TeXmacs:一个真正"所见即所得"的排版系统



1.4 Formal proof with mapping reduction

Suppose we have D_{HALT} , a decider for HALT. We define the mapping reduction as:

$$f(\langle M, w \rangle) = \langle M', w \rangle$$

where the TM M' is constructed as computing the function:

$$M'(x) = if(eval(\langle M, x \rangle))$$

then $accept$
else $loop$

This means exactly (but concisely) what the above informal description says:

Construct the following machine M'.

"On input x:

- 1. Run M on x.
- 2. If M accepts, accept.
- 3. If M rejects, enter a loop."

Notice although M' takes input named "x", the actual input is w when it is simulated. This is because when $\langle M', w \rangle$ is passed as input to a decider of HALT, it uses the w part as the *actual argument* for M'. This is like the difference between a formal parameter and the actual argument in a function call when using a programming language such as Java or Python.

We can see the behavior of M, M' and D_{HALT} in the following table:

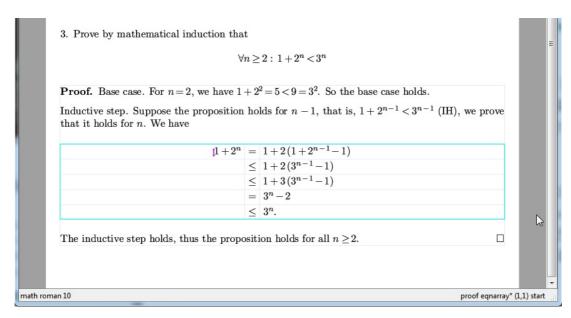
M accepts w		D_{HALT} accepts $\langle M', w \rangle$
M does not accept w	M' loops	D_{HALT} rejects $\langle M', w \rangle$

Table 1. Behavior table for M, M' and D_{HALT}

好久没有推荐过自己喜欢的软件了,现在推荐一款我在美国做数学作业的私家法宝:TeXmacs。我恐怕不可能跟以前那么有闲心写个长篇的 TeXmacs 说明文档了,不过这东西如此的简单好用,所以基本上不用我写什么文档了。鉴于知道的人很少,不理解它的人很多,这里只是帮它打个广告,吊一下胃口。

TeXmacs 的主要特点是:

- 跟 Lyx 等不同,它不是一个 TeX 的"前端",而是一个完全独立的,超越 TeX 的系统。TeXmacs 拥有跟 TeX 相同,甚至更好的排版美观程度。这是因为它采用跟 TeX 一样的排版算法,并且用 C++ 重新实现。据说分页的算法比 TeX 的还要好些。
- 拥有超越 Word (或者任何一款字处理软件)的,真正的"所见即所得"(WYSIWYG)。Word 所谓的"所见即所得"其实是假的。所见即所得的含义应该是,屏幕上显示的内容,跟打印下来的完全一样。可是 Word 能做到吗?打印一个文档出来你就发现跟屏幕上显示的有很大区别,一般来说屏幕上显示的要粗糙一些。一些 TeX 的前端,比如 Lyx, Scientific Workspace 等也是类似的,它们都不能达到真正的所见即所得。
- 直接可在屏幕文档里绘图。完全可视化的表格,公式编辑环境。这些都是比 TeX 方便高效很多的方式。需要当心的是,用过 TeXmacs 一段时间之后,你会发现回到 TeX 的公式编辑方式简直就像回到原始社会。
- 非常人性化的按键设计。比如,在数学公式环境下,你按任意一个字符,然后就可以用多次 TAB 键相继选择"拓扑相同"的字符。举个例子,如果你按 @,然后再按几下 TAB,就会发现这个字符变成各种各样的圆圈形的字符。如果你按 >,再按 = ,就会出现大于等于号,之后再按 TAB,就会相继出现大于等于号的各种变体。
- 在直观的同时不失去对底层结构的控制。比如,(见下图)窗口右下角的状态栏,显示出当前光标位置的"上下文"是"proof eqnarry* (1,1) start",这表示的是这是在一个 proof 环境里的 eqnarry 的坐标 (1,1) 的开始处。当你使用 Ctrl-Backspace,最靠近光标的那层"环境"会被删除。比如,如果你现在的字体是斜体,那么在 Ctrl-Backspace 之后,字体就立即还原成正体。



- 结构化的浏览功能。比如,按 Ctrl-PgUp, Ctrl-PgDn 就可以在"相同类型"的结构里上下跳转。比如,如果你在小节标题里按这个键,就可以迅速的浏览所有的小节标题。如果你在数学公式里按这个键,就可以迅速浏览所有的数学公式。
- 与交互式程序接口。支持很多种计算机代数系统,和交互式软件,比如 MAXIMA,Octave,…… 这些系统返回的数学公式会直接被 TeXmacs 显示为"TeX 效果"。使用 Scheme 作为嵌入式语言,并且可以使用它来扩展系统。这比起 TeX 的语言是非常大的进步。

目前由于 TeX 的垄断地位,以及由于 TeXmacs 是法国人做的,这个系统在美国还不是很流行,很多人都没听说过有这种东西存在。学术圈的很多人由于受到某种错误思想的"洗脑",都不理解这种图形化编辑软件的价值。希望中国人民和法国人民一样后来居上,超越美国。

想要迅速的掌握 TeXmacs 的基本用法,可以参考我绘制的 TeXmacs 思维导图:

