

# 汇报

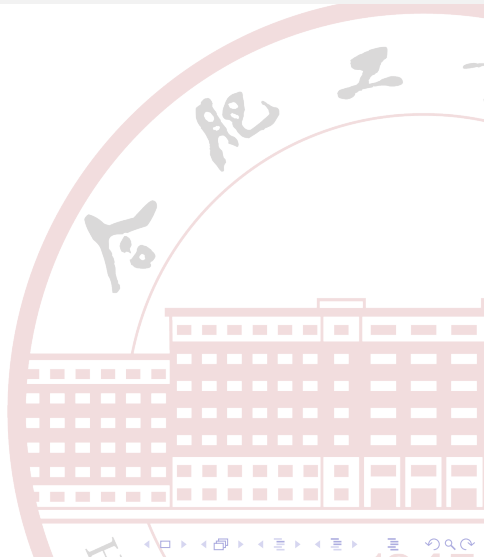
王子颢

School of Computer and Informatics, HFUT

2024 年 8 月 1 日

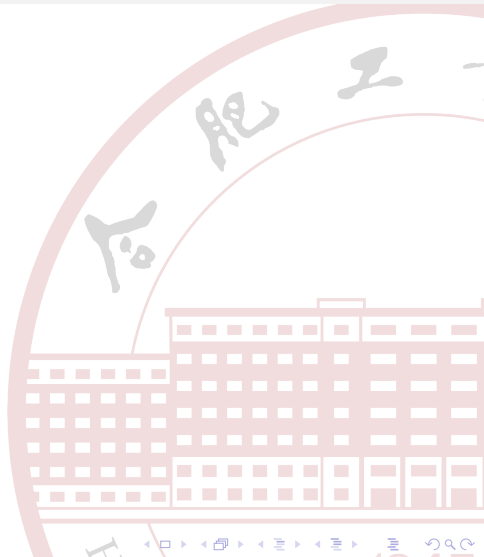
# 目录

- 1 BEAMER 简单入门
- 2 动画
- 3 引用
- 4 BEAMER 的一些坑



# 目录

- 1 BEAMER 简单入门
- 2 动画
- 3 引用
- 4 BEAMER 的一些坑



# 标题

## 和副标题

一帧幻灯片的标题可以用如下方式指定：

- `\begin{frame}{标题}{副标题}`
- `\frametitle{标题}`
- `\framesubtitle{副标题}`

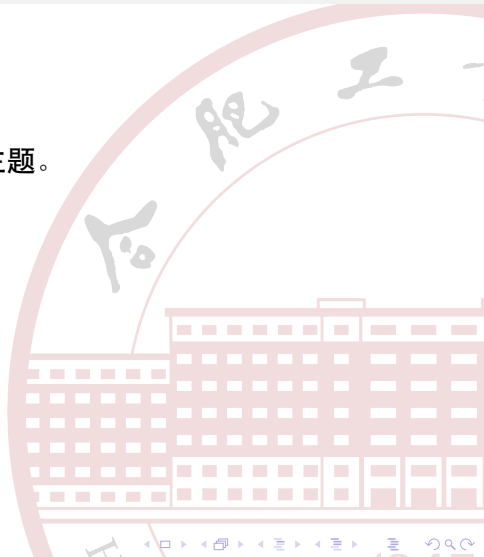


# 主题

BEAMER 的主题有很多，我们提供了三套完整的主题。

- hfut-CambridgeUS
- hfut-sx
- hfut-BinChen

这种完整的主题使用`\usetheme`命令指定。



## 主题（续）

同时，BEAMER 还有很多模块化的颜色（color）、内部（inner）、外部（outer）和字体（font）主题。一般来说完整的主题由这些模块化的子主题构成。我们可以通过以下指令指定这些主题：

```
color \usecolortheme
inner \useinnertheme
outer \useoutertheme
font \usefonttheme
```

由于维护者的精力有限，本 repo 提供的主题并未进行这样的模块化，欢迎提交 PR 进行修正。

# Block 环境

Block 环境是 BEAMER 的一个特色功能，可以用来引导读者视线，突出幻灯片中的重点内容。

这是 Block 的标题

这是 Block 的内容

定理

*There exists an infinite set.*

# Block 环境

Block 环境是 BEAMER 的一个特色功能，可以用来引导读者视线，突出幻灯片中的重点内容。

这是 Block 的标题

这是 Block 的内容

定理

*There exists an infinite set.*

证明.

