NumericPlots - plot numeric data with latex

Thomas König, Alexander Michel, Michael Baumgart

September 19, 2013

Plotting numeric data is a task which often has to be done for scientific papers. In LaTeX, however, it is only possible to include graphics created with an external program. The pstricks-packages provide many commands to generate graphics in LaTeX. To generate simple graphics from numeric data, however, it is difficult to use. This package provides a simpler interface for the pstricks-package to plot numeric data.

The authors may be reached at numericplots@tikey.de.

Contents

Part I.
Introduction

Plotting numeric data is a task which often has to be done for scientific papers. In LaTeX, however, it is only possible to include graphics created with an external program. The pstricks-packages provide very many commands to generate graphics in LaTeX. To generate simple graphics from numeric data, however, it is difficult to use. This package provides a simpler interface for the pstricks-package to plot numeric data.

NumericPlots is free software: you can redistribute it and/or modify it under the terms

Part II.
Using the package

1. Basic Functionality

The package NumericPlots
`\usepackage{NumericPlots}`
is intended to be used to plot numeric data which may, e.g., be exported from Matlab by export2latex.m. The data must be defined in the form

```
\def\IdentI{
  1.0 1.0e2
  1.1 11e1
  1.2 1.25e2
  1.3 110
  1.4 100
  1.5 90
  1.6 80
  }
```

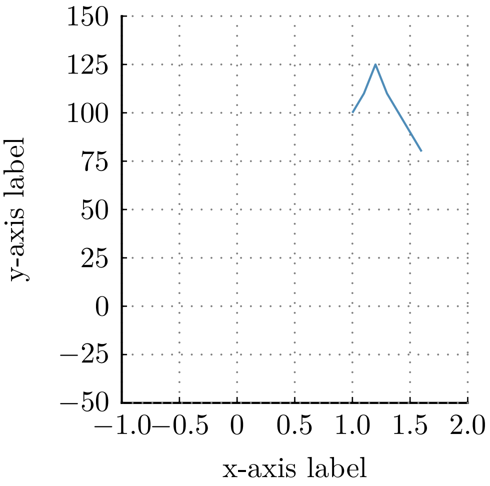
where the first column contains the x, the second column the y-data.

Please note that the package relies on the `listplot`-command from pstricks!

1.1. plots

The easiest plot may be done by

```
\begin{NumericDataPlot}{\textwidth
    }{6.5cm}
  \setxAxis{xMin=-1, xMax=2, Dx=0.5,
       x0=0}
  \setyAxis{yMin=-50, yMax=150, Dy
      =25, y0=0}

  \plotxAxis{x-axis label}
  \plotyAxis{y-axis label}

  \listplot[style=StdLineStyA]{\
      IdentI}
\end{NumericDataPlot}
```

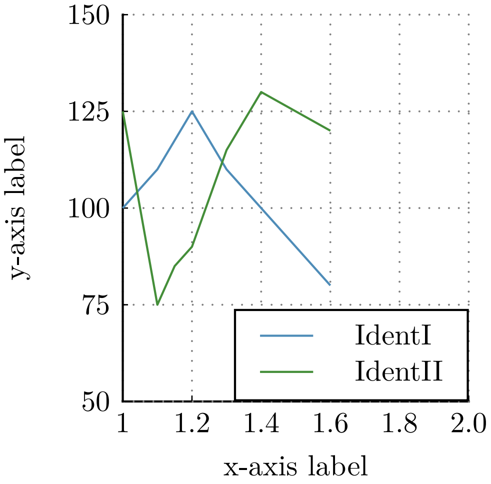


if you want to add a legend, you simply call

```
\LegendDefinition{
  \LegLine{style=StdLineStyA} &
      IdentI
}
```

To plot multiple data in one plot call

```
\begin{NumericDataPlot}
   {\textwidth}{6.5cm}
  \setxAxis
    {xMin=1, xMax=2, Dx=0.2}
  \setyAxis
    {yMin=50, yMax=150, Dy=25}

  \plotxAxis{x-axis label}
  \plotyAxis{y-axis label}

  \listplot[style=StdLineStyA]
    {\IdentI}
  \listplot[style=StdLineStyB]
    {\IdentII}

  \putSE{\LegendDefinition{
    \LegLine{style=StdLineStyA} &
        IdentI \\
    \LegLine{style=StdLineStyB} &
        IdentII
  }}
\end{NumericDataPlot}
```



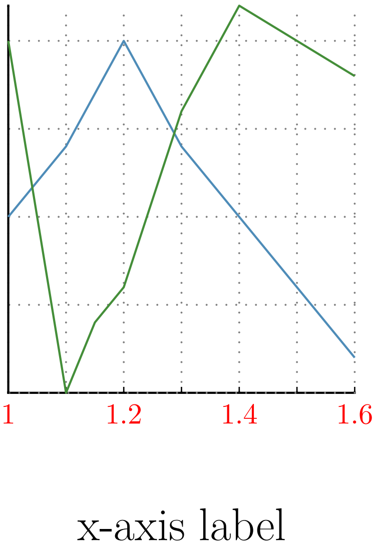
1.2. Label and TickLabels

The commands `plotxAxis` and `plotyAxis` take the options `NoLabel`, `NoTicks`, `NoTickLabel` as well as `LabelOption` and `TickLabelOption` which may be used to eliminate or change the look of the labels.

Standard values for `LabelOption` and `TickLabelOption` may be set
by `\newcommand{\StdLabelOption}{\color{blue}`
and `\newcommand{\StdTickLabelOption}{\small}`.

The option `LabelSep` may be used for `\plotxAxis` and `\plotyAxis` to set the seperation between the axis and the label. Standard value is `\baselineskip+1ex` for the x-label and `7ex` for the y-label.

```
\begin{NumericDataPlot}{\textwidth
    }{6.5cm}
  \setxAxis{xMin=1, xMax=1.6, Dx
      =0.1, DDx=0.2}
  \setyAxis{yMin=75, yMax=130, Dy
      =12.5}

  \plotxAxis
  [LabelOption=\LARGE,%
  TickLabelOption=\color{red},%
  LabelSep=40pt]
  {x-axis label}
  \plotyAxis
  [NoLabel, NoTicks, NoTickLabel]
  {y-axis label}

  \listplot[style=StdLineStyA]{\
      IdentI}
  \listplot[style=StdLineStyB]{\
      IdentII}

\end{NumericDataPlot}
```



It is furthermore possible to change the position and the rotation of the labels and tick labels, see the example in section 5.1.

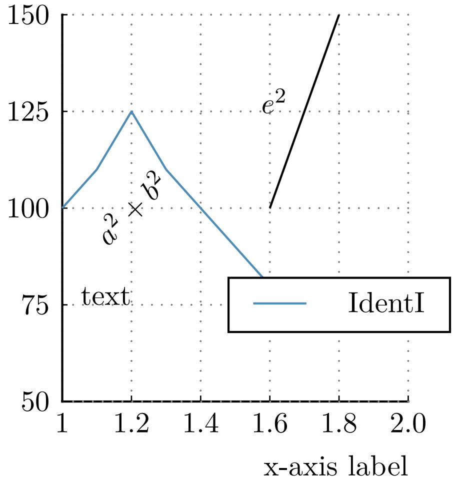
It is also possible to put single tick labels, see the example in section 5.1.

1.3. Place “Objects” in the plot.

There are basically two different options to place objects in the plot. To understand the difference one has to keep in mind that the axis have two different coordinate systems. One is the system defined by xMin, xMax, yMin and yMax (refered to as “DataCoordinateSystem”), the other ist the system defined by xCoordMin, xCoordMax, yCoordMin and yCoordMax (refered to as “PictureCoordinateSystem”), see section 2.

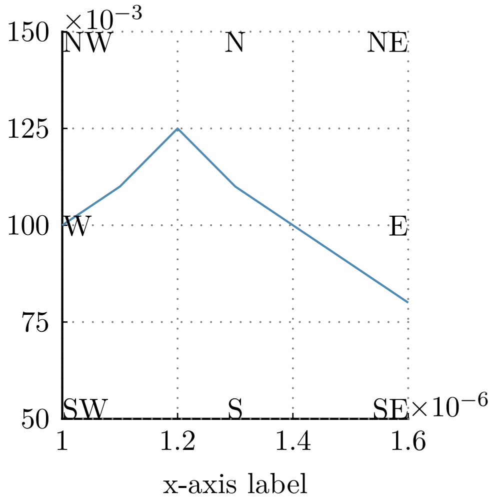
It is now possible to place stuff in the graph with the DataCoordinates with the command NDPput, see the following example.

```
\begin{NumericDataPlot}
   {\textwidth}{6.5cm}
  \setxAxis
     {xMin=1, xMax=2, Dx=0.2}
  \setyAxis
     {yMin=50, yMax=150, Dy=25}

  \plotxAxis[LabelPos=1, LabelRefPt=
      tr]%
    {x-axis label}
  \plotyAxis[NoLabel]{}

  \listplot[style=StdLineStyA]
     {\IdentI}

  % put some stuff somewhere
  \NDPput[x=1.2, y=75, RefPoint=br]{
      text}
  \NDPput[x=1.2, y=100, Rot=45]{$a
      ^2+b^2$}

  % or put nodes...
  \NDPput[x=1.6, y=100]{\pnode{A}}
  \NDPput[x=1.8, y=150]{\pnode{B}}
  % ...and draw a line between them
  \ncline{A}{B}
  % .. with text
  \naput{$e^2$}

  % or put the legend at a specific
      position
  \NDPput[x=1.8, y=75]{\
      LegendDefinition{
    \LegLine{style=StdLineStyA} &
        IdentI
  }}
\end{NumericDataPlot}
```



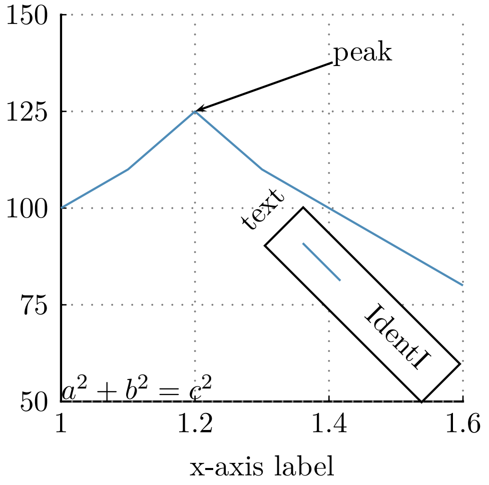
For convenience the commands `\putXX{object}` where $XX \in (N, S, E, W, NW, NE, SW, SE)$ are defined to place something in the North, South,. . . , SouthEast corner of the plot. Also, the command `\putExpY{xx}` and `\putExpX{xx}` may be used to place exponents at the axes.

```
\putExpY{$\times 10^{-3}$}
\putExpX{$\times 10^{-6}$}

\putN{N}
\putS{S}
\putW{W}
\putE{E}
\putNW{NW}
\putNE{NE}
\putSW{SW}
\putSE{SE}
```



