

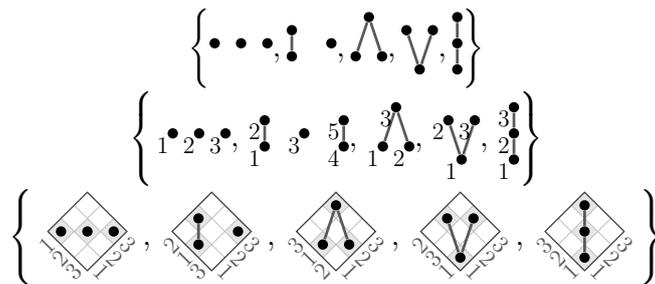
package

# causets

version 1.2

Christoph Minz

February 07, 2022



## What Is the Purpose of This Package?

Causal set theory is a proposed framework for quantum gravity. Causal sets (**causets** for short) are locally finite, partially ordered sets. Though one can describe them mathematically as a set with a partial order, it is usually much simpler to just draw (Hasse) diagrams.

This package implements some simple functionality to produce such diagrams for causets by drawing them as TikZ graphics. The drawing algorithm makes use of the fact that every finite causet that embeds in 1 + 1 dimensional Minkowski spacetime can be represented by a product of two total orders (2D-orders), where one order can be the total order of the integers labeling the causet elements, while the other is a permutation of the labels<sup>1</sup>. The idea is then used as a blueprint to draw diagrams for any generic (finite) causet, so that it becomes easy to draw causets and even use them in mathematical expressions.

---

<sup>1</sup>See <https://arxiv.org/abs/0706.0375> and <https://arxiv.org/abs/2011.02965> for more details on 2D-orders and causal sets.

## 1 Introduction - the Main Commands

This package is built on the package `tikz` to draw the diagrams. To import the `causets` package into a  $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$  file, write

```
1 \usepackage{causets}
```

in the preamble of the document.

For a more demonstrative introduction in this first section, I use the package options `black` (only black and white diagrams) and `permutation` (also shows the permutation),

```
1 \usepackage[black,permutation]{causets}
```

The three main commands of this package are

```
1 \pcauset{...i,...k,...j,...}
```

```
2 \rcauset{...}{...i/j,...}
```

```
3 \causet{...}{...i/j,...}
```

The first command takes a permutation of the integers  $[1, N]$  (where  $N$  is the cardinality of the causet). The permutation determines the event positions and causal structure such that events  $i$  and  $j$  are linked if  $i < j$  and there is no  $k > j$  between  $i$  and  $j$ . The second command does the same, but ignores links between event pairs  $i/j$  as given in the second argument. The third command only uses the permutation to position the events, the links have to be explicitly specified by event pairs  $i/j$  in the second argument.

Any causet that can be embedded in  $1 + 1$  dimensional Minkowski spacetime can also be described by the product of two total orders (a 2D-order). Let the causets elements be labeled by an increasing null coordinate  $u$  (pointing upwards along the right axis of the grids under the diagrams below), then the causal relation (partial order) implies a label permutation along the null coordinate  $v$  (pointing upwards along the left axis of the grids under the diagrams below). This permutation is the input of the `\pcauset` command that computes the causal relation from the permutation. See example 1 in [Table 1](#).

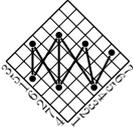
#	command	inline output
1	<code>\pcauset{4,7,2,6,1,5,3}</code>	
2	<code>\rcauset{4,7,2,6,1,5,3}{2/5}</code>	
3	<code>\causet{4,7,2,6,1,5,3}{1/3,1/5,2/3,2/6,4/5,4/6,4/7}</code>	

Table 1: Examples for the three main commands of the package.

Every causet that does not embed in  $1 + 1$  dimensional Minkowski spacetime can be made embeddable by adding links to it. Adding the minimal number of links yields permutations that should be used to determine the coordinates of the diagram. The links that have been added to obtain the permutation need to be removed from the generated causet, which is specified by link pairs  $i/j$ . See example 2 in [Table 1](#).

In some situations, it might be easier to specify the list of link pairs that exist, instead of removing links. For this purpose, use the `\causet` command. As an example, we rewrite the example 2 by specifying all existing links as pairs, see example 3 in [Table 1](#). Note that the `\rcauset` command was actually much shorter for this particular example. However, if the list of link pairs in the `\rcauset` command is very long, specifying all existing link pairs instead with the `\causet` yields a faster performance.

