LBT by Example

Gavin Sinclair

Document under active development from May 2025
Current version: 23 June 2025

Contents

List of Examples

Skipping non-draft content (eID=101). Title is ‘Preface’

Skipping non-draft content (eID=102). Title is ‘Introduction’

Part I

Core templates

Skipping non-draft content (eID=103). Title is ‘Core commands’

Skipping non-draft content (eID=104). Title is ‘Lists and tables’

Skipping non-draft content (eID=105). Title is ‘Mathematical text (lbt.Math) – various macros’

1 Mathematical text (lbt.Math) – the MATH command

The MATH command gets its own chapter so that its various features can be displayed one section at a time.

MATH provides for a variety of display equations. It is a portal to various amsmath and mathtools environments like split, gather, align, and so on. The examples here give a good primer on their use, but readers should consult the relevant documentation to develop greater awareness of the details.

1.1 Opening remarks

Setting a display equation with \[... \][1] is enough for a great many cases. If you want your equation to be numbered, you upgrade to the equation environment. If the math content to be displayed is more complicated than that, the author should decide which of the following applies:

- there is one logical equation with several parts (separated by = or > or …) that should appear on separate lines (split);
- there is one logical equation that is too long to fit on one line (multline);
- there are several logical equations to be displayed together (gather);
- there are several logical equations to be displayed reasonably with alignment (align);
- there are more complicated alignment requirements, perhaps involving comments to the side (also align).

Based on that, the author can choose an amsmath environment, which are demonstrated in Table 1.1. The table does not show *all* available environments, but it gives a good overview for readers who are not already familiar.

The sections of this chapter give more detailed information on these environments and more.

[1] Or the Tex command $$ ... $$, which is lower-level and may produce different vertical spacing from \[... \].

Note

In normal Latex code, equations are numbered by default. If you use the `align` environment then all lines are numbered. If you use `align*` then none of them are.
LBT is similar: use `MATH` to get numbered equations and `MATH*` to suppress numbereding. If you want unnumbered equations by default, set the oparg `MATH.eqnum = false`.

The LBT examples that follow will demonstrate fully numbered, partially numbered, and unnumbered equations, as appropriate to the environment being demonstrated.

Table 1.1 Some environments provided by amsmath

Environment	Example
`equation`	$a^2 + b^2 = c^2 \quad (1.1)$
`gather`	$a^2 + b^2 = c^2 \quad (1.1)$ $E = mc^2 \quad (1.2)$ $F = k\frac{q_1 q_2}{r^2} \quad (1.3)$
`align` (1)	$a^2 + b^2 = c^2 \quad (1.1)$ $E = mc^2 \quad (1.2)$ $F = k\frac{q_1 q_2}{r^2} \quad (1.3)$
`align` (2)	$a^2 + b^2 = c^2 \qquad E = mc^2 \quad (1.1)$ $F = k\frac{q_1 q_2}{r^2} \qquad F = ma \quad (1.2)$
`align` (3)	$2^{n+1} = 2 \cdot 2^n \quad (1.1)$ $> 2 \cdot n^2 \qquad$ by assumption $\quad (1.2)$ $= n^2 + \frac{1}{2}n^2 + \frac{1}{2}n^2 \quad (1.3)$ $> n^2 + 2n + 1 \qquad$ reader to confirm $\quad (1.4)$ $= (n+1)^2 \quad (1.5)$
`split` (inside `equation`)	$$\begin{aligned} f'(x) &= \lim_{h\to 0} \frac{f(x+h) - f(x)}{h} \\ &= \lim_{h\to 0} \frac{(x+h)^2 - x^2}{h} \\ &= \lim_{h\to 0} \frac{(x^2 + 2xh + h^2) - x^2}{h} \\ &= \lim_{h\to 0} \frac{2xh + h^2}{h} \\ &= \lim_{h\to 0} 2x + h \\ &= 2x \end{aligned} \quad (1.1)$$
`multline`	$$(1+x)^n = \sum_{r=0}^{n} \binom{n}{r} x^r = 1 + \binom{n}{1}x + \binom{n}{2}x^2 + \cdots + \binom{n}{r}x^r + \cdots + \binom{n}{n}x^n \quad (1.1)$$

