The package TooYoung Version 2

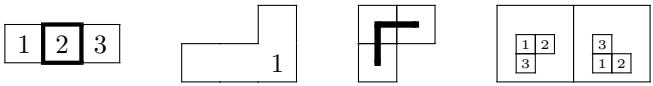
Rui Xiong

April 26, 2023

1 Introduction

Young tableaux are combinatorial, easy-visualized objects which are widely used in algebraic geometry and representation theory. A huge number of papers were devoted to build enough understanding of it. Though, at the stage of writing a real paper, it would be a common feeling that the support of Young tableaux is probably not enough on LaTeX. Usually, the problems were settled by inserting an individual image or typing equivalent codes to illustrate one (for example `tikz`). But due to its multi-functionality, these methods are not sufficiently effective. The purpose of this article is to advertise a new package TooYoung to typeset Young tableaux.

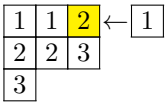
There are a couple of existing packages doing this, for example, ytableau and youngtab. In fact, the author is inspired from them extremely, and feel great honor to make complement. Besides realizing most of functions of them, perhaps one of the biggest advantage of our package TooYoung is the flexibility. For example, by a one-line code, we can simply typeset each of the following diagrams.



We will explain the usage and give samples at length in the coming sections.

The API is designed to be of xymatrix style which we hope to be of the most convenience for the users. The following example reveals the fundamental usage (the package color is required)

```
\Young[1pc]{
1&1&2*(yellow)&[\leftarrow]&1\\
2&2&3\\3}
```



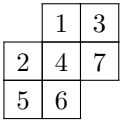
We also supply micro `\hook`, `\wall`, `\black` for a faster usage. The author would kindly remark that to use the package, one needs to make a copy of `TooYoung.sty` in the same folder of your `*.tex` documents, so that `\usepackage{TooYoung}` works in the preamble of your document.

Lastly, the full package is written by 400-line codes where certain black techniques are used. Please e-mail me (`XiongRui_Math@126.com`) if you have any comments, questions or functions you hope to realize.

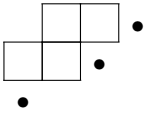
2 Fundamental Usage

Firstly, the main micro is `\Young{..}`. The codes included in the bracket are supposed to be "`matrix`" like. You can skip boxes taking advantage of `[..]`. For example, we can type skew shapes as follows.

```
\Young{[]&1&3\\2&4&7\\5&6}
```

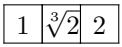


```
\Young{[]&&&[\bullet]\\
&&[\bullet]\\[\bullet]}
```



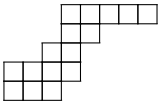
As a price, to use an command with `[..]`, one should protect it in `{..}`, for instance the following.

```
\Young{1&{\sqrt[3]{2}}&2}
```

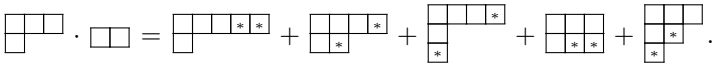


Sometimes the size of `\Young{..}` is too big in context, we provide `\young{..}` to give a smaller tableau which is carefully resized.

```
\young{[]&[]&[]&&&&&\\
[]&[]&[]&&\\[]&[]&&\\&&&\\&&}
```

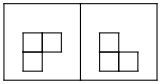


This would be useful when stating formulas like



Similarly, `\YOUNG{..}` will give a bigger one.

```
\YOUNG{\young{&\\}&\young{\\&}}
```

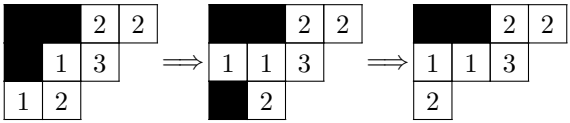


We will see how to modify the size manually in the next section.

We provide a command `\black` to fulfill a box,

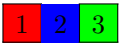
```
\Young{1&\black\\\black}
```

which is useful when illustrating jeu de taquin.

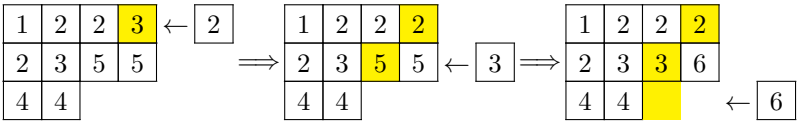


After loading the package `color` (or any package providing micro `\color{..}`), we can use colors by `*(color)` in each cell of our tableaux.

```
\Young{1*(red)&*(blue)[2]&*(green)3}
```



This is useful when it is necessarily to emphasize boxes, for example, the following RSK algorithm



3 Sizes and Formats

The default size for a cell in `\Young{..}` is `1.2pc`, and it is half for that in `\young{..}`. One can use `\Young[size]{..}` to customize the size

```
\Young[0.3pc]{&&&&&&&&&&\\&&&&&&&&&}
```

