The numodel-plot package*

Paul Zuurbier
mail@paulzuurbier.nl

May 26, 2026

Abstract

A PGFPlots engine that auto-sizes plots to a whole number of tick intervals, supports configurable axis-label formats (IEEE-style by default; ISO 80000-1 also supported), and automatically selects label placement for 1-, 2-, and 4-quadrant graphs. Part of the numodel package suite, but can be loaded standalone as an independent PGFPlots styling layer.

Contents

1 Introduction

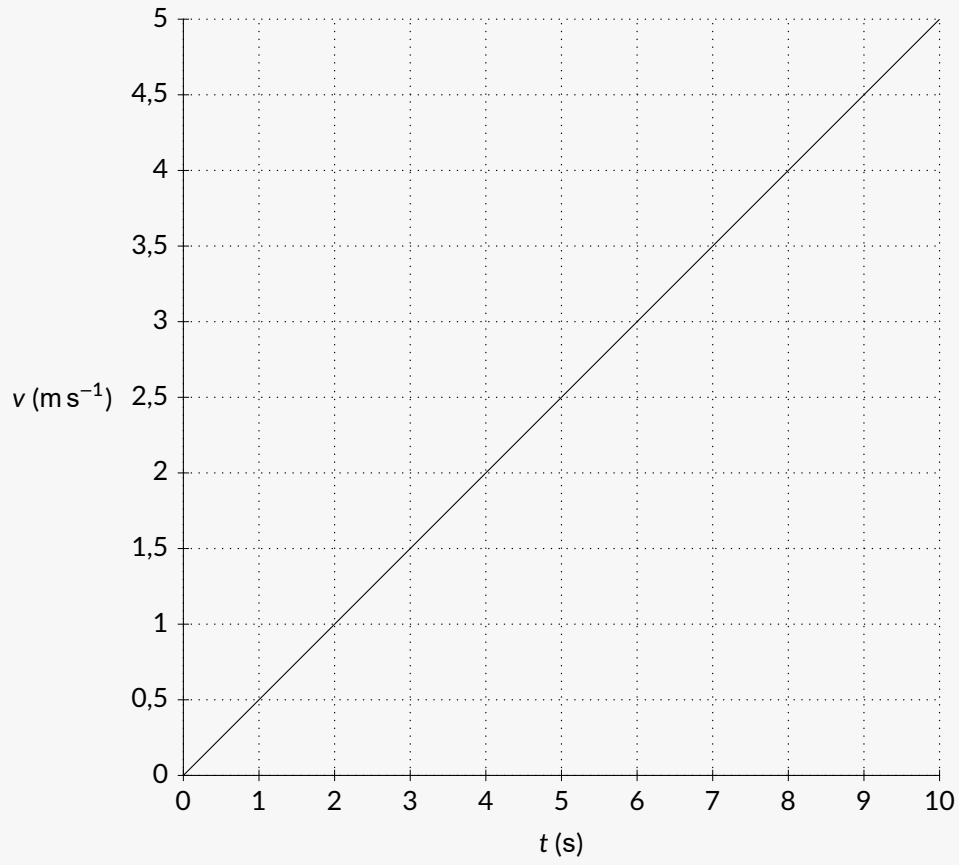
numodel-plot fills a gap between bare PGFPlots and the heavy styling required for physics-teaching material: it sizes every axis to an integer number of centimetre ticks, lays out the axis origin according to which quadrants of the coordinate plane contain data, and renders axis labels as either `quantity (unit)` (IEEE, the default) or one of four alternative conventions selectable at package level. It was extracted from a Dutch high-school physics test set where uniform plot appearance across hundreds of graphs is more valuable than per-graph tweaking, and hence adopts an opinionated default style. Users who need one-off deviations are expected to drop to plain PGFPlots with the variable macros `\xmin`, `\xmax`, … exposed by this package.

*This document corresponds to numodel-plot v0.6.0, dated May 26, 2026.

2 Usage

Minimum working example (assuming `\usepackage{numodel-plot}` in the preamble):