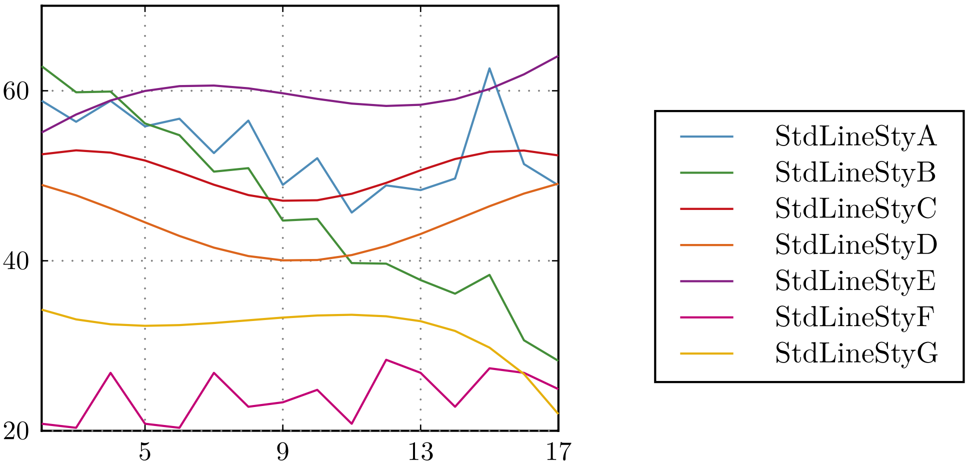
Alternatively, stuff can be placed within the plot with `\rput`.

```
% put text in the middle of the plot
\rput{45}(500,500){text}
% put a formula in the lower left
    corner
\rput[bl](0,0){$a^2+b^2=c^2$}

% or put nodes...
\NDPput[x=1.2, y=125]{\pnode{A}}
\rput(750,900){\Rnode{B}{peak}}
% ...and draw a line between them
\ncline{<-}{A}{B}

% or put the legend at a specific
    position
\rput{-45}(750,250){\LegendDefinition{
  \LegLine{style=StdLineStyA} & IdentI
}}
```

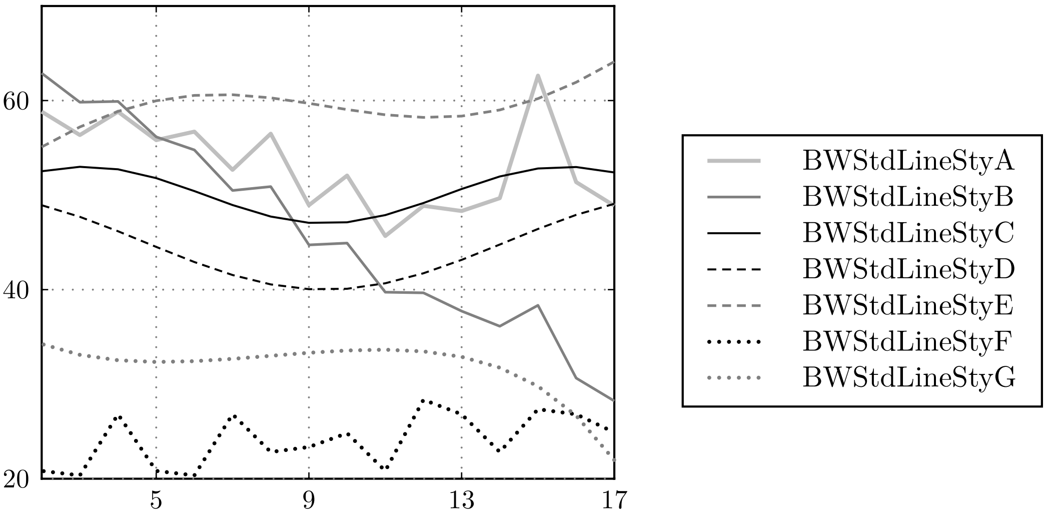


1.4. Linestyles and colors

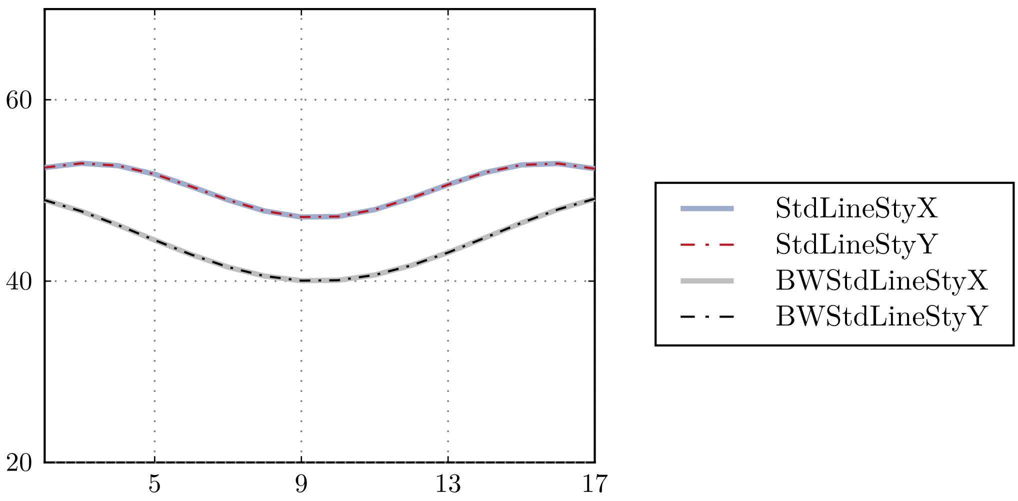
While using the package, there are predefined linestyles which may be used:



When using the package option `BW` the standard line styles will be replaced by their black and white counterparts:

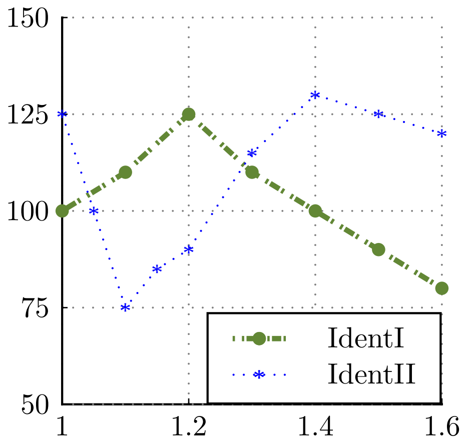


For values which are nearly the same (reference and measurement, e.g.) the following line styles may be used:



It is, of course, possible to redefine the available linestyles or to define new linestyles.

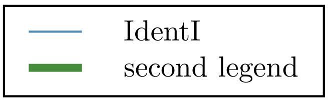
```
 \definecolor{MyColor}{cmyk}{0.6
     0.21 1.0 0.2}
 \newpsstyle{MyLine}{linecolor=
     MyColor, linewidth=2pt,
     linestyle=dashed,
 dash=1pt 1pt 4pt 1pt 1pt 3pt,
     dotstyle=*, showpoints=true,
     dotsize=5pt}
 \newpsstyle{MyLineA}{linecolor=blue
     , linestyle=dotted, dotstyle=
     asterisk,
 showpoints=true}

\setxAxis
   {xMin=1, xMax=1.6, Dx=0.2}
\setyAxis
   {yMin=50, yMax=150, Dy=25}

\plotxAxis[NoLabel]{}
\plotyAxis[NoLabel]{}

\listplot[style=MyLine]
   {\IdentI}
\listplot[style=MyLineA]
   {\IdentII}

\putSE{\LegendDefinition{
  \LegLine{style=MyLine} & IdentI\\
  \LegLine{style=MyLineA} & IdentII
}}
```



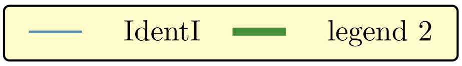
1.5. Legend

The legend may be created with `\LegendDefinition`. The command takes the two optional arguments `nrCols` and `LabelOrientation=[l|c|r]`. The mandatory argument is the definition of a table as demonstrated in the follwing examples.

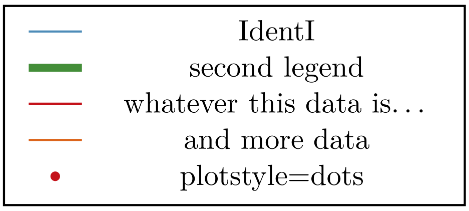
```
\LegendDefinition{
  \LegLine{style=StdLineStyA} & IdentI\\
  \LegLine{style=StdLineStyB, linewidth
      =3pt} & second legend
}
```



```
\newpsstyle{LegendBoxStyle}%
  {framearc=0.2, fillstyle=solid,
      fillcolor=yellow, opacity=0.2}
\LegendDefinition[nrCols=2]{
  \LegLine{style=StdLineStyA} & IdentI
       &
  \LegLine{style=StdLineStyB,
      linewidth=3pt} & legend 2
}
\newpsstyle{LegendBoxStyle}%
  {fillstyle=solid, fillcolor=white}
```



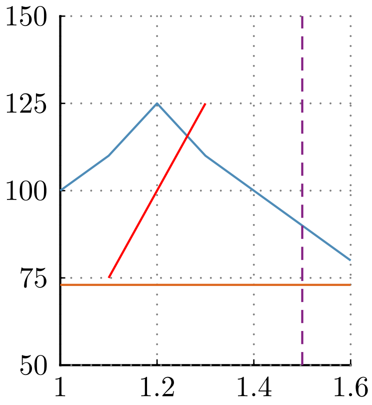
```
\LegendDefinition[LabelOrientation=c]{
  \LegLine{style=StdLineStyA} & IdentI\\
  \LegLine{style=StdLineStyB, linewidth
      =3pt} & second legend\\
  \LegLine{style=StdLineStyC} & whatever
       this data is\ldots\\
  \LegLine{style=StdLineStyD} & and more
       data\\
  \LegDot{style=StdLineStyC, plotstyle=
      dots} & plotstyle=dots
}
```



1.6. Add Lines to the Plot

Horizontal and vertical lines may be added to the plot with the commands `\NDPhline{coord}`, `\NDPvline{coord}` and `\NDPline{coord}`. It is also possible to put nodes and draw lines between them, see section 1.3.

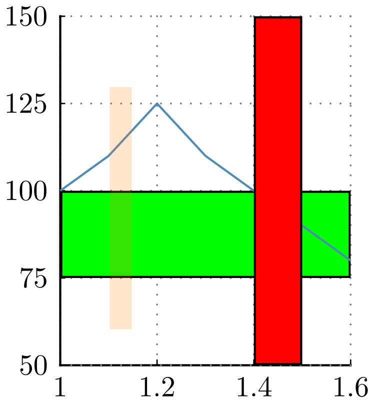
```
\NDPhline[linecolor=LineColorD]{73}
\NDPvline[linecolor=LineColorE, %
  linestyle=dashed]{1.5}
\NDPline[linecolor=red
    ]{1.1}{75}{1.3}{125}
```



1.7. Add Boxes to the Plot

Horizontal and vertical boxes may be added to the plot with the commands `\NDPhbox{coord}`, `\NDPvbox{coord}` and `\NDPbox{coord}`. It is also possible to put nodes and draw lines between them, see section 1.3.

```
\NDPhbox[fillstyle=solid,fillcolor =
    green]%
         {75.0}{100.0}%

\listplot[style=StdLineStyA] {\IdentI}

\NDPvbox[fillstyle=solid,fillcolor=red
    ]
         {1.4}{1.5}

  \NDPbox[fillstyle=solid,%
     fillcolor=orange, opacity=0.2,
         linestyle=none]%
  {1.1}{60}{1.15}{130}
```



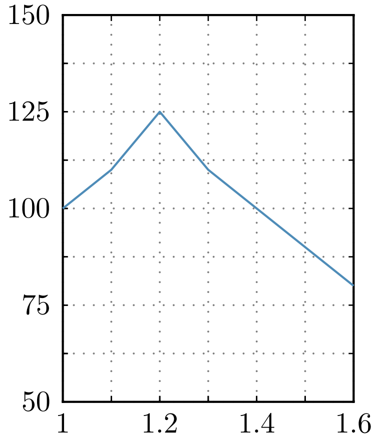
1.8. Grid

As shwon in the example below, it is possible to plot a fine grid but not have a tick label at each of the grid lines. While the options `Dx` and `Dy` define the distance between the grid lines, the options `DDx` and `DDy` define the distance between the tick labels. If `DDx` or `DDy` are not set, they take the values of `Dx` and `Dy`.

