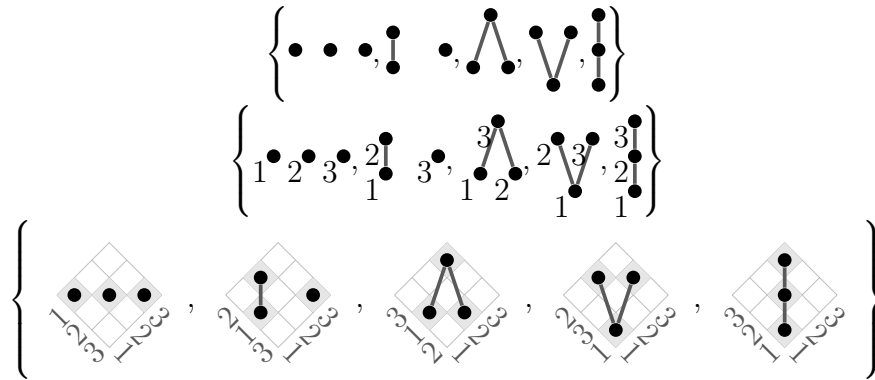package

causets

version 1.0

Christoph Minz

October 28, 2020

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. The idea is then used as a blueprint to draw diagrams for any generic causet, so that it becomes easy to draw causets and even use them in mathematical expressions.

1

1 Introduction - the Main Commands

This package requires the packages `tikz` (for the main drawing tasks), and `listofitems` (for handling the command inputs as lists). Import the package into LaTeX with:

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

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 an integer permutation that 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 and links all event pairs i-j as specified 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.
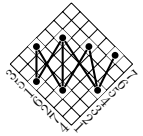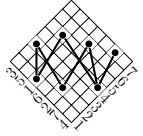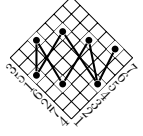
#	command	inline output
1	\pcauset{4,7,2,6,1,5,3}	
2	\rcauset{4,7,2,6,1,5,3}{2!5}	
3	\causet{4,7,2,6,1,5,3}{1-3,1-5,2-3,2-6,4-5,4-6,4-7}	

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 embedable by adding links to it. Adding the minimal number of links yields a

permutation that is 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 with a exclamation mark (often used for the logical not in programming). 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 separated by a minus sign, 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 ...
3 \begin{align*}
4   \left\{
5     \pcauset{3,2,1}, \pcauset{3,1,2}, \pcauset{2,1,3},
    \pcauset{1,3,2}, \pcauset{1,2,3}
6   \right\}.
7 \end{align*}
```

$$\left\{ \cdots, \vdots \cdot, \wedge, \vee, \vdots \right\}.$$

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 ...
3 \rcauset{4,2,7,6,1,5,3}{2!5}
```

The package options and their macros are shown in Table 2, using the same example.

All these options can also be combined, but only one of the size and color options. The last option (`spacelike`) is only supported by the `\pcauset` commands (and ignored by the other commands), so that the last example uses `\pcauset{2,4,1,3}`.
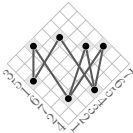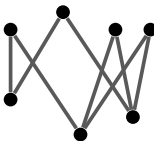
| package option	inline output

`\usepackage[permutation]{causets}` shows permutations
  (`\causetsDrawPermutationtrue` activates the option)
  (`\causetsDrawPermutationfalse` deactivates the option)	

`\usepackage[labeled]{causets}` shows labeled events
  (`\causetsDrawLabelstrue` activates the option)
  (`\causetsDrawLabelsfalse` deactivates the option)	

`\usepackage[unlinked]{causets}` hides links
  (`\causetsDrawLinksfalse` activates the option)
  (`\causetsDrawLinkstrue` deactivates the option)	

`\usepackage[small]{causets}` for smaller diagrams
  (`\causetsSetSizesSmall` activates the option)
`\usepackage[large]{causets}` for larger diagrams
  (`\causetsSetSizesLarge` activates the option)
  (`\causetsSetSizesNormal` resets to normal size)
(`\causetsSetSizes{}{}{}{}{}` for user-defined sizes, see below)	

`\usepackage[black]{causets}` for black and white causets
  (`\causetsSetColorsBlack` activates the option)
`\usepackage[gray]{causets}` for gray scale causets (default)
  (`\causetsSetColorsGray` activates the option)
`\usepackage[blue]{causets}` for blue causets
  (`\causetsSetColorsBlue` activates the option)
`\usepackage[neon]{causets}` for brightly colored causets
  (`\causetsSetColorsNeon` activates the option)
(more color formatting options below)	

`\usepackage[spacelike]{causets}` shows a partial order for space-like separated events, but only with the `\pcauset` commands
  (`\causetsDrawSeparationstrue` activates the option)
  (`\causetsDrawSeparationsfalse` deactivates the option)	

Table 2: All package options.