This follows from the axiom of infinity.





# Block 环境

Block 环境是 BEAMER 的一个特色功能，可以用来引导读者视线，突出幻灯片中的重点内容。

这是 Block 的标题

这是 Block 的内容

定理

*There exists an infinite set.*

证明.

This follows from the axiom of infinity. □

例 (Natural Numbers)

The set of natural numbers is infinite.

# 排版

很多 MS Powerpoint 文档拥有复杂的排版，BEAMER 其实也可以做到类似的效果。

比如说，columns 环境就可以提供多栏的排版格式，这允许你进行左右对照。

```
\begin{columns}
  \begin{column}{0.45\textwidth}
    % ...
  \end{column}
  \begin{column}{0.45\textwidth}
    % ...
  \end{column}
\end{columns}
```

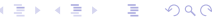
# 幻灯片脚注

- 脚注用`\footnote`命令<sup>1</sup>。
- 脚注引用用`\footcite`命令<sup>2</sup>。
- 脚注完整引用用`\footfullcite`命令<sup>3</sup>。

---

<sup>1</sup>这是一个脚注

<sup>2</sup>, 需要 BIBLATEX, 下同.

<sup>3</sup>Philip Kime, Moritz Wemheuer, and Philipp Lehman. *The biblalex Package - Programmable Bibliographies and Citations*. 2022. 

# 图片、表格、算法

这些浮动体环境在 BEAMER 下并无什么不同，我们可以通过正常的方法放置这些环境。但是有时候（特别是算法环境）放置策略需要采用 [H]。



图 1: 合肥工业大学校徽

# 图片、表格、算法

这些浮动体环境在 BEAMER 下并无什么不同，我们可以通过正常的方法放置这些环境。但是有时候（特别是算法环境）放置策略需要采用 [H]。

表 1: Table caption

| Treatments  | Response 1 | Response 2 |
|-------------|------------|------------|
| Treatment 1 | 0.0003262  | 0.562      |
| Treatment 2 | 0.0015681  | 0.910      |
| Treatment 3 | 0.0009271  | 0.296      |

# 图片、表格、算法

这些浮动体环境在 BEAMER 下并无什么不同，我们可以通过正常的方法放置这些环境。但是有时候（特别是算法环境）放置策略需要采用 [H]。

---

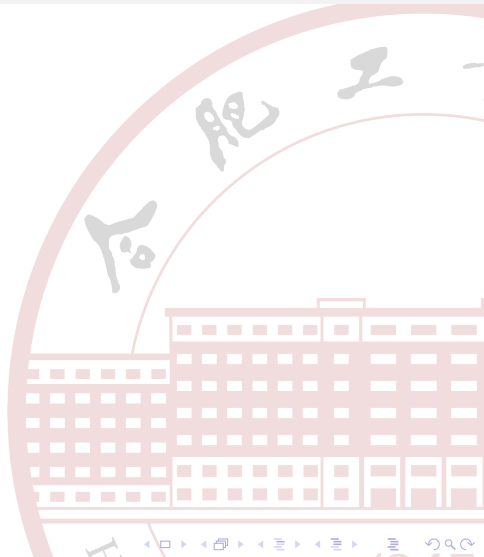
## 算法 1 神经网络训练过程

---

- 1: 初始化网络权值  $\mathcal{W}$
  - 2: **for**  $t = 0 \rightarrow \max\_train\_steps$  **do**
  - 3:     输入一批次  $x$  以计算  $L(t)$
  - 4:     计算梯度  $\nabla_{\mathcal{W}} L(t)$
  - 5:     使用梯度  $\nabla_{\mathcal{W}} L(t)$  更新  $\mathcal{W}(t) \mapsto \mathcal{W}(t+1)$
  - 6: **end for**
-

