash:n
118     { $tl_if_eq:NNTF #1\ {\}\{#1} }
119     $gdef$wolframTex{
120         $char_set_catcode_letter:n { 92 }
121         $wolframTex@getarg
122     }
123     $gdef$wolframTex@getarg#1{
124         $tl_set:Ne $l_tmpa_tl
125         {
126             $tl_map_function:nN {#1}
127             $__double_backslash:n
128         }
129         $__ztikz_wolfram_excute:onn
130         {TeXResult = TeXForm[ToExpression["$l_tmpa_tl", TeXForm]]}
131         {txt}{TeXResult}
132         $char_set_catcode_escape:n { 92 }
133     }
134     $char_set_catcode_escape:n { 92 }
135     $char_set_catcode_letter:n { 36 }
136 \group_end:
137
138 % --> wolfram table (extended the interface of 'latexalpha2')
139 \flag_new:N \l_wolfram_table_column_int
140 \cs_set:Npn \__table_item_handle:n #1
141     {% the inner '\exp_not:N' prevent expansion from 'tabularray'.
142     \exp_not:n {
143         \exp_not:N \__wolfram_table_cell_cmd:n {#1}
144     },
145 }
146 \cs_set:Npn \__table_row_handle:n #1 % #1='1, 2, 3'
147 {
148     \clist_use:en
149     {
150         \clist_map_function:oN #1

```

```

151      \table_item_handle:n      151
152      }{ & } \\                152
153      }                        153
154 \cs_generate_variant:Nn \clist_use:nn { en }      154
155 \cs_generate_variant:Nn \clist_map_function:nN { oN } 155
156 \cs_new:Npn \__part_table_from_file:nN #1#2      156
157   {% #1:file; #2:data var      157
158     \ztool_gread_file_as_seq:neN {\c_true_bool}    158
159     { #1 } \l_tmpa_seq      159
160     \tl_set:Ne #2      160
161     {      161
162       \seq_map_function:NN \l_tmpa_seq      162
163       \__table_row_handle:n      163
164     }      164
165   }      165
166 \tl_new:N \l_part_table_data_tl      166
167 \tl_new:N \l_full_table_data_tl      167
168 \cs_set:Npn \__full_table_from_file:nn #1#2      168
169   {% #1:file; #2:table header      169
170     \__part_table_from_file:nN      170
171     { #1 } \l_part_table_data_tl      171
172     \tl_set:Ne \l_full_table_data_tl      172
173     {      173
174       \tl_if_empty:eF {#2}{#2 \\}      174
175       \l_part_table_data_tl      175
176     }      176
177     \tl_set:Ne \l_part_table_data_tl      177
178     { \l_part_table_data_tl }      178
179   }      179
180 \cs_generate_variant:Nn \__full_table_from_file:nn { VV } 180
181 \cs_set:Npn \__typeset_table:nnn #1#2#3      181
182   {% #1:table format; #2:table header; #3:table part data 182
183     \begin{tabular}{#1}      183
184     \hline      184
185     \bool_if:NT \l_wolfram_table_hdbt_rule_bool      185
186       { #2\\ \hline }      186
187     #3      187
188     \hline      188
189     \end{tabular}      189
190   }      190
191 \cs_generate_variant:Nn \__typeset_table:nnn { VVV }      191
192 \ztikz_keys_define:nn { wolfram / table }      192
193   {      193
