

longmath

Documentation for Version 0.1

Hans-Jürgen Matschull

hjm.tex@matschull.net

2024/03/25

The longmath package provides yet another solution to some well known typesetting problems solved in a variety of ways: multi line formulas with paired and nested delimiters. It tackles the problem at the Lua level, which has some advantages over solutions implemented in \TeX . In particular, the \TeX code need not be executed multiple times, and there is no interference between \TeX grouping and the nesting of delimiter groups.

As a byproduct, delimiters can be scaled in various ways, inner delimiters come in different flavours like relational and binary operators, punctuation symbols etc., and outer delimiters can be selected automatically according to the nesting level. Last but not least, delimiter groups can even extend across several array cells or across the whole document.

A special environment is provided as well, which allows multi line expressions to be placed inside a displayed equation and make \TeX do the line splitting and alignment.

This is the first and in parts still experimental version. If anyone actually uses it, the author would be happy to receive bug reports, critics and suggestions.

1 Delimiters

The `longmath` package provides “long” versions of the `\left` and `\right` primitives.

- `\lleft [scale] delimiter` marks the beginning of a delimiter group and inserts an opening *delimiter*.
- `\rright [scale] delimiter` marks the end of a delimiter group and inserts a closing *delimiter*.

The optional *scale* argument will be explained below. The *delimiter* can be anything that is also accepted by `\left` and `\right`, including a dot representing an invisible or null delimiter, any character with a `\delcode`, or an explicit `\delimiter` or `\Udelimiter`. If you want to use the `longmath` package with existing documents, or just stick to the familiar notation, and provided that this does not cause conflicts with other packages manipulating the original primitives, it is even save to say `\let\left\lleft` and `\let\right\rright`.

<p>The Sellmann-Yokuzi contribution to the Johansson Entropy is given by</p> $S^{\text{top}} = k \cdot \left[A + \left(B_C + D^{E^2} \right)^{3+\kappa^2} + F \right]$	<p>The Sellmann-Yokuzi contribution to the Johansson Entropy is given by</p> $S^{\text{T}} = k \cdot \left[A + \left(B_C + D^{E^2} \right)^{(3+\kappa^2)^2} + F \right]$
--	--

The `\lleft` and `\rright` delimiters enclose a *delimiter group*. Within the group, you can have arbitrarily many inner or middle delimiters. As these often have different interpretations, several “long” variants of the `\middle` primitive are provided, associated with different types of math atoms.

- `\mmiddle [scale] delimiter` inserts an inner *delimiter* into a delimiter group.
- `\mmrel [scale] delimiter` interprets the *delimiter* as a relational operator.
- `\mmbin [scale] delimiter` interprets the *delimiter* as a binary operator.
- `\mmord [scale] delimiter` interprets the *delimiter* as an ordinary symbol.
- `\mmop [scale] delimiter` interprets the *delimiter* as a large operator.
- `\mmpunct [scale] delimiter` interprets the *delimiter* as a punctuation character.

They differ in the way TEX inserts spaces before and after these delimiters, just as if you had written `\mathrel{delimiter}`, `\mathbin{delimiter}`, etc. Like the `\middle` primitive, the `\mmiddle` variant creates a `\mathinner{delimiter}`.

<p>The scalar product is written as</p> $\left\langle A_1 \middle \tilde{A}_2 \right\rangle .$	<p>The scalar product is written as</p> $\langle A_1 \tilde{A}_2 \rangle .$
<p>There are three parallel lines</p> $A_1 B_1 \parallel A_2 \hat{B}_2 \parallel \hat{A}_2 B_2 .$	<p>There are three parallel lines</p> $A_1 B_1 \parallel A_2 \hat{B}_2 \parallel \hat{A}_2 B_2 .$
<p>The centre of mass is defined as</p> $\vec{R} = \left(\sum_{k=1}^n m_k \vec{r}_k \middle/ \sum_{k=1}^n m_k \right) .$	<p>The centre of mass is defined as</p> $\vec{R} = \left(\sum_{k=1}^n m_k \vec{r}_k \middle/ \sum_{k=1}^n m_k \right) .$
<p>Vectors can be written as tuples like</p> $\left(\tilde{X}_1 \middle \tilde{X}_2 \middle \tilde{X}_3 \middle \dots \right) .$	<p>Vectors can be written as tuples like</p> $(\tilde{X}_1 \tilde{X}_2 \tilde{X}_3 \dots) .$
<p>This expression makes no sense:</p> $\left(\int \prod \left(\sum \prod \right) \right) .$	<p>This expression makes no sense:</p> $\int (\sum [\prod])$

Unlike the standard `\left`-`\middle`-`\right` primitives, the long variants do not create subformulas or induce grouping at the TEX level. TEX nically, each delimiter is actually its own group, paired with an invisible delimiter, and with an empty `\hbox` between them. As such, it behaves like a single mathematical symbol of the respective type, or like an

atom in $\text{T}_\text{E}\text{X}$'s language. In particular, sub- and superscripts can be attached to individual delimiters, not only to the whole group, and the operator version really behaves like a big operator.

```
Applying the triangular square limit, we get

$$\lim_{n \rightarrow \infty} \frac{1-x^n}{1-x} = \log(x).$$

```

Applying the triangular square limit, we get

$$W(x) + \int_n^\infty \frac{1-x^n}{1-x} = \log(x).$$

```
The relations 1^* and 2^* imply the existence of

$$\left[ \left( A_k \mid B_k \mid C_k \right)^{(1^*)} \right]^{(2^*)}$$

