Manual

April 14, 2021

The package consists of `graphdrawingpackage.sty` and `spectralcppcode.cpp` files, and requires LuaLaTeX compilation version 1.07.0 and TikZ package to be installed. Following packages must be included: `tikz`, `graphdrawingpackage`. The used commands should be included in the scope of `tikzpicture`:

```
usepackage{tikz}
usepackage{graphdrawingpackage}
...
\begin{tikzpicture}
\PathGraph{4}{}{}
\end{tikzpicture}
```

The compilation is done with LuaLaTeX, in following command in Windows10:

```
lualatex <texfile.tex>
```

However if to use the function using c++ code, it must be compiled with `--shell-escape` flag. The `spectralcppcode.cpp` file will automatically compiled with `g++` compiler, and executed on OS cmd. Currently it is set up to only work for Windows10 cmd.

```
lualatex --shell-escape <texfile.tex>
```

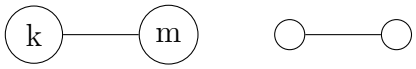
Therefore it needs `g++` compiler to be installed, and also need additional cpp package, '`eigen`' to be included. When you download the package from `http://eigen.tuxfamily.org`, take the file '`Eigen`' in the package directory(for example, `/eigen-3.3.9/Eigen`) and put it on the location where compiler can `#include` them in the program(in directory `/include` for instance).

The node is drawn by `\node` command and edge by `\draw` command in TikZ. Options like `nodestyle` and `edgestyle` are applied to these TikZ macros.

PathGraph[*<option>*]{*<number>*}{*<lables>*}{*<edges>*}
<number>: number of nodes (must have)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<edges>: list of user specific edges (leave it blank if not needed, by default cyclic edges), using node index to indicate nodes. (able to check with 'nodename' options)
<option>

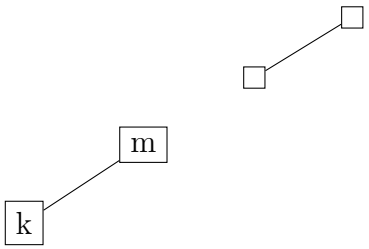
nodename: flag to indicate node names as labels (default=disabled)
direction: nodes drawing direction (default=right) (possible options: below, right, left, above, and combination of these for diagonal directions ex: below right)
vertdistance: node distance in vertical way (default=1cm)
horidistance: node distance in horizontal way (default=1cm)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)

```
\begin{tikzpicture}
\PathGraph{4}{}{}
\end{tikzpicture}
```




```
\begin{tikzpicture}
\PathGraph{4}{k,m}{{1,2},{3,4}}
\end{tikzpicture}
```


```
\begin{tikzpicture}
\PathGraph[nodename,nodeonly,prefix=p]{4}{k,m}{{1,2},{3,4}}
\end{tikzpicture}
```



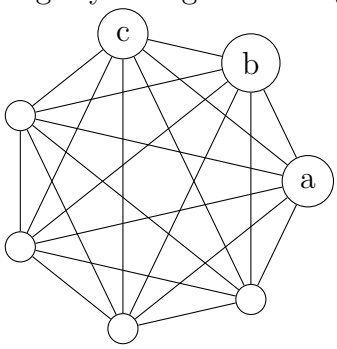

```
\begin{tikzpicture}
\PathGraph[direction=above right,nodestyle={draw,rectangle},vertdistance=5mm]{4}{k,m}{
\end{tikzpicture}
```

CycleGraph[*<option>*]{*<number>*}{*<lables>*}{*<edges>*}
<number>: number of nodes (must have)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<edges>: list of user specific edges (leave it blank if not needed, by default cyclic edges), using node index to indicate nodes. (able to check with 'nodename' options)

<option>
nodename: flag to indicate node names as labels (default=disabled)
radius: cycle graph's radius (default=3)
startangle: cycle graph's first node's angle in degree (default=0)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)

StarGraph[*<option>*]{*<number>*}{*<lables>*}{*<edges>*}
<number>: number of nodes (must have)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<edges>: list of user specific edges (leave it blank if not needed, by default star edges), using node index to indicate nodes. (able to check with 'nodename' options)
<option>
nodename: flag to indicate node names as labels (default=disabled)
radius: cycle graph's radius (default=3)
startangle: cycle graph's first node's angle in degree (default=0)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)

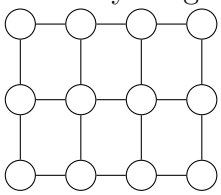
CompleteEdge[*<option>*]{*<number>*}{*<prefix>*}
<number>: number of nodes in graph to apply complete edge (must have)
<prefix>: prefix of target graph (must have) the target graph must be in the same tikzpicture scope and called before this macro.
<option>
edgestyle: edge's drawing style (default={})




```
\begin{tikzpicture}
\CycleGraph[prefix=Q,radius=2,nodeonly]{7}{a,b,c}{}
\CompleteEdge{7}{Q}
\end{tikzpicture}
```