3   (Optional) Arguments for the Main Commands

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` and `\causetsDrawLabelstrue`, 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 ...
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 .

All main commands have an optional argument in which any TikZ commands can be used to change the style of the causet. By default, the optional argument is set to `causet`, which is a TikZ style defined by the package to align the output with the baseline of the text, see the examples here.

```
1 Without an optional argument \pcauset{2,4,5,1,3} aligns the
     center just above the baseline. Using \pcauset[]{2,4,5,1,3},
     the bottom of the diagram will be aligned with the baseline;
     and with \pcauset[baseline=1mm]{2,4,5,1,3} the center is
     placed 1mm below the baseline.
```

Without an optional argument aligns the center just above the baseline. Using , the bottom of the diagram will be aligned with the baseline; and with the center is placed 1mm below the baseline.

4   Some Common Causets

There are a few short hand macros for small common causets:

```
1 \causetFence{2}
2 \causetFence{3}
3 \causetFence{4}
4 \causetFence{5}
5 \causetClosedFence{2}
6 \causetClosedFence{3}
7 \causetClosedFence{4}
8 \causetCrown
```

5 Changing Sizes

Apart from the pre-defined sizes `small`, `large` (and `normal`), sizes can be defined with a macro similar to

```
1 \causetsSetSizes{0.45ex}{0.01ex}{0.25ex}{0.08ex}{0.27} % small
2 \causetsSetSizes{0.90ex}{0.02ex}{0.50ex}{0.15ex}{0.50} % normal
3 \causetsSetSizes{1.80ex}{0.04ex}{1.00ex}{0.27ex}{1.00} % large
```

The arguments set the following size macros in order:

1. `\causetTileSize` as size of the tiles in the permutation (measured in any absolute unit, like cm, or any relative unit, like ex)

2. `\causetPermutationLine` as line thickness of the permutation grid (measured in any absolute unit, like cm, or any relative unit, like ex)

3. `\causetEventSize` as diameter of the events (measured in any absolute unit, like cm, or any relative unit, like ex)

4. `\causetLinkWidth` as link thickness (measured in any absolute unit, like cm, or any relative unit, like ex)

5. `\causetTextScale` as the scaling factor of text relative to the normal font size

   For example:

```
1 \causetsSetSizes{2.50ex}{0.06ex}{1.30ex}{0.35ex}{1.20}
2 \pcausetL{13,8,14,9,3,15,12,1,6,5,11,7,2,10,4}
```
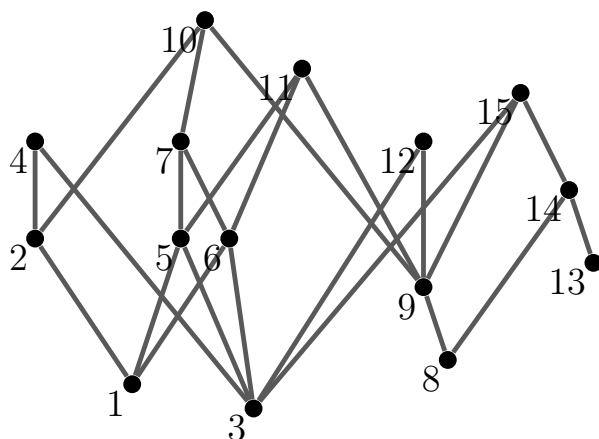
Causets diagrams can also be used in mathematical expression, where the size may be adjusted when they are used in indices or limits, for example:

```
1  \begin{equation}
2      \sum_{C \in \causetsSetSizesSmall
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}
```

$$\sum_{C \in \left\{ \, \mathord{\vdots}, \, \mathord{\Diamond}, \, \mathord{\Diamond}, \, \mathord{\Diamond} \right\}} |C| = \left| \, \vdots \, \right| + \left| \, \Diamond \, \right| + \left| \, \Diamond \, \right| + \left| \, \Diamond \, \right| \tag{1}$$

6   Changing Colors

This package uses the following colors

- `causetRegionColor` for the lines of the permutation grid

- `causetTileColor` for the occupied tiles

- `causetULabelColor` for the u labels

- `causetVLabelColor` for the v labels

- `causetEventColor` for the events

- `causetLabelColor` for the label text

- `causetLinkColor` for the links

- `causetSeparationColor` for the spacelike separations.