1.2 Simple equations with MATH (.o equation)

The equation environment provides for a simple numbered equation. Example 1.1 demonstrates this in LBT. equation is in fact the default environment, so you can just write MATH F = ma, as the example shows.

```
TEXT Newton's second law is known to
↪  many.
MATH F = ma

TEXT You can suppress numbering in two
↪  ways.
MATH* F = ma
MATH .o noeqnum :: F = ma

TEXT \code{equation} is the
↪  \emph{default} environment for
↪  \code{MATH}, but you can be
↪  explicit if you wish.
MATH .o equation :: F = ma
```

Newton's second law is known to many.

$$F = ma \tag{1.1}$$

You can suppress numbering in two ways.

$$F = ma$$

$$F = ma$$

equation is the *default* environment for MATH, but you can be explicit if you wish.

$$F = ma \tag{1.2}$$

Example 1.1 MATH .o equation to format a simple equation

1.3 Long equations with MATH .o multiline

If an equation is too long for one line, you can insert linebreaks and the amsmath environment multline will handle formatting with a mixture of left, center and right justification. Example 1.2 demonstrates.

Both MATH .o multiline and MATH .o multline work, and do the same thing.

Note that the example includes the oparg sm = false to disable simplemath. This is necessary to prevent be from being rendered as β.

```
TEXT The display environment below has its margins adjusted so that the effect of
↪  \code{multiline}

MATH .o multiline, sm = false, adjustwidth = 2cm 2cm
:: (a+b+c+d+e)^2 =
:: a^2 + 2ab + 2ac + 2ad + 2ae + b^2 + 2bc + 2bd + 2be
:: + c^2 + 2cd + 2ce + d^2 + 2de + e^2
```

The display environment below has its margins adjusted so that the effect of `multiline`

$$\begin{aligned}(a+b+c+d+e)^2 = \\ a^2 + 2ab + 2ac + 2ad + 2ae + b^2 + 2bc + 2bd + 2be \\ + c^2 + 2cd + 2ce + d^2 + 2de + e^2 \end{aligned} \tag{1.3}$$

Example 1.2 A long equation with `multiline`

1.4 gather

If you have a few equations that you want to display at once, and aren't too fussed about how they are presented, then you `gather` them and they will each be centered and numbered. Two examples suffice. Example 1.3 demonstrates the default numbering. Example 1.4 demonstrates selective numbering. If you want no numbering, just use `MATH*`.

Note that the first example (Example 1.3) demonstrates the oparg `linespace`, which provides some extra space between the lines.

```
TEXT Some interesting series results.
MATH .o gather, linespace = 1ex
:: sum_{n=1}^infty frac 1 {n^2} = frac {pi^2} 6
:: sum_{n=0}^infty frac {x^n} {n!} = e^x
```

Some interesting series results.

$$\sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{\pi^2}{6} \tag{1.4}$$

$$\sum_{n=0}^{\infty} \frac{x^n}{n!} = e^x \tag{1.5}$$

Example 1.3 Displaying several equations with `gather`

```
TEXT Area formulas you must know:
MATH .o gather, eqnum = 2 4
:: A = pi r ^2
:: A = pi a b       \label{eq:a1}
:: A = \tfrac12 b h
:: A = \tfrac12 x y   \label{eq:a2}
TEXT We'll discuss \cref{eq:a1} and \cref{eq:a2} in more detail\dots
```

Area formulas you must know:

$$A = \pi r^2$$
$$A = \pi a b \tag{1.6}$$
$$A = \tfrac{1}{2} b h$$
$$A = \tfrac{1}{2} x y \tag{1.7}$$