```

The relations 1^* and 2^* imply the existence of

$$\left[(A_k \mid^{(1^*)} B_k \mid^{(2^*)} C_k) \right].$$

Within a formula or subformula, you can have unbalanced groups, or even inner delimiters only. The formula is then interpreted and typeset as if there were invisible delimiters at one or both ends.

```

$$\prod_{p \in P} G(p) \uparrow \sum_{q \in Q} H(q) \downarrow \lim_{x \rightarrow \infty} F(x)$$

```

$$\prod_{p \in P} G(p) \uparrow \sum_{q \in Q} H(q) \downarrow \lim_{x \rightarrow \infty} F(x)$$

```
A skewed-rational fraction can be written as

$$\frac{x \times (1 - x^2)}{x^2 + 1} \times x$$

```

A skewed-rational fraction can be written as

$$\frac{x \times (1 - x^2)}{x^2 + 1} \times x.$$

When the formula is processed, each actual delimiter group is temporarily put in a box and the size of that box is determined. Then the dimensions of the empty `\hboxes` inside the fake delimiter groups are set accordingly, so that $\text{T}_\text{E}\text{X}$ can apply its algorithm to finally fix the sizes of the delimiters. As this takes place after the whole formula has been transformed into a list of math nodes, no $\text{T}_\text{E}\text{X}$ code has to be processed multiple times. It also allows to add some features that cannot be implemented otherwise, unless you are willing to go really deep into macro and token magic.

For example, the limits of large operators like sums or integrals can be excluded from the box that determines the delimiter size. This feature can be turned on and off with two control sequences.

- `\includelimits` includes the limits of large operators (default).
- `\ignoreslimits` ignores the limits of large operators.

It is possible to use the switch in the middle of a formula or even inside a delimiter group. The switch position at the moment the operator is created determines whether its limits are included or ignored.

```

$$\vec{R} = \left( \sum_{k=1}^n m_k \vec{r}_k \mid \sum_{k=1}^n m_k \right)$$

```

$$\vec{R} = \left(\sum_{k=1}^n m_k \vec{r}_k \mid \sum_{k=1}^n m_k \right).$$

```

$$\left[ \sum_{A_X^3} F(Z) + \sum_a^c f(x) \right]$$

```

$$\left[\sum_{A_X^3} F(Z) + \sum_a^c f(x) \right]$$

```

$$A = 3 \times \int_1^5 (1 + \sqrt{1+x}) \, dx$$

```

$$A = 3 \times \int_1^5 (1 + \sqrt{1+x}) \, dx$$

```

$$A = 3 \times \int_1^5 (1 + \sqrt{1+x}) \, dx$$

```

$$A = 3 \times \int_1^5 (1 + \sqrt{1+x}) \, dx$$

There are several ways to fine tune the sizes of delimiters. A priori, the minimal size of all delimiters within a group is determined by the total height and depth of everything in this group, except for the delimiters themselves. $\text{T}_\text{E}\text{X}$ then

checks if a predefined character with the required size exists in the current math font and uses it. Otherwise it creates a stretched symbol with the required size.

Two plain TeX parameters modify the finally chosen sizes of all delimiters. By default, they are used to shrink them slightly, which makes the final formula look less bulky while keeping it readably. This can be seen in the integral example above. An additional third parameter is provided by the `longmath` package, which applies to middle delimiters only.

- `\delimiterfactor` *factor* scales all delimiters by *factor*/1000 (default 901).
- `\delimitershortfall` *dimension* limits the shrinkage of delimiters to the *dimension* (default 5.0pt).
- `\delimiterscale` *factor* additionally scales middle delimiters by *factor*/1000 (default 902).

The parameters `\delimiterfactor` (a count register) and `\delimitershortfall` (a dimension register) are explained in the TeXbook. Briefly summarised, every delimiter is finally resized by the given *factor*/1000, but the shrinkage is limited to the given *dimension*. The `\delimiterscale` (another count register) is applied to all middle delimiters before the final scaling is applied, and there is no limited shrinkage at this stage.

<pre>\delimiterscale 400 \$\$ \vec R = \lleft(\sum_{k=1}^n m_k \, \vec r_k \, \mmbin / \sum_{k=1}^n m_k \, \rrright) . \$\$</pre>	$\vec{R} = \left(\sum_{k=1}^n m_k \vec{r}_k / \sum_{k=1}^n m_k \right).$
--	---

Instead of changing `\delimiterscale`, the scale factor can also be applied to each delimiter individually, using the optional *scale* argument. In this case, the `\delimiterscale` is ignored and replaced by the given number.

<pre>\$\$ X = \lleft(\sum_k^{\infty} A \, \mmbin 900 \sum_k^{\infty} B \, \mmbin 700 \sum_k^{\infty} C \, \mmbin 500 \sum_k^{\infty} D \, \mmbin 300 \sum_k^{\infty} E \, \rrright) \$\$</pre>	$X = \left(\sum_k^{\infty} A \mid \sum_k^{\infty} B \mid \sum_k^{\infty} C \mid \sum_k^{\infty} D \mid \sum_k^{\infty} E \right)$
--	--

If you want to produce strange and ugly results, you can even scale the opening and closing delimiters independently, or choose factors greater than 1000.

<pre>\$\$ \vec R = \lleft 900 (\sum_{k=1}^n m_k \, \vec r_k \, \mmbin 1300 \divslash \sum_{k=1}^n m_k \, \rright 700) \$\$</pre>	$\vec{R} = \left(\sum_{k=1}^n m_k \vec{r}_k \, / \, \sum_{k=1}^n m_k \right)$
--	--

2 Alignment

There are two more special delimiter types not listed in section 1. They are treated as middle delimiters regarding the logical grouping, but when they are finally typeset, they are treated as opening and closing delimiters.