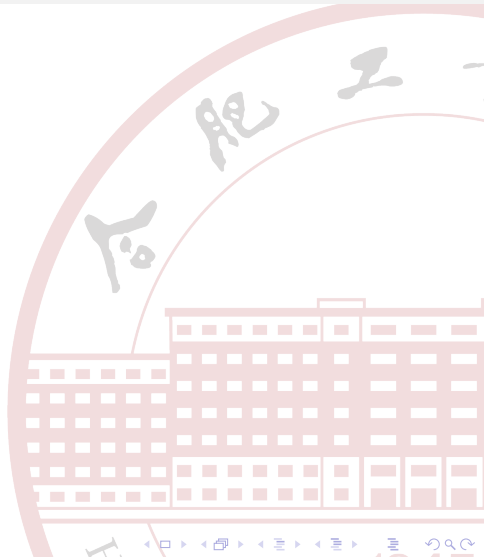
# 目录

- 1 BEAMER 简单入门
- 2 动画**
- 3 引用
- 4 BEAMER 的一些坑



# 简单动画

BEAMER 比较方便的是实现一些简单的动画效果。  
比方说：

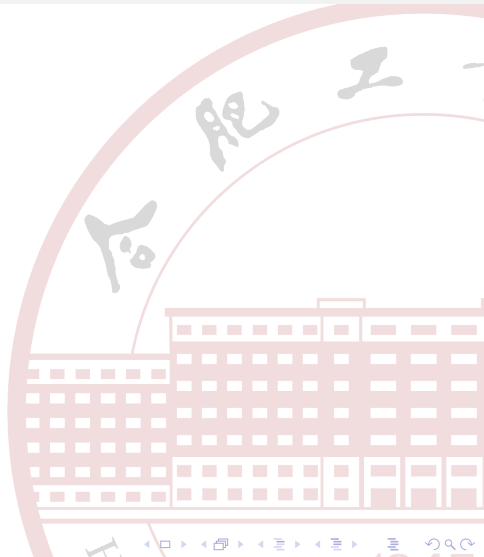




# 简单动画

BEAMER 比较方便的是实现一些简单的动画效果。  
比方说：

- 单击显示/隐藏：\pause命令

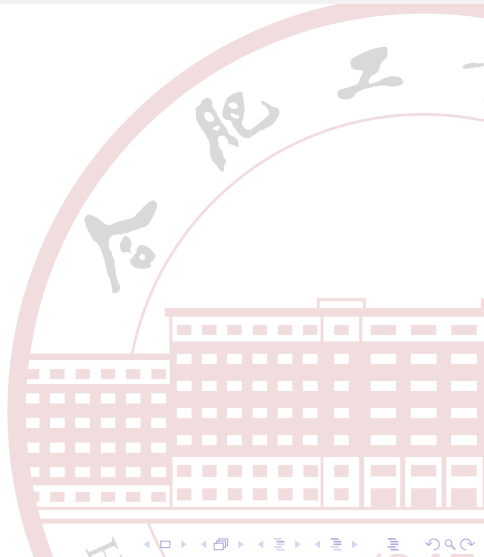


# 简单动画

BEAMER 比较方便的是实现一些简单的动画效果。

比方说：

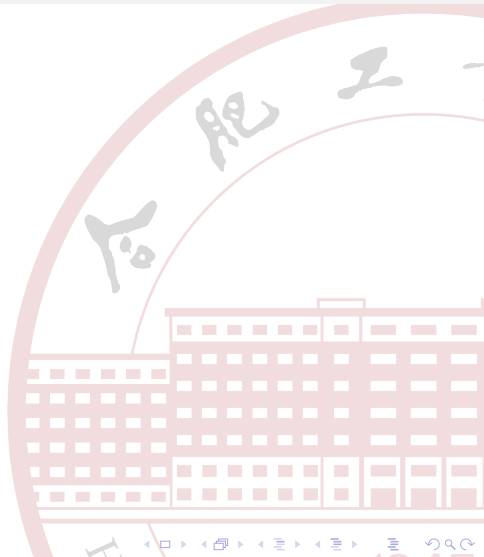
- 单击显示/隐藏：\pause命令
- 单击变色：\alert命令



# 简单动画

BEAMER 比较方便的是实现一些简单的动画效果。  
比方说：

- 单击显示/隐藏：\pause命令
- 单击变色：\alert命令



# 简单动画

同样的事情也可以用\uncover命令做到

```
^^I^^Iint main (void)
^^I^^I{
^^I^^I    std::vector<bool> is_prime (100, true);
^^I^^I    for (int i = 2; i < 100; i++)
^^I^^I
^^I^^I
^^I^^I
^^I^^I
^^I^^I
^^I^^I
^^I^^I    return 0;
^^I^^I}
^^I
```

# 简单动画

同样的事情也可以用\uncover命令做到

```
^^I^^Iint main (void)
^^I^^I{
^^I^^I  std::vector<bool> is_prime (100, true);
^^I^^I  for (int i = 2; i < 100; i++)
