

The interval package

Lars Madsen*

(on behalf of By the Danish T_EX collective)

May 24, 2013

Motivation

In mathematics there are two syntax' when it comes to specifying open and closed intervals.

The first use parantheses to mark an open end

$$[a, b] \quad (a, b] \quad [a, b) \quad (a, b),$$

while the other use brackets throughout

$$[a, b] \quad]a, b] \quad [a, b[\quad]a, b[,$$

The former poses no problem in T_EX, but the later does, as, e.g., a closing bracket is being used in place of an opening fence, and thus have the wrong category when it comes to spacing:

$$] - a, b[+ c \quad \text{versus} \quad] - a, b[+ c.$$

One could use

`\mathopen{[]}-a,b\mathclose{[]}+c`

to solve the problem, but then `\left...\right` can no longer be used to auto scale the fences.

The `\interval` command

The following is the result of a discussion on the Danish T_EX Users groups mailing list. Kudos to Martin Heller, for proposing the original version using `pgfkeys`.

We provide a macro and a way to globally configure it

```
\interval[<options>]{<start>}{<end>}
\intervalsetup{<options>}
```

We note that the interval separator symbol is hidden inside the `\interval` macro and can be changed using an option.

*Email: daleif@imf.au.dk

Configuration options

separator symbol

symbol that separates the start and end of the interval. Default: `{,}`, note that as comma is the separating character in the options specification, the symbol is enclosed in braces, these are automatically removed.

left open fence

Default: `]`

left closed fence

Default: `[`

right open fence

Default: `[`

right closed fence

Default: `]`

soft open fences

This is just a fast way of saying

```
left open fence=(,  
right open fence=)
```

Usage options

By default `\interval{<start>}{<end>}` will produce a closed interval. Other types are provided via options:

open

an open interval

open left

interval open on the left side

open right

interval open on the right side

scaled

auto scale interval fences

scaled=<scaler>

scale fences using `<scaler>`, i.e. using `scaled=\Big`

As some might be guessed, the `interval` package depends on the `pgfkeys` package to handle its key-value configuration.

Examples

```
\begin{align*}
&A\in\interval{a}{b}\\
&A\in\interval[open]{a}{b}\\
&A\in\interval[open left]{a}{b}\\
&A\in\interval[open right,
  scaled]{a}{\frac{1}{2}b}=B\\
&A\in\interval[scaled=\big]{a}{b}
\end{align*}
```

$$\begin{aligned}
 A &\in [a, b] \\
 A &\in]a, b[\\
 A &\in]a, b] \\
 A &\in \left[a, \frac{1}{2}b \right] = B \\
 A &\in [a, b]
 \end{aligned}$$

And using soft open fences:

```
\intervalconfig{
  soft open fences,
  separator symbol=;,
}
\begin{align*}
&A\in\interval{a}{b}\\
&A\in\interval[open]{a}{b}\\
&A\in\interval[open left]{a}{b}\\
&A\in\interval[open right,
  scaled]{a}{\frac{1}{2}b}=B\\
&A\in\interval[scaled=\big]{a}{b}
\end{align*}
```

$$\begin{aligned}
 A &\in [a; b] \\
 A &\in (a; b) \\
 A &\in (a; b] \\
 A &\in \left[a; \frac{1}{2}b \right) = B \\
 A &\in [a; b]
 \end{aligned}$$