As an example for the use in mathematical expressions, here is the code for the first set of causets on the title page:

```

1 \usepackage{causets}
2 ... % ... \begin{document} ...
3 \begin{align*}
4   \left\{
5     \pcauset{3,2,1}, \pcauset{3,1,2}, \pcauset{2,1,3}, \pcauset{1,3,2},
6     \pcauset{1,2,3}
7   \right\}.
\end{align*}

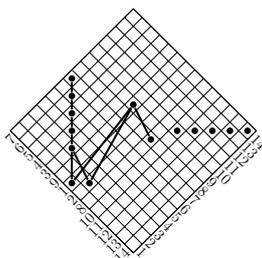
```

$$\{\dots, \downarrow, \cdot, \wedge, \vee, \uparrow\}.$$

(Note that here the package is imported without any options.)

Of course, a given causet might have more than one representing permutation, so that any of those permutations may be used. However, you may consider it a good practice to choose the permutation that starts with the larger integer, so that disjoint elements appear to the right in the graphs. For example, you may write `\pcauset{3,1,2}`,  $\downarrow \cdot$ , as opposed to `\pcauset{2,3,1}`,  $\cdot \downarrow$ .

It is possible to use dots in a permutation so that TikZ will auto-complete the list. For example, the command `\pcauset{14,\dots,10,8,2,1,9,3,4,\dots,7}` will generate the diagram



which is the same as `\pcauset{14,13,12,11,10,8,2,1,9,3,4,5,6,7}`. (This graphic uses the package options `black` and `permutation` again.)

## 2 Package Options

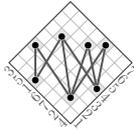
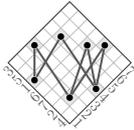
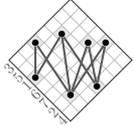
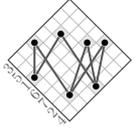
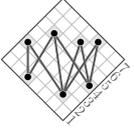
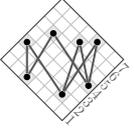
Each of the package options can either be specified in square brackets when declaring the package (option applies globally) or be changed with a macro at any point in the document (option applies to the local scope).

If no options are used, an output like  is obtained with the commands

```
1 \usepackage{causets}
2 ... % ... \begin{document} ...
3 \rcauset{4,2,7,6,1,5,3}{2/5}
```

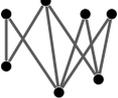
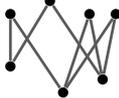
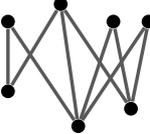
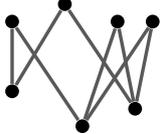
The options – given in the following table – are passed to the `causets` package as optional parameter and can be combined. If not specified otherwise, any package option works with the command `\causet` as well as with the two shown examples created with the commands

```
1 \pcauset{4,2,7,6,1,5,3}
2 \rcauset{4,2,7,6,1,5,3}{2/5}
```

Package option	<code>\pcauset</code>	<code>\rcauset</code>
<code>\usepackage[permutation]{causets}</code> shows permutations ( <code>\causetsDrawPermutationtrue</code> activates this option) ( <code>\causetsDrawPermutationfalse</code> deactivates this option)		
<code>\usepackage[permutation, unlabeled u]{causets}</code> shows permutations without u-labels ( <code>\causetsDrawULabelstrue</code> activates u-labels) ( <code>\causetsDrawULabelsfalse</code> deactivates u-labels)		
<code>\usepackage[permutation, unlabeled v]{causets}</code> shows permutations without v-labels ( <code>\causetsDrawVLabelstrue</code> activates v-labels) ( <code>\causetsDrawVLabelsfalse</code> deactivates v-labels)		
<code>\usepackage[labeled]{causets}</code> shows labeled events ( <code>\causetsDrawLabelstrue</code> activates labels) ( <code>\causetsDrawLabelsfalse</code> deactivates labels)		
<code>\usepackage[continuously linked]{causets}</code> allows links to cross over events. Broken links (that is the default option: <code>\usepackage[brokenly linked]{causets}</code> ) affects only the <code>\causet</code> command, but not the examples of <code>\pcauset</code> and <code>\rcauset</code> , shown on the right. Details on broken links are given below.		

