

Polyhedra L^AT_EX package - v. 0.1

Zafeirakis zafeirakopoulos

July 29, 2015

Abstract

This package provides macros for creation of polyhedral objects in 2D and 3D. It is build on top of Tikz and tikz-3dplot. The macros provided can be used for drawing vertices, edges, rays, polygons and cones.

1 Introduction

2 Functionality

2.1 The polyhedron environment

2.2 Input

The following macros work within the `polyhedron` environment.

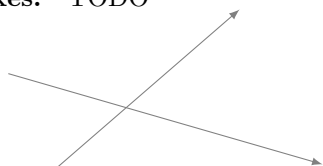
Point. To let A be a point (x,y) or (x,y,z)

`\point{(x,y,z)}{A}`

After defining a point, we can use the handle (variable name) A, instead of writing down the coordinates in the macros that follow.

2.3 Drawing commands

Axes. TODO



Vertex. To draw a vertex at a point (x,y,z) or A

`\vertex{x,y,z}`

`\vertex{A}`

The vertex macro takes a second optional argument, which may have any of the values:

normal This is the default point. If the argument is not passed, the result is the same.

```
\vertex {0,0}[normal]
```

focus This argument makes the point a focus point. The focus style is global

```
\vertex {1,0}[focus]
```

alert This argument makes the point an alert point. The alert style is global

```
\vertex {2,0}[alert]
```

empty This argument makes the point empty. Currently this mean the point is fill with white.

```
\vertex {3,0}[empty]
```

The second argument can also be any already defined color. In that case the vertex is filled by that color, e.g.,

```
\vertex {4,0}[orange]
```

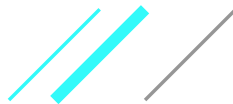
In the following polyhedron environment we see the five points defined above.



Edge. To draw an edge from point A to point B

For edges, in addition to the focus and alert options we had for vertices, we also have the *dashed* and *dashedAlert* options. A fourth optional argument is available to define the line width of the edge.

```
\begin{polyhedron}[2d]
  \edge {0,0}{2,2}[][normal]
  \edge {1,0}{3,2}[][normal][4.4]
  \edge {2,0}{4,2}[][alert]
  \edge {3,0}{5,2}[][focus]
  \edge {4,0}{6,2}[][dashed]
  \edge {5,0}{7,2}[][dashedAlert]
  \edge {6,0}{8,2}[][purple]
\end{polyhedron}
```



Ray. To draw a ray from point A to point B The options are exactly the same as with ray.

```
\begin{polyhedron}[2d]
  \ray {0,0}{2,2}[normal]
  \ray {1,0}{3,2}[normal][4.4]
  \ray {2,0}{4,2}[alert]
  \ray {3,0}{5,2}[focus]
  \ray {4,0}{6,2}[dashed]
```

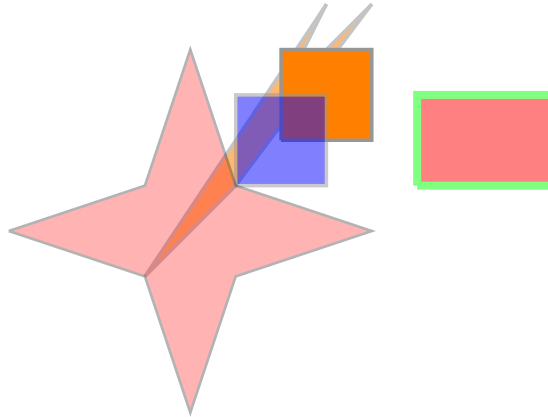
```

\ray{5,0}{7,2}[dashedAlert]
\ray{6,0}{8,2}[purple]
\end{polyhedron}

```



Polygon. To draw a polygon given as a list of vertices. Attention must be paid in the sequence of the points given. The command does not take the cinvex hull, but rather draws the polygonal line defined by the sequence of points. The first point is also assumed to be the last point, so that the polygonal line is a closed curve. The macro takes two arguments, the first is a list of points defining the polygonal line. The second argument is the last point of the polygonal line. It is expected (but not forced) to be the same as the first point given in the first argument. The optional third argument is the fill color for the polygon (assuming the polygonal line is closed). The fourth argument is the opacity (between 0 and 1). The last two optional arguments are the color of the and the thickness of the contour.



```

\begin{polyhedron}[2d]
\polygon{{3,5},{-1,-1},{1,1},{4,5},{2,3}}{3,5}[orange][0.56]
\polygon{{2,2},{2,4},{4,4},{4,2}}{2,2}[orange][1]
\polygon{{1,1},{1,3},{3,3},{3,1}}{1,1}[blue][0.5]
\polygon{{5,1},{5,3},{8,3},{8,1}}{5,1}[red][0.5][green][3]
\polygon{{-4,0},{-1,-1},{0,-4},{1,-1},{4,0},{1,1},{0,4},{-1,1}}
{-4,0}[red][0.3][black][1]
\end{polyhedron}

```

3 Known limitations and errors