

Как рисовать графики функций в \LaTeX -документах с помощью таблицы значений функции и пакета *TikZ*

Е. А. Максименко

19 апреля 2008 г.

Аннотация

В этой небольшой заметке рассказано, как вставлять графики функций в \LaTeX -документы, используя таблицу значений функции и графический \LaTeX -пакет *TikZ*.

1 Основной принцип

Вычислительные возможности системы $\text{\TeX}/\text{\LaTeX}$ слабоваты для того, чтобы вычислять значения произвольной функции, заданной с помощью аналитического выражения. Но $\text{\TeX}/\text{\LaTeX}$ вполне подходит для того, чтобы по заданной таблице значений создать график функции и добавить оформление: оси координат, текстовые подписи и т. п.

Таким образом, текстовый файл с таблицей значений функции будет создаваться отдельной программой, а затем эта таблица значений будет с помощью подходящего \LaTeX -пакета превращаться в график функции.

2 Выбор \LaTeX -пакета

Будем использовать \LaTeX -пакет *TikZ* (автор — Till Tantau). *TikZ* — мощный графический пакет для программирования векторных рисунков внутри \LaTeX -документа. Этот пакет совместим с программами *dvips* и *pdflatex*, т. е. позволяет получать на выходе файлы в форматах PostScript и PDF.

Пакет *TikZ* представляет собой удобный \LaTeX -интерфейс к графическому \TeX -пакету *pgf*. Пакеты *TikZ* и *pgf* распространяются вместе, поэтому можно говорить о составном пакете *TikZ+pgf*. Пакет *TikZ+pgf* можно скачать со следующего адреса:

<http://sourceforge.net/projects/pgf>.

Возможно, перед установкой пакета *TikZ+pgf* нужно будет скачать и установить новую версию пакета *xcolor*. См. подробнее в инструкции по установке (файл *INSTALL*), которая прилагается к пакету *TikZ+pgf*.

Пакет *TikZ+pgf* содержит подробную документацию на английском языке в виде PDF-файла.

Чтобы использовать пакет *TikZ+pgf*, нужно в преамбуле \LaTeX -файла написать:

```
\usepackage{tikz}
```

3 Таблица значений функции

Таблица значений функции должна быть задана в виде текстового файла следующего вида:

```
# Это файл atan.table, который содержит
# таблицу значений функции "арктангенс"
# в пределах  $x = -3 \dots 3$  с шагом 0.1.
-3.0000 -1.2490
-2.9000 -1.2387
...
2.9000 1.2387
3.0000 1.2490
```

Как видно, строки с комментарием начинаются с символа `#`. Из каждой строки файла `TikZ` пытается взять два числа (абсциссу и ординату точки графика), оставшаяся часть строки не принимается во внимание. «Экспоненциальная запись» чисел (например, « $5e-1$ » вместо «0.5») недопустима.

Для создания такого текстового файла с таблицей значений функции подходят многие языки программирования. Например, вот программа на языке Gnuplot (<http://www.gnuplot.info>):

```
set terminal table
set format "%.4f"
set output "atan.table"
set samples 61
plot [x=-3:3] atan(x)
```

Программа на языке Python (<http://www.python.org>):