One may choose not to plot the grid with the option `NoGrid` for the commands `\plotxAxis` and `\plotyAxis`.

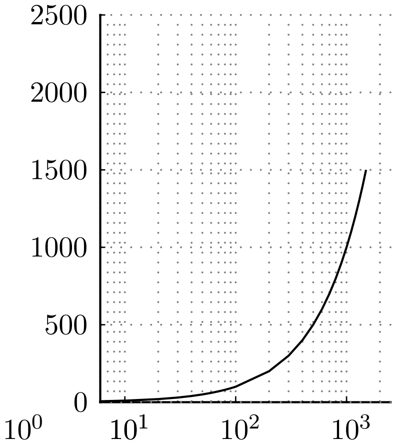
If the grid is plottet with the axis it may happen that the grid is plottet over the axis. To avoid this, plot the grid first and then plot the axis as shown.

```
\begin{NumericDataPlot}
   {\textwidth}{6.5cm}
  \setxAxis{xMin=1, xMax=1.6, Dx=0.1,
      DDx=0.2}
  \setyAxis{yMin=50, yMax=150, Dy=12.5,
      DDy=25}

  \plotxGrid
  \plotyGrid
  \plotxAxis[NoLabel, NoGrid, AxisStyle=
      Boxed]{}
  \plotyAxis[NoLabel, NoGrid, AxisStyle=
      Boxed]{}

  \listplot[style=StdLineStyA]{\IdentI}

\end{NumericDataPlot}
```

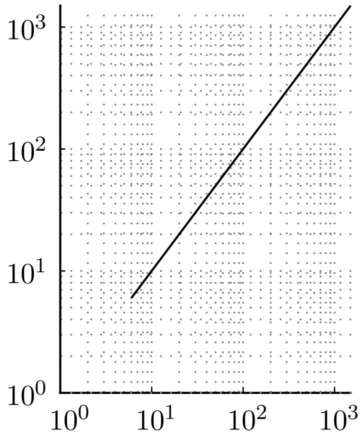


1.9. Logarithmic axes

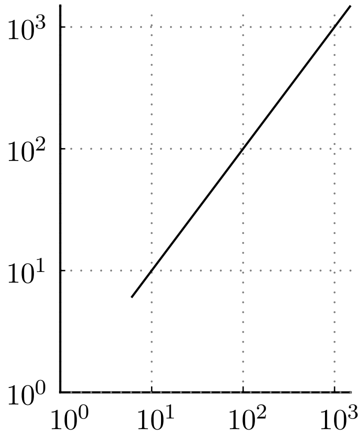
```
\begin{NumericDataPlot}{\textwidth}{6.5
    cm}
  \setxAxis{xMin=6, xMax=2500, xLog}
  \setyAxis{yMin=0, yMax=2500, Dy=500}
  \plotxAxis{}
  \plotyAxis{}

  \listplot{\LogData}
\end{NumericDataPlot}
```



```
\begin{NumericDataPlot}{\textwidth}{6.5
    cm}
  \setxAxis{xMin=1, xMax=1500, xLog}
  \setyAxis{yMin=1, yMax=1500, yLog}
  \plotxAxis{}
  \plotyAxis{}

  \listplot{\LogData}
\end{NumericDataPlot}
```



```
\begin{NumericDataPlot}{\textwidth}{6.5
    cm}
  \setxAxis{xMin=1, xMax=1500, xLog,
      xLogNoSubGrid}
  \setyAxis{yMin=1, yMax=1500, yLog,
      yLogNoSubGrid}
  \plotxAxis{}
  \plotyAxis{}

  \listplot{\LogData}
\end{NumericDataPlot}
```



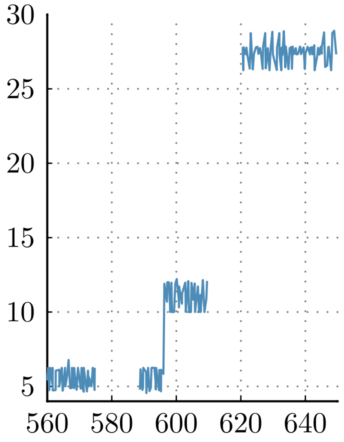
1.10. plots with holes

If plotting data, one might not want to plot some part of this data, e.g. if only the data in a certain y-range is interesting. If one chooses to eliminate the invalid data either by using `yStart` or `yEnd` or by eliminating the data before the export, pstricks will always connect the last valid point of an interval with the first valid point of the next interval. In Matlab e.g., invalid data (NaN) is just ignored. This behavior is also possible with NumericPlots and export2latex. If export2latex is called with data which contains NaN and with the option.NaNsplit=true, it will split the data in several intervals and append a consecutive number to the identifier for each interval. The exported yMax, yMin values etc. will still be valid for the whole data-vector. The exported data might look like:

```
\expandafter\def\csname DataIdentNrRanges\endcsname{3}
\expandafter\def\csname DataIdent1\endcsname{
560.00 5.40
...
574.72 6.25
}

\expandafter\def\csname DataIdent2\endcsname{
588.73 4.78
...
609.53 12.09
}


\expandafter\def\csname DataIdent3\endcsname{
620.56 27.81
...
649.56 27.32
}
```

This data can then easily be plotted using the command `\multilistplot{<options>}{<Identifier>}`[1]. The command will call `\listplot[<options>]{\<Identifier>}` for each of the intervals. The first argument is directly passed to the listplot command which is part of the pstricks package. The second argument is the identifier of the data that should be plotted.

```
\begin{NumericDataPlot}{\textwidth}{6.5
    cm}
  \setxAxis{xMin=560, xMax=650, Dx=20}
  \setyAxis{yMin=4, yMax=30, Dy=5, y0=5}
  \plotxAxis{}
  \plotyAxis{}

  \multilistplot{style=StdLineStyA}{
      DataIdent}
\end{NumericDataPlot}
```



---

[1] Please notice, that the first argument of multilistplot is a mandatory argument. Also notice, that the second argument ist only the name of the data including "Data" but not the command as it would be for the listplot command

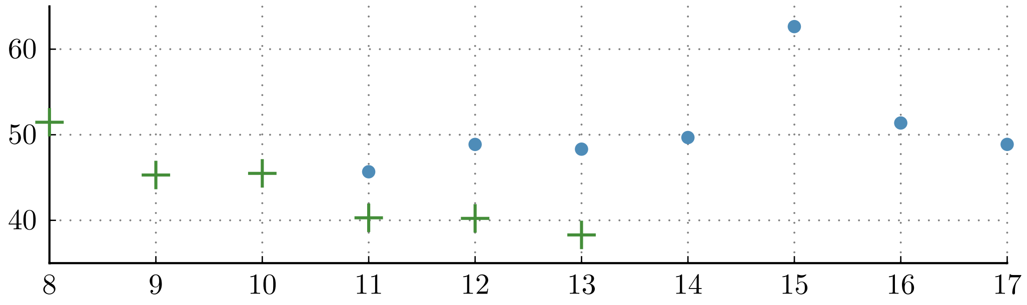
2. Multiple plots in one picture

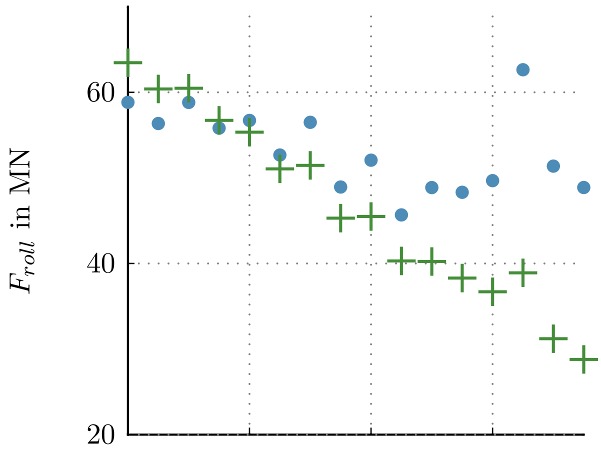
xPicMin, xPicMax, yPicMin and yPicMax are the inner coordinates of one picture. The position of the axes are defined in this coordinate system via xCoordMin, xCoordMax, yCoordMin and yCoordMax.

Example:

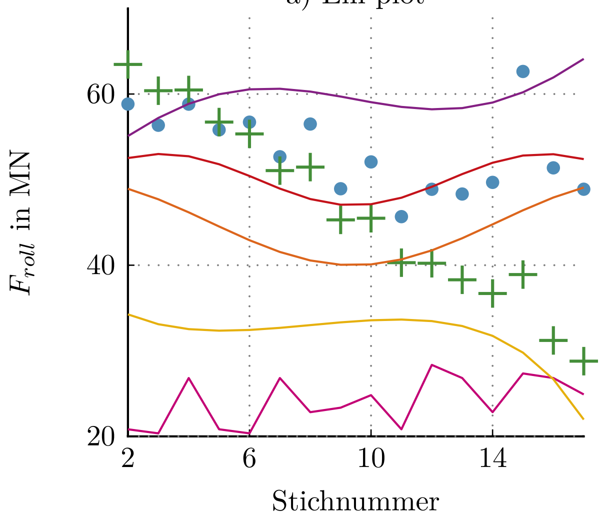
```
\begin{NumericDataPlot}[xPicMin=0, xPicMax=1050,
yPicMin=0, yPicMax=1450]{\textwidth}{0.85\textheight}

  % --- definition of the axis and the grid ---
  % set the axis of the lower left corner
  \setxAxis{xMin=2, xMax=17, Dx=4, xCoordMin=0, xCoordMax=500}
  \setyAxis{yMin=20, yMax=70, Dy=20, yCoordMin=0, yCoordMax=500}

  % plot the axis of the lower left corner
  \plotxAxis{Stichnummer}
  \plotyAxis{$F_{roll}$ in $\si{\mega\newton}$}

  \listplot[style=Database]{\DataA}
  \listplot[style=Result]{\DataB}
  \listplot[style=StdLineStyC]{\DataC}
  \listplot[style=StdLineStyD]{\DataD}
  \listplot[style=StdLineStyE]{\DataE}
  \listplot[style=StdLineStyF]{\DataF}
  \listplot[style=StdLineStyG]{\DataG}

  % set the y-axis for the plot in the middle of the left side
  % x-axis remains the same
  \setyAxis{yMin=20, yMax=70, Dy=20, yCoordMin=550, yCoordMax=1050}
  % plot the axis (x-axis without ticklabels)
  \plotxAxis[NoTickLabel,LabelSep=1ex]{a) Ein plot}
  \plotyAxis{$F_{roll}$ in $\si{\mega\newton}$}

  \listplot[style=Database]{\DataA}
  \listplot[style=Result]{\DataB}

  % set axis for the plot at the right side
  \setxAxis{xMin=2, xMax=17, Dx=4, xCoordMin=600, xCoordMax=1050}
  \setyAxis{yMin=20, yMax=70, Dy=10, yCoordMin=0, yCoordMax=1050}

  % plot the axis at the right side (y-axis without label)
  \plotxAxis{Stichnummer}
  \plotyAxis[NoLabel]{$F_{roll}$ in $\mega\newton$}

  \listplot[style=Database]{\DataA}
  \listplot[style=Result]{\DataB}
```