- `\mleft` [*scale*] *delimiter* typesets a left *delimiter*, but does not open a new group.
- `\mright` [*scale*] *delimiter* typesets a right *delimiter*, but does not close the current group.

To see what they are useful for, consider the following example.

<pre>\$\$ \lleft(A + \sum B \, \rrright) \lleft(C + \int D \, \rrright) \$\$</pre>	$(A + \sum B)(C + \int D)$
--	----------------------------

Can we make the brackets to have the same size? A few `\vphantoms` would probably fix it. But then we need to know in advance which symbols are the responsible ones, or we have to place the content of each bracket as a phantom inside the other one and make TeX process everything twice. Alternatively, we could include everything in a single group and make the two brackets in the middle inner delimiters. But then the correct spacing between them and the adjacent symbols is lost.

<pre>\$\$ \lleft(A + \sum B \, \mmiddle) \mmiddle (C + \int D \, \rrright) \$\$</pre>	$(A + \sum B)(C + \int D)$
--	----------------------------

With `\mleft` and `\mright`, we get the same sizes and the correct spacing. Since in this case we only have these two pairs of brackets, we can simple make the whole formula a single group.

```


$$\left( A + \sum B \right) \left( C + \int D \right)$$


```

$$(A + \sum B)(C + \int D)$$

What `\left` and `\right` basically do is not to adapt the opening and closing delimiters to the content between them, but align them with the enclosing group.

```


$$\left( \left( 1 + 2 \right) \left( 1 + 2^2 \right) \left( 1 + 2^{2^2} \right) \right) + \left( 1 + 3 \right) \left( 1 + 3^3 \right) + \left( 1 + 5 \right) \left( 1 + 5^5 \right)$$


```

$$\left(\left(1 + 2 \right) \left(1 + 2^2 \right) \left(1 + 2^{2^2} \right) \right) + \left(1 + 3 \right) \left(1 + 3^3 \right) + \left(1 + 5 \right) \left(1 + 5^5 \right)$$

But what if there is more stuff inside the enclosing group and we only want to align particular brackets with each other? Something like this, and we want both the square and the round brackets to be aligned.

```


$$F \left[ X \right] \left( 1 + 2^2 \right) \left( 1 + 2^{2^2} \right) + F \left[ X^2 \right] \left( 1 + 3^3 \right) \left( 1 + 3^{3^3} \right) + F \left[ X^3 \right] \left( 1 + 5^5 \right) \left( 1 + 5^{5^5} \right)$$


```

$$F[X](1 + 2^2)(1 + 2^{2^2}) + F[X^2](1 + 3^3)(1 + 3^{3^3}) + F[X^3](1 + 5^5)(1 + 5^{5^5})$$

Now we definitely need `\vphantoms`, right? In principle, yes, but the `longmath` package provides a way to set them up automatically and hide them inside the delimiters. To use this feature, we have to attach a tag to those delimiters we want to align. The tag is enclosed in braces and placed between the control sequence and the delimiter symbol.

- `\type {tag} [scale] delimiter` aligns this *delimiter* with all delimiters with the same *tag*.

Here, `\type` is any of the above delimiter types, and the `tag` is any string that would also do as, say, an argument to `\label`. The basic idea is to make all delimiters with the same `tag` attached to them behave as if they belong to the same group. So, the problem just posed can be solved as follows.

```


$$F \left[ X \right] \left( 1 + 2^2 \right) \left( 1 + 2^{2^2} \right) + F \left[ X^2 \right] \left( 1 + 3^3 \right) \left( 1 + 3^{3^3} \right) + F \left[ X^3 \right] \left( 1 + 5^5 \right) \left( 1 + 5^{5^5} \right)$$


```

$$F[X](1 + 2^2)(1 + 2^{2^2}) + F[X^2](1 + 3^3)(1 + 3^{3^3}) + F[X^3](1 + 5^5)(1 + 5^{5^5})$$

It is enough to tag only the left delimiters, or just one delimiter per group, because those belonging to the same group are aligned anyway. What we actually do is to align groups with each other, not individual delimiters.

The groups being linked need not belong to a common subformula. Here we want to align the brackets and the division slashed in the numerators.

```


$$\frac{\left( a + 1 \right)}{a} + \frac{\left( \sum_k b^k \right)}{b} + \frac{\left( c^2 + 1 \right)}{c}$$


```

$$\frac{(a + 1)/x}{a} + \frac{(\sum_k b^k)/y}{b} + \frac{(c^2 + 1)/z}{c}$$

This is how it can be done.

```


$$\frac{\left( a + 1 \right)}{a} + \frac{\left( \sum_k b^k \right)}{b} + \frac{\left( c^2 + 1 \right)}{c}$$


```

$$\frac{(a + 1)/x}{a} + \frac{(\sum_k b^k)/y}{b} + \frac{(c^2 + 1)/z}{c}$$

Note that the division slashes are outside the linked groups. But in each numerator, the slash belongs to the enclosing group, which is the subformula formed by the numerator. As this group also contains the brackets, the size of the slash is also adapted to them and thus all slashes get the same size as well. Alternatively, we could have done it like this.

```


$$\frac{(a+1)}{x} + \frac{(\sum_k b^k)}{y} + \frac{(c^2+1)}{z}$$


