Get started with the UIC Beamer Theme

Using LaTeX to prepare slides

Usama Muneeb

October 4, 2026





Beamer for UIC presentations

A quick start

If you would like LaTeX in your presentation, Beamer is a great way to go!

- Beamer has a detailed user manual, but we will go over the most common features.
- This template is designed for a 16:9 aspect ratio, which is the default in PowerPoint and the most common amongst projectors.
- The most common of all slide types involve bulleted points, like these.



Beamer for UIC presentations

A quick start

If you would like LaTeX in your presentation, Beamer is a great way to go!

- Beamer has a detailed user manual, but we will go over the most common features.
- This template is designed for a 16:9 aspect ratio, which is the default in PowerPoint and the most common amongst projectors.
- The most common of all slide types involve bulleted points, like these.
  - Placing `\pause` after items will allow you to sequentially unroll points.
  - Changing `\begin{itemize}` to `\begin{itemize}[<+->]` achieves a similar effect, although `\pause` gives finer control on the unrolling.

Slide Layouts

Some standard slide layouts



Images, columns and tips for good slides

Do use this optional subtitle where appropriate

- **Do keep** a good balance between text and figures.
  - **Do use** nested lists wherever appropriate.
- **Do use** the `\underbrace` and `\overbrace` commands to help explain your equations:

$$\max_{\lambda,\nu} . \quad \underbrace{\inf_{x} \overbrace{L(x,\lambda,\nu)}^{\text{Lagrangian}}}_{\text{Lagrange Dual Function, } g(\lambda,\nu)}$$

$$\text{s.t.} \quad \lambda \geq 0$$

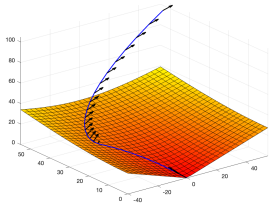


Figure: Minimum fuel trajectory



More with bullet points

Alerts and repeats

Sections 12.1 through 12.3 of the Beamer user manual demonstrate many more features, like alerts and repeats.

- The `\alert{}` feature can be particularly useful.
- Like this.
- The highlighting may span multiple slides in a frame.
  - Previous one did not, but this one will.
- This is accomplished by the hyphen (i.e. `-`) in the `\alert<3->` command.



More with bullet points

Alerts and repeats

Sections 12.1 through 12.3 of the Beamer user manual demonstrate many more features, like alerts and repeats.

- The `\alert{}` feature can be particularly useful.
- Like this.
- The highlighting may span multiple slides in a frame.
  - Previous one did not, but this one will.
- This is accomplished by the hyphen (i.e. `-`) in the `\alert<3->` command.



More with bullet points

Alerts and repeats

Sections 12.1 through 12.3 of the Beamer user manual demonstrate many more features, like alerts and repeats.

- The `\alert{}` feature can be particularly useful.
- Like this.
- The highlighting may span multiple slides in a frame.
  - Previous one did not, but this one will.
- This is accomplished by the hyphen (i.e. `-`) in the `\alert<3->` command.



More with bullet points

Alerts and repeats

Sections 12.1 through 12.3 of the Beamer user manual demonstrate many more features, like alerts and repeats.

- The `\alert{}` feature can be particularly useful.
- Like this.
- The highlighting may span multiple slides in a frame.
  - Previous one did not, but this one will.
- This is accomplished by the hyphen (i.e. `-`) in the `\alert<3->` command.

Another images/columns example

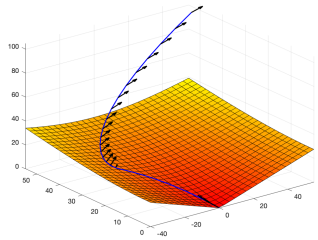


Figure: Minimum fuel trajectory

Left: Solving for optimal fuel consumption **Right:** Solving for optimal time taken

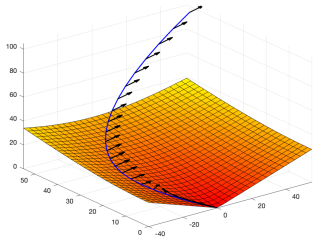


Figure: Minimum time trajectory



Blocks

Some content will just look better in blocks.

Block title

This is a `block`. Its color will match the color of the footline. The `block` environment is native to Beamer.

Block title

This Beamer theme also provides a `colorblock` environment which is a wrapper on top of the usual `block` environment in Beamer. It allows you to specify a custom background and font color.

Here's an example usage:

The discrete case

...

The continuous case

...



Blocks for theorems

Beamer also uses blocks by default to wrap theorems.

Theorem

This is a theorem.

Side-Picture Slides

This Beamer theme also provides a `sidepic` environment which is a wrapper on top of the `frame` environment.

- It has an optional `image` argument which will help you achieve a layout of this type.





Pseudocode Example

Algorithm Bellman-Kalaba (adapted from algorithmicx documentation).