^^I^^I    if (is_prime[i])
^^I^^I    {
^^I^^I
^^I^^I
^^I^^I
^^I^^I  }
^^I^^I  return 0;
^^I^^I}
^^I
```

# 简单动画

同样的事情也可以用\uncover命令做到

```
^^I^^Iint main (void)
^^I^^I{
^^I^^I    std::vector<bool> is_prime (100, true);
^^I^^I    for (int i = 2; i < 100; i++)
^^I^^I        if (is_prime[i])
^^I^^I        {
^^I^^I            std::cout << i << " ";
^^I^^I            for (int j = i; j < 100;
^^I^^I                is_prime [j] = false, j+=i);
^^I^^I        }
^^I^^I    return 0;
^^I^^I}
^^I
```

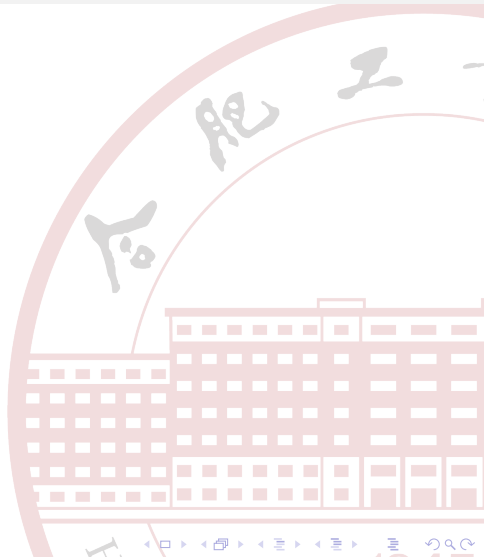
# 简单动画

同样的事情也可以用\uncover命令做到

```
^^I^^Iint main (void)
^^I^^I{
^^I^^I  std::vector<bool> is_prime (100, true);
^^I^^I  for (int i = 2; i < 100; i++)
^^I^^I    if (is_prime[i])
^^I^^I    {
^^I^^I      std::cout << i << " ";
^^I^^I      for (int j = i; j < 100;
^^I^^I        is_prime [j] = false, j+=i);
^^I^^I    }
^^I^^I  return 0;
^^I^^I}
^^I
```