The sizes are recorded in the dimensions `\Youngwidth`, `\Youngheight` and `\Youngdepth`. We provide the commands

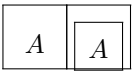
`\setYsize`, `\setYwidth`, `\setYheight`, `\Ybigger`, and `\Ysmaller`

to adjust them. Technically, the depth will be automatically modified to 1/4 of the height using the aforementioned commands. The thicknesses of the horizontal edges and the vertical ones are differently treated, named by `\YoungHthickness` and `\YoungVthickness`. We provide

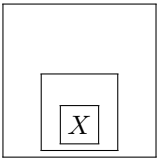
`\setYth`, `\setYhth`, `\setYvth`, `\Ythicker`, and `\Ythinner`

to modify them. Here are some samples.

```
\setYsize{2pc}\Young{A&
{\setYsize{1.5pc}\Young{A}}}
```



```
\Ybigger\Ybigger\Young{
    {\Ysmaller\Young{
        {\Ysmaller\Young{X}}}
    }
}
```



```
\setYwidth{2pc}\setYheight{1pc}
\setYhth{0.2pc}\setYvth{0.2pt}
\Young{X&Y&Z}
```

```
\setYth{0.1pc}\Young{\texttt{?}}
```

```
\Ythicker\Young{{\bf L}}
\Ythinner\Young{{L}}
```

For example, we can typeset Young tabloid by `\setYvth{0pc}`.

$$\begin{array}{|c|c|c|c|}\hline 1&2&3&4\\\hline 5&6&7\\\cline{1-3} 8\\\cline{1-1}\end{array}\quad=\quad\begin{array}{|c|c|c|c|}\hline 4&2&1&3\\\hline 6&5&7\\\cline{1-3} 8\\\cline{1-1}\end{array}$$

The uniform format in all boxes can be easily customized by command `\setYstyle{..}`. Or, perhaps more flexibly by redefining `\Youngstyle#1{..}`.

```
\setYstyle{\bf}\Young{a&b\\c}
```

$$\begin{array}{|c|c|}\hline \mathbf{a}&\mathbf{b}\\\hline \mathbf{c}\\\cline{1-1}\end{array}$$

```
\def\Youngstyle#1{#1^{#1}}
\Young{a&b\\c}
```

$$\begin{array}{|c|c|}\hline a^a&b^b\\\hline c^c\\\cline{1-1}\end{array}$$

For example, in case one want to investigate the relation between tableaux and Chemistry.

$$\begin{array}{|c|c|c|c|c|c|c|c|}\hline \mathbf{H}&&&&&&&\mathbf{He}\\\hline \mathbf{Li}&\mathbf{B}&\mathbf{Be}&\mathbf{C}&\mathbf{N}&\mathbf{O}&\mathbf{F}&\mathbf{Ne}\\\hline \mathbf{Na}&\mathbf{Mg}&\mathbf{Al}&\mathbf{Si}&\mathbf{P}&\mathbf{S}&\mathbf{Cl}&\mathbf{Ar}\\\hline \mathbf{K}&\mathbf{Ca}\\\cline{1-2}\end{array}$$

We provide a way to individually modify each single box, say, writing the formats assumption before `||` (or `[]` if applicable).

```
\Young{x&\setYstyle{\bf}||y\\\bf z}
```

$$\begin{array}{|c|c|}\hline x&\mathbf{y}\\\hline \mathbf{z}\\\cline{1-1}\end{array}$$

We will see later what `|..|` mean. Note that in the above example, the results for **y** and **z** are the same. But it does make a difference in the following example.

```
\Young{&\Ythicker||&\\&&}
```

$$\begin{array}{|c|c|c|}\hline &\boldsymbol{|}&\\\hline &&\\\cline{1-3}\end{array}$$

This is useful when the color is not allowed.

$$\begin{array}{|c|c|}\hline 1&1\\\hline 2\\\cline{1-1} 3\\\cline{1-1}\end{array}\xrightarrow{1}\begin{array}{|c|c|}\hline 1&2\\\hline 2\\\cline{1-1} 3\\\cline{1-1}\end{array}\xrightarrow{2}\begin{array}{|c|c|}\hline 1&3\\\hline 2\\\cline{1-1} 3\\\cline{1-1}\end{array}$$

Actually when it involves colors, there are some subtle difference, compare

```
\Young{x&\color{red}||y\\z}
```

$$\begin{array}{|c|c|}\hline x&\color{red}{y}\\\hline z\\\cline{1-1}\end{array}$$

```
\Young{x&||\color{red}y\\z}
```

$$\begin{array}{|c|c|}\hline x&\color{red}{y}\\\hline z\\\cline{1-1}\end{array}$$

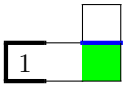
4 Strips and Hooks

This section is the main upload of the last version TooYoung. Now we provide ways to make boxes not necessarily surrounded by four walls. The grammar is

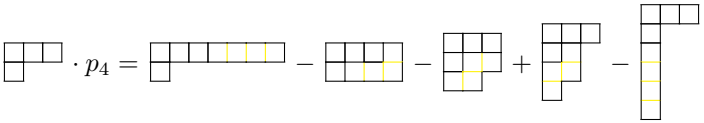
`(formats) | walls | (contents)` or `walls | (contents)` .

