The luaplot package in LaTeX

Chetan Shirore and Ajit Kumar

July 6, 2023

1 Introduction

The luaplot package is developed using Lua to plot graphs of real-valued functions of a real variable in LaTeX. It is developed with the MetaPost system [4] and luamplib [3] and luacode [2] packages. It provides an easy way for plotting graphs of standard mathematical functions and their finite combinations. There is no particular environment in the package for plotting graphs. It also works inside floating environments of LaTeX like tables and figures. The compilation time to plot several graphs in LaTeX using the luaplot package is significantly less with LuaLaTeX engine.

The package is based on the core idea of loading mathematical functions inside Lua and determining plot points using different methods available in Lua. After determining plot points in Lua, two different approaches are used:

• parse plot points to the MetaPost system via luampblib.
• parse plot points to the tikz package.

The MetaPost system is based on the Metafont to produce precise technical illustrations. Donald Knuth designed Metafont for TeX. John Hobby designed the MetaPost system to produce scalable PostScript or scalable vector graphics. The output from MetaPost can be directly included with LaTeX. The first approach thus offers a native way of plotting graphs inside LaTeX using Lua and MetaPost.

Tikz is designed by Till Tantau for producing vector graphics from different expressions. Drawing lines, arrows, paths, geometric shapes, etcetera is possible using Tikz [5]. Tikz commands can be considered TeX macros, but Tikz itself is a language. LaTeX users widely use Tikz to produce different graphics. The second approach combines Lua and Tikz to plot graphs inside LaTeX.

2 Installation and License

The installation of luaplot package is similar to plain latex package, where the .tex file is in LaTeX directory of texmf tree. The package can be included with \usepackage{luaplot} command in the preamble of the LaTeX document. A TeX file is to be compiled using the LuaLaTeX engine.

The luaplot package is released under the LaTeX Project Public License v1.3c or later. The complete license text is available at http://www.latex-project.org/lppl.txt. It is developed in Lua. Lua is available as a certified open-source software. Its license is simple and liberal, which is compatible with GPL.

3 The luaplot command

The luaplot command plots graph of a real valued function of a real variable in LaTeX using Lua and the MetaPost system. The plot points are exclusively determined using Lua. The algorithm is quite efficient to calculate coordinates of points for plotting in the MetaPost. It is deliberately written in Lua. The graphing is done using the MetaPost system. The luaplot command has the following syntax. It uses xkeyval package [6] for dealing with optional parameters.
The only compulsory argument is the function(s). Multiple functions can be given as input separated by a comma. The standard functions available from mathematics library are abs, acos, asin, atan, ceil, cos, exp, floor, log, sin, sqrt, tan, etcetera. The custom functions can easily be defined inside the luacode package. Table 1 lists optional parameters in the \texttt{luaplot} command with their brief description.

<table>
<thead>
<tr>
<th>Optional Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>xmin</td>
<td>Specifies the minimum value of $x$ coordinate used for calculating plot points. The default value of xmin is 0.</td>
</tr>
<tr>
<td>xmax</td>
<td>Specifies the maximum value of $x$ coordinate used for calculating plot points. The default value of xmax is 5.</td>
</tr>
<tr>
<td>ymin</td>
<td>Specifies the minimum value of $y$ coordinate used for calculating plot points. If not specified, it is determined by values of xmin and xmax.</td>
</tr>
<tr>
<td>ymax</td>
<td>Specifies the maximum value of $y$ coordinate used for calculating plot points. If not specified, it is determined by values of xmin and xmax.</td>
</tr>
<tr>
<td>plotpts</td>
<td>Specifies the number of plot points used to plot graphs of functions. The default value of \texttt{noofpoints} is 100.</td>
</tr>
<tr>
<td>hor</td>
<td>Specifies the width of plotting window. The default value of width is 5cm. It can be specified in any unit that LaTeX accepts: pt(point), pc(pica), in(inch), cm(centimeter), mm(millimeter), etcetera.</td>
</tr>
<tr>
<td>ver</td>
<td>Specifies the height of plotting window. The default value of height is 3cm. It can be specified in any unit that LaTeX accepts: pt(point), pc(pica), in(inch), cm(centimeter), mm(millimeter), etcetera.</td>
</tr>
<tr>
<td>plotsty</td>
<td>Specifies plotting styles of curves. Styles for different curves are to be given in curly braces separated by comma. The basic syntax is \texttt{withpen&lt;pen expression&gt;</td>
</tr>
<tr>
<td>clr</td>
<td>Specifies colors for plotting. Colors for different curves are to be given in curly braces separated by comma. The default color is black. Colors can be specified in a variety of ways. Standard colours can be specified by using names such as “red”, “blue”, “green” etcetera. The RGB format (r,g,b) can also be used. The custom colors can be defined in different ways and can be used with the package.</td>
</tr>
<tr>
<td>plotoptions</td>
<td>Specifies different options for plotting. The input is to be in curly braces. If the expression is a big string and multi-line then it can be enclosed in two square brackets (string in Lua). The different plot options include grids, labels, legends, axes styles, point markings on axes, framing styles etcetera. For all possible options, the MetaPost package documentation \cite{4} and guide \cite{1} can be referred.</td>
</tr>
</tbody>
</table>