*continued on the next page*

*continued from the previous page*

Package option	<code>\pcauset</code>	<code>\rcauset</code>
<code>\usepackage[cap linked]{causets}</code> sets caps as link tips		
<code>\usepackage[arrow linked]{causets}</code> sets arrows (rounded, open stealth) as link tips		
<code>\usepackage[unlinked]{causets}</code> does not draw links ( <code>\causetsDrawLinkstrue</code> activates links) ( <code>\causetsDrawLinksfalse</code> deactivates links)		
<code>\usepackage[tiny]{causets}</code> for tiny diagrams ( <code>\tikzcausetsset{tiny}</code> sets tiny sizes)		
<code>\usepackage[small]{causets}</code> for small diagrams ( <code>\tikzcausetsset{small}</code> sets small sizes)		
<code>\usepackage[large]{causets}</code> for large diagrams ( <code>\tikzcausetsset{large}</code> sets large sizes)		
<code>\usepackage[huge]{causets}</code> for huge diagrams ( <code>\tikzcausetsset{huge}</code> sets huge sizes) ( <code>\tikzcausetsset{normal}</code> resets to normal sizes)		
Details on size options are given in <a href="#">Sec. 4</a> .		
<code>\usepackage[black]{causets}</code> for black and white causets ( <code>\tikzcausetsset{black colors}</code> activates the option)		
<code>\usepackage[gray]{causets}</code> for gray scale causets (default) ( <code>\tikzcausetsset{gray colors}</code> activates the option)		
<code>\usepackage[blue]{causets}</code> for blue causets ( <code>\tikzcausetsset{blue colors}</code> activates the option)		
<code>\usepackage[neon]{causets}</code> for brightly colored causets ( <code>\tikzcausetsset{neon colors}</code> activates the option)		
Details on color options are given in <a href="#">Sec. 5</a> .		
<code>\usepackage[spatially linked]{causets}</code> shows dashed lines between spatially separated events (which is a secondary partial order for the causets drawn with <code>\pcauset</code> ). This option only affects the <code>\pcauset</code> and <code>\rcauset</code> commands, but not the <code>\causet</code> command. ( <code>\causetsDrawSpatialLinkstrue</code> activates the option) ( <code>\causetsDrawSpatialLinksfalse</code> deactivates the option)		

By default, the links in a diagram created with the `\causet` command are interrupted if they hit another event (exactly in their center). Any such crossing of links over events

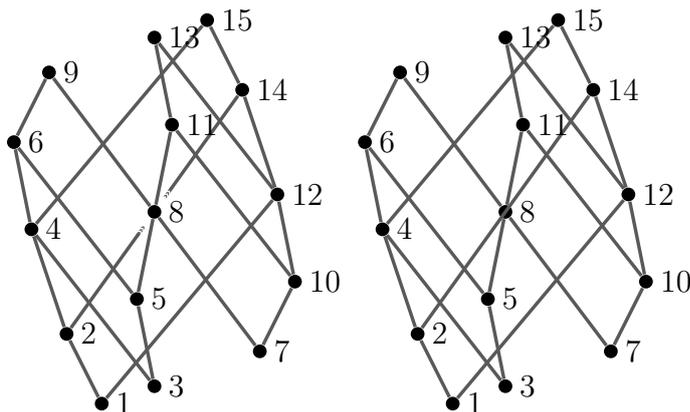
can only be caused by the manually drawn links, explicitly given in the second argument of `\causet`, but not with the links drawn by `\pcauset` or `\rcauset`. This feature is activated by default (`\causetsBreakLinkstrue`) and can be deactivated with `\causetsBreakLinksfalse`, or with the option `continuously linked`.

For example, here is a causet that embeds in 3-dimensional but not in 2-dimensional Minkowski spacetime. This causet has one event (8) in the center that is linked to events 7 and 9, but not to events 2 and 14, while the link from 2 to 14 crosses the event 8 in the center.

```

1 \usepackage[huge, labeled]{causets}
2 \tikzcausetsset{labels={right=0.5ex}}
3 ... % ... \begin{document} ...
4 \causet[brokenly linked]{7,10,3,1,12,5,2,8,14,11,4,15,13,6,9}
   {1/2,1/12,2/4,2/14,3/4,3/5,4/6,4/15,5/6,5/8,6/9,7/8,7/10,8/9,8/11,
   10/11,10/12,11/13,12/13,12/14,14/15}
5 \causet[continuously linked]{7,10,3,1,12,5,2,8,14,11,4,15,13,6,9}
   {1/2,1/12,2/4,2/14,3/4,3/5,4/6,4/15,5/6,5/8,6/9,7/8,7/10,8/9,8/11,
   10/11,10/12,11/13,12/13,12/14,14/15}

```



Note that the option `brokenly linked` is not necessary if the default option has not been changed.

### 3 Short Hand Macros

The permutations and labels for causets can be switched on globally with the package options `permutation` and `labeled`, or at any point in the document with the switches `\causetsDrawPermutationtrue` (same as the command `\tikzcausetsset{show permutation}` used in the preamble) and `\causetsDrawLabelstrue` (or `\tikzcausetsset{labeled}` used in the preamble), respectively.