194     format .tl_set:N = \l_ztikz_wolfram_table_format_tl,      194
195     format .initial:n = { *{12}{1} },      195
196     header .tl_set:N = \l_ztikz_wolfram_table_header_tl,      196
197     header .initial:n = { },      197
198     hdbt-rule .bool_set:N = \l_wolfram_table_hdbt_rule_bool,      198
199     hdbt-rule .initial:n = { false },      199
200     hdbt-rule .default:n = { true },      200
201     cell-cmd .cs_gset:Np = \__wolfram_table_cell_cmd:n #1,      201
202     cell-cmd .initial:n = { #1 },      202

```

203	}	203
204	\NewDocumentCommand{\wolframTable}{s0{m}}	204
205	{% #1:if typeset; #2:key-value; #3:code	205
206	\group_begin:	206
207	\ztikz_keys_set:nn { wolfram/table } {#2}	207
208	__ztikz_wolfram_excute:enn	208
209	{ TeXResult = #3; }	209
210	{ txt }{ TeXResult }	210
211	__full_table_from_file:VV \wolframOuputFile	211
212	\l__ztikz_wolfram_table_header_tl	212
213	\IfBooleanT{#1}	213
214	{	214
215	__typeset_table:VVV	215
216	\l__ztikz_wolfram_table_format_tl	216
217	\l__ztikz_wolfram_table_header_tl	217
218	\l_part_table_data_tl	218
219	}	219
220	\exp_args:NNo \gdef\wolframTablePData{ \l_part_table_data_tl }	220
221	\exp_args:NNo \gdef\wolframTableFData{ \l_full_table_data_tl }	221
222	\group_end:	222
223	}	223
224		224
225	% --> equation solve	225
226	\ztikz_keys_define:nn { wolfram/solve }	226
227	{	227
228	var .tl_set:N = \l__ztikz_wolfram_var_tl,	228
229	var .initial:n = {},	229
230	domain .tl_set:N = \l__ztikz_wolfram_domain_tl,	230
231	domain .initial:n = {},	231
232	}	232
233	\NewDocumentCommand\wolframSolve{som}	233
234	{	234
235	\group_begin:	235
236	\IfValueT {#2} { \ztikz_keys_set:nn { wolfram/solve } {#2} }	236
237	\tl_if_empty:VF \l__ztikz_wolfram_domain_tl	237
238	{ \tl_set:Ne \l__ztikz_wolfram_tmp_arg_tl {,\l__ztikz_wolfram_domain_tl} }	238
239	__ztikz_wolfram_excute:enn	239
240	{	240
241	\IfBooleanTF {#1}{	241
242	TeXResult = Row[Solve[#3]//Flatten, "@"]	242
243	/.{Rule -> Equal}//TeXForm//ToString;	243
244	}{	244
245	TeXResult = Row[245
246	Solve[#3, {\l__ztikz_wolfram_var_tl	246
247	\l__ztikz_wolfram_tmp_arg_tl]//Flatten,	247
248	"@"	248
249]/.{Rule -> Equal}//TeXForm//ToString;	249
250	}{txt}{TeXResult}	250
251	\group_end:	251
252	}	252
253		253

```

254 % --> differential equation solve
255 \ztikz_keys_define:nn { wolfram/dsolve }
256 {
257     depend .tl_set:N = \l__ztikz_wolfram_de_var_tl,
258     depend .initial:n = { y[x] },
259     independ .tl_set:N = \l__ztikz_wolfram_in_var_tl,
260     independ .initial:n = { x },
261 }
262 \NewDocumentCommand\wolframDSolve{som}
263 {
264     \group_begin:
265     \IfValueT {#2} { \ztikz_keys_set:nn { wolfram/dsolve } {#2} }
266     \tl_if_empty:VF \l__ztikz_wolfram_in_var_tl
267     { \tl_set:N \l__ztikz_wolfram_in_var_tl {,\l__ztikz_wolfram_in_var_tl} }
268     \__ztikz_wolfram_excute:enn
269     {
270         \IfBooleanTF {#1}{