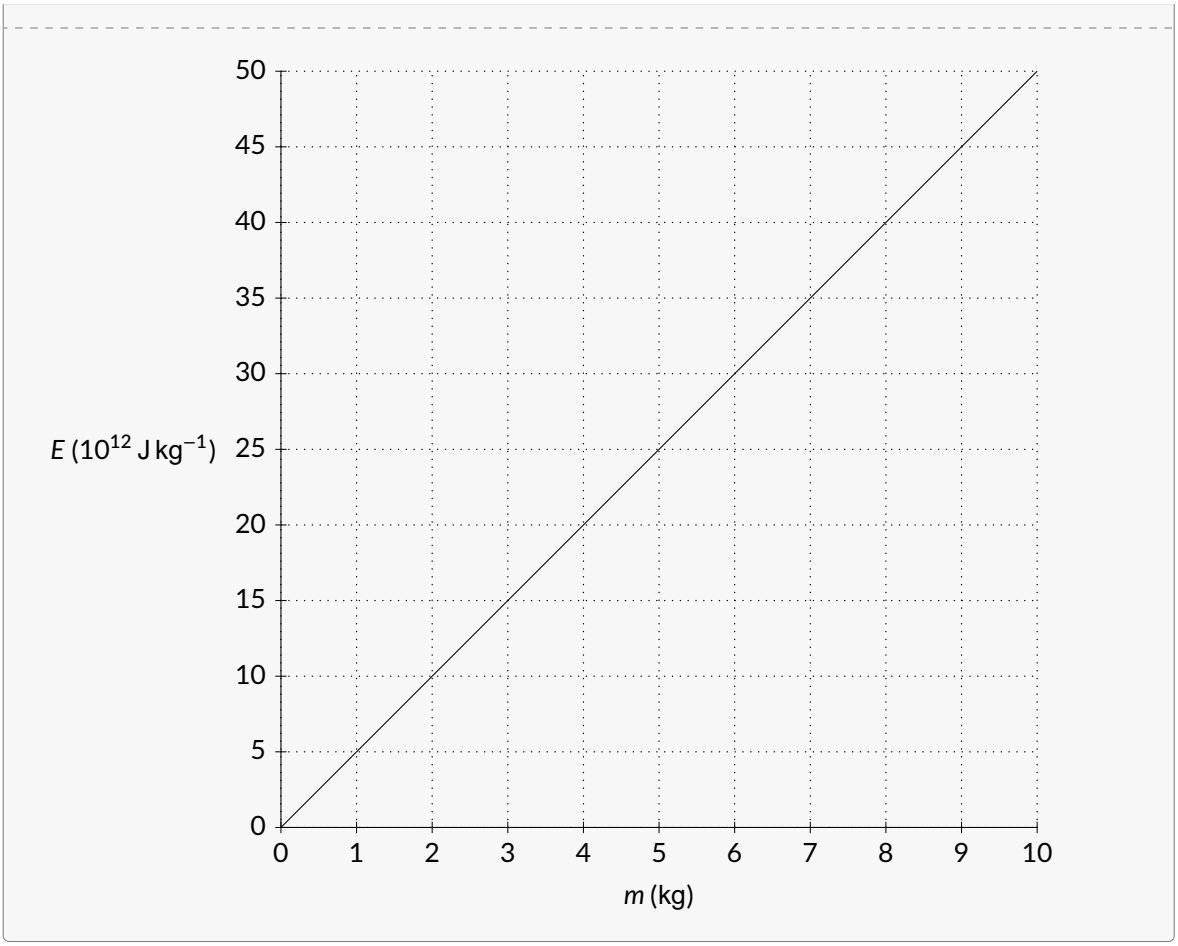
```
\def\xmin{0}   \def\xmax{10}
\def\ymin{0}   \def\ymax{5}
\def\xlabelqty{t}  \def\xlabelunit{\s}
\def\ylabelqty{v}  \def\ylabelunit{\m\per\s}
\drawplot{\addplot[domain=\xmin:\xmax]{0.5*x};}
```



The user sets the data range (`\xmin`…`\ymax`) and optionally a quantity symbol plus siunitx unit for each axis. `\drawplot` internally calls `\calcplotdims` to round the range to a clean tick lattice and compute the axis size in centimetres, then renders a full `tikzpicture+axis` environment whose body is the argument (one or more `\addplot` lines).

Labels are built automatically from `\xlabelqty+\xlabelunit` (and likewise for the y-axis). If the data magnitude exceeds 10^4 or is below 10^{-2}, a factor 10^n is injected into the label and PGFPlots' own `scaled ticks` are configured so that tick numbers remain small. In the next example the data magnitude is 5×10^7 and the unit `\mega\joule\per\kilo\gram` already carries two engineering prefixes; the package extracts every prefix and combines them with the magnitude into a single power-of-ten factor:

```
\def\xmin{0}   \def\xmax{10}
\def\ymin{0}   \def\ymax{5e7}
\def\xlabelqty{m}  \def\xlabelunit{\kilo\gram}
\def\ylabelqty{E}  \def\ylabelunit{\mega\joule\per\kilo\gram}
\drawplot{\addplot[domain=\xmin:\xmax]{5e6*x};}
```



Users preferring full control can omit `\xlabelqty`/`\xlabelunit` and set `\xlabel`/`\ylabel` directly; the package will use them verbatim.