Table 1: Optional parameters in the \texttt{luaplot} command
4 The luatikzpath command

The \texttt{luatikzpath} command produces a sequence of points for plotting that can be used inside tikz. The best feature is that all plotting options in tikz remain with the user. The all custom formatting can be done with draw options, node options, tikz styles etcetera in tikz. The sequence of points is entirely determined using Lua. The \texttt{luatikzpath} command in luaplot package has the following syntax.

\begin{verbatim}
\luaplot{function}{xmin}{xmax}{plot points}
\end{verbatim}

The first compulsory argument is the function. The standard functions available from mathematics library: \texttt{abs}, \texttt{acos}, \texttt{asin}, \texttt{atan}, \texttt{ceil}, \texttt{cos}, \texttt{exp}, \texttt{floor}, \texttt{log}, \texttt{sin}, \texttt{sqrt}, \texttt{tan}, etcetera can be input. The second and third compulsory arguments are \texttt{xmin} and \texttt{xmax}. These specify the beginning and end values of \texttt{x} coordinate. The last compulsory argument is the number of points used for plotting.

5 Plotting graphs with the luaplot command

Table 2 illustrates plotting graphs of functions with the \texttt{luaplot} command.

<table>
<thead>
<tr>
<th>LaTeX Input</th>
<th>Graph Produced</th>
</tr>
</thead>
<tbody>
<tr>
<td>\luaplot{exp(x)}</td>
<td><img src="image1.png" alt="Graph 1" /></td>
</tr>
</tbody>
</table>
| \luaplot[  
  xmin=-1, xmax=11,  
  clr={ 'blue' },  
  plotsty={'dashed evenly'}
]{cos(x)} | ![Graph 2](image2.png) |
| \luaplot[  
  xmin=-1, xmax=11,  
  ymin=-2, ymax=2,  
  plotpts=1000,  
  clr={ 'red, blue' }
]{cos(x),sin(x)} | ![Graph 3](image3.png) |
Listing 1 illustrates the use of optional argument \texttt{plotoptions} in the \texttt{luaplot} command. It generates graphs shown in Figure 1.

\begin{verbatim}
\luaplot[
  xmin=-10, xmax=20, plotpts=300, hor='6cm',
  ver='5cm', clr={'red, blue'},
  plotsty={'dashed evenly'},
  plotoptions=
  \[
  [\[
    \text{glabel(tex${\sin(x)+x}$,atex,(7,11.5));}
    \text{glabel(tex${\cos(x)}$,atex,(12,2));}
    \text{glabel.bot(tex Some Graphs atex, OUT);}
    \text{autogrid(grid.bot, withcolor .85white;}
    \text{autogrid(grid.lft, withcolor .85white;}
    \text{frame.dashed evenly;}
  ]]
  ]
\]
]{\sin(x)+x,\cos(x)}
\end{verbatim}

Table 2: Plotting graphs with the luaplot command
6 Plotting graphs with the luatikzpath command

Listing 2 illustrates plotting graph of a function with the \texttt{luatikzpath} command. Multiple graphs can be plotted in a single picture environment.

\begin{tikzpicture}
\draw[thin,->] (-4.2,0)--(5,0)node[right]{$x$};
\draw[thin,->] (0,-3)--(0,2.5)node[above]{$y$};
\draw[red] \luatikzpath{sin(x^2)}{-4}{4}{100} node at (1,1.3) {$y=\sin(x^2)$};
\draw[blue] \luatikzpath{log(x)}{0.1}{4.5}{100} node at (4,1.7) {$y=\log(x)$};
\end{tikzpicture}

Listing 2 generates graphs shown in Figure 2.

Figure 2: Plotting with the luatikzpath command
7 Known issues and limitations

The package does not use any external library supporting arbitrary precision arithmetic. The luaplot package can handle big and small numbers within the range of Lua that it supports. However, the MetaPost system does not support numbers in scientific notation. The coordinates of plot points produced in scientific notation are rounded off to 12 significant decimal places within the package. This may cause slight deviations from actual values. The same issue is not faced while parsing points to tikz as it supports input in scientific notation.

References