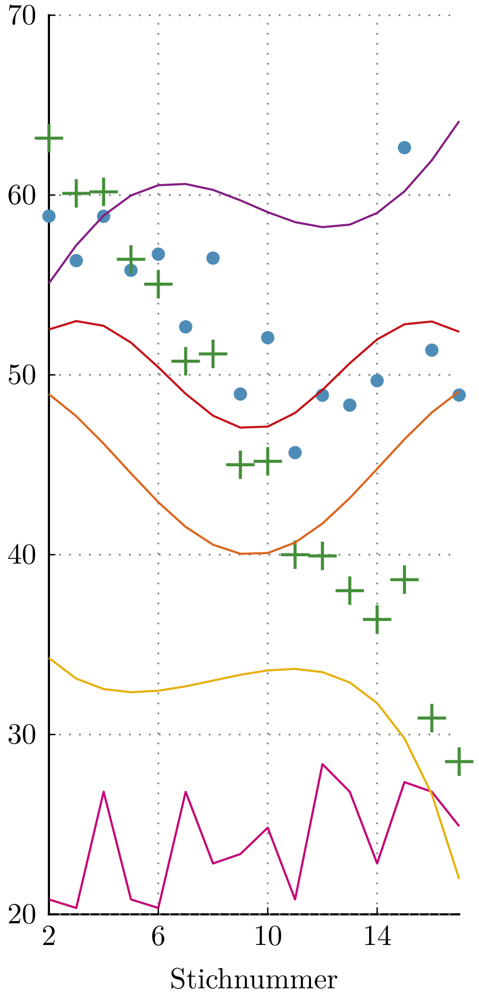
```
    \listplot[style=StdLineStyC]{\DataC}
    \listplot[style=StdLineStyD]{\DataD}
    \listplot[style=StdLineStyE]{\DataE}
    \listplot[style=StdLineStyF]{\DataF}
    \listplot[style=StdLineStyG]{\DataG}

    % set the axis for the plot at the top
    \setxAxis{xMin=8, xMax=17, Dx=1, xCoordMin=0, xCoordMax=1050}
    \setyAxis{yMin=35, yMax=65, Dy=10, y0=40, yCoordMin=1150, yCoordMax=1450}

    % plot the axis for the plot at the top
    \plotxAxis[NoLabel]{Stichnummer}
    \plotyAxis[NoLabel]{$F_{roll}$ in $\mega\newton$}

    % plot only part of the data
    \listplot[style=Database, xStart=11, xEnd=17]{\DataA}
    \listplot[style=Result, xStart=8, xEnd=13]{\DataB}

  \end{NumericDataPlot}

  % put legend outside of the plot
  \LegendDefinition[nrCols=1, LabelOrientation=r]{
    \LegLine{style=Database} & Werte aus der Datenbank \\
    \LegLine{style=Result} & Modell}
\end{center}
```

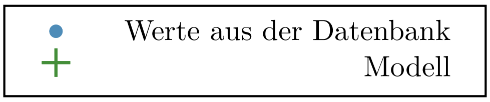




a) Ein plot

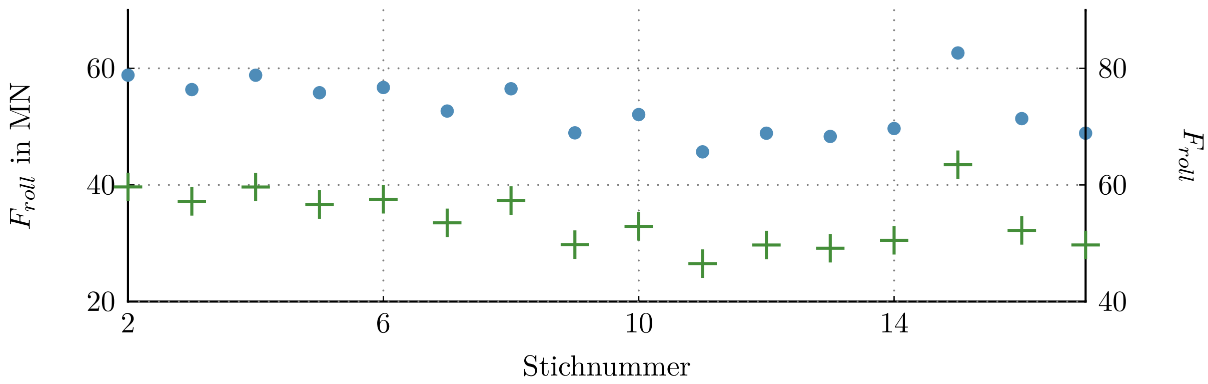






An example with different y-axes on the left and on the right side:

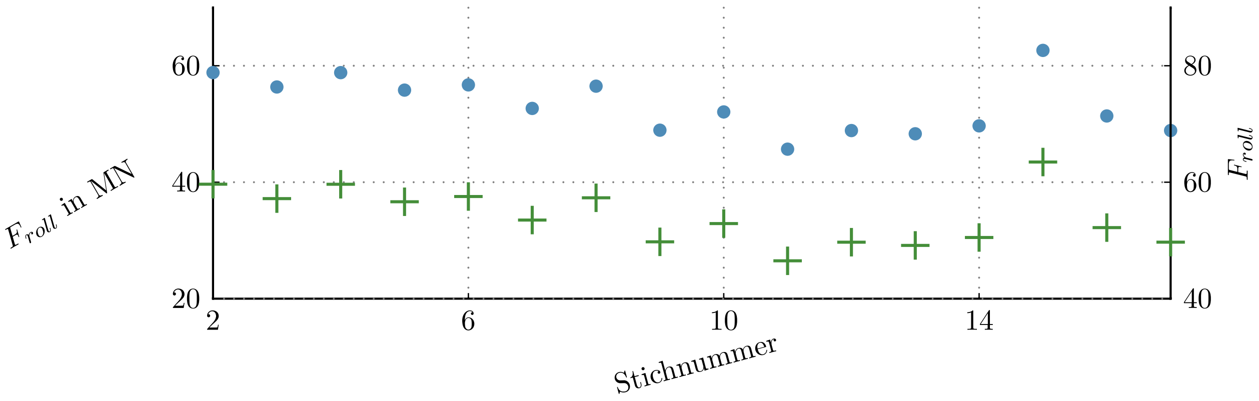
```
\newpsstyle{Database}{linecolor=LineColorA, linestyle=none, dotstyle=*,
showpoints=true, dotsize=5pt}
\newpsstyle{Result}{linecolor=LineColorB, linestyle=none, dotstyle=+,
showpoints=true, dotsize=10pt}

\begin{NumericDataPlot}{\textwidth}{0.25\textheight}

  % --- definition of the axis and the grid ---
  \setxAxis{xMin=2, xMax=17, Dx=4}
  \setyAxis{yMin=20, yMax=70, Dy=20}

  % plot the axis of the lower left corner
  \plotxAxis{Stichnummer}
  \plotyAxis{$F_{roll}$ in $\si{\mega\newton}$}

  \listplot[style=Database]{\DataA}

  % define a second y-axis
  \setyAxis{yMin=40, yMax=90, Dy=20}
  \plotyAxis[AxisStyle=Right, NoGrid]{$F_{roll}$}

  \listplot[style=Result]{\DataA}

\end{NumericDataPlot}
```



It is also possible to rotate the labels of the axes (LabelRot and LabelRefPt have to be set after AxisStyle!):

```
  % plot the axis of the lower left corner
  \plotxAxis[LabelRot=15]{Stichnummer}
  \plotyAxis[LabelRot=30, LabelRefPt=tr]{$F_{roll}$ in $\si{\mega\newton}$}

  \listplot[style=Database]{\DataA}

  % define a second y-axis
```

```
\setyAxis{yMin=40, yMax=90, Dy=20}
\plotyAxis[AxisStyle=Right, NoGrid, LabelRot=90]{$F_{roll}$}
```



3. Matlab Support

3.1. export2latex.m

The function `export2latex(data, filename, [options])` may be used to export data from Matlab to be used with NumericPlots.sty. The function takes the two parameters data and filename, where filename is the name of the file where the data should be stored with full path but without extension. Data is a structure with the three entries x, y, and ident, where x are the x-data, y the y-data and ident is an identifier to have access to the data in Latex. To access the data in Latex the command `\Data<ident>` is used.

The parameter data may be provided as an 1xK array if each entry of the array is a structure with the entries x, y and ident. There may be an additional entry descr.

- data(1,i).x = X_i
- data(1,i).y = Y_i
- data(1,i).ident = Identifier
- (data(1,i).descr = Description)
- (data(1,i).group = GroupNr)

Identifier must be a valid Latex command name, which basically means it must be a string without numbers.

For every entry of data, X and Y must be arrays of the same size MxN. If $M > 1$, each column of the array is exported to the file and the identifier is expanded by the roman representation of the column number.

It might be useful to do a `interp1` before exporting the data to get smaller files and to use less of TEX's memory. (→ script Patrick?!?)

The optional parameter `options` might be used to control which additional information is provided in the output file.

- `options.DataBoundaries` [true] If set to true, the output file contains the commands `\DataXmin`, `\DataXmax`, `\DataYmin` and `\DataYmax` which may be used to define the axis.
- `options.AxisBoundaries` [false] If set to true, the output file contains the com-

mands `\DataXminAxis`, `\DataXmaxAxis`, `\DataYminAxis` and `\DataYmaxAxis` which may be used to define the axis. In contrary to the DataBoundaries the AxisBoundaries add an additional gap of `options.AxisBoundariesGap` [10] percent of the full scale to the data boundaries such that the plot doesn't touch the axis.

- `options.SuppressWarning` [false] suppresses the warning about max/min values being to close together
- `options.precision` [empty] how many decimal places should be printed for x and y values. Will be calculated automatically if left empty.
- `options.NaNsplit [false]` if true, the data will be split at NaN values. See `\multilistplot` for how to plot them.