# 目录

- 1 BEAMER 简单入门
- 2 动画
- 3 引用
- 4 BEAMER 的一些坑



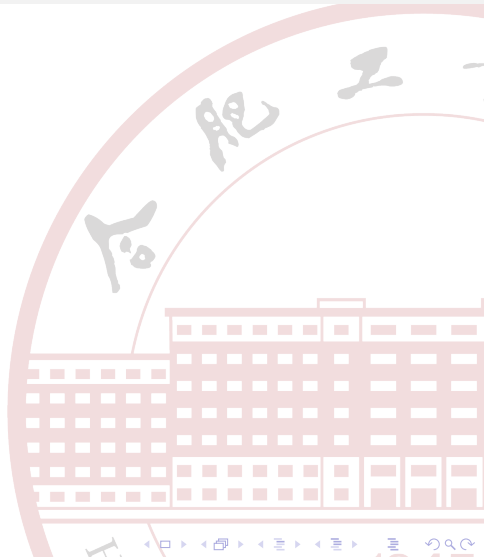


# 引用最好用 $\text{BIB}\text{\LaTeX}$

其实  $\text{BIB}\text{\TeX}$  不一定不可以在  $\text{BEAMER}$  中使用，但是它在  $\text{BEAMER}$  下很可能不如  $\text{BIB}\text{\LaTeX}$  方便。

# 目录

- 1 BEAMER 简单入门
- 2 动画
- 3 引用
- 4 BEAMER 的一些坑



## [fragile] 选项

如果帧包含脆性文本 (fragile text), 在排版帧时将使用不同的内部机制 (mechanisms) 以确保在帧内重置字符代码 (reset character codes)。此时需要给 frame 添加 [fragile] 选项。

所谓的脆性文本通常包括：

- verbatim 环境
- lstlisting 环境
- minted 环境
- ... ..

怎么判断呢？一般来说，需要在编译时加入 `-shell-escape` 的命令或者环境就很可能需要添加 [fragile] 选项。

# 不兼容 `enumitem` 宏包

BEAMER 不兼容 `enumitem` 宏包。原因很简单，因为 BEAMER 本身定义了很多和 `enumerate` 和 `itemize` 环境相关的宏，所以是不兼容的。

# 最好提前准备好便携的 PDF 阅读器

BEAMER 和其他  $\text{\LaTeX}$  文档一样生成的是 PDF，不要等到需要展示的时候才想起来需要准备支持某种特性的 PDF 阅读器。

# 打印参考文献列表需要特殊处理

参考文献列表通常比较长，一帧幻灯片通常无法满足其空间需求。这时候需要给 frame 环境添加 `[allowframebreaks]` 选项。

# 参考书目

- [1] Indian T<sub>E</sub>X Users Group. *L<sub>A</sub>T<sub>E</sub>X Tutorials*. Ed. by E. Krishnan. Indian T<sub>E</sub>X Users Group, 2003. URL: <https://www.tug.org/twg/mactex/tutorials/ltxprimer-1.0.pdf>.
- [2] Philip Kime, Moritz Wemheuer, and Philipp Lehman. *The bibl<sub>at</sub>ex Package - Programmable Bibliographies and Citations*. 2022.
- [3] Tobias Oetiker et al. 一份（不太）简短的L<sub>A</sub>T<sub>E</sub>X 2<sub>ε</sub>介绍. 2021. URL: <https://github.com/CTeX-org/lshort-zh-cn/>.
- [4] Till Tantau et al. *beamer* 文档类用户手册. 2011. URL: <https://raw.githubusercontent.com/latexstudio/LaTeXPackages-CN/master/beamer/beamer%E7%94%A8%E6%88%B7%E6%89%8B%E5%86%8C%E7%BC%88V3.24%E7%BC%89%E4%B8%AD%E8%AF%91%E7%89%88.pdf>.
- [5] 王然. 一份简短的关于 L<sub>A</sub>T<sub>E</sub>X 安装的介绍. 2022. URL: <https://github.com/OsbertWang/install-latex-guide-zh-cn>.

# HFUTTUG 需要你!

我们在: <https://github.com/HFUTTUG>, 欢迎进行贡献!  
向[hfuttug@163.com](mailto:hfuttug@163.com)发邮件以加入我们