We'll discuss eq. (1.6) and eq. (1.7) in more detail…

Example 1.4 gather with selective numbering

1.5 Several-part equations with eqsplit

The amsmath environment split is designed for a single logical equation that is broken into two or more lines, like the example below.

$$\begin{aligned}(a+b)^2 &= (a+b)(a+b) \\ &= a^2 + ab + ab + b^2 \\ &= a^2 + 2ab + b^2\end{aligned} \tag{1.8}$$

However, split is a sub-environment that can only occur within a display environment like equation or gather or align. The most common use would be for a split appear alone in an equation environment, so LBT provides MATH .o eqsplit for that purpose.

An eqsplit equation gets one number overall, not one number per line, because it is one logical equation. It is the equation environment that provides the number, not the contained split.

Example 1.5 demonstrates an unnumbered split equation that is aligned on = and >. Example 1.6 shows a split equation that is numbered and referenced.

Note that it could be desirable to add aligned comments off to the right in Example 1.5. Unfortunately that is not possible with eqsplit (or the underlying split). Later, in Section 1.6, this example will be revisited. The problem there is that align gives one number per line, because it sees each line as a separate equation.

```
TEXT Part of a proof by induction.
STO half :: 1 :: $\tfrac 1 2$
MATH* .o eqsplit
:: 2^{n+1} &= 2 cdot 2^n
::         &> 2 cdot n^2
::      &= n^2 + ◊half n^2 + ◊half n^2
::         &> n^2 + 2n + 1
::         &= (n+1)^2
```

Part of a proof by induction.

$$\begin{aligned} 2^{n+1} &= 2 \cdot 2^n \\ &> 2 \cdot n^2 \\ &= n^2 + \tfrac{1}{2}n^2 + \tfrac{1}{2}n^2 \\ &> n^2 + 2n + 1 \\ &= (n+1)^2 \end{aligned}$$

Example 1.5 Using `MATH .o eqsplit` for a multi-step equation

```
MATH .o eqsplit, label = eq1
:: (a+b)^2 &= (a+b)(a+b)
::         &= a^2 + ab + ab + b^2
::         &= a^2 + 2ab + b^2

TEXT As shown in \eqref{eq1}, $(a+b)^2$
↪  does not equal $a^2 + b^2$!
```

$$\begin{aligned} (a+b)^2 &= (a+b)(a+b) \\ &= a^2 + ab + ab + b^2 \\ &= a^2 + 2ab + b^2 \end{aligned} \tag{1.9}$$

As shown in (1.9), $(a+b)^2$ does not equal $a^2 + b^2$!

Example 1.6 A `MATH .o eqsplit` equation that is referenced

1.6 `align` and `alignat` and `flalign`

The amsmath environments `align` and its variants are supported directly by `MATH .o align` and friends to produce mathematical output using specified alignment points. Note that `split` (**Section 1.5**) does this as well, but it only allows one alignment point per line. `align` and `alignat` allow as many alignment points as you wish.

The difference between `align` and the others is as follows:

- `align` determines the horizontal spacing between alignment columns itself, while centering the material overall;
- `flalign` ("full-length align") spreads the columns out as far as possible, using the full text width of the page;
- `alignat` requires the author to specify the number of columns and to control the spacing between them.

There are different logical uses for the alignment environments, as the subsections below demonstrate.

Presenting a group of equations with simple alignment

The mathematical content in Example 1.7, Example 1.8 and Example 1.9 is the same, but different numbering choices are demonstrated.

These examples use only one alignment point, so the material could technically be typeset by eqsplit. This would be the wrong logical choice, however, because these are three equations, not one. Accordingly, align produces (up to) three equation numbers whereas eqsplit would only produce one.

```
TEXT All equations numbered (the default).
MATH .o align
:: (a+b)^3    &= a^3 + 3a^2b + 3ab^2 + b^3
:: (a-b)(a+b) &= a^2 - b^2
::        c^2 &= a^2 + b^2
```

All equations numbered (the default).

