

抠出 yhmath 的部分东西

kuing

January 20, 2014

1 为何要抠?

yhmath 宏包除了提供弧符号、特大根号、特大括号等等的东西外，其实还将数学符号中的 largesymbols 部分全部改成用他自己的符号，除了特大型的东西外，其他的虽然看起来都跟原先 L^AT_EX 默认的差不多，但是细细比较下还是有不少差别的，之前在群里跟 555 聊起 yhmath 时就对比过那些括号的大小和位置差别，下面来看看。

这段公式代码：

```

$$\left(\frac{ab}{right}\right)+\left(\frac{1x}{right}\right)+\left(\frac{ay}{right}\right)+\left(\frac{1y}{right}\right)+\left(\frac{\sqrt{2}c}{right}\right)+\left(\frac{\sqrt{\sqrt{2}}c}{right}\right)$$

```

在 L^AT_EX 默认状态下显示为：

$$\left(\frac{a}{b}\right) + \left(\frac{1}{x}\right) + \left(\frac{a}{y}\right) + \left(\frac{1}{y}\right) + \left(\frac{\sqrt{2}}{c}\right) + \left(\frac{\sqrt{\sqrt{2}}}{c}\right)$$

加载 yhmath 后变为：

$$\left(\frac{a}{b}\right) + \left(\frac{1}{x}\right) + \left(\frac{a}{y}\right) + \left(\frac{1}{y}\right) + \left(\frac{\sqrt{2}}{c}\right) + \left(\frac{\sqrt{\sqrt{2}}}{c}\right)$$

嗯，这差别的确是没什么所谓，yhmath 做得还更细致的样子。但是，我们来看特大的括号的情形，比如说常见的矩阵。

默认：

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

加载 yhmath 后：

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

咦？为什么加载 yhmath 后的第二个矩阵没有使用 yhmath 的特大括号？居然变成跟默认一样？那是因为，yhmath 虽然提供了特大的括号，但是始终有上限，不会无穷大，当超过上限高度时，就会用回默认了，于是当上面这样的式子出现时，括号就不统一，显得有点不和谐的感觉。

当然，你或者觉得括号不统一也没什么，那我们再来看看根号会怎么样。

默认:

$$\sqrt{\frac{2x}{y+z}} + \sqrt{\frac{2y}{z+x}} + \sqrt{\frac{2z}{x+y}}$$

加载 yhmath 后:

$$\sqrt{\frac{2x}{y+z}} + \sqrt{\frac{2y}{z+x}} + \sqrt{\frac{2z}{x+y}}$$

请留意中间的根式，差别很明显了，如果你还是觉得无所谓，那你可以直接加载 yhmath 了，反正我呢就是接受不鸟嘴。

但是，yhmath 的弧符号和特大根号的确是有用的，弧符号就不用说了，默认根本就没有，而特大根号指的是这种：

$$\sqrt{\sum_{i=1}^n a_i} + \sqrt[n]{\sum_{i=1}^n a_i}$$

效果做得很好，它在默认情况下是这样的：

$$\sqrt{\sum_{i=1}^n a_i} + \sqrt[n]{\sum_{i=1}^n a_i}$$

于是我就想在不加载 yhmath 的情况下，将它们单独抠出来用，尽量不影响原先的默认符号。

2 抠出弧符号

先看看 yhmath 宏包的内容，里面内容很短，首先看到的是

```
\DeclareSymbolFont{largesymbols}{OMX}{yhext}{m}{n}
```

隐约记得 `\DeclareSymbolFont` 的意思，虽然不是很清晰，但是应该就是这句，将 `largesymbols` 这类符号全改变了，再看后面的，都是新定义符号，找到弧符号的定义方式是

```
\DeclareMathAccent{\wideparen}{\mathord}{largesymbols}{"F3}
```

于是，要抠出 yhmath 的弧符号就很简单了，只要在导言区加上

```
\DeclareSymbolFont{yh}{OMX}{yhext}{m}{n}
```

```
\DeclareMathAccent{\hu}{\mathord}{yh}{"F3}
```

这样，`$$\hu{AB}+\hu{ABC}+\hu{ABCD}$$` 就会给出 $\widehat{AB} + \widehat{ABC} + \widehat{ABCD}$ ，而且 `largesymbols` 没被改变，亦即不会影响所有数学符号。

要想抠出分界符的话，方法也差不多，不过它们的定义就要到 `amsmath` 或者 `LATEX 2ε` 的说明文档 (`source2e.pdf`) 里面找，只要是用到 `largesymbols` 的都可以类似变更，不过其实也没什么必要变那些，所以这里就不细讲了。

3 抠出根号

这个比较复杂，虽然我们能够在 `source2e.pdf` 里找到根号的定义，但是后来发现 `amsmath` 重写了部分，好像是为了控制 n 次根号的指数位置，以致于代码变得更复杂了。