```

$$\frac{(a+1)/x}{a} + \frac{(\sum_k b^k)/y}{b} + \frac{(c^2+1)/z}{c}$$

But not like this, which aligns the slashes, but not the brackets.

```


$$\frac{(a+1)}{x} + \frac{(\sum_k b^k)}{y} + \frac{(c^2+1)}{z}$$


```

$$\frac{(a+1)/x}{a} + \frac{(\sum_k b^k)/y}{b} + \frac{(c^2+1)/z}{c}$$

For the alignment to work, the linked delimiters need not be visible. An invisible inner delimiter basically behaves like a `\vphantom`, whose content is the whole group to which this delimiter belongs. More precisely, it is actually the empty `\hbox` that is present in our special delimiters which replaces the `\vphantom`. We can use this to align the square roots in the following example.

```


$$\sqrt{a+1} + \sqrt{\sum_k b_k} + \sqrt{c^2+1}$$


```

$$\sqrt{a+1} + \sqrt{\sum_k b_k} + \sqrt{c^2+1}$$

Why did we use `\mleft` here, and not one of the middle delimiters? The first reason is that putting an opening atom at the beginning of a subformula will get the horizontal spacing correct, independent of what kind of symbol follows. The second reason is that a middle delimiter would be scaled by `\delimiterscale`, so that it wouldn't have the required size to align the heights of the square roots. The `\delimiterfactor` is irrelevant here, because it doesn't affect an invisible delimiter.

We can even go a step further and link delimiters in different equations and, in fact, all across the document.

```

Equations that start with

$$\int_1^\infty f(x) dx + \sum_k a_k + 1000 + 999 + \dots$$

often go on for pages and end with

$$\dots + 1 = 0$$


```

```

Equations that start with

$$\left( \int_1^\infty f(x) dx \right) \left( \sum_k a_k + 1000 + 999 + \dots \right)$$

often go on for pages and end with

$$\dots + 1 = 0$$


```

To make this possible, the \LaTeX method of storing information in the aux file is used. The dimensions of all tagged delimiter groups are saved and reloaded when the document is typeset the next time. It therefore requires a second run to get the sizes of linked delimiters correct. If the size of any delimiter changes while the typesetting is in progress, a warning message is issued at the end, similar to the message that you need to rerun the document to get all the `\references` to `\labels` right.

You will also get a warning message if you set up the tags so that some delimiter group is linked to its own parent or subgroup. In such a case, there is no consistent way to determine the required sizes. With each run, they will change randomly or even start to increase and diverge. It's okay, however, to link delimiters in different math styles. You can link an exponent in script style to its own basis in display style, although this might not make much sense.

For all this to work, the tags must be document wide unique. But unlike equation labels or bibliography entries, they typically only link delimiters within a single equation or an array of equations. It is therefore preferable to have short tags like the ones used above, or even simple letters or numbers. To make this possible, you can locally define a *prefix*, which is added to each tag used within its scope.

- `\delimiterprefix{prefix}` prepends the *prefix* to the tags of all tagged delimiters.

It is then sufficient to use a document wide unique prefix for each section, equation, or whatever scope suits best. The setting is local to the \TeX group, but if the same prefix is used at different places, the tagged delimiters within those scopes are linked to each other. For example, if the prefix is `introduction-` and the tag `a2` is attached to a delimiter, then the actual tag used internally is `introduction-a2`, and this is a global name.

You can even automate this, by setting a dynamical prefix like this in the preamble.

```
\delimiterprefix{\theequation.}
```

Provided that you use unique equations numbers, and link delimiters within single equations only, your tags will then always be local to the current equation, something like 5.16.a if the tag a is used in equation (5.16).

The perhaps most useful application of tagged delimiters is to typeset formulas that extend over several cells or lines in an array. Here is a little challenge. In the following array, we want the multiplication and addition operators to be aligned vertically, and of course the brackets to be adapted to the content they enclose. We use the `array` package to define a few special column types to get the math style and the spacing correct.

```
\newcolumntype{d}{ >{\displaystyle} c }
\newcolumntype{s}{ @{} >{\}} c <{\}} @{} }
```

```
$$ \begin{array}{dsdsdsd}
  \lleft [ Z & - & \sum_k A & \times & & & \\
  \lleft ( B & + & C^X \rright ) \rright ] \ll[3ex]
  \lleft [ P & \times & \lleft ( U & + & & & \\
  \int_R V \rright ) & - & W^Y \rright ] \\
\end{array} $$
```

$$\left[Z - \sum_k A \times (B + C^X) \right]$$

$$\left[P \times \left(U + \int_R V \right) - W^Y \right]$$

In an `array`, each cell is typeset as a separate formula. Therefore, the delimiters in each cell are adapted only to the content of this particular cell. Apparently, it would be quite a complicated task to fix this with `\vphantoms`. Let's try to simply link the delimiters that belong together.

```
$$ \begin{array}{dsdsdsd}
  \lleft {a1} [ Z & - & \sum_k A & \times & & & \\
  \lleft {b1} ( B & + & C^X \rright {b1} ) \rright {a1} ] \ll[3ex]
  \lleft {a2} [ P & \times & \lleft {b2} ( U & + & & & \\
  \int_R V \rright {b2} ) & - & W^Y \rright {a2} ] \\
\end{array} $$
```