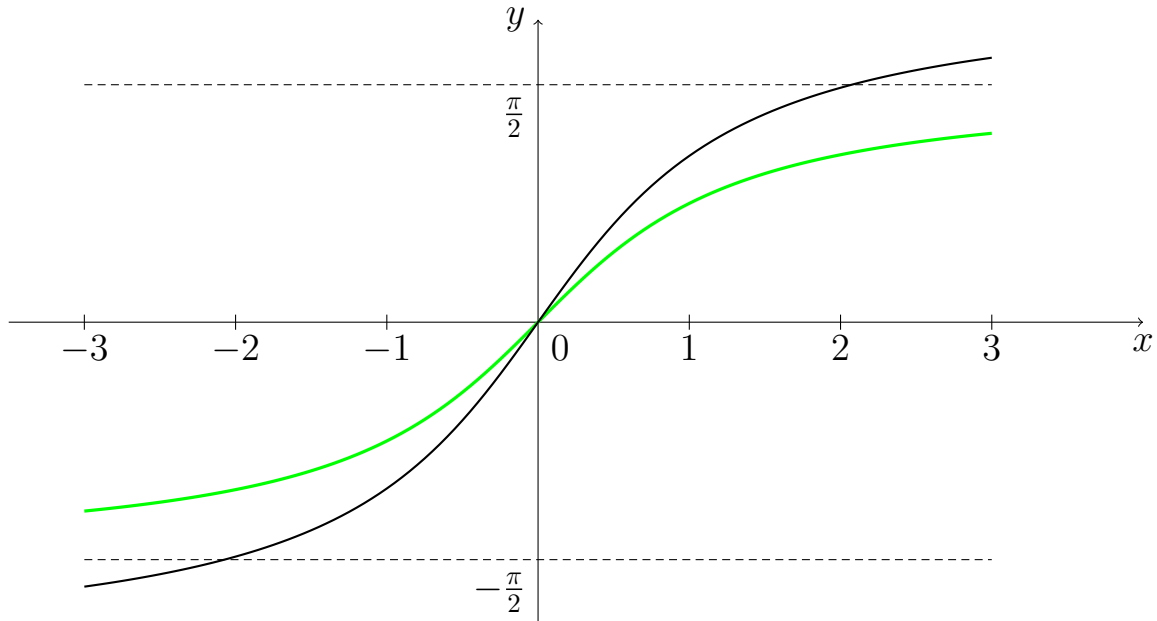
```
from math import *
fout = open("atan.table", "wt")
x1 = -3.0; x2 = 3.0; n = 60
h = (x2 - x1) / n
for k in range(n + 1):
    x = x1 + k * h
    print >>fout, "%.4f %.4f" % (x, atan(x))
```

4 Пример вставки графика

Внутри TikZ-рисунка, который задаётся окружением `tikzpicture`, график функции можно вставить с помощью команды следующего вида:

```
\draw plot[smooth] file{atan.table};
```

В следующем примере показано, как нарисовать несколько графиков, начертить систему координат и расставить подписи:



```
\begin{tikzpicture}[scale=2] % легко изменить масштаб
\draw[->] (-3.5,0) -- (4,0) node[anchor=north] {$x$};
\draw[->] (0,-2) -- (0,2) node[anchor=east] {$y$};
\foreach \x in {-3,-2,-1,1,2,3} {
  \draw (\x,0.05) -- (\x,-0.05);
  \draw (\x,0) node[anchor=north] {$\x$};
}
\draw (0,0) node[anchor=north west] {$0$};
\draw[black,very thick] plot[smooth] file{atan.table};
\draw[green,thick,yscale=1.4] plot[smooth] file{atan.table};
\draw[densely dashed,thin] (-3,1.57) -- (3,1.57);
\draw (0,1.57) node[anchor=north east] {$\frac{\pi}{2}$};
\draw[densely dashed,thin] (-3,-1.57) -- (3,-1.57);
\draw (0,-1.57) node[anchor=north east] {$-\frac{\pi}{2}$};
\end{tikzpicture}
```

5 Особенности этого подхода

Рассмотренный подход имеет следующие особенности:

- + Можно компилировать \LaTeX -файл как в формат PDF (с помощью программы `pdflatex`), так и в формат PostScript (с помощью программ `latex` и `dvips`).
- + При масштабировании рисунков размеры шрифтов и толщина линий остаются постоянными.
- + В подписях к рисунку автоматически используются те же гарнитуры и размеры шрифтов, что и в тексте.
- + Всё оформление рисунка определяется внутри \LaTeX -документа, поэтому легко согласовать оформление рисунков с оформлением текста.
- + Легко менять оформление осей координат и метки на осях. В подписях к рисунку легко помещать сложные \LaTeX -формулы.
- + Можно вычислять значения функции с помощью любимого языка программирования.
- + Задание элементов рисунка внутри \LaTeX -файла удобно при создании электронных презентаций (например, с помощью \LaTeX -пакета Beamer), так как при таком подходе легко устроить пошаговый показ элементов рисунка: сначала изобразить оси координат, затем одну часть графика, потом другую, потом график другой функции на том же рисунке.
- Диапазоны изменения координат приходится прописывать в двух местах: в программе, которая строит таблицу значений, и в \LaTeX -документе, для правильного оформления осей координат.
- Нет вспомогательных файлов с картинками графиков, которые можно было бы посмотреть отдельно.
- Есть отдельные table-файлы с таблицами значений функции.
- Оси координат и подписи на осях не создаются автоматически, их нужно задавать внутри TikZ -рисунка.
- Немного увеличивается время компиляции \LaTeX -файла.