把相关代码看来看去，可以肯定的是根号所使用的符号最终是通过

```
\DeclareMathRadical{\sqrtsign}{symbols}{"70}{largesymbols}{"70}
```

来确定的，也就是说，如果将 `largesymbols` 改成 `yh` 就可以将根号换成 `yhmath` 的根号了，但是，我现在只要特大的根号，因此这种方式并不是我想要的。（否则我干脆就加载 `yhmath` 算了）

想来想去，干脆自己重新写一套出来，这里我就仿照 `source2e.pdf` 里面的来写，比如：

```
\makeatletter
\DeclareMathRadical{\kksqrtsign}{symbols}{"70}{yh}"70}

\newbox\kkrootbox

\def\kkroot#1\kkof{%
\setbox\kkrootbox\hbox{\m@th\scriptscriptstyle{#1}$}%
\mathpalette\kk@t}

\def\kk@t#1#2{%
\setbox\z@\hbox{\m@th#1\kksqrtsign{#2}$}%
\dimen@ \ht\z@ \advance\dimen@-\dp\z@
\mkern5mu\raise.6\dimen@\copy\kkrootbox
\mkern-10mu\box\z@}

\DeclareRobustCommand\kksqrt{\@ifnextchar[\kk@sqrt\kksqrtsign}
\def\kk@sqrt[#1]{\kkroot #1\kkof}
\makeatother
```

（注：将上述代码中的所有 `kk` 去掉就是 `source2e.pdf` 里面关于根号的定义）

这样就不会影响原先的根号，也就是说现在会有两套根号——默认的 `\sqrt` 和 `yhmath` 的 `\kksqrt`，随你用哪个。

下面来测试一下，代码：

```
\begin{gather*}
\sqrt{\sum_{i=1}^n a_i}=\kksqrt{\sum_{i=1}^n a_i}
=\sqrt[n]{\sum_{i=1}^n a_i}=\kksqrt[n]{\sum_{i=1}^n a_i}
\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{1}}}}}}
=\kksqrt{\kksqrt{\kksqrt{\kksqrt{\kksqrt{\kksqrt{\kksqrt{\kksqrt{1}}}}}}}}
=\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{1}}}}}}}
=\kksqrt[3]{\kksqrt[3]{\kksqrt[3]{\kksqrt[3]{\kksqrt[3]{\kksqrt[3]{\kksqrt[3]{1}}}}}}}\
\sqrt{\frac{2x}{y+z}}+\sqrt{\frac{2y}{z+x}}+\sqrt{\frac{2z}{x+y}}
=\kksqrt{\frac{2x}{y+z}}+\kksqrt{\frac{2y}{z+x}}+\kksqrt{\frac{2z}{x+y}}
\end{gather*}
```

显示：

$$\sqrt{\sum_{i=1}^n a_i} = \sqrt[n]{\sum_{i=1}^n a_i} = \sqrt[n]{\sum_{i=1}^n a_i} = \sqrt[n]{\sum_{i=1}^n a_i}$$

$$\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{\sqrt{1}}}}} = \sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{1}}}}} = \sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{\sqrt[3]{1}}}}}}}$$

$$\sqrt{\frac{2x}{y+z}} + \sqrt{\frac{2y}{z+x}} + \sqrt{\frac{2z}{x+y}} = \sqrt{\frac{2x}{y+z}} + \sqrt{\frac{2y}{z+x}} + \sqrt{\frac{2z}{x+y}}$$

暂时没发现什么问题，这样就可以平时用 `\sqrt`，遇到根号内的东东很高，需要特大根号时才用 `\kksqrt`。

其实我还有一个想法，就是再定义一个命令，让其根据根号内的东东的总高度自动判断用哪套根号，不过在设计时遇到些困难，而且似乎也不是很有必要这样做，毕竟要用特大根号的情况相对来说很少，暂时先放着吧，有空再续……

4 另一种方法，反过来

前面所讲的方法都是在不加载 `yhmath` 的情况下将部分东西抠出来用，这适合于小量的抠，比如像我只要弧符号和特大根号。而如果想抠的部分很多，比不抠的还多，那要写的代码就很长了，这时应该反过来想，即在加载 `yhmath` 的情况下将不想被改变的东西保留下来，比如仅仅不想改变小括号和根号，那可以这样：

```
\usepackage{yhmath}
\DeclareSymbolFont{oldlargesymbols}{OMX}{cmex}{m}{n}
\DeclareMathRadical{\sqrtsign}{symbols}{"70}{oldlargesymbols}{"70}
\DeclareMathDelimiter{(\}{\mathopen}{operators}{"28}{oldlargesymbols}{"00}
\DeclareMathDelimiter{)}{\mathclose}{operators}{"29}{oldlargesymbols}{"01}
```

此时如果又想用特大根号，也可以将那套 `\kksqrt` 抄上去，不过就要改点东西，`yh` 改成 `largesymbols`。时间关系不解释那么多了，测试的事自己来吧……闪先……