To show only single causets with either of these options, there exist short hand macros for all three main commands. Each main command can be followed by a capital letter `P` (to show the permutation), `L` (to show event labels), or `x` (to show both).

```

1 \usepackage{causets}
2 ... % ... \begin{document} ...
3 Causets with permutations (and labels) \pcausetP{4,1,2,3},

```

```
\pcausetX{1,2,3,4}, \pcausetL{1,2,4,3}, and without any
\pcauset{2,4,1,3}.
```

Causets with permutations (and labels) , , , and without any . These results are also obtained by using the optional parameter, for example:

```
1 \usepackage{causets}
2 ... % ... \begin{document} ...
3 Causets with permutations (and labels) \pcauset[permutation]{4,1,2,3},
   \pcauset[permutation,labeled]{1,2,3,4}, \pcauset[labeled]{1,2,4,3},
   and without any \pcauset{2,4,1,3}.
```

The optional argument of the main commands accepts any causets or TikZ keys to change the style of the causet. For example, the baseline can be moved with

```
1 \pcauset[baseline=2mm]{2,4,5,1,3}
```

such that the diagram is placed with its center at 2mm below the text base, 

There are a few short hand macros for small common causets as shown below.

```
1 \causetFence{1}, \causetFence{2}, \ldots, \causetFence{10},
2 \causetClosedFence{1}, \causetClosedFence{2}, \ldots,
   \causetClosedFence{10},
3 \causetCrown
```

, , ..., , , ..., , 

## 4 Changing Sizes (Esp. for Mathematical Expressions)

Apart from the pre-defined sizes `tiny`, `small`, `normal`, `large` and `huge`, sizes can be specified in the optional argument of a single causet diagram or with `\tikzcausetsset{...}` for the current scope. Use the keys

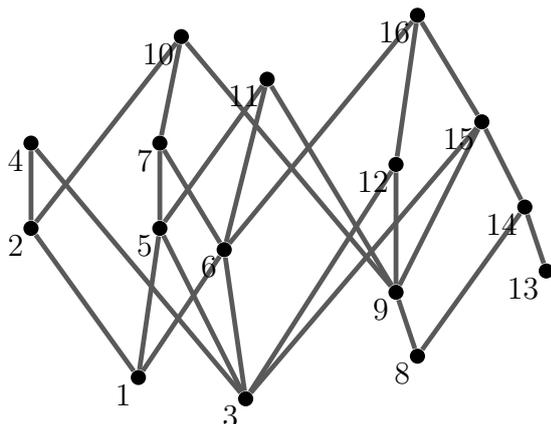
- `tile size=<value>` stores `<value>` in the macro `\causetTileSize` as the size of the tiles in the permutation
- `region line width=<value>` stores `<value>` in the macro `\causetRegionLine` as the thickness for the line surrounding the permutation region
- `grid line width=<value>` stores `<value>` in the macro `\causetGridLine` as the line thickness of the permutation grid
- `event size=<value>` stores `<value>` in the macro `\causetEventSize` as the diameter of the events
- `link width=<value>` stores `<value>` in the macro `\causetLinkWidth` as the link thickness
- `broken link gap=<value>` stores `<value>` in the macro `\causetBrokenLinkGap` as the gap before and after an event that interrupts a link
- `text scale=<value>` stores `<value>` in the macro `\causetTextScale` as the scaling factor of all the text labels (relative to the normal font size)

For example, the normal size is a pre-defined style with

```
1 \tikzcausetsset{normal/.style={tile size=0.90ex, region line
   width=0.06ex, grid line width=0.03ex, event size=0.48ex, link
   width=0.15ex, text scale=0.48}}
```

However, if you want to draw a single diagram with your own sizes, use

```
1 \pcausetL[tile size=2.20ex, region line width=0.15ex, grid line
   width=0.06ex, event size=1.10ex, link width=0.35ex, text
   scale=1]{13,8,14,9,3,15,12,1,6,16,5,11,7,2,10,4}
```