3 Configuration

`\numodelplotsetup` Configuration is set through a single key–value interface:

```
\numodelplotsetup{axis-label-format=ieee, grid=mm-dots}
```

3.1 Keys

`axis-label-format` Default `ieee`. Determines the notation emitted for axis labels built from `\xlabelqty` and `\xlabelunit`:

`ieee`	`v (m/s)`	IEEE (default)
`iso`	`v / (m/s)`	ISO 80000-1
`brackets`	`v [m/s]`	older physics convention
`qty-only`	`v`	quantity symbol only
`unit-only`	`m/s`	unit only

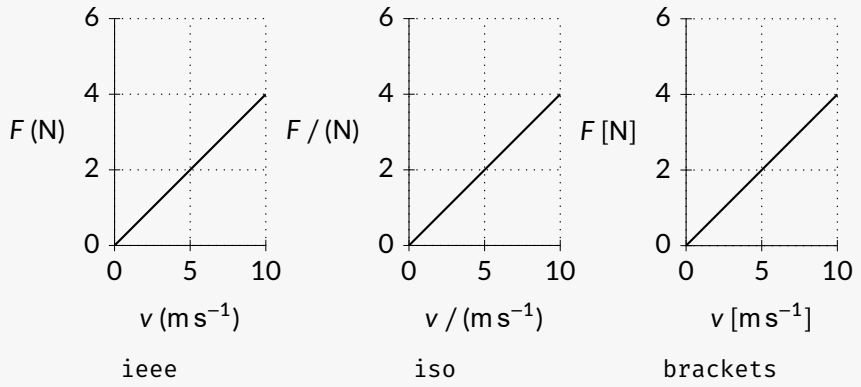
When scaling is applied (data exceeds 10^4 or below 10^{-2}), the factor is integrated into the label, e.g. `v (10^4 m/s)` for IEEE. Under `qty-only` the exponent remains in PGFPlots' scaled-tick label instead (otherwise the scale information would be lost).

`grid` Default `mm-dots` (black millimetre dots, matching engineering millimetre paper). Accepts `none`, or any PGFPlots style list which will be passed verbatim to the `numodel/grid` style.

`xcmmax`, **`ycmmax`** Maximum axis width and height in centimetres (defaults 12 and 10).

The first three axis-label formats render as follows. Each plot uses `\numodelplotsetup{xcmmax=3, ycmmax=3}` so the axis itself is trimmed to a 2 cm by 3 cm tick lattice (the package's invariant 1 cm major-grid spacing is preserved):

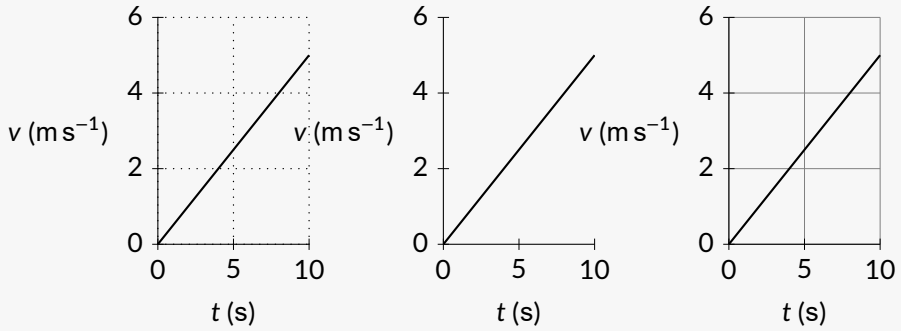
```
\captionsetup{type=figure}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3, axis-label-format=ieee}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{v}\def\xlabelunit{\m\per\s}%
\def\ylabelqty{F}\def\ylabelunit{\N}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.4*x};}
\caption*{\texttt{ieee}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3, axis-label-format=iso}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{v}\def\xlabelunit{\m\per\s}%
\def\ylabelqty{F}\def\ylabelunit{\N}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.4*x};}
\caption*{\texttt{iso}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3, axis-label-format=brackets}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{v}\def\xlabelunit{\m\per\s}%
\def\ylabelqty{F}\def\ylabelunit{\N}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.4*x};}
\caption*{\texttt{brackets}}
\end{subfigure}
```



Three grid variants, sized the same way:

```
\captionsetup{type=figure}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3, grid=mm-dots}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{mm-dots}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3, grid=none}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{none}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\numodelplotsetup{xcmmax=3, ycmmax=3,
```