$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3 \tag{1.10}$$
$$(a-b)(a+b) = a^2 - b^2 \tag{1.11}$$
$$c^2 = a^2 + b^2 \tag{1.12}$$

Example 1.7 Aligning a group of equations with align

```
TEXT Numbering suppressed.
MATH* .o align
:: (a+b)^3    &= a^3 + 3a^2b + 3ab^2 + b^3
:: (a-b)(a+b) &= a^2 - b^2
::        c^2 &= a^2 + b^2
```

Numbering suppressed.

$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3$$
$$(a-b)(a+b) = a^2 - b^2$$
$$c^2 = a^2 + b^2$$

Example 1.8 Alignment without numbering

Example 1.9 shows a special feature of MATH: selective numbering. In ordinary Latex, you use \notag on any line you do not want numbered. (You can do that in MATH too if you wish.) The MATH oparg eqnum gives you convenient control over which lines are numbered, without editing the lines themselves.

```
TEXT Selective numbering.
MATH .o align, eqnum = 1 3
:: (a+b)^3    &= a^3 + 3a^2b + 3ab^2 + b^3
:: (a-b)(a+b) &= a^2 - b^2
::        c^2 &= a^2 + b^2
```

Selective numbering.

$$(a+b)^3 = a^3 + 3a^2b + 3ab^2 + b^3 \tag{1.13}$$
$$(a-b)(a+b) = a^2 - b^2$$
$$c^2 = a^2 + b^2 \tag{1.14}$$

Example 1.9 Alignment with selective numbering

Aligning equations in multiple columns

Suppose you wanted to demonstrate three kinds of derivative: polynomial, trigonometric and exponential. And you wanted to do so in minimal vertical space. Then you might typeset something like Example 1.10.

```
MATH* .o align
:: f(x)  &= x^3 - 7x^2 + 4x + 1  & g(x)  &= sin(2x) - tan x  & h(x)  &= 3^x
:: f'(x) &= 6x^2 - 14x + 4       & g'(x) &= 2cos(2x) - sec2x & h'(x) &= 3^x\:ln 3
```

$$f(x) = x^3 - 7x^2 + 4x + 1 \qquad g(x) = \sin(2x) - \tan x \qquad h(x) = 3^x$$
$$f'(x) = 6x^2 - 14x + 4 \qquad g'(x) = 2\cos(2x) - \sec^2 x \qquad h'(x) = 3^x \ln 3$$

Example 1.10 Alignment in multiple columns

The space between the "columns" is determined by the amsmath package – see the relevant documentation for details. If you want to really spread things out, you can use `flalign`, which uses the "full length" of the page, as shown in Example 1.11.

And if you want to determine your own spacing, you can: the `alignat` environment gives the author that control. Example 1.12 demonstrates the use of `\qquad` to separate the two columns. When you use `alignat`, you need to provide the oparg `ncols` to specify how many columns there are.[2]

[2]Note that in this example, there are two columns and $2(2) - 1 = 3$ ampersands per line. It is helpful to keep this relationship in mind.

```
MATH* .o flalign
:: f(x)  &= x^3 - 7x^2 + 4x + 1  & g(x)  &= sin(2x) - tan x  & h(x)  &= 3^x
:: f'(x) &= 6x^2 - 14x + 4       & g'(x) &= 2cos(2x) - sec2x & h'(x) &= 3^x\:ln 3
```

$$f(x) = x^3 - 7x^2 + 4x + 1 \qquad g(x) = \sin(2x) - \tan x \qquad h(x) = 3^x$$
$$f'(x) = 6x^2 - 14x + 4 \qquad g'(x) = 2\cos(2x) - \sec^2 x \qquad h'(x) = 3^x \ln 3$$