Changing sizes becomes important when you use causet diagrams in mathematical expression, where the size may be adjusted for indices or limits. For example:

```
1 \begin{equation}
2   \sum_{C \in \tikzcausetsset{small}}
3   \left\{
4     \pcauset{1,2,3},
5     \pcauset{1,3,2,4},
6     \pcauset{1,4,3,2,5},
7     \pcauset{1,5,4,3,2,6}
8   \right\}
9   | C |
10  = \left| \pcauset{1,2,3} \right|
11  + \left| \pcauset{1,3,2,4} \right|
12  + \left| \pcauset{1,4,3,2,5} \right|
13  + \left| \pcauset{1,5,4,3,2,6} \right|
14 \end{equation}
```

$$\sum_{C \in \left\{ \begin{array}{c} \bullet \\ \bullet \\ \bullet \end{array}, \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \end{array}, \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{array}, \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{array} \right\}} |C| = \left| \begin{array}{c} \bullet \\ \bullet \\ \bullet \end{array} \right| + \left| \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \end{array} \right| + \left| \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{array} \right| + \left| \begin{array}{c} \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \\ \bullet \end{array} \right| \quad (1)$$

## 5 Changing Colors

Colors can be set with the following keys:

- `region color=value` for the frame of the permutation grid
- `grid color=value` for the lines of the permutation grid

- `tile color=<value>` for the occupied tiles
- `ulabel color=<value>` for the  $u$  labels
- `vlabel color=<value>` for the  $v$  labels
- `event color=<value>` for the events
- `label color=<value>` for the event labels
- `link color=<value>` for the links
- `spatial link color=<value>` for the spatial links.

When using any of these key-value pairs in an optional arguments of the causet macros, they apply to that diagram. To change colors within a scope or the entire document, use the `\tikzcausetsset` macro, for example

```
1 \tikzcausetsset{event color=purple, link color=cyan}
```

changes the event color to purple and the link color to cyan.

For the pre-defined color schemes, use the keys `gray colors` (default colors), `black colors` (only black and white), `blue colors` (blue color tones), or `neon colors` (bright colors). These keys set all the colors listed above to pre-defined values.

## 6 Standalone Causets

For any document that contains many (large) causet diagrams or the same causets that are used repeatedly, the compilation time can be reduced by putting the commands in external files using the `standalone` package.

For example, create a file named `mycauset.tex` in the same directory as the main file with the following code:

```
1 \documentclass[tikz]{standalone}
2 \usepackage{causets}
3 \begin{document}
4 \pcauset{3,2,4,1}
5 \end{document}
```

The compiled version is then included with the macro `\causetfile{mycauset}`. The macro `\causetfile[#1]{#2}` also accepts one optional argument `[#1]` and redirects to

```
1 \includegraphics[#1]{\causetspath/#2}
```

The macro `\causetspath` is pre-defined to `.` (the directory of the main  $\text{\TeX}$  file), but may be renewed to any desired directory with

```
1 \renewcommand{\causetspath}{my_causet_directory}
```

## 7 Advanced Style Changes

In the optional argument of the causet macros, you may use other TikZ options to change the output of the macros, for example:

```

1 \usepackage{causets}
2 ... % ... \begin{document} ...
3 A labeled causet with red-framed semi-transparent labels,
   \pcausetL[labels={draw=red, fill=white, opacity=0.8, inner sep=1pt,
   scale=1.5}]{2,4,5,1,3}.

```

A labeled causet with red-framed semi-transparent labels, .

In the same way, it is possible to modify the existing styles of all parts of the diagrams, which means that the default drawing styles are extended by the options passed as value to these keys:

1. `region=<value>` shows the permutation (`show permutation`) and appends to the style of the permutation frame,
2. `grid=<value>` shows the permutation (`show permutation`) and appends to the style of the permutation grid,
3. `tiles=<value>` shows the permutation (`show permutation`) and appends to the style of the occupied permutation tiles,
4. `events=<value>` appends to the style of the event nodes (graph vertices),
5. `links=<value>` shows the links (`show links`) and appends to the style of the links (main graph edges),
6. `link starts=<value>` sets the starting line tip of each link,
7. `link ends=<value>` sets the ending line tip of each link,
8. `link pauses=<value>` sets the ending line tip of each link at an interruption,
9. `link resumes=<value>` sets the starting line tip of each link at an interruption,
10. `cap links` sets the starting and ending line tip of each link to a (reversed) cap,
11. `arrow links` sets the ending line tip of each link to a rounded, open, stealth arrow,
12. `spatial links=<value>` shows the spatial links (`show spatial links`) and appends to the style of the spatial links (secondary graph edges),
13. `all labels=<value>` appends to the general style of all text labels (applies to the following three label types),
14. `labels=<value>` shows the event labels (`show labels`) and appends to the style that applies only to the event labels,
15. `ulabels=<value>` shows the *u* labels (`show ulabels`) and appends to the style that applies only to the *u* labels, and
16. `vlabels=<value>` shows the *v* labels (`show vlabels`) and appends to the style that applies only to the *v* labels.