For example

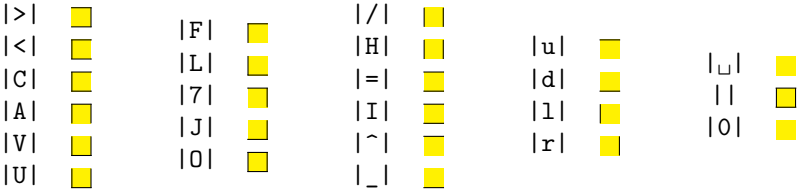
```
\Young{[]&[]&ulr|\\
\Ythicker|udl|1&du|&
{\color{blue}\Ythicker\wall{u}}
|dr|*(green)}
```



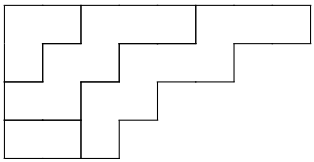
Note that `wall{..}` illustrate correctly before `|..|` or `[..]`. For example, Murgnahan–Nakayama rule



The author would kindly warn that the commands `|..|` and `[..]` are not supposed to used simultaneously. Moreover, we provide the following syntactic sugars

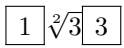


Moreover, all box drawings symbol (double) can be used if your compile is XeLaTeX which allows unicodes. For example, `U+2569` (see `https://unicode-table.com/en/2569/`) gives ╩. Here is an example (it will fail using pdfLateX)



If one uses more than one `[..]`'s, only the first one contributes.

```
\Young{1&[]\sqrt[2]{3}&3}
```



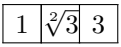
If one use more than two `|`, only the first two count.

```
\Young{|u|d|l|r}
```

$\overline{d|l|r}$

If one use both `|` and `[..]`, then `|` takes priority.

```
\Young{1&|\sqrt[2]{3}&3}
```



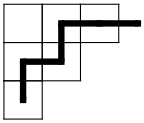
So it provides another way to protect commands involving `[..]`, though it is not recommended. The `*(..)` can be inserted anywhere but only the first counts.

```
\Young{|u*(red)d|l*(blue)r|}
```

$l * (blue)r|$

We can use `\hook{..}` in a tableau,

```
\Young{
&\hook{rd}&\hook{rl}&[\hook{l}]\\
\hook{dr}&\hook{ul}\\\hook{u}}
```



Moreover, we provide syntactic sugars

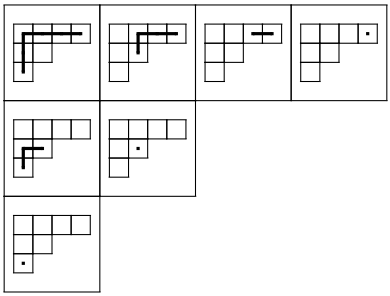
```
\hook{-}  \hook{--}  \hook{/}  \hook{|}  \hook{+}
\hook{F}  \hook{L}  \hook{7}  \hook{J}  \hook{T}
\hook{^}  \hook{_}  \hook{- }  \hook{ -}
\hook{n}  \hook{e}  \hook{w}  \hook{s}  \hook{}
```

Moreover, all box drawings symbol (light) can be used, for example `U+252C` (see `https://unico` gives  .

Lastly, we can use `\sethookth` to set the thickness of the hook.

```
\sethookth{0.5pc}
\setYstyle{\color{blue}}
\Young{\hook{-}&\hook{|}\\
\hook{|}&\hook{-}}
```

Note that our package admits a useful feature that we can insert tableaux into tableaux, for example,



5 Summary

We provide

`\Young[size]{...&...\\...}` or `\young{...&...\\...}`;

`\TooYoung{...&...\\...}`.

In each cell, the grammar is

`formats | walls | contents`.

Note that

```
                    contents = {} |udlr| contents;
formats [contents] contents = formats |0| contents;
          walls | contents = {} | walls | contents.
```

Inside `\Young`, `\young` or `\TooYoung`, we provide

`\wall{..}`, `\hook{..}`, and `\black`.

To modify the dimensions and styles, we have

`\setYsize`, `\setYwidth`, `\setYheight`, `\Ybigger`, and `\Ysmaller`;

`\setYth`, `\setYhth`, `\setYvth`, `\Ythicker`, and `\Ythinner`;

`\sethookth`;

`\setYstyle`.