$$\left[Z - \sum_k A \times (B + C^X) \right]$$

$$\left[P \times \left(U + \int_R V \right) - W^Y \right]$$

A first step, but not the expected result. It seems that we have to look at it more closely and sort out which parts of which cells actually belong to the same groups. But there is in fact a simple solution that doesn't require any deeper thinking or sorting out. We just need to connect the end of each cell to the beginning of the next one. We can do this by inserting invisible delimiters.

```
$$ \begin{array}{dsdsdsd}
  \lleft [ Z \mright{c1}. & - & \lleft{c1}. \sum_k A \mright{c2}. \\
  & \times & \lleft{c2}. \lleft( B \mright{c3}. & + & \\
  \lleft{c3}. C^X \rright ) \rright ] \ll[3ex] \\
  \lleft [ P \mright{d1}. & \times & \lleft{d1}. \lleft ( U \mright{d2}. \\
  & + & \lleft{d2}. \int_R V \rright ) \mright{d3}. & - & \\
  \lleft{d3}. W^Y \rright ] \end{array} $$
```

$$\left[Z - \sum_k A \times (B + C^X) \right]$$

$$\left[P \times \left(U + \int_R V \right) - W^Y \right]$$

This is the desired output, but the code starts to look complicated. But it is a simple rule, so there should be a way to automate it. Can we include the invisible linking delimiters into the cell template? The problem is that we need a unique tag for each transition from one cell to the next. For this purpose, two macros are provided.

- `\pushdelimiter` inserts an invisible closing delimiter with an automatically generated unique tag.
- `\pulldelimiter` inserts an invisible opening delimiter with the previously generated tag.

The generated tags are stored on a stack, so that multiple push and pull operations can be nested. They are globally unique and survive the current \TeX group. If we include the macros into the cell template, they connect each cell to the next one. We can also define templates that start a new sequence or end the current one. As $\text{Lua}\TeX$ allows this, let's use Greek letters for the beginning, the middle, and the end of a row.

```
\newcolumntype{a}{ >{ \displaystyle } c <{ \pushdelimiter } }
\newcolumntype{μ}{ >{ \displaystyle \pulldelimiter } c <{ \pushdelimiter } }
\newcolumntype{ω}{ >{ \displaystyle \pulldelimiter } c }
```

Now the code is as simple as it was in the beginning. We can just ignore the cell boundaries and write the formula as if it was a single line in `$$`s, or whatever single line math environment.

```



$$\begin{array}{c}
\left[ Z - \sum_k A \times (B + C^X) \right] \\
\left[ P \times \left( U + \int_R V \right) - W^Y \right]
\end{array}$$



```

$$\begin{array}{c} \left[Z - \sum_k A \times (B + C^X) \right] \\ \left[P \times \left(U + \int_R V \right) - W^Y \right] \end{array}$$

If you want the formula to continue on the next line with pending delimiters, you can include the push and pull macros into the line separator as well.

```


\newcommand\continue[1][0pt]{ \pushdelimiter \[#1] \pulldelimiter }

$$\left\{ Z - \sum_k (A \times B^3) + \left( U + \int_R V \right) - W^Y \right\}$$



```

$$\left\{ Z - \sum_k (A \times B^3) + \left(U + \int_R V \right) - W^Y \right\}$$

A construction like this can also be used with `amsmath` environments like `multiline` or `split`, which require explicit line breaks to be inserted into long formulas. The `longmath` environment described in section 4 provides an alternative method to typeset long formulas without the need to specify the breakpoints manually.

3 Automatic Nesting

It is possible to select the kinds of brackets used as outer delimiters automatically depending on the nesting level. This might not add much value for formulas that are explicitly written into the document. But it could be useful for semi-automatically produced formulas, where the author doesn't have direct control over individual symbols.

The basic idea is to define a collection of delimiter pairs, and let \TeX pick them according to the current nesting level.

- `\autodelimiters{sample}` defines a collection of nested delimiters pairs.

The *sample* must be an otherwise empty, nested structure of delimiters enclosing an asterisk `*` as a single object.

```

\autodelimiters{ \lfloor [ \{ ( * ) \} ] \rceil }

```

Once this is set up, you can pick one of them by using `*` as a variable delimiter.

```


The corrected Sellmann-Yokuzi contribution to the
Johansson Entropy is given by

$$S^{\text{top}} = k \cdot \left( A + \left( B_C + \left( D^E \right)^2 \right)^{3+\kappa} + F \right)$$



```

The corrected Sellmann-Yokuzi contribution to the Johansson Entropy is given by

$$S^{\text{top}} = k \cdot \left[A + \left\{ B_C + (D^E)^2 \right\}^{(3+\kappa)^2} + F \right]$$

If the nesting level exceeds the number of sample pairs provided, the outermost pair is repeated.

```



$$\left( A + \left[ B + \left[ C + \left( D + \frac{(X+Y) \times Z}{W} \right) \cdot E \right] \cdot F \right] \cdot G \right)$$



```

$$\left(A + \left[B + \left[C + \left(D + \frac{(X+Y) \times Z}{W} \right) \cdot E \right] \cdot F \right] \cdot G \right)$$

If variable delimiters are mixed with explicit delimiters, there is an interaction between them. If you use one of the pairs from the collection, it is treated as if it was inserted automatically. Any enclosing bracket will be the next one from the collection. Let's define a larger collection to demonstrate this.

```

\autodelimiters{ < \lceil \lfloor [ \{ ( * ) \} ] \rfloor \rceil > }

```


This is what we get with automatic brackets only. The numbers indicate the actually used level. A negative value for `\delimitershortfall` makes the bracket size increase linearly with the nesting level.

```
\delimitershortfall -2pt
$$ \lleft* 5 + \lleft* 4 + \lleft* 3
+ \lleft* 2 + \lleft* 1 + A \rright* + B \rright*
+ C \rright* + D \rright* + E \rright* $$
```

$$\left[5 + \left[4 + \left[3 + \left\{ 2 + (1 + A) + B \right\} + C \right] + D \right] + E \right]$$