271             TeXResult = Row[DSolve[#3]//Flatten, ","]
272             /.{Rule -> Equal}//TeXForm//ToString;
273         }{
274             TeXResult = Row[
275                 DSolve[{#3},
276                     {\l__ztikz_wolfram_de_var_tl}\l__ztikz_wolfram_in_var_tl]//Flatten,
277                     "@"]
278             /.{Rule -> Equal}//TeXForm//ToString;
279         }{txt}{TeXResult}
280     \group_end:
281 }
282
283 % --> any wolfram/mathics code
284 \NewDocumentEnvironment{wolframAny}{m}
285 {
286     \cs_seq_eq:NN \NewLine \iow_newline:
287     \cs_set_eq:NN \PrefixPath \g__ztikz_wolfram_path_tl
288     \__ztikz_wolfram_excute:nne
289     {}{}{}
290 }{
291     \cs_seq_eq:NN \NewLine \scan_stop:
292     \cs_set_eq:NN \PrefixPath \scan_stop:
293 }
294
295 % check if integer
296 \prg_new_protected_conditional:Npnn \__ztikz_if_int:n #1 { T, F, TF }
297 {
298     \regex_match:nnTF { ^[\+|-]?[\d]+$ } {#1}
299     { \prg_return_true: }
300     { \prg_return_false: }
301 }
302 \prg_generate_conditional_variant:Nnn \__ztikz_if_int:n { e } { T, F, TF }

```

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

B	
<code>\BarPlot</code>	<i>11, 15</i>
<code>\begin</code>	<i>24, 25, 31, 33</i>
C	
<code>\ContourPlot</code>	<i>17, 18, 20</i>
<code>\CurrentFp</code>	<i>12</i>
<code>\currentTikzIndex</code>	<i>20</i>
D	
<code>\draw</code>	<i>11, 17</i>
draw commands:	
<code>\draw_begin:</code>	<i>33</i>
<code>\draw_end:</code>	<i>33</i>
<code>\draw_path_scope_begin:</code>	<i>33, 34</i>
<code>\draw_path_scope_end:</code>	<i>33, 34</i>
E	
<code>\end</code>	<i>24, 25, 31, 33</i>
exp commands:	
<code>\exp_not:N</code>	<i>65</i>
F	
file internal commands:	
<code>\g_file_read_ior</code>	<i>61</i>
<code>\filldraw</code>	<i>4</i>
G	
<code>\gnudata</code>	<i>20</i>
H	
<code>\hline</code>	<i>28</i>
I	
<code>\includegraphics</code>	<i>24, 25, 31</i>
<code>\input</code>	<i>24, 25</i>
iow commands:	
<code>\iow_close:N</code>	<i>61</i>
M	
<code>\makeindex</code>	<i>4</i>
<code>\midrule</code>	<i>28</i>
N	
<code>\node</code>	<i>11</i>
P	
<code>\ParamPlot</code>	<i>17, 18</i>
<code>\Plot</code>	<i>17–19, 21</i>
<code>\PlotPrecise</code>	<i>17, 18</i>
<code>\Plotz</code>	<i>19</i>
<code>\PolarPlot</code>	<i>17, 18</i>
<code>\Polygon</code>	<i>13</i>
<code>\printindex</code>	<i>4</i>
<code>\py</code>	<i>23</i>
<code>pycode</code>	<i>25</i>
<code>\pycodeOutputFile</code>	<i>25</i>
<code>pyfig</code>	<i>24</i>
<code>\pyfigOutputFile</code>	<i>24</i>
S	
<code>\ShowAxis</code>	<i>12, 13</i>
<code>\ShowGrid</code>	<i>13</i>
<code>\ShowIntersection</code>	<i>12</i>
<code>\ShowPoint</code>	<i>4, 11, 12</i>
<code>\StairsPlot</code>	<i>14</i>
<code>\StemPlot</code>	<i>14</i>
str commands:	
<code>\str_mdfive_hash:n</code>	<i>40</i>
<code>\svec</code>	<i>34</i>
<code>\sympy</code>	<i>22, 24</i>
T	
<code>ztikz/2dplot/domain</code>	<i>17</i>