GridGraph[*<option>*]{*<number1>*}{*<number2>*}{*<lables>*}{*<edges>*}

<*number1*>: number of nodes in rows (must have)
<*number2*>: number of nodes in colums (must have)
<*labels*>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<*edges*>: list of user specific edges (leave it blank if not needed, by default grid edges), using node index to indicate nodes. (able to check with 'nodename' options)
<*option*>
nodename: flag to indicate node names as labels (default=disabled)
vertdistance: node distance in vertical way (default=1)
horidistance: node distance in horizontal way (default=1)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)



```
\begin{tikzpicture}
\GridGraph[horidistance=0.8]{3}{4}{}{}
\end{tikzpicture}
```

BipartiteEdge[<*option*>]{<*number1*>}{<*prefix1*>}{<*number2*>}{<*prefix2*>}
<*number1*>: number of nodes in first path graph (must have)
<*prefix1*>: prefix of target graph (must have) the target graph must be in the same tikzpicture scope and called before this macro.
<*number2*>: number of nodes in second path graph (must have)
<*prefix2*>: prefix of target graph (must have) the target graph must be in the same tikzpicture scope and called before this macro.
<*option*>
edgestyle: edge's drawing style (default={})

BipartiteGraph[<*option*>]{<*number1*>}{<*labels1*>}{<*number2*>}{<*labels2*>}
<*number1*>: number of nodes in first path graph (must have)
<*labels1*>: labels of nodes in first path graph (leave it blank if not needed, no need to match number of nodes)
<*number2*>: number of nodes in second path graph (must have)
<*labels2*>: labels of nodes in second path graph (leave it blank if not needed, no need to match number of nodes)
<*option*>
nodename: flag to indicate node names as labels (default=disabled)
prefixA: first graph node's prefix (default=a)
prefixB: second graph node's prefix (default=b)
direction: both graphs nodes drawing direction
distance: distance between two graphs (default=3cm)

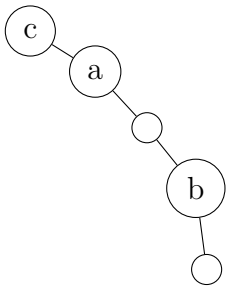
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)

ButterflyGraph[*<option>*]{*<dimension>*}{*<lables>*}
<dimension>: dimension of butterfly graph (must have)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<option>
nodename: flag to indicate node names as labels (default=disabled)
vertdistance: node distance in vertical way (default=5mm)
horidistance: node distance in horizontal way (default=5mm)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={->}, able to change to {},{<-},{<->} or with other stylings)
nodeonly: flag of drawing nodes only (default=disabled)

Hypercube[*<option>*]{*<dimension>*}{*<lables>*}
<dimension>: dimension of hypercube (must have)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<option>
nodename: flag to indicate node names as labels (default=disabled)
vertdistance: node distance in vertical way (default=1)
horidistance: node distance in horizontal way (default=1)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)

GeneralGraph[*<option>*]{*<nodes>*}{*<lables>*}{*<edges>*}
Uses spring model graph drawing algorithm.
<nodes>: nodes of graph (must have) it if only for indicating separate nodes, does not effect labels or node names (can use any symbols)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<edges>: edges of graph (must have) use symbols in *<nodes>* to make edge connection (the graph must be all connected for the algorithm to work)
<option>
startlayout: the base graph to apply the algorithm (default=cycle, possible obtions: cycle, star)
nodename: flag to indicate node names as labels (default=disabled)
radius: radius for start graphs -which are cycle graph and star graph (default=3) if the nodes are positioned too close, the algorithm might not work. Therefore with many nodes, it may need bigger radius.
L: side length of graph display area (default set up as decent length for number of nodes)

e: local minimum of overall energy applied in spring of string model (default=0.001) (Algorithm will try to reach this minimum value by ultering node positions in each loop. So if e is too small, it may loop infinitly and if too big, it might not produce fully neat graph)
K: Tension/strength of springs overall (default=1) (Also effect the resulting graph like e, if too small, likely to have less tidy graph and if too big, might have infinite loop depending on e value)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)




```
\begin{tikzpicture}
\GeneralGraph[L=4,e=0.1,K=1]{5,3,4,2,1}{a,b,c}{{1,3},{2,3},{2,5},{4,5}}
\end{tikzpicture}
```

SpectralGraph[*<option>*]{*<nodes>*}{*<lables>*}{*<edges>*}
Uses spectral graph drawing algorithm, originaly 3-dimensional graph. It requires use of c++ compilation using g++, with external c++ library Eigen. The tex file also needs to be run with –shell-escape flag on LuaLaTeX.
<nodes>: nodes of graph (must have) it if only for indicating separate nodes, does not effect labels or node names (can use any symbols)
<labels>: labels of nodes (leave it blank if not needed, no need to match number of nodes)
<edges>: edges of graph (must have) use symbols in *<nodes>* to make edge connection (the graph must be all connected for the algorithm to work)
<option>
nodename: flag to indicate node names as labels (default=disabled)
viewaxis: the direction of axis to view the graph (default=x, possible options: x, y, z)
scale: node distance scale of the graph (default=10)
prefix: node's prefix (default=a)
nodestyle: node's drawing style (default={draw,circle})
edgestyle: edge's drawing style (default={})
nodeonly: flag of drawing nodes only (default=disabled)