In order to manually define the full style of the parts in the diagram - not extending, but *overwriting all* drawing options - set the style of the following keys:

1. `every causet` for the general style of the causet diagram (like `baseline`),
2. `every tile` for the occupied permutation tiles,
3. `every event` for the event nodes (graph vertices),
4. `every grid` for the permutation grid,
5. `every region` for the permutation region,
6. `every label base` for all text labels,
7. `every ulabel` for the permutation  $u$  labels,
8. `every vlabel` for the permutation  $v$  labels,
9. `every link` for the links (graph edges),
10. `every spatial link` for the spacelike separations, and
11. `every label` for the event labels.

With the macro `\tikzcausetsset{...}`, any of these styles can also be set in the preamble of the document. So, for example,

```

1 \usepackage{causets}
2 \tikzcausetsset{events={rectangle}, links={-Triangle Cap[]}}
3 ... % ... \begin{document} ...
4 \pcauset{2,4,5,1,3}

```

will generate all causets with square nodes as events and triangle caps on the future link ends, like .

## 8 Using this Package with TikZ

Since the package is TikZ based, using a causet within a TikZ picture is simple. Each of the three main commands is based on a draw command in TikZ that has the same name but starts with `\draw...`. The `\draw` macros, however, do not support an optional argument (since this can be specified directly in a TikZ picture) and they also do not have any short hand macros ending in `P`, `L` or `X`.

The two following examples are given by external (standalone) TikZ graphics and included with `\causetfile{...}`. More details on using causets in standalone files are given in [Sec. 6](#).

Note that the output of the `\draw` macros has the point (0,0) in the center, so in order to shift it, we may enclose it in a TikZ scope, for example:

```

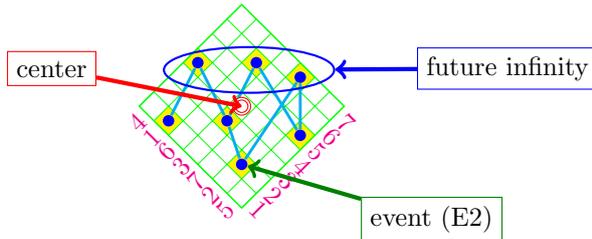
1 \documentclass[tikz]{standalone}
2 \usepackage[permutation,huge,neon]{causets}
3 \usetikzlibrary{fit,shapes.geometric}
4 \begin{document}
5 \begin{tikzpicture}
6   \begin{scope}[xshift=3cm, yshift=-1cm]
7     \drawpcauset{5,2,7,3,6,1,4}
8     \draw[red, double] (0, 0) circle[radius=0.1];
9   \end{scope}

```

```

10 \node[draw=blue, inner sep=1pt, thick, ellipse, fit=(E4) (E6) (E7)]
    (Finf) {};
11 \node[draw=blue] (FinfLabel) at (6.5, -0.5) {future infinity};
12 \node[draw=red] (centerLabel) at (0.5, -0.5) {center};
13 \node[draw=green!50!black] (myEventLabel) at (5.5, -2.5) {event (E2)};
14 \draw[ultra thick, blue, ->] (FinfLabel) -- (Finf);
15 \draw[ultra thick, red, ->] (centerLabel) -- (3, -1);
16 \draw[ultra thick, green!50!black, ->] (myEventLabel) -- (E2);
17 \end{tikzpicture}
18 \end{document}

```



Note that this example also shows how to use the event nodes that are generated by the `\drawpcauset` macro. The green arrow points to the node of the second causet event (E5). The causet events, event labels,  $u$ -axis labels, and  $v$ -axis labels are the nodes  $(E_i)$ ,  $(EL_i)$ ,  $(EUL_i)$ , and  $(EVL_i)$ , respectively, where  $i \in [1, N]$  for a causet with  $N$  events.