<code>ztikz/2dplot/marker</code>	<i>17</i>
<code>ztikz/2dplot/style</code>	<i>17</i>
<code>ztikz/3dplot/domain</code>	<i>19</i>
<code>ztikz/3dplot/palette</code>	<i>19</i>
<code>ztikz/3dplot/pm3d</code>	<i>19</i>
<code>ztikz/3dplot/width</code>	<i>19</i>
<code>ztikz/axis/axisColor</code>	<i>12</i>
<code>ztikz/axis/axisRotate</code>	<i>12</i>
<code>ztikz/axis/mainStep</code>	<i>12</i>
<code>ztikz/axis/mainTickColor</code>	<i>12</i>
<code>ztikz/axis/mainTickLabel</code>	<i>12</i>
<code>ztikz/axis/mainTickLabelColor</code>	<i>12</i>
<code>ztikz/axis/mainTickLabelPosition</code>	<i>12</i>
<code>ztikz/axis/mainTickLength</code>	<i>12</i>
<code>ztikz/axis/subStep</code>	<i>12</i>
<code>ztikz/axis/subTickColor</code>	<i>12</i>
<code>ztikz/axis/subTickLength</code>	<i>12</i>
<code>ztikz/axis/tickEnd</code>	<i>12</i>
<code>ztikz/axis/tickLabelShift</code>	<i>12</i>
<code>ztikz/axis/tickStart</code>	<i>12</i>
<code>ztikz/axis/tickStyle</code>	<i>12</i>
<code>ztikz/point/color</code>	<i>11</i>
<code>ztikz/point/opacity</code>	<i>11</i>

ztikz/point/radius	11	X	
ztikz/point/rotate	11	\xAxis	13
ztikz/point/type	11	xsim commands:	
ztikz/polygon/edgeColor	13	\xsim_file_write_start:nn	40
ztikz/polygon/fillColor	13	Y	
ztikz/polygon/fillOpacity	13	\yAxis	13
ztikz/polygon/marker	13	\yvec	34
ztikz/polygon/radius	13	Z	
ztikz/polygon/rotate	13	\zbg	34
ztikz/polygon/shift	13	\zcapbutt	34
ztikz/wolfram/dsolve/depend	30	\zcaprect	34
ztikz/wolfram/dsolve/independ	30	\zcaproun	34
ztikz/wolfram/solve/domain	29	\zcirc	34
ztikz/wolfram/solve/var	29	\zclosepath	34
ztikz/wolfram/table/cell-cmd	28	\zcoor	34
ztikz/wolfram/table/format	28	Zdraw	33
ztikz/wolfram/table/hdbt-rule	28	\zdrawSetPathWidth	32
ztikz/wolfram/table/header	28	\zdrawSetUnit	32
ztikz/wolfram/cloud	8	\zeg	34
ztikz/wolfram/engine	8	\zfcOLOR	33
ztikz/zdraw/zplot/action	33	Zgroup	33
ztikz/zdraw/zplot/axis	33	\zlineto	33
ztikz/zdraw/zplot/domain	33	\zmoveto	33
ztikz/zdraw/zplot/endColor	33	\znewtext	34
ztikz/zdraw/zplot/range	33	\zplot	32
ztikz/zdraw/zplot/startColor	33	\zpolar	34
ztikz/zdraw/zrule/endColor	32	\zputtext	34
ztikz/zdraw/zrule/height	32	\zrect	34
ztikz/zdraw/zrule/startColor	32	\zrule	32
ztikz/zdraw/zrule/step	32	\zscaletext	34
ztikz/zdraw/zrule/width	32	\zscolor	33
ztikz/library	8	\zsethtext	34
\tikz	4	\zsetvtext	34
\tikzpicture	19	\zshift	35
		\ztexloadlib	10
U		ztikz commands:	
\usepath	35	\ztikz_hash_if_change:nn	40
W		\ztikzForceToRun	22, 40
\wolfram	27, 28	\ztikzForceToSkip	22, 40
\wolframanimation	27	\ztikzHashClean	22
\wolframDSolve	30	\ztikzHashCurrent	22
wolframGraphics	31	\ztikzloadlib	8, 10
\wolframOuputFile	22, 27, 31	\ztikzMkdir	9
\wolframResult	22, 27	ztool commands:	
\wolframSolve	29	\ztool_replace_file_line:nnn	17
\wolframTable	28	\ztrans	35
\wolframTableFData	28	\zxscale	35
\wolframTablePData	28	\zxvec	34
\wolframTex	28	\zyscale	35
		\zyvec	34