Example 1.11 Full-length in multiple columns

```
MATH* .o alignat, ncols = 2
:: f(x)  &= x^3 - 7x^2 + 4x + 1  &\hspace{4em} g(x)  &= sin(2x) - tan x
:: f'(x) &= 6x^2 - 14x + 4       &              g'(x) &= 2cos(2x) - sec2x
```

$$f(x) = x^3 - 7x^2 + 4x + 1 \qquad g(x) = \sin(2x) - \tan x$$
$$f'(x) = 6x^2 - 14x + 4 \qquad g'(x) = 2\cos(2x) - \sec^2 x$$

Example 1.12 Manual control of inter-column spacing

Generally speaking, the default spacing should be sufficient. A more useful purpose for `alignat` is shown in *Multiple alignment points among equations* on page 15, where polynomials have their like terms lined up regardless of the width of coefficients.

Improving the display of a single long equation

Earlier, we saw how `multiline` can be used to manually break up an equation that doesn't fit on one line. The mixed-justification formatting that `multiline` applies may suit some equations but not others. If you prefer a left-justified equation as shown in **Example 1.13**, this can be achieved with `alignat` and a single column. Note the use of `\MoveEqLeft` from the mathtools package to place the first line correctly. Also note the use of `\phantom{=\ }` to align the continuation lines nicely.

Providing comments to the right

It is common that an author wants to write some brief commentary to the right of a line of working in a multi-step equation. We can achieve this using `align` with two columns, as **Example 1.14** demonstrates.

```
STO ph :: 1 :: \phantom{=\ }
MATH .o alignat, ncols = 1, eqnum = 5
:: \MoveEqLeft (x - r_1)(x - r_2)(x - r_3)(x - r_4)
:: quad &= x^4 - (r_1 + r_2 + r_3 + r_4)x^3
::      &\◊ph + (r_1r_2 + r_1r_3 + r_1r_4 + r_2r_3 + r_2r_4 + r_3r_4)x^2
::      &\◊ph - (r_1r_2r_3 + r_1r_2r_4 + r_1r_3r_4 + r_2r_3r_4)x
::      &\◊ph + r_1r_2r_3r_4
```

$$\begin{aligned}(x-r_1)(x-r_2)(x-r_3)(x-r_4)&\\ &= x^4-(r_1+r_2+r_3+r_4)\,x^3\\ &\quad+(r_1r_2+r_1r_3+r_1r_4+r_2r_3+r_2r_4+r_3r_4)\,x^2\\ &\quad-(r_1r_2r_3+r_1r_2r_4+r_1r_3r_4+r_2r_3r_4)\,x\\ &\quad+r_1r_2r_3r_4 \end{aligned} \tag{1.15}$$

Example 1.13 Using `alignat` and `MoveEqLeft` to improve the formatting of a long equation

The inter-column spacing in Example 1.14 is too large, so we assert manual control using `alignat`, as shown in Example 1.15. Note that we insert a `\qquad` in the *longest* line of working.

In these examples we have been typesetting a single multi-step equation (for which `split` is designed) using `align`, which is designed for multiple logical equations. It is perhaps a shame that the amsmath package does not support this use-case—providing commentary on a split equation—more directly.

Having said that, there is the option to use `alignedat`, instead of `alignat`, and place that inside `equation`. `alignedat` does the logical layout without doing any numbering, and `equation` displays the result and assigns a number. LBT supports this combination with `eqalignedat`, as shown in Example 1.16.