3.2. struct2latex.m

The function
`struct2latex(data,[xname],[downsample],[filtertype],[filename],[postfix],[options])` may be used to export a Matlab struct to be used with NumericPlots.sty. The function is based on export2latex and the `options` argument is given to export2latex. The optional argument `xname` specifies the fieldname of the x-data. If no `xname` is given, you have to chose one via popup. All other fields are treated like y-data. Furthermore, the identifier of the y-data are equal to the fieldnames and have to be valid Latex commands (fieldnames with numbers are not supported). The struct2latex command only supports 1 dimensional arrays (row vectors). Column vectors are mapped for propper export without warning. With the optional argument `downsample` (integer$>$ 0) it is possible to downsample the data for smaller files. A warning is given when data points exceed 5000. Usually 1000 data points are enough for a propper plot in latex. The optional argument `filename` specifies the outputname with path of the texfile. If no filename is given, it has to be chosen via popup. The optional argument `postfix` can be used to put a postfix after the fieldname in the identifier of the y-data. The `options` argument is also optional and equal to the options argument of the function export2latex.

Filter may be one of `{'none','MinMax','Hull'}`. The corresponding filter will be applied to the data before exporting it. See 6.3 for further details.

Matlab example.

```
% data for struct2latex should be row-vectors. anyway, column-vectors are
% mapped to row-vectors. matrices are not supported
```

```
%% export row-vectors
t = [1:0.1:10];

data.time = t;
data.sin = sin(t);
data.cos = cos(t);
data.exp = exp(t);
data.tan = tan(t);

%struct2latex(data,xname,downsample,filename,filtertype,postfix,options)
%downsample,filename,postfix,options are optional arguments
struct2latex(data,'time',1,'none','texdataA','row');
```

3.3. dspace2struct

The function `dspace2struct(dataname,version)` may be used to import `.mat` created by dSPACE to MATLAB. It returns a MATLAB struct out of the provided variables `dataname` and `version`. Thereby `dataname` is the name of the dSPACE `.mat` file and the optional string argument `version` has to be equal `version = 'NG'` if dSPACE CONTROL DESK NEXT GENERATION is used. It is convenient to chose a unique signal name in MATLAB/SIMULINK because they will be the same in dSPACE. The function `dspace2struct(dataname,version)` just adopts the names provided by dSPACE and use them as fieldnames. The time vector will always be called `.time`.

Matlab example:

```
% Old Control Desk:
dataOG = dspace2struct(measurements)
% Control Desk Next Generation:
dataNG = dspace2struct(measurements,'NG')
```

The resulting structs `dataOG` and `dataNG` differ in way of structuring. In the older versions of dSPACE it was only possible to save one time vector. The dSPACE CONTROL DESK NEXT GENERATION allows to save several time vectors. So it was necessary to save the index of the time vector with the signal name. Typical plot commands for the two data structures look as follows:

```
%Old Control Desk
plot(dataOG.time,dataOG.signalname);
```

```
%Control Desk Next Generation
plot(dataNG.time(dataNG.signalname.timeidx),dataNG.signalname.Data);
```

3.4. dspace2latex

The function `dspace2latex` is just a combination of `dspace2struct` and `struct2latex` with commonly used arguments. It is a simple and quick way to export measurements from dSpace to LaTeX without any revision of the data. However, the current implementation allows only data structures of the old version of dSpace Control Desk.

4. Options

4.1. General

These are the options for the environment `NumericDataPlot`. Note that the padding values llx, lly, urx, ury depend on the type settings.

- xPicMin [0], yPicMin [0], xPicMax [1000], yPicMax [1000]
- TickLength [2] defines the length of the ticks in mm
- `\origXLabelSep` [`\baselineskip`+1ex], `\origYLabelSep` [7ex]
- `\StdLLX` [7ex +`\baselineskip`+2pt] `\StdLLY`[2`\baselineskip`+1ex+2pt] `\StdURX`[2ex] `\StdURY`[0.5em]

4.2. Automatic list with xkvview

The following list of keys defined by the package was created with xkvview.

Key	Prefix	Family	Type	Default
xMax	NumericDataPlot	Axis	command	[none]
xMin	NumericDataPlot	Axis	command	[none]
Dx	NumericDataPlot	Axis	command	[none]
dx	NumericDataPlot	Axis	command	[none]
xCoordMax	NumericDataPlot	Axis	command	[none]
xCoordMin	NumericDataPlot	Axis	command	[none]
yMax	NumericDataPlot	Axis	command	[none]
yMin	NumericDataPlot	Axis	command	[none]
Dy	NumericDataPlot	Axis	command	[none]
dy	NumericDataPlot	Axis	command	[none]
yCoordMax	NumericDataPlot	Axis	command	[none]
yCoordMin	NumericDataPlot	Axis	command	[none]
xLog	NumericDataPlot	Axis	boolean	true
yLog	NumericDataPlot	Axis	boolean	true
xLogNoSubGrid	NumericDataPlot	Axis	boolean	true
yLogNoSubGrid	NumericDataPlot	Axis	boolean	true
yNearlyTight	NumericDataPlot	Axis	boolean	false
x0	NumericDataPlot	AxisWait	command	[none]
y0	NumericDataPlot	AxisWait	command	[none]

Continued on next page

Continued from previous page

Key	Prefix	Family	Type	Default
`DDy`	`NumericDataPlot`	`AxisWait`	`command`	`[none]`
`DDx`	`NumericDataPlot`	`AxisWait`	`command`	`[none]`
`NoGrid`	`NumericDataPlot`	`xAxis`	`boolean`	`true`
`NoTicks`	`NumericDataPlot`	`xAxis`	`boolean`	`true`
`NoLabel`	`NumericDataPlot`	`xAxis`	`boolean`	`true`
`NoTickLabel`	`NumericDataPlot`	`xAxis`	`boolean`	`true`
`NoGrid`	`NumericDataPlot`	`yAxis`	`boolean`	`true`
`NoTicks`	`NumericDataPlot`	`yAxis`	`boolean`	`true`
`NoLabel`	`NumericDataPlot`	`yAxis`	`boolean`	`true`
`NoTickLabel`	`NumericDataPlot`	`yAxis`	`boolean`	`true`
`AxisStyle`	`NumericDataPlot`	`xAxis`	`choice`	`[none]`
`AxisStyle`	`NumericDataPlot`	`yAxis`	`choice`	`[none]`
`xPicMin`	`NumericDataPlot`	`General`	`command`	`[none]`
`yPicMin`	`NumericDataPlot`	`General`	`command`	`[none]`
`xPicMax`	`NumericDataPlot`	`General`	`command`	`[none]`
`yPicMax`	`NumericDataPlot`	`General`	`command`	`[none]`
`TickLength`	`NumericDataPlot`	`General`	`command`	`[none]`
`llx`	`NumericDataPlot`	`General`	`ordinary`	`\StdLLX`
`urx`	`NumericDataPlot`	`General`	`ordinary`	`\StdURX`
`lly`	`NumericDataPlot`	`General`	`ordinary`	`\StdLLY`
`ury`	`NumericDataPlot`	`General`	`ordinary`	`\StdURY`
`x`	`NumericDataPlot`	`put`	`ordinary`	`[none]`
`y`	`NumericDataPlot`	`put`	`ordinary`	`[none]`
`RefPoint`	`NumericDataPlot`	`put`	`ordinary`	`c`
`Rot`	`NumericDataPlot`	`put`	`ordinary`	`0`
`LabelPos`	`NumericDataPlot`	`putxLabel`	`command`	`[none]`
`LabelOption`	`NumericDataPlot`	`putxLabel`	`command`	`[none]`
`LabelOrientation`	`NumericDataPlot`	`putxLabel`	`command`	`[none]`
`LabelRot`	`NumericDataPlot`	`putxLabel`	`command`	`[none]`
`LabelRefPt`	`NumericDataPlot`	`putxLabel`	`command`	`[none]`
`LabelPos`	`NumericDataPlot`	`putyLabel`	`command`	`[none]`
`LabelOption`	`NumericDataPlot`	`putyLabel`	`command`	`[none]`
`LabelOrientation`	`NumericDataPlot`	`putyLabel`	`command`	`[none]`
`LabelRot`	`NumericDataPlot`	`putyLabel`	`command`	`[none]`
`LabelRefPt`	`NumericDataPlot`	`putyLabel`	`command`	`[none]`
`LabelSep`	`NumericDataPlot`	`putxLabel`	`ordinary`	`\origXLabelSep`
`LabelSep`	`NumericDataPlot`	`putyLabel`	`ordinary`	`\origYLabelSep`
`ax`	`NumericDataPlot`	`putxLabel`	`choice`	`[none]`
`ax`	`NumericDataPlot`	`putyLabel`	`choice`	`[none]`
`x`	`NumericDataPlot`	`putxTickLabel`	`ordinary`	`[none]`
`y`	`NumericDataPlot`	`putyTickLabel`	`ordinary`	`[none]`

Continued on next page

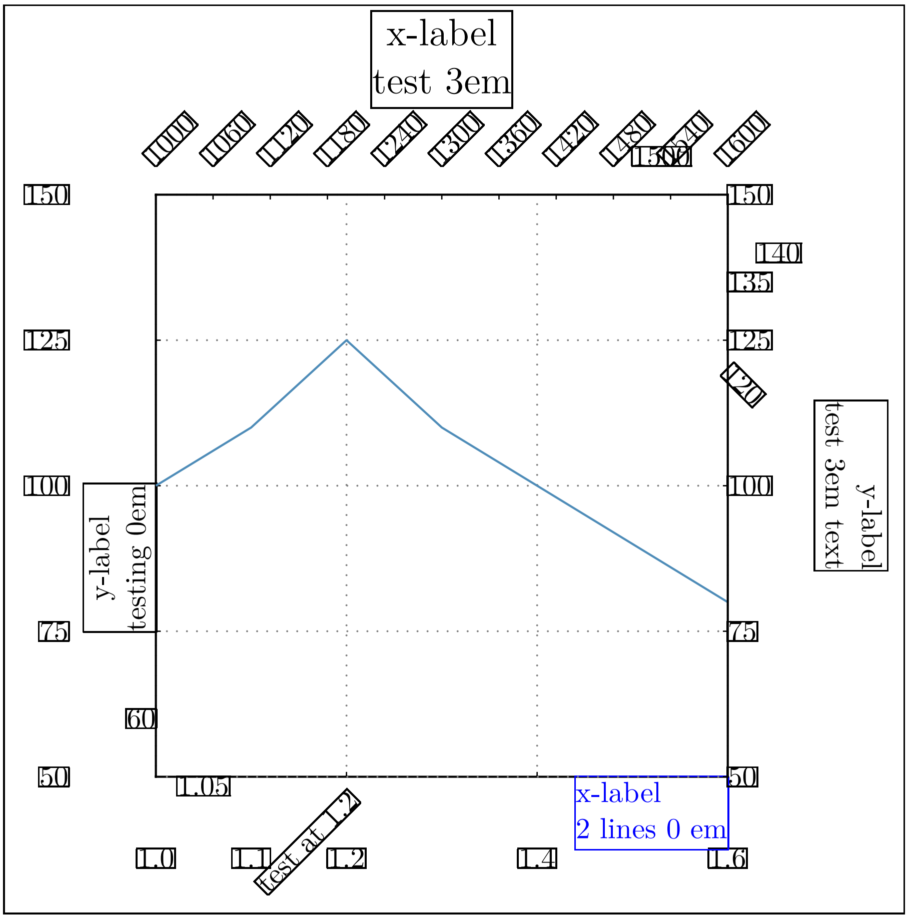
Continued from previous page

Key	Prefix	Family	Type	Default
`TickLabelRefPt`	`NumericDataPlot`	`putxTickLabel`	`ordinary`	`[none]`
`TickLabelRefPt`	`NumericDataPlot`	`putyTickLabel`	`ordinary`	`[none]`
`TickLabelOption`	`NumericDataPlot`	`putxTickLabel`	`command`	`[none]`
`TickLabelRot`	`NumericDataPlot`	`putxTickLabel`	`command`	`[none]`
`TickLabelOption`	`NumericDataPlot`	`putyTickLabel`	`command`	`[none]`
`TickLabelRot`	`NumericDataPlot`	`putyTickLabel`	`command`	`[none]`
`TickLabelSep`	`NumericDataPlot`	`putxTickLabel`	`ordinary`	`\origXTickLabelSep`
`TickLabelSep`	`NumericDataPlot`	`putyTickLabel`	`ordinary`	`\origYTickLabelSep`
`ax`	`NumericDataPlot`	`putxTickLabel`	`choice`	`[none]`
`ax`	`NumericDataPlot`	`putyTickLabel`	`choice`	`[none]`
`nrCols`	`NumericDataPlot`	`Legend`	`ordinary`	`1`
`LegLineWidth`	`NumericDataPlot`	`Legend`	`ordinary`	`20pt`
`LabelOrientation`	`NumericDataPlot`	`Legend`	`choice`	`[none]`