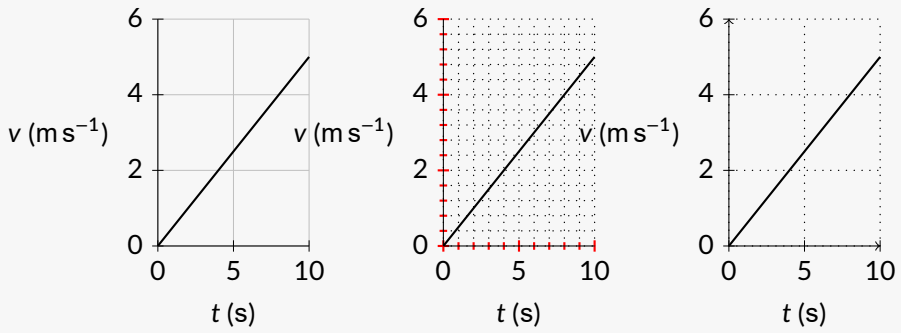
```
  grid={grid=major, grid style={gray, very thin}}}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{\{major,gray\}}}
\end{subfigure}
```



3.2 PGFPlots styles

The package defines three PGFPlots styles applied by `\drawplot`: `numodel/grid`, `numodel/ticks`, `numodel/axis`. These can be overridden wholesale through `\pgfplotsset{numodel/axis/.style={...}}` from the calling preamble, giving projects a single choke point for house-style customisation. One override per style, on the same plot:

```
\captionsetup{type=figure}%
\begin{subfigure}{0.33\textwidth}\centering
\pgfplotsset{numodel/grid/.style={grid=major,
  grid style={gray!50, very thin}}}%
\numodelplotsetup{xcmmax=3, ycmmax=3}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{numodel/grid}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\pgfplotsset{numodel/ticks/.style={tick style={red, thick},
  minor tick num=4}}%
\numodelplotsetup{xcmmax=3, ycmmax=3}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{numodel/ticks}}
\end{subfigure}\hspace{-1cm}%
\begin{subfigure}{0.33\textwidth}\centering
\pgfplotsset{numodel/axis/.append style={axis line style={->}}}%
\numodelplotsetup{xcmmax=3, ycmmax=3}%
\def\xmin{0}\def\xmax{10}\def\ymin{0}\def\ymax{5}%
\def\xlabelqty{t}\def\xlabelunit{\s}%
\def\ylabelqty{v}\def\ylabelunit{\m\per\s}%
\drawplot{\addplot[domain=\xmin:\xmax,thick]{0.5*x};}
\caption*{\texttt{numodel/axis}}
\end{subfigure}
```



4 Public API

`\drawplot` Renders a `tikzpicture` containing an `axis` whose body is the single mandatory argument. Typically a block of `\addplot` and `\addlegendentry` lines. Calls `\calcplotdims` internally, so the user does not need to invoke it separately.

`\calcplotdims` Reads `\xmin`, `\xmax`, `\ymin`, `\ymax`, and (if set) `\xlabelqty`/`\xlabelunit`/`\ylabelqty`/`\ylabelunit`. Writes `\xcm`, `\ycm`, `\xtickdistance`, `\ytickdistance`, `\xlabel`, `\ylabel`, and may rewrite `\xmin`…`\ymax` to align with the tick lattice (floor/ceil to the nearest tick). It also appends axis-positioning styles to `numodel/axis` based on which quadrants the data occupies. `\drawplot` invokes it automatically; expose for advanced cases where dimensions are needed before rendering (overlay TikZ, custom `axis` environment).

`\xlabelqty` `\xlabelunit` `\ylabelqty` `\ylabelunit` Input hooks for automatic label construction. `\xlabelqty` is the mathematical quantity symbol (e.g. `v`, `F`, `E`); the corresponding `\xlabelunit` is a bare siunitx unit macro sequence (e.g. `\m\per\s`, `\J`, `\N\m`) *without* a surrounding `\si{}` or `\qty{}` wrapper.

`\xcmmax` `\ycmmax` Maximum axis dimensions in centimetres. Can be set directly through `\def` for backwards compatibility, or via `\numodelplotsetup`.

`\qtyPlain` Like siunitx's `\qty` but prints no numeric mantissa when the final output after prefix extraction has mantissa 1 and exponent 0. Used internally to inject scale factors into axis labels; exposed because the same need recurs in other scaled-axis contexts.

`pzuIfUnitNonEngTF` Boolean conditional testing whether a unit macro sequence contains a non-engineering SI prefix (`\centi`, `\deci`, `\deca`, `\hecto`, plus the siunitx short forms `\cm`, `\dm`, `\hPa`, …). Used internally to suppress scaling on units where the user has already encoded the order of magnitude; exposed for completeness.

5 Requirements

numodel-plot requires expl3, xparse, l3keys2e, siunitx (mandatory, for quantities in labels), and pgfplots (with the `fillbetween` library). LuaLaTeX is not required at the plot layer (it is required by the sibling numodel package).