Incorporating lines of text

Multiple alignment points among equations

Aligning equations near the left margin

```
TEXT Part of a proof by induction.
STO half :: 2 :: $\tfrac 1 2$
MATH* .o align
:: 2^{n+1} &= 2 cdot 2^n
::         &> 2 cdot n^2                    && text {(by assumption)}
::         &= n^2 + ◊half n^2 + ◊half n^2
::         &> n^2 + 2n + 1                  && text {(reader to confirm)} \tag{*}
::         &= (n+1)^2

TEXT The reader who is interested in tackling (*) might like to consider how we know
↪  that $◊half n^2 > 2n$ and $◊half n^2 > 1$.
```

Part of a proof by induction.

$$
\begin{aligned}
2^{n+1} &= 2 \cdot 2^n \\
&> 2 \cdot n^2 && \text{(by assumption)} \\
&= n^2 + \tfrac{1}{2}n^2 + \tfrac{1}{2}n^2 \\
&> n^2 + 2n + 1 && \text{(reader to confirm)} && (*) \\
&= (n+1)^2
\end{aligned}
$$

The reader who is interested in tackling (*) might like to consider how we know that $\frac{1}{2}n^2 > 2n$ and $\frac{1}{2}n^2 > 1$.

Example 1.14 Using `MATH .o align` for a multi-step equation with commentary

```
TEXT Part of a proof by induction.
STO half :: 2 :: $\tfrac 1 2$
MATH* .o alignat, ncols = 2
:: 2^{n+1} &= 2 cdot 2^n
::         &> 2 cdot n^2                    && text {(by assumption)}
::         &= n^2 + ◊half n^2 + ◊half n^2   \qquad && text {}
::         &> n^2 + 2n + 1                  && text {(reader to confirm)} \tag{*}
::         &= (n+1)^2

TEXT The reader who\dots
```

Part of a proof by induction.

$$
\begin{aligned}
2^{n+1} &= 2 \cdot 2^n \\
&> 2 \cdot n^2 && \text{(by assumption)} \\
&= n^2 + \tfrac{1}{2} n^2 + \tfrac{1}{2} n^2 \\
&> n^2 + 2n + 1 && \text{(reader to confirm)} && (*) \\
&= (n+1)^2
\end{aligned}
$$

The reader who…

Example 1.15 Using `MATH .o alignat` to improve the previous example

```
TEXT Part of a proof by induction.
STO half :: 1 :: $\tfrac 1 2$
MATH .o eqalignedat, ncols = 2, label = eq:induc
:: 2^{n+1} &= 2 cdot 2^n
::         &> 2 cdot n^2                    && text {(by assumption)}
::         &= n^2 + ◊half n^2 + ◊half n^2   \qquad && text {}
::         &> n^2 + 2n + 1                  && text {(reader to confirm)}
::         &= (n+1)^2

TEXT The techniques in \eqref{eq:induc} should be mastered by all students.
```

Part of a proof by induction.

$$\begin{aligned} 2^{n+1} &= 2 \cdot 2^n \\ &> 2 \cdot n^2 && \text{(by assumption)} \\ &= n^2 + \tfrac{1}{2}n^2 + \tfrac{1}{2}n^2 \\ &> n^2 + 2n + 1 && \text{(reader to confirm)} \\ &= (n+1)^2 \end{aligned} \tag{1.16}$$

The techniques in (1.18) should be mastered by all students.

Example 1.16 Using `MATH .o eqalignedat` to align a single logical equation

1.7 Other environments

split

Revisit this text in light of it being in the "other" section

MATH provides the split option to access the split environment, but it is not likely to be all that useful, because of the need to enclose it in another environment. The example below shows the LBT code and resulting Latex code.

```
MATH .o split
:: (a+b)^2 &= (a+b)(a+b)
::         &= a^2 + ab + ab +
↪  b^2
::         &= a^2 + 2ab + b^2
```

```
\begin{split}
\ensuremath{\left(a+b\right)^2 &= \left(a+b\right)\left(
\ensuremath{&= a^2 + ab + ab + b^2} \\
\ensuremath{&= a^2 + 2ab + b^2}
\end{split}
\par
```

1.8 Combinations

1.9 Summary of the MATH command

Part II

Non-core built-in templates

Skipping non-draft content (eID=107). Title is ‘Worksheet or exam questions with lbt.Questions’

Part III

Creating a new template

Part IV

Extra features