It is also possible to use multiple causets in more complicated graphics, like a tree for example, but note that whenever a causet is actually a text of a node within a TikZ picture (like in the following), we need to use the `\pcauset` (`\rcauset` or `\causet`) commands without the `draw` prefix again:

```

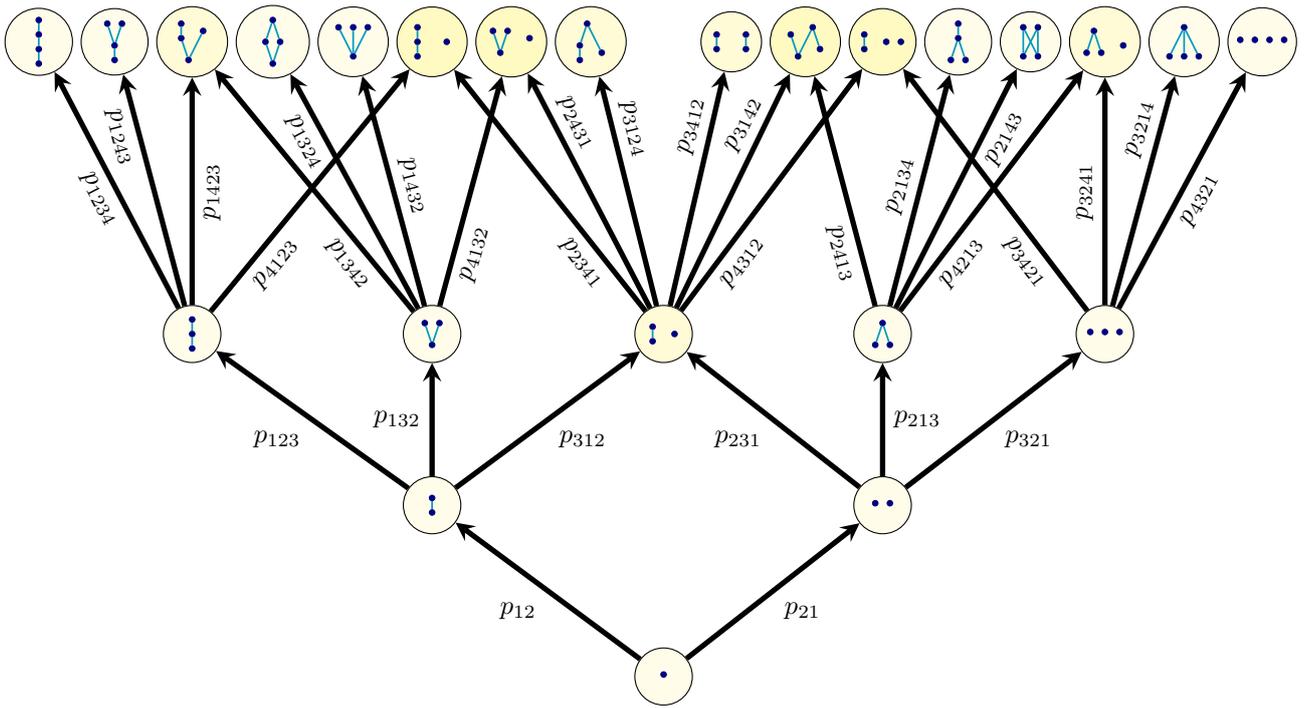
1 \documentclass[tikz]{standalone}
2 \usepackage[blue]{causets}
3 \usetikzlibrary{fit, shapes.geometric}
4 \begin{document}
5 \begin{tikzpicture}[-stealth, line width=2pt]
6 \matrix[nodes={draw, fill=yellow!10, thin, circle, inner sep=0.6ex,
7 minimum size=5ex}, row sep=1.5cm, column sep=0.1cm]
8 {
9 \node (C1234) {\pcauset{1,2,3,4}};
10 & \node (C1243) {\pcauset{1,2,4,3}};
11 & \node[fill=yellow!20] (C1423) {\pcauset{1,4,2,3}};
12 & \node (C1324) {\pcauset{1,3,2,4}};
13 & \node (C1432) {\pcauset{1,4,3,2}};
14 & \node[fill=yellow!30] (C4123) {\pcauset{4,1,2,3}};
15 & \node[fill=yellow!30] (C4132) {\pcauset{4,1,3,2}};
16 & \node[fill=yellow!20] (C3124) {\pcauset{3,1,2,4}};
17 & & \node[fill=yellow!20] (C3412) {\pcauset{3,4,1,2}};
18 & \node[fill=yellow!30] (C3142) {\pcauset{3,1,4,2}};
19 & \node[fill=yellow!30] (C4312) {\pcauset{4,3,1,2}};
20 & \node (C2134) {\pcauset{2,1,3,4}};
21 & \node (C2143) {\pcauset{2,1,4,3}};
22 & \node[fill=yellow!20] (C4213) {\pcauset{4,2,1,3}};