If we replace the second level by explicit square brackets, these are identified as the third level, and the counting continues from there.

```
$$ \lleft* 6 + \lleft* 5 + \lleft* 4
+ \lleft[ 3 + \lleft* 1 + A \rright* + B \rright]
+ C \rright* + D \rright* + E \rright* $$
```

$$\left\langle 6 + \left[5 + \left[4 + \left[3 + (1 + A) + B \right] + C \right] + D \right] + E \right\rangle$$

And here the counting restarts further out.

```
$$ \lleft* 2 + \lleft( 1 + \lleft* 3
+ \lleft* 2 + \lleft* 1 + A \rright* + B \rright*
+ C \rright* + D \rright) + E \rright* $$
```

$$\left\{ 2 + \left(1 + \left[3 + \left\{ 2 + (1 + A) + B \right\} + C \right] + D \right) + E \right\}$$

A pair that is not in the collection is ignored and the counting continues at the next level.

```
$$ \lleft* 4 + \lleft* 3 + \lleft\uparrow 0
+ \lleft* 2 + \lleft* 1 + A \rright* + B \rright*
+ C \rright\downarrow + D \rright* + E \rright* $$
```

$$\left[4 + \left[3 + \uparrow 0 + \left\{ 2 + (1 + A) + B \right\} + C \right] + D \right] + E \right]$$

If variable delimiters are tagged and aligned, this also applies to the chosen delimiter from the collection. Here is a typical example. Without tagging, we get this.

```
$$ \lleft * \frac{ n+1 }{ n^2 } \rright * ^2
+ \lleft * \frac{ n }{ \lleft * n - 1 \rright * ^2 }
\rright * ^2 $$
```

$$\left(\frac{n+1}{n^2} \right)^2 + \left\{ \frac{n}{(n-1)^2} \right\}^2$$

If the outer brackets are aligned, both become second level brackets.

```
$$ \lleft {qx} * \frac{ n+1 }{ n^2 } \rright * ^2
+ \lleft {qx} * \frac{ n }{ \lleft * n - 1 \rright * ^2 }
\rright * ^2 $$
```

$$\left\{ \frac{n+1}{n^2} \right\}^2 + \left\{ \frac{n}{(n-1)^2} \right\}^2$$

As a last challenge, let's consider the array from the previous section with a slight modification. If we replace all brackets with variable delimiters, but do not link the cells, we get this.

```
$$ \begin{array}{dsdsdsd}
\lleft * Z & - & \sum_k A & \times & & \\
\lleft * B & + & C^X \rright * & \rright * & \ll[3ex] \\
\lleft * P & \times & \lleft * \lleft * U^2 \rright * & & + & & \\
& \int_R V \rright * & & - & & W^Y \rright * & \\
\end{array} $$
```

$$\left(Z - \sum_k A \times (B + C^X) \right) \\ \left(P \times \left\{ (U^2) + \int_R V \right\} - W^Y \right)$$

As expected, only the brackets within each cell are set up properly, but with no relation to those in the other cells. If we use the version with linked cells, we get the intended result.

```
$$ \begin{array}{asmspsw}
\lleft * Z & - & \sum_k A & \times & & \\
\lleft * B & + & C^X \rright * & \rright * & \ll[3ex] \\
\lleft * P & \times & \lleft * \lleft * U^2 \rright * & & + & & \\
& \int_R V \rright * & & - & & W^Y \rright * & \\
\end{array} $$
```

$$\left\{ Z - \sum_k A \times (B + C^X) \right\} \\ \left[P \times \left\{ (U^2) + \int_R V \right\} - W^Y \right]$$

4 Multi Line Formulas

Unlike the \TeX primitives, the `longmath` macros do not create a subformula that cannot be broken into lines. In inline math mode, long formulas with delimiter groups can extend over several lines just like ordinary text.

```
\newcommand\polyA{ x^{10} + x^9 - x^8 + 2 x^6 + 9 x^5 + 2 x^4 - 4 x^3 - 2 x^2 + 6 x + 4 }
\newcommand\polyB{ x^4 - x^3 + x + 2 } \newcommand\polyC{ x^6 + 2 x^5 + x^4 - 2 x^2 + 2 x + 2 }
```

Factorising the polynomial
 $\$ P \llleftarrow(\text{\polyA} \rright) \$$
 gives $\$ P \llleftarrow(\text{\polyB} \rright)$
 $\cdot P \llleftarrow(\text{\polyC} \rright) \$$.

Factorising the polynomial $P(x^{10} + x^9 - x^8 + 2x^6 + 9x^5 + 2x^4 - 4x^3 - 2x^2 + 6x + 4)$ gives $P(x^4 - x^3 + x + 2) \cdot P(x^6 + 2x^5 + x^4 - 2x^2 + 2x + 2)$.

There is no interference between nested delimiter groups and logical \TeX groups. Delimiters can be placed deep inside `\begingroup-\endgroup` constructs, and they do not create such groups themselves. The following example uses the `multido` package. It wouldn't work with `\left`, `\right` and `\middle`.

Applying the Hofstad Zebelajev theorem, we conclude that
 $\$ \llleftarrow(\hat{A}_0 \uparrow \multido{\i=1+1}{10}{\mmbin \uparrowarrow \hat{A}_{\i}} \rright) \in \llleftarrow[\mathcal{H}^{\infty} \rright]^{\{11\}} \$$.