5. Details

5.1. Labels and TickLabels

The following example shows some possibilities to format the axis labels and the tick labels. The example is not intended to be pretty nor useful in any other way than just showing some labels. More examples of customized tick labels are listed in Sec. 6.4.



```
\begin{center}
\renewcommand{\testframe}[1]{\frame{#1}}
\testframe{
\begin{NumericDataPlot}%
  [urx=2.2cm,ury=2.5cm,llx=2.0cm,lly=1.8cm]%
  {0.8\textwidth}{12cm}

  \setxAxis{xMin=1.0, xMax=1.6, Dx=0.2}
  \setyAxis{yMin=50, yMax=150, Dy=25}

  \plotxAxis%
    [TickLabelSep=2\baselineskip, LabelSep=0em, LabelOption=\color{blue},
    LabelPos=1, LabelRefPt=tr, LabelOrientation=l]%
    {x-label\\2 lines 0 em}

  \plotyAxis%
    [TickLabelSep=3em, LabelSep=0em, LabelPos=0.25, %
     LabelRefPt=bl]%
    {y-label\\testing 0em}

  \PutTickLabelYaxis[y=60,TickLabelSep=0pt]{$60$}
  \PutTickLabelXaxis[x=1.1]{$1.1$}
  \PutTickLabelXaxis[x=1.05,TickLabelSep=0em]{$1.05$}
  \PutTickLabelXaxis%
    [x=1.2,TickLabelSep=1ex,TickLabelRot=45,TickLabelRefPt=tr]%
    {test at $1.2$}

  \listplot[style=StdLineStyA]{\IdentI}

  \setxAxis{xMin=1000, xMax=1600, Dx=60}
  \setyAxis{yMin=50, yMax=150, Dy=25}

  \plotxAxis%
    [TickLabelSep=1em, LabelSep=3em, AxisStyle=Upper, NoGrid, %
     LabelOption=\Large,TickLabelRot=45,TickLabelRefPt=bl]%
    {x-label\\test 3em}
  \plotyAxis%
    [TickLabelSep=0em, LabelSep=3em, AxisStyle=Right, NoGrid, %
    LabelOrientation=r]%
    {y-label\\test 3em text}

  \PutTickLabelXaxis[x=1500]{$1500$}
  \PutTickLabelYaxis[ax=right,y=135]{$135$}
  \PutTickLabelYaxis[y=140,TickLabelSep=1em]{$140$}
  \PutTickLabelYaxis[y=120,TickLabelSep=0em,TickLabelRot=-45]{$120$}
\end{NumericDataPlot}
}
%\renewcommand{\testframe}[1]{#1}
\end{center}
```

5.2. Coordinate Systems

This section should just give some hints how to use the different coordinate systems.

A new plot is created with the environment `NumericDataPlot`. The options are illustrated in the following example.

Consider the plot drawn below which is defined by the following code. Note that the example parameters deviate from the standard values to clarify the general possibilities.

```
\begin{NumericDataPlot}
 [xPicMin=300, xPicMax=700, yPicMin=200, yPicMax=700,
  llx=2cm, lly=1.5cm, urx=1cm, ury=1cm]
   {8cm}{6.5cm}
...
% some axes and stuff
...
\end{NumericDataPlot}
```

The total width and total height of the plot object are set with the mandatory arguments of `NumericDataPlot`. These dimensions define the bounding box of the plot object (dashed box in the plot below), as it is seen by the TEX interpreter[2]. The internal, dimensionless coordinate system, in which the actual axes will be placed, is defined by the optional xPicMin, xPicMax, yPicMin and yPicMax values. The location of the horizontal and vertical axes x=xPicMin, x=xPicMax, and y=yPicMin, y=yPicMax, respectively, relative to the bounding box is controlled by the padding values llx, lly, urx and ury. Thus, the actual length Wx of the range xPicMax-xPicMin on the page is calculated by

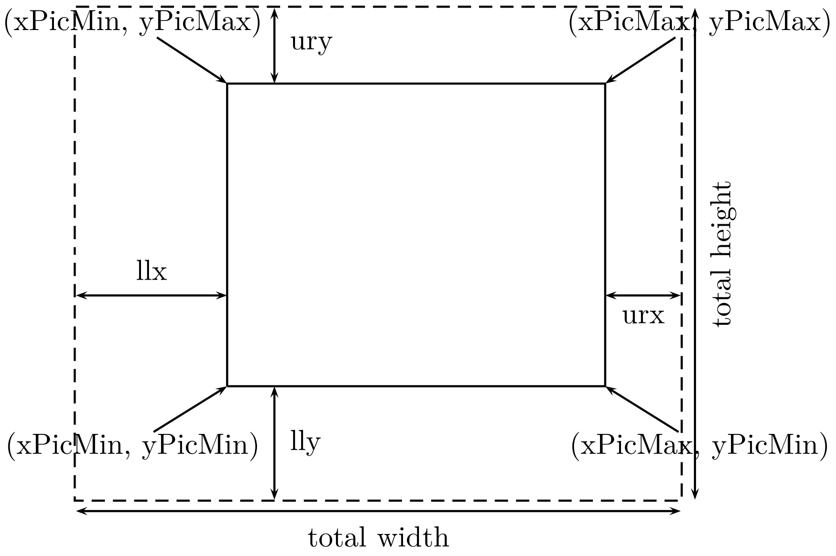
$$\text{Wx} = \text{total width} - \text{llx} - \text{urx},$$

and alike for the y direction. Obviously, the padding values should be positive.

The standard values of the padding values are chosen such that single line axis labels and 3-4 digit tick labels should be within the bounding box.

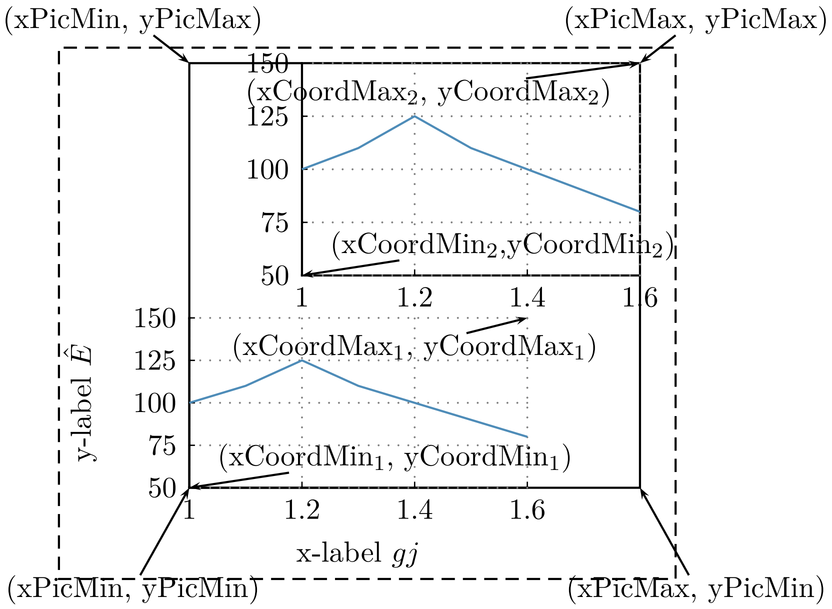
---

[2] Because the elements inside a `NumericDataPlot` environment are set relative to an internal coordinate system, content may be placed outside the bounding box and may overlap with surrounding elements.



In the next example, the placement of multiple axes in the internal coordinate frame is illustrated. The coordinate frame settings are the same as above, but standard padding values are used.

```
\begin{NumericDataPlot}
   [xPicMin=300, xPicMax=700, yPicMin=200, yPicMax=700]
   {0.8\textwidth}{7cm}

% set and draw 1. (lower) axes
\setxAxis{xCoordMin=300, xCoordMax=600, xMin=1, xMax=1.6, Dx=0.2}
 \setyAxis{yCoordMin=200, yCoordMax=400, yMin=50, yMax=150, Dy=25}

\plotxAxis{x-label}
\plotyAxis{y-label}

\listplot[style=StdLineStyA]{\IdentI}

% set and draw 2. (upper) axes
\setxAxis{xCoordMin=400, xCoordMax=700, xMin=1, xMax=1.6, Dx=0.2}
\setyAxis{yCoordMin=450, yCoordMax=700, yMin=50, yMax=150, Dy=25}

\plotxAxis[NoLabel]{}
\plotyAxis[NoLabel]{}

\listplot[style=StdLineStyA]{\IdentI}

\end{NumericDataPlot}
```