```

```

22 & \node (C3214) {\pcauset{3,2,1,4}};
23 & \node (C4321) {\pcauset{4,3,2,1}};
24 \\
25 \\
26 & & \node (C123) {\pcauset{1,2,3}};
27 & & & \node (C132) {\pcauset{1,3,2}};
28 & & & \node[fill=yellow!20] (C312) {\pcauset{3,1,2}};
29 & & & \node (C213) {\pcauset{2,1,3}};
30 & & & \node (C321) {\pcauset{3,2,1}};
31 \\
32 & & & & \node (C12) {\pcauset{1,2}};
33 & & & & & \node (C21) {\pcauset{2,1}};
34 \\
35 & & & & & & & \node (C1) {\pcauset{1}};
36 \\
37 };
38 \draw (C1) -- node[below left] {$p_{12}$} (C12);
39 \draw (C12) -- node[below left] {$p_{123}$} (C123);
40 \draw (C123) -- node[sloped, midway, below] {$p_{1234}$} (C1234);
41 \draw (C123) -- node[sloped, near end, below] {$p_{1243}$} (C1243);
42 \draw (C123) -- node[sloped, midway, below] {$p_{1423}$} (C1423);
43 \draw (C123) -- node[sloped, near start, below] {$p_{4123}$} (C4123);
44 \draw (C12) -- node[left] {$p_{132}$} (C132);
45 \draw (C132) -- node[sloped, near start, below] {$p_{1342}$} (C1423);
46 \draw (C132) -- node[sloped, near end, below] {$p_{1324}$} (C1324);
47 \draw (C132) -- node[sloped, midway, above] {$p_{1432}$} (C1432);
48 \draw (C132) -- node[sloped, near start, below] {$p_{4132}$} (C4132);
49 \draw (C12) -- node[below right] {$p_{312}$} (C312);
50 \draw (C312) -- node[sloped, near end, above] {$p_{3124}$} (C3124);
51 \draw (C312) -- node[sloped, near end, above] {$p_{3412}$} (C3412);
52 \draw (C312) -- node[sloped, near end, above] {$p_{3142}$} (C3142);
53 \draw (C312) -- node[sloped, near start, below] {$p_{4312}$} (C4312);
54 \draw (C1) -- node[below right] {$p_{21}$} (C21);
55 \draw (C21) -- node[below left] {$p_{231}$} (C312);
56 \draw (C312) -- node[sloped, near start, below] {$p_{2341}$} (C4123);
57 \draw (C312) -- node[sloped, near end, above] {$p_{2431}$} (C4132);
58 \draw (C21) -- node[right] {$p_{213}$} (C213);
59 \draw (C213) -- node[sloped, midway, above] {$p_{2134}$} (C2134);
60 \draw (C213) -- node[sloped, near start, below] {$p_{2413}$} (C3142);
61 \draw (C213) -- node[sloped, near end, below] {$p_{2143}$} (C2143);
62 \draw (C213) -- node[sloped, near start, below] {$p_{4213}$} (C4213);
63 \draw (C21) -- node[below right] {$p_{321}$} (C321);
64 \draw (C321) -- node[sloped, near start, below] {$p_{3421}$} (C4312);
65 \draw (C321) -- node[sloped, midway, above] {$p_{3241}$} (C4213);
66 \draw (C321) -- node[sloped, near end, above] {$p_{3214}$} (C3214);
67 \draw (C321) -- node[sloped, midway, below] {$p_{4321}$} (C4321);
68 \end{tikzpicture}
69 \end{document}

```



## 9 Updates

### Version 1.1

- All colors, size and style options have been moved to PGF keys in the path

```
1 /tikz/causets/
```

### Version 1.2

- The color option `blue` has been modified so that a diagram is now only colored in different shades of blue, gray and black.
- Every event label is now positioned relative to the center point of an event, so that the position of event labels can be modified with `left=<value>`, `right=<value>`, etc.
- Arrow tips may now be added to the start and end of links via the style settings `link starts=<value>` and `link ends=<value>`, or via the pre-defined styles `cap links` and `arrow links`.
- By default, links drawn by the command `\causet` are now broken if they cross over an event. The style of the start and end of the link interruption are set with `link pauses=<value>` and `link resumes=<value>`.
- Links between spacelike separated events are now also supported by the command `\rcauset`.
- More details have been added to this manual, especially to [Sec. 2](#).

## 10 Bug Reports and Package Requests

Problems with the package are reported here:

<https://github.com/c-minz/LaTeX-causets/issues>

If you have a problem when using the package or you would like to have another feature to be implemented, please write a message to christoph(dot)minz(AT)gmail(dot)com

For reporting a bug, I much appreciated if you do the following:

- Check the issues reported previously if your problem is already listed.
- Describe your problem including the errors and warning messages, information about the drivers and programs versions.
- Provide a minimal working test file (only a single T<sub>E</sub>X file) that demonstrates the problem.