

TeXlattice - draw accelerator lattices with L^AT_EX

using pgf/tikz

Jan Schmidt <schmidt@physik.uni-bonn.de>

v2.1 (January 19, 2015)

Contents

1	Installation	2
1.1	Copy lattice.sty	2
1.2	Required packages	2
2	What is missing?	2
3	Known issues	2
4	lattice environment	3
5	Within lattice environment	3
5.1	Elements	3
5.2	Modify your lattice/elements	4
5.2.1	Labels	4
5.2.2	Colors	4
5.3	Access lattice coordinates	5
6	Remarks	5

1 Installation

1.1 Copy lattice.sty

You just need to copy the lattice.sty file to a place where your L^AT_EX installation can recognize it. This can be

- the same folder as your .tex document
- in the L^AT_EX system or user tree

e.g. to add it to the system tree for texlive under ubuntu:

```
sudo mkdir -p /usr/local/share/texmf/tex/latex/lattice/  
sudo cp lattice.sty /usr/local/share/texmf/tex/latex/lattice/  
sudo mktexlsr (or sudo texhash)
```

For this path there is also a Makefile prepared, so just enter

```
sudo make install
```

Otherwise read the documentation of your L^AT_EX distribution.

1.2 Required packages

- tikz, pgf
- siunitx
- ifthen
- xargs

2 What is missing?

- manually adding (and maybe changing?) legend entries
- The look of the elements can be improved
- More element types can be added
- ...

3 Known issues

- no error handling implemented
- no dedicated scoping of internal macros (use of lattice with documentclass standalone recommended)
- please report bugs...

4 lattice environment

To draw a lattice just add

```
\usepackage{lattice}
```

to your preamble and use the lattice environment. the lattice environment has 2 optional arguments:

1. `[tikz options]` give any options for the tikzpicture (e.g. `overlay`)
2. `[scale]` scale whole picture (default: 1)

5 Within lattice environment

5.1 Elements

- `\drift{length/m}[name (default: none)]`
- `\dipole{name}{arc length/m}{bending angle/deg}[type (default: s)][thickness/m (default 0.6)]`
the `type` option allows to select different dipole shapes. It can be:
 - `br` for a bend rectangle magnet (parallel entrance/exit surfaces)
 - `r` for a rectangle magnet
 - `s` for a sector magnet (entrance/exit surface 90 degree to beampipe)

If you use any other letters, also the default (`s`) is used.

- `\quadrupole{name}{length/m}[thickness/m (default 0.5)]`
- `\sextupole{name}{length/m}[thickness/m (default 0.3)]`
- `\corrector{name}{length/m}[thickness/m (default 0.25)]`
- `\kicker{name}{length/m}[thickness/m (default 0.25)]`
- `\cavity{name}{length/m}[thickness/m (default 0.45)]`
- `\solenoid{name}{length/m}[thickness/m (default 0.2)]`
- `\beamdump{name}{length/m}[thickness/m (default 0.5)]`
- `\source{name}{length/m}[thickness/m (default 0.5)]`
- `\screen{name}[length/m (default 0.2)]`
- `\valve{name}`
- `\marker{name}[length/m (default 0.35)]` a line perpendicular to beamline of given length

5.2 Modify your lattice/elements

- `\rotate{angle/deg}` "bends" the beamline., offset from current angle
- `\setangle{angle/deg}` "bends" the beamline., set absolute angle
- `\goto{coordinate name}` sets current position and angle to values saved with `\savecoordinate`
hint: e.g. use to draw injection/extraction
- `\start{coordinate/m}` sets starting point of lattice. use before first element coordinate in form (x,y) or any tikz label, e.g. (mylabel.east) hint: use with `\savecoordinate` to connect lattices! (compile twice!)
- `\drawrule{position/m}[tick distance/m (default: 1)]...`
`[scale (default: 1)][height/m (default:0.1)]`
a rule to visualize lattice size. coordinate in form (x,y) or any tikz label, e.g. (mylabel.east)
- `\legend{position/m}[scale (default: 1)]` a legend with all element types that occur in the lattice before this command. position is north west (upper left corner) of the legend box. the scale option scales the whole box including the text, which has the usual label textsize for scale=1
- `\completelegend{position/m}[scale (default: 1)]` similar to `\legend`, but shows all existing element types.

5.2.1 Labels

- `\turnlabels` moves labels to other side of elements (swap with marker labels)
- `\rotatelabels{angle/deg}[anchor (default: automatic)]` allows rotation of element labels. the anchor sets the center of rotation (north, center, south west, ...). west corresponds to labels first character.
- environment `{labeldistance{distance/m}}` sets distance of text labels to element center for all elements within this environment (default is 0.35)
- `\setlabeldistance{fontsize}` sets distance of text labels to element center for following elements (default is 0.35)
- `\resetlabeldistance{fontsize}` resets distance of text labels to element center for following elements to default 0.35
- `\setlabelfont{fontsize}` text label fontsize (default is `\normalsize`)

5.2.2 Colors

The color can be changed at any point in the lattice. A setting is valid until the next color setting command.

- `\setlabelcolor{color}` for textlabels (set to white to hide labels).
- `\setlinecolor{type}{color}` for type drift and marker.

- `\setelementcolor{type}{color}[gradient color (default: white)]` for all element types. set gradient color = color to "disable" gradient
- `\resetlinecolor{type}` reset to default color.
- `\resetelementcolor{type}` reset to default color.
- environment `\begin{fade}[opacity (default: 0.25)]` sets the opacity of all elements within the environment to fade out regions of the lattice - e.g. for presentations. Also sets all colors to gray. This can also be used to completely hide regions by setting opacity to zero.

5.3 Access lattice coordinates

You can use element coordinates to draw anything you want using pgf/tikz. You can even connect lattices to draw injection/extraction or a complete accelerator facility.

- `\savecoordinate{name}[position (default: east)]` saves coordinate of previous element to access it later.
 - position specifies the exact place of the element. East (default) and center are available. East is always downstream.
 - you can use all tikz/pgf commands within lattice environment to draw anything.
 - You can use this to connect multiple beamlines within a lattice environment with `\goto{name}`
 - You can use this to connect multiple lattices with `\start{name}`. use tikz overlay option (1. argument of lattice)
 - see example 3

6 Remarks

- lengths are set in meter, so you write `{1.32}` for 1.32m.
- beamline with angle 0 goes to the right, positive angles bend counter clockwise.
- settings (colors, font, rotatelabel,...) changed within a `scope` environment are set back to the previous values outside of `scope`
- picture scale: for lattice scale=1 an element of 1m length is plotted with 2cm length
- if you refer to a coordinate from another lattice (another tikzpicture) you have to compile twice