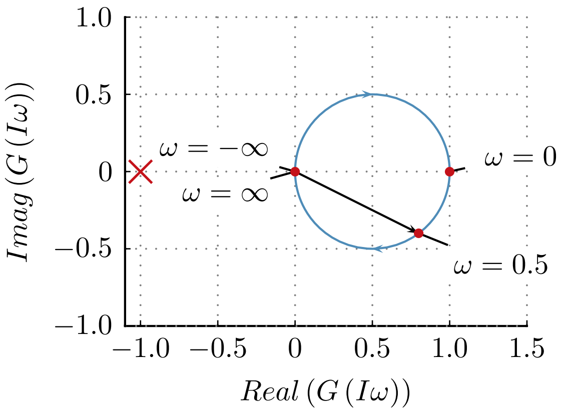
5.3. Undistorted plot

To get an undistorted plot, the following example may be helpful. The width w of the graphic is set manually. The calculation of the total height h is

$$h = \left(\frac{yMax - yMin}{xMax - xMin}\right)\left(\frac{xCoordMax - xCoordMin}{yCoordMax - yCoordMin}\right)$$
$$\left(\frac{yPicMax - yPicMin}{xPicMax - yPicMin}\right)(w - llx - urx) + lly + ury,$$

which simplifies for default values ($xCoordMax = yCoordMax = xPicMax = yPicMax = 1000$, $xCoordMin = yCoordMin = xPicMin = yPicMin = 0$) to

$$h = \left(\frac{yMax - yMin}{xMax - xMin}\right)(w - llx - urx) + lly + ury$$

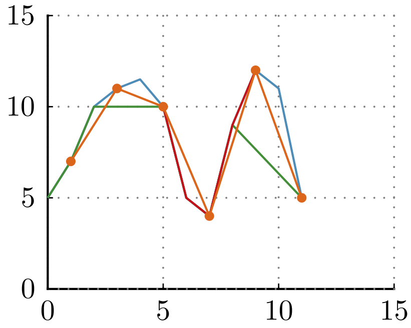


```
\newlength{\GRheight}
\newlength{\GRwidth}

% desired width of the graphic
\setlength{\GRwidth}{0.5\textwidth}
% calculate height:
      % first intermediate step
      % axis width: GRwidth - (llx+urx)
\setlength{\GRheight}%
  {\GRwidth-7ex-\baselineskip-2pt-2ex}
      % sec. intermediate step
      % scale with data range ration
% factor 0.7692 = (yMax-yMin)/(xMax-xMin)
             \setlength{\GRheight}{0.7692\GRheight}
      % third intermediate step
      % add padding values to axis height to obtain total height.
\begin{NumericDataPlot}%
  {\GRwidth}{\GRheight+2\baselineskip+1ex+2pt+0.5em}

  \setxAxis{xMin=-1.1, xMax=1.5, Dx=0.5, x0=0}
  \setyAxis{yMin=-1, yMax=1, Dy=0.5, y0=0}
```

6. Further Examples

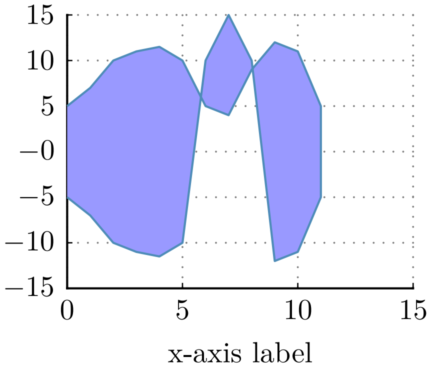
Here, some features of the pst-plot command **`listplot`** are illustrated. With xStart, xEnd (yStart, yEnd), the data can be truncated. Note that all visible data points are connected by a straight line which may render a misleading plot, cf. the blue and green line in the next plot below.

```
\listplot
[style=StdLineStyA]
{\MoreData}
\listplot
[style=StdLineStyB, yEnd=10]
{\MoreData}
\listplot
[style=StdLineStyC, xStart=5, xEnd=9]
{\MoreData}
\listplot
[style=StdLineStyD, showpoints=true, nStep=2]
{\MoreData}
```

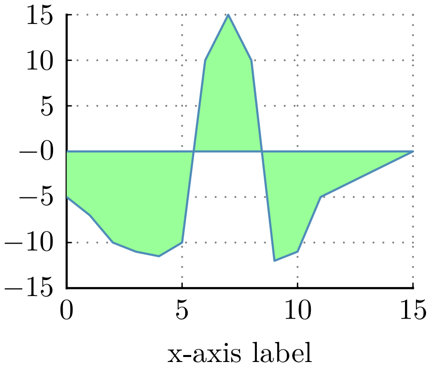


6.1. Fill area between plots

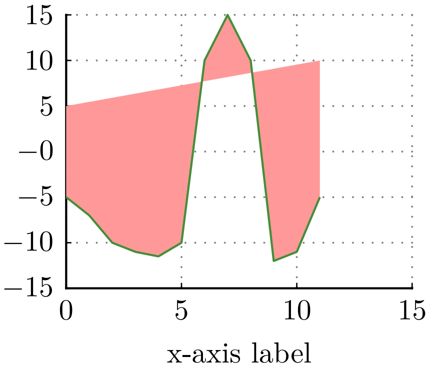
```
\begin{NumericDataPlot}%
{\textwidth}{5cm}
\setxAxis
{xMin=0, xMax=15, Dx=5, x0=0}
\setyAxis
{yMin=-15, yMax=15, Dy=5, y0=10}

\plotxAxis{x-axis label}
\plotyAxis[NoLabel]{}

\pscustom%
[style=StdLineStyA, fillstyle=solid, %
fillcolor=blue!40]{%
\listplot{\MoreData}%
\listplot[ChangeOrder]{\MoreDataI}%
}
\end{NumericDataPlot}
```



```
...
\pscustom%
[style=StdLineStyA, fillstyle=solid, %
fillcolor=green!40]{%
\NDPhline{0}
\listplot[ChangeOrder]{\MoreDataI}%
}
...
```



```
...
\pscustom%
[style=StdLineStyA, fillstyle=solid, %
fillcolor=red!40]{%
\NDPline{0}{5}{11}{10}
\listplot[ChangeOrder]{\MoreDataI}%
}
...
```



6.2. Custom Grid

To plot a custom grid (grid-lines are not equidistant) the following procedure may be used if working with Matlab:

Add the following code after export2latex (XTick is a vector containing the positions of the desired x-gridlines):

```
% export positions of the x-Tick-Marks
fid = fopen(TargetTexFile, 'at');
fprintf(fid, '\n\\def\\CustomXGrid{\n');
for i=1:length(XTick)
    fprintf(fid, ...
     '\\NDPvline[style=CustomXGridStyle]{%f}\n', adj.XTick(i));
end
fprintf(fid, '}\n');
fclose(fid);
```

In Latex just define the style “CustomXGridStyle” and use the command `\CustomXGrid` after defining the axis.

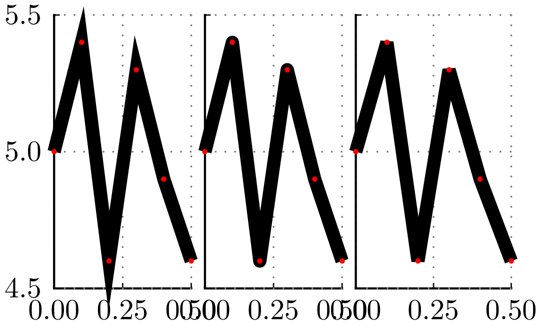
6.3. Plotting noisy data

Plotting measurement data with a lot of noise may be a little tricky. First, you often have a lot of data to plot which may lead to memory problems in TEX. When filtering the data, the appearance of the noise may change which is not what you intend to do. To solve this problem, the two Matlab-functions “LatexFilterMinMax” and “LatexFilterHull” are provided. Both take x- and y-data and divide it in NrOfPoints equally intervals. The filter returns the maximum and the minimum value of each interval.

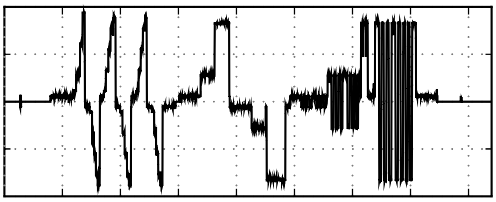
The filter “LatexFilterHull” returns a vector which contains the minima first and then the maxima. This data may be plotted with “fillstyle=solid” as in section 6.1.

The filter “LatexFilterMinMax” returns maximum and minimum value of each interval (with corresponding x-data). This data may be plotted as a “normal line”. As pstricks-lines are normally closed lines, this may lead to a plot where the noise seems exagerated. You may therefore use the option linejoin as demonstrated in the following examples. The red dots mark the actual data. Notice that the measurement data seems noisier than it is if it is plotted with linejoin=0. Plotting with linejoin=2 represents the data much better. The standard for linejoin is 0. The StdLineStyXX defined in this package are defined with linejoin=1.

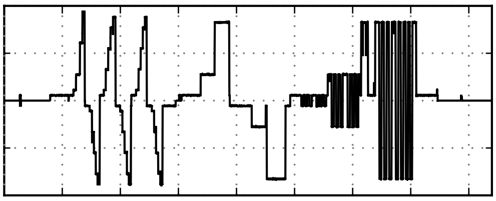
```
...
% linejoin=0:
\listplot[linewidth=5pt]%
{\NoiseData}%
...
% linejoin=1:
\listplot[linejoin=1, %
linewidth=5pt]{\NoiseData}%
...
% linejoin=2:
\listplot[linejoin=1, %
linewidth=5pt]{\NoiseData}%
...
```



linejoin=0 linejoin=1 linejoin=2



linejoin=0

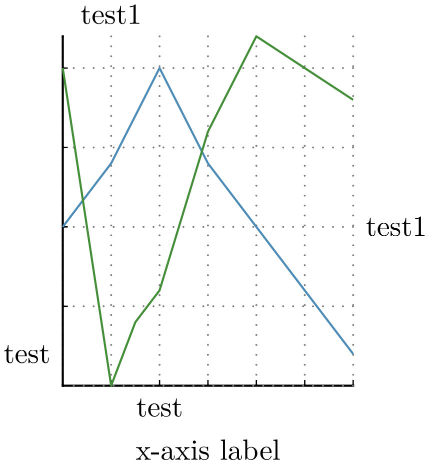


linejoin=2

6.4. Customized Tick Labels

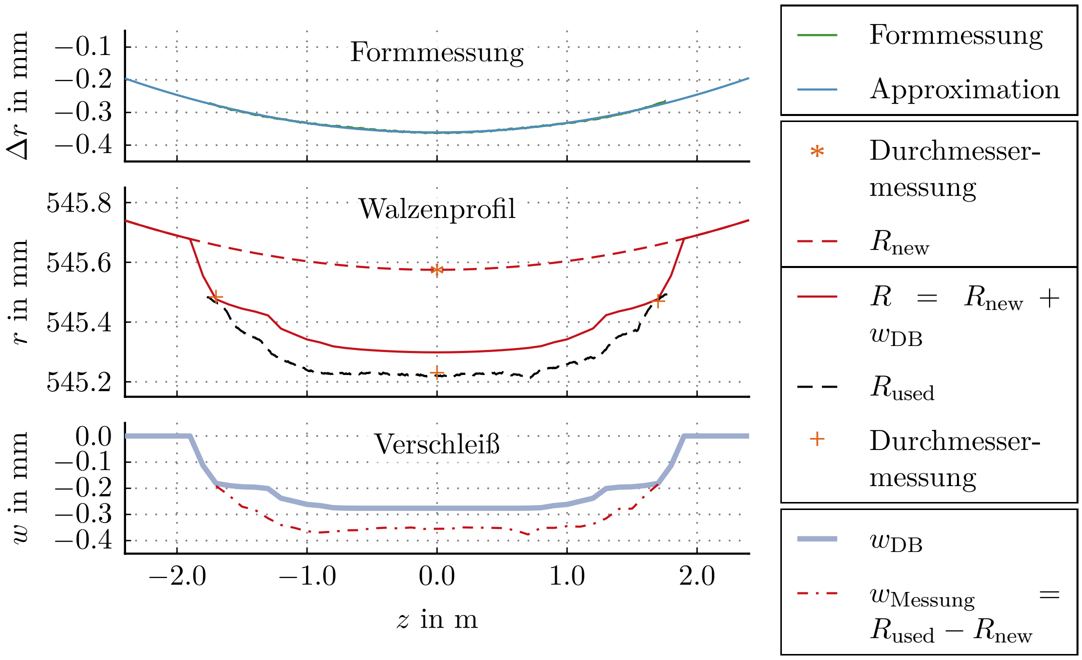
A common task is to plot a graph without numeric tick labels but with selected customized labels. This is easily accomplished by setting the option `NoTickLabel` when plotting the axis and adding the customized labels with the commands `PutTickLabelXaxis` and `PutTickLabelYaxis`. Note that when setting `NoTickLabel` for an axis, the option `ax` has to be set in each call of `PutTickLabelXaxis` and `PutTickLabelYaxis` according to the axis one wishes to place the label at. This is also required if only the lower x-axis or left y-axis are plotted and tick labelled.

```
\plotxAxis[NoTickLabel]{x-axis label}
\plotyAxis[NoLabel, NoTickLabel]{y-
   axis label}

\PutTickLabelXaxis[x=1.2, ax=lower]{
   test}
\PutTickLabelXaxis[x=1.1, ax=upper]{
   test1}

\PutTickLabelYaxis[y=80, ax=left]{test
   }
\PutTickLabelYaxis[y=100, ax=right]{
   test1}
```