Applying the Hofstad Zebelajev theorem, we conclude that $(\hat{A}_0 \uparrow \hat{A}_1 \uparrow \hat{A}_2 \uparrow \hat{A}_3 \uparrow \hat{A}_4 \uparrow \hat{A}_5 \uparrow \hat{A}_6 \uparrow \hat{A}_7 \uparrow \hat{A}_8 \uparrow \hat{A}_9 \uparrow \hat{A}_{10}) \in [\mathcal{H}^\infty]^{11}$.

For multi line formulas in display mode, `longmath` provides a special environment.

- `\begin{longmath}[align]{width}` begins a multi line expression in display mode.
- `\end{longmath}` ends a multi line expression in display mode.

It works like the `minipage` environment. You have to specify the *width* of the box the formula will be placed into, and how to *align* it vertically with the rest of the displayed equation. The optional alignment argument is either *c* for centre, which is the default, or *t* for top or *b* for bottom.

```
\newcommand\harm[2]{ \frac{1}{\#1} \multido{\n=#1+1}{\#2}{ + \frac{1}{\n} } }
```

The harmonic series starts with
 $\$ \$ Q = \begin{longmath}[t]{70mm}$
 $1 + \text{\harm{2}{21}} + \text{\cdots}$
 $\end{longmath} \$ \$$
 and diverges very slowly.

The harmonic series starts with

$$Q = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} + \frac{1}{9} + \frac{1}{10} +$$

$$\frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15} + \frac{1}{16} +$$

$$\frac{1}{17} + \frac{1}{18} + \frac{1}{19} + \frac{1}{20} + \frac{1}{21} + \frac{1}{22} + \frac{1}{23} + \dots$$
 and diverges very slowly.

No markers are needed to force line breaks at particular points. \TeX 's line breaking algorithm is used to find the "best" breakpoints. If you're familiar with it, you can insert penalties into the formula to optimize it.

The *width* can be specified either as an absolute dimension or as fraction of the available width of the display.

One can easily prove that
 $\$ \$ \begin{longmath}[b]{0.7}$
 $1 + \text{\harm{2}{21}} + \text{\cdots}$
 $\end{longmath} \text{\to} \infty$, $\$ \$$
 which comes as a surprise for some students.

One can easily prove that

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} + \frac{1}{9} +$$

$$\frac{1}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15} + \frac{1}{16} +$$

$$\frac{1}{17} + \frac{1}{18} + \frac{1}{19} + \frac{1}{20} + \frac{1}{21} + \frac{1}{22} + \frac{1}{23} + \dots \rightarrow \infty,$$
 which comes as a surprise for some students.

As formulas typically have fewer possible breakpoints than ordinary text, no attempt is made to justify the lines. Instead, some flexible glue is inserted at both sides of each line to avoid extreme stretching or shrinking. Only the beginning of

the first line and the end of the last line are forced to align with the boundary of the box specified by the *width* parameter. If the whole formula fits into a single line, a box with only the required width is created. This ensures that, regardless of the number of lines, a top aligned `longmath` box always aligns neatly with an expression to the left, and the formula in a bottom aligned `longmath` box can be continued to the right.

Since the `longmath` environment creates a vertical box, you can have several of them in a single display.

```



$$\begin{aligned} & \text{\polyA} \\ & \text{\polyB} \cdot \text{\polyC} \end{aligned}$$



```

$$\left[\begin{array}{l} x^{10} + x^9 - x^8 + \\ 2x^6 + 9x^5 + 2x^4 - \\ 4x^3 - 2x^2 + 6x + 4 \end{array} \right] = \left[\begin{array}{l} (x^4 - x^3 + x + 2) \cdot \\ (x^6 + 2x^5 + \\ x^4 - 2x^2 + 2x + 2) \end{array} \right]$$

To make this look a bit nicer, you can add some extra space to the width of the box and arrange the lines from the top left to the bottom right. If the *width* parameter is given as *linewidth+extrawidth*, the lines are stacked so that the total width of the box is the sum of *linewidth* and *extrawidth*. Both values can be given as absolute dimensions or as fractions of the total display width.

```



$$\begin{aligned} & \text{\polyA} \\ & \text{\polyB} \cdot \text{\polyC} \end{aligned}$$



```

$$\left[\begin{array}{l} x^{10} + x^9 - x^8 + \\ 2x^6 + 9x^5 + 2x^4 - \\ 4x^3 - 2x^2 + 6x + 4 \end{array} \right] = \left[\begin{array}{l} (x^4 - x^3 + x + 2) \cdot \\ (x^6 + 2x^5 + \\ x^4 - 2x^2 + 2x + 2) \end{array} \right]$$

A `longmath` box can replace any kind of subformula. Just specify the maximal width the subformula is allowed to have. If it fits into this width, it will be typeset normally. Otherwise, it is split into several lines.

```



$$\frac{\text{\polyA}}{\text{\polyB}} = \text{\polyC}$$



```

$$\frac{x^{10} + x^9 - x^8 + 2x^6 + 9x^5 + 2x^4 - 4x^3 - 2x^2 + 6x + 4}{x^4 - x^3 + x + 2} = x^6 + 2x^5 + x^4 - 2x^2 + 2x + 2$$

```


\newcommand\polyX{ x^5 - x^4 + x^3 - x^2 + x - 1 }
\newcommand\polyY{ x^{10} - 2 x^9 + 3 x^8 - 4 x^7 + 5 x^6 - 6 x^5 + 5 x^4 - 4 x^3 + 3 x^2 - 2 x + 1 }

$$\sqrt{\text{\polyY}} = \text{\polyX}$$



```

$$\sqrt{\frac{x^{10} - 2x^9 + 3x^8 - 4x^7 + 5x^6 - 6x^5 + 5x^4 - 4x^3 + 3x^2 - 2x + 1}{6x^5 + 5x^4 - 4x^3 + 3x^2 - 2x + 1}} = x^5 - x^4 + x^3 - x^2 + x - 1$$

There is also a smarter `longmath*` environment, which sets the width of the box “intelligently”, so that the formula is split into equally long segments and the available space is used efficiently. The idea is to specify a maximal width, then let \TeX find the number of lines needed, and adapt the width to the total length divided by the number of lines. This will usually result in a box that is slightly shorter than the given width.