Any of these colors can be set to a new value with the `\colorlet` macro, for example

```
1 \colorlet{causetEventColor}{purple}
```

redefines the event color to purple.

7 Advanced Style Changes

The optional argument of the main commands is not restricted to baseline adjustments. TikZ options will change the output of the macros, for example:

```
1 \usepackage{causets}
2 ...
3 A labeled causet with red-framed semi-transparent labels,
    \pcausetL[causet,causetLabels/.style={draw=red, fill=white,
    opacity=0.8, inner sep=1pt, scale=1.2}]{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 to

- the permutation grid (`causetRegion/.style={...}`),

- the occupied permutation tiles (`causetTiles/.style={...}`),

- the permutation u labels (`causetULabels/.style={...}`),

- the permutation v labels (`causetVLabels/.style={...}`),

- the events (`causetEvents/.style={...}`),

- the labels (`causetLabels/.style={...}`),

- the links (`causetLinks/.style={...}`),

- and the spacelike separations (`causetSeparations/.style={...}`).

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

- the permutation grid (`causetRegionStyle/.style={...}`),

- the occupied permutation tiles (`causetTilesStyle/.style={...}`),

- the permutation u labels (`causetULabelsStyle/.style={...}`),

- the permutation v labels (`causetVLabelsStyle/.style={...}`),

- the events (`causetEventsStyle/.style={...}`),

- the labels (`causetLabelsStyle/.style={...}`),

- the links (`causetLinksStyle/.style={...}`),

- and the spacelike separations (`causetSeparationsStyle/.style={...}`).

Of course, it is also possible to redefine any of these styles with a \tikzset{...} in the preamble of the document. So for example

```
1 \usepackage{causets}
2 \tikzset{causetEvents/.style={rectangle}}
3 ...
4 \pcauset{2,4,5,1,3}
```

will generate ⟨symbol⟩.

8 Using causets 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 ouput of the \draw macros has the point (0,0) in the center, so to shift it, we can enclose it in a TikZ scope, for example:

```
1  \usetikzlibrary{fit,shapes.geometric}
2  ...
3  \begin{tikzpicture}
4    \begin{scope}[xshift=3cm, yshift=-1cm]
5      \causetsSetSizesLarge
6      \causetsDrawPermutationtrue
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}
```
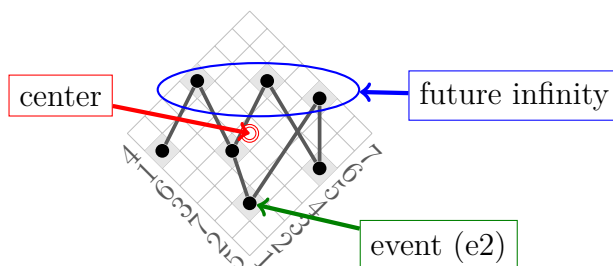
Note that this example also shows how to use the event nodes that are generated by the \drawpcauset macro. The green arrow points to the node of the second causet event (e5). Similarly, all events of a causet with n events are the nodes (ei), where $i \in [1, n]$.

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 (and similar) commands without the draw prefix again:

```
 1 \causetsSetColorsBlue
 2 \begin{tikzpicture}[-stealth, line width=2pt]
 3   \matrix[nodes={draw, fill=yellow!10, thin, circle, inner
       sep=1.7ex}, row sep=1cm, column sep=1cm]
 4   {
 5     \node (C123) {\pcauset{1,2,3}};
 6   & \node (C132) {\pcauset{1,3,2}};
 7   & \node[fill=yellow!20] (C312) {\pcauset{3,1,2}};
 8   & \node (C213) {\pcauset{2,1,3}};
 9   & \node (C321) {\pcauset{3,2,1}};
10   \\
11   & \node (C12) {\pcauset{1,2}};
12   &
13   & \node (C21) {\pcauset{2,1}};
14   \\
15   &
16   & \node (C1) {\pcauset{1}};
17   \\
18   };
19   \draw (C1) -- node[below left] {$p_{12}$} (C12);
20   \draw (C12) -- node[below left] {$p_{123}$} (C123);
21   \draw (C12) -- node[left] {$p_{132}$} (C132);
22   \draw (C12) -- node[below right] {$p_{312}$} (C312);
23   \draw (C1) -- node[below right] {$p_{21}$} (C21);
24   \draw (C21) -- node[below left] {$p_{231}$} (C312);
25   \draw (C21) -- node[right] {$p_{213}$} (C213);
26   \draw (C21) -- node[below right] {$p_{321}$} (C321);
27 \end{tikzpicture}
```