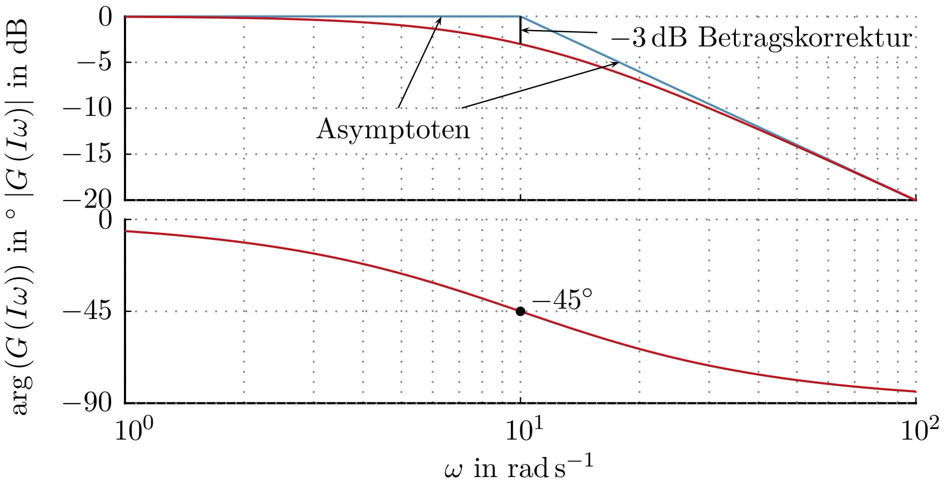


7. Some test plots

7.1. example



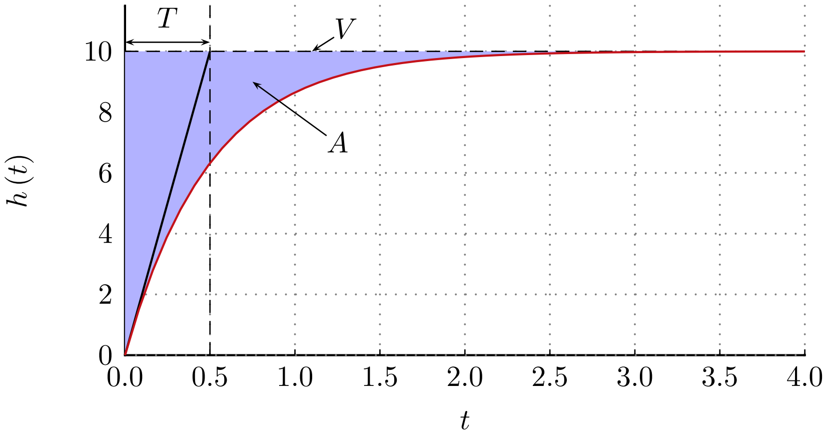
7.2. bode plot



```
\begin{NumericDataPlot}{0.85\textwidth}{6.5cm}
```

```
  \setxAxis{xMin=1, xMax=100, Dx=10, xLog}
  \setyAxis{yMin=-20, yMax=0, Dy=5, y0=0, yCoordMin=525}

  \plotxAxis[NoTickLabel,NoLabel]{}
  \plotyAxis{$\left|G\left(I\omega\right)\right|$ in \si{\decibel}}

  \NDPline[style=StdLineStyA]{0}{0}{1}{0}
  \NDPline[style=StdLineStyA]{1}{0}{2}{-20}

  \listplot[style=StdLineStyC]{\DataBetrag}

  \NDPput[x=1, y=0]{\pnode{Pa}}
  \NDPput[x=1, y=-3]{\pnode{Pb}}
  \ncline{-}{Pb}{Pa}
  \NDPput[x=1, y=-1.5]{\pnode{Pc}}
  \NDPput[x=2, y=-2.5, RefPoint=r]%
    {\Rnode{Pd}{
    \psframebox[fillstyle=solid,fillcolor=white,linestyle=none,framesep=0pt]{%
    \SI{-3}{\decibel} Betragskorrektur}}}
  \pscustom[linewidth=0.5pt]{\ncline[angleA=0]{->}{Pd}{Pc}}

  \NDPput[x=0.8, y=0]{\pnode{Pf}}
  \NDPput[x=1.25, y=-5]{\pnode{Pg}}
  \NDPput[x=0.9, y=-12.5, RefPoint=r]%
    {\Rnode{Pe}{
    \psframebox[fillstyle=solid,fillcolor=white,linestyle=none]%
    {Asymptoten}}}
  \pscustom[linewidth=0.5pt]{\ncline{->}{Pe}{Pf}}
  \pscustom[linewidth=0.5pt]{\ncline{->}{Pe}{Pg}}

  \setyAxis{yMin=-90, yMax=0, Dy=45, y0=0, yCoordMax=475}

  \plotxAxis{$\omega$ in $\si{\radian\per\second}$}
  \plotyAxis{$\arg\left(G\left(I\omega\right)\right)$ in $\si{\degree}$}

  \listplot[style=StdLineStyC]{\DataPhase}

  \listplot[plotstyle=dots]{10 -45}
  \NDPput[x=1, y=-45, RefPoint=lb]{ \SI{-45}{\degree}}

\end{NumericDataPlot}
```

7.3. areas in a plot



```
\begin{NumericDataPlot}{0.75\textwidth}{6cm}

  \setxAxis{xMin=0.0, xMax=4.0, Dx=0.5}
  \setyAxis{yMin=0, yMax=11.5, Dy=2}

  \plotxAxis{$t$}
  \plotyAxis{$h\left(t\right)$}

  % Markiere Fläche
  \pscustom[fillstyle=solid, fillcolor=blue!30, linestyle=none]{
    \listplot{\DataSprungantwort}
    \NDPline{4}{10}{0}{10}
  }

  % Hilfslinien
  \NDPline{0}{0}{0.5}{10}
  \NDPvline[linestyle=dashed, linewidth=0.5pt]{0.5}
  \NDPhline[linestyle=dashed, linewidth=0.5pt]{10}

  % plotte Sprungantwort
  \listplot[style=StdLineStyC]{\DataSprungantwort}

  % Beschriftung
  \NDPput[x=0,y=10.3]{\pnode{A}}
  \NDPput[x=0.5,y=10.3]{\pnode{B}}
  \ncline[linewidth=0.5pt]{<->}{A}{B}
  \naput{$T$}

  \NDPput[x=0.75, y=9]{\pnode{F11}}
  \NDPput[x=1.25, y=7]{\Rnode{F12}{$A$}}
  \ncline[linewidth=0.5pt]{->}{F12}{F11}
  \NDPput[x=1.1,y=10]{\pnode{V1}}
  \NDPput[x=1.3, y=10.75]{\Rnode{V2}{$V$}}
  \ncline[linewidth=0.5pt]{->}{V2}{V1}

\end{NumericDataPlot}
```

Part III.
Version

8. History

- 18.05.2010
  - added LabelOption to plotAxis
  - added TickLabelOption to plotAxis
  - added possibility to change StdLabelOption
  - added possibility to change StdTickLabelOption
- 11.06.2010
  - expanded documentation
  - changed the command `LegLine`.
  - added the commands `plotxGrid` and `plotyGrid`. The grids may now be plotted before the axis to avoid overlapping of the grid and the axis.
- 17.06.2010
  - changed export2latex: check for the necessary precision before exporting the data
- 21.06.2010
  - added option for logarithmic axes
- 05.07.2010
  - added command NDPline
  - added instructions to fill areas between plots to documentation
- 03.08.2010

  - added options to `export2latex.m`
- 05.08.2010
  - new calculation of tick lengths
  - length of the ticks may now be given in mm, see 4.1
- 14.09.2010
  - added `struct2latex` to the package
  - added `dspace2struct` to the package
  - added `dspace2latex` to the package
  - changed `struct2latex`: mapping column vectors
  - changed `struct2latex`: downsampling option added
- 15.11.2010
  - changed the lineends of the axis
- 07.01.2011
  - adjusted linestyles
  - example with the available linestyles
- 04.05.2011
  - added commands to place exponents at the axes
  - added documentation for put North, . . . , SouthWest
- 14.05.2011
  - added command to put a box in the graph
- 17.05.2011
  - added ps-style LegendBoxStyle to adjust the style of the box underlying the legend

- 12.07.2011
    - changes in documentation
    - adopted `dspace2struct` for dSpace Next Generation
    - added `\NDPvline`, `\NDPhline`, `\NDPvbox` and `\NDPhbox`
- 01.06.2012
    - changes in documentation
    - general bug fixing
    - bug fixing for `dspace2struct`
    - suppressing notification of the calculations made by the fp package during the compilation.
- 14.06.2012
    - added rotation option for `\NDPput` (Rot=45)
- 03.08.2012
    - added documentation for plotting noisy data
- 23.08.2012
    - added command `\LegDot`
- 29.08.2012
    - added command `\multilistplot`
    - added `options.NaNsplit` to export2latex
- 30.08.2012
    - added options DDx and DDy for minor grid
    - added option LabelBP
- 19.03.2013

  - added Customized Tick Labels

- 11.04.2013
  - Corrected calculation of bounding box
    - removed additional horizontal phantom space
    - Previously, the padding values lly and ury where added to the given height parameter to set the object height. Now, the given height parameter is correctly set as height of the bounding box.
    - new standard values for padding parameters lly, llx, ury, urx, which now take into account the current font size settings (unit `em`, `\baselineskip`)
  - added more detailed documentation of the bounding box and coordinate frame behavior
- 16.04.2013
  - Changed positioning of axis labels and tick labels and added more options for customization.
  - Changed option LabelBP to LabelRefPt.
  - Changed options xLabelSep and yLabelSep to LabelSep (with different meaning).
  - Changed options xTickLabelSep and yTickLabelSep to TickLabelSep (with different meaning).
- 19.04.2013
  - added and improved documentation
- 25.04.2013
  - changed implementation of subgrids for logarithmic axes
- 04.06.2013
  - corrected and added documentation concerning the use of `PutTickLabelXaxis` and `PutTickLabelYaxis` for customized tick labels

- 18.09.2013
  - corrected error ocurring for xLog and yLog labels. In some occasions Latex wrote 10^{-0} instead of 10^{0}.