Whether this really succeeds depends on a sufficient number of possible breakpoints and stretchable and shrinkable glue in the formula, so that \TeX 's algorithm can actually split it into equally long parts. Take it with a grain of salt and don't mind if it doesn't work properly in difficult cases. As an example, here is what we get without this feature.

To prove it, observe that each bracket is larger than $\frac{12}{100}$:

```


$$\left( 1 + \frac{1}{2} \right) + \left( \frac{1}{3} + \frac{1}{4} \right) + \left( \frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} \right) + \left( \frac{1}{9} + \frac{1}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15} + \frac{1}{16} \right) + \left( \frac{1}{17} + \frac{1}{18} + \frac{1}{19} + \frac{1}{20} + \frac{1}{21} + \frac{1}{22} + \frac{1}{23} + \frac{1}{24} + \frac{1}{25} + \frac{1}{26} + \frac{1}{27} + \frac{1}{28} + \frac{1}{29} + \frac{1}{30} + \frac{1}{31} + \frac{1}{32} \right) + \left( \frac{1}{33} + \frac{1}{34} + \frac{1}{35} + \frac{1}{36} + \dots \right)$$


```

To prove it, observe that each bracket is larger than $\frac{1}{2}$:

$$\left(1 + \frac{1}{2} \right) + \left(\frac{1}{3} + \frac{1}{4} \right) + \left(\frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} \right) + \left(\frac{1}{9} + \frac{1}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15} + \frac{1}{16} \right) + \left(\frac{1}{17} + \frac{1}{18} + \frac{1}{19} + \frac{1}{20} + \frac{1}{21} + \frac{1}{22} + \frac{1}{23} + \frac{1}{24} + \frac{1}{25} + \frac{1}{26} + \frac{1}{27} + \frac{1}{28} + \frac{1}{29} + \frac{1}{30} + \frac{1}{31} + \frac{1}{32} \right) + \left(\frac{1}{33} + \frac{1}{34} + \frac{1}{35} + \frac{1}{36} + \dots \right)$$

If we replace `longmath` by `longmath*`, but stick to the same width specification, we get this.

There is even a slightly “shorter” proof:

$$\left(1 + \frac{1}{2} \right) + \left(\frac{1}{3} + \frac{1}{4} \right) + \left(\frac{1}{5} + \frac{1}{6} + \frac{1}{7} + \frac{1}{8} \right) + \left(\frac{1}{9} + \frac{1}{10} + \frac{1}{11} + \frac{1}{12} + \frac{1}{13} + \frac{1}{14} + \frac{1}{15} + \frac{1}{16} \right) + \left(\frac{1}{17} + \frac{1}{18} + \frac{1}{19} + \frac{1}{20} + \frac{1}{21} + \frac{1}{22} + \frac{1}{23} + \frac{1}{24} + \frac{1}{25} + \frac{1}{26} + \frac{1}{27} + \frac{1}{28} + \frac{1}{29} + \frac{1}{30} + \frac{1}{31} + \frac{1}{32} \right) + \left(\frac{1}{33} + \frac{1}{34} + \frac{1}{35} + \frac{1}{36} + \dots \right)$$

The spacing between the lines is controlled by a dimension register that can be adjusted.

- `\longmathlinesep` dimension to be added to the line skip in the `longmath` environment (default 3.0pt).

This dimension is added to the current `\baselineskip` and `\lineskip` when the content of the box is split into lines and placed in a vertical box, basically using \TeX 's `\openup` macro. Some additional effort is made to adjust the line spacing if the `longmath` environment is placed in a sub- or superscript as in this final example.

```


$$\Omega = \left( T \cdot H \cdot I^{(a+b)^4} \cdot S \right) + I \cdot S + A + \left\{ V \cdot E^{(a^2+2ab+b^2)^2} \cdot R \cdot Y + L \cdot O \cdot N \cdot G + \int_I^N T \cdot (E^{G^R}) \cdot A_L \right\} - \frac{D^I \cdot V^I}{D \cdot E \cdot D} \left/ \left\{ B \cdot Y + A \times \sum_{-\infty}^{\infty} O^F \langle S_a^c I_r^a | \prod^s \rangle \right\} \right\} + C^\infty$$


```

$$\Omega = \left(T \cdot H \cdot I^{(a+b)^4} \cdot S \right) + I \cdot S + A + \left\{ V \cdot E^{(a^2+2ab+b^2)^2} \cdot R \cdot Y + L \cdot O \cdot N \cdot G + \int_I^N T \cdot (E^{G^R}) \cdot A_L \right\} - \frac{D^I \cdot V^I}{D \cdot E \cdot D} \left/ \left\{ B \cdot Y + A \times \sum_{-\infty}^{\infty} O^F \langle S_a^c I_r^a | \prod^s \rangle \right\} \right\} + C^\infty$$