Input: G, u, l, p
  for all $v \in V(G)$ **do**
    $l(v) \leftarrow \infty$
  end for
  $l(u) \leftarrow 0$
  repeat
    for $i \leftarrow 1, n$ **do**
      $min \leftarrow l(v_i)$
      for $j \leftarrow 1, n$ **do**
        if $min > \text{Edge}(v_i, v_j) + l(v_j)$ **then**
          $min \leftarrow \text{Edge}(v_i, v_j) + l(v_j)$ ▷ Example comment
          $p(i) \leftarrow v_j$
        end if
      end for
      $l'(i) \leftarrow min$
    end for
    $changed \leftarrow l \neq l'$
    $l \leftarrow l'$
  until $\neg changed$



Typesetting Algorithms

Common Issues

- There are multiple environments for typesetting algorithms.
  - This template uses `algorithmicx` via `algpseudocode`.
    - `algpseudocode` is (one of the) layouts for `algorithmicx`.
    - Others exist, as mentioned in the documentation, section 2.1.
  - `algorithmic` is also widely used, which is an alternative to using `algorithmicx` (via a layout package).
- While `algpseudocode` tries to be close to `algorithmic`, they use different capitalization schemes
  - `algorithmic` commands are fully capitalized (e.g. `\STATE`, `\FORALL`)
  - `algpseudocode` commands are not (e.g. `\State`, `\ForAll`).
  - **If moving an algorithm from one to the other, you will need to fix the capitalization, or the compiler will run into problems.**



Typesetting Algorithms

More control

- If you must use `algorithmic` instead of `algpseudocode`, some commands will need to be manually defined
  - For example the function `\Call` (or rather, the `\CALL`) command.
    - Do it as `\newcommand*\CALL[2]{\textsc{#1}(#2)}`
    - Remember that not all fonts have a small caps variant needed by `\textsc`
    - You can look at `uicfont.sty` to see how we augment our font with another small caps font (i.e. **PlayFair Display SC**)
  - Additionally, the `\Comment` of `algpseudocode` might look better than the `\COMMENT` of `algorithmic`
    - You can define a custom one to mimic `\Comment`
    - Do it as `\newcommand*\ANNOTATE[1]{\hfill\(\triangleright\) #1}`



Common issues (this is $\hat{P}$ laceholder math)

- Math in title will cause its height to be inconsistent with that of other frames (try viewing these example slides in a full screen mode)
  - This theme provides a `\titlemanualoffset` register which can be used to manually adjust this offset.
    - Putting `\setlength{\titlemanualoffset}{<some calibrated offset>}` before `\begin{frame}` for this frame will fix this issue.
    - **Make sure to zero it out after** `\end{frame}`
- If you use `\verb` command (or `\begin{verbatim}` environment) in a frame, make sure to specify the `[fragile]` option in `\begin{frame}` otherwise the compiler will run into problems.
  - In most cases, `\texttt` can be used instead of `\verb`, and it does not require the `[fragile]` option.

Avoid duplicate logos

If using a logo based background, you can disable the logo in the top left by temporarily switching to a color scheme that does not use a logo. The next slide is such an example.

Personalization

Fonts and color schemes

UIC UNIVERSITY OF ILLINOIS CHICAGO



If you need more space in slides

If you would like more space, you can control the font size by specifying an option in the `\documentclass` command at the beginning of this file.

- Use `\documentclass[smaller]{beamer}` to reduce font size.
- For all options, refer to Section 18.2.1 of the Beamer user manual.



Color schemes

Three fundamental features greatly affect the look and feel of your slides. Two of them can be controlled with the `\themecolor` command (provided by this theme).

- The background color.
  - The default is `light` theme, which has a light background, which we are currently using.
- The foreground color.
  - The `light` theme has a `dark` foreground (i.e. text color) and vice versa.
- The third, the footer color can be set using `\footlinecolor` command.
  - We are currently using `uicblue` color for our footer.
  - The default is no footline, but I believe page numbers are incredibly helpful for your audience to ask questions later.



Color schemes

- I just called `\themecolor{dark}` before `\begin{frame}` for this slide.
  - Ideally you should set the theme globally in the preamble (i.e. before `\begin{document}`). Default is light.
- You can also change the footer color with `\footlinecolor{color}`, as we did for this slide.
- All the changes we talked about here happen for the current **as well as subsequent** frames.
  - I will manually revert all the changes for the next slide.



UIC Colors

- Font color can be set with the `\textcolor{<color name>}{text}` command.
- The colors are defined in the `uiccolor` package, in accordance with the UIC Visual Elements guide:
  - UIC's primary colors are Navy Pier Blue (`uicblue`) and Fire Engine Red (`uicred`).
  - UIC's secondary colors are Chicago Blue (`chicagoblue`), UI Health Teal (`uihteal`) and Champions Gold (`championsgold`).
  - UIC's neutral colors are Expo White (`expowhite`) and Steel Gray (`steelgray` or `steelgrey`).
- A *simpler alternative* to font colors is often just `\emph{}`.



Fonts

Fonts are categorized as Serif and Sans-Serif (see this link for when to use which).

- **Open Sans** and **IBM Plex Serif** have been provided with this template.
- Beamer uses Sans-Serif mode by default.
  - To switch to Serif mode, please change `\usefonttheme[onlymath]{serif}` to `\usefonttheme{serif}` in the preamble.
- If you want to mix and match Serif and Sans-Serif
  - You can still typeset in Serif font in Sans-Serif mode (and vice versa) using `\textrm` and `\textsf` commands.

For more options with fonts, you need to modify the self explanatory `uicfont.sty` file. Next few slides will be helpful in this regard.



Font formats over time

Here are the **old** formats (still used in the LaTeX ecosystem):

- *PostScript Type 1* format developed by Adobe in the 1980s
  - Authoring support officially ended by Adobe in January 2023
- *TrueType format* (`.ttf`) developed by Apple, also in 1980s, licensed to Microsoft

The **new** font formats are *OpenType*, developed jointly by Microsoft and Adobe in the 1990s as an extension of Apple's TrueType font format. *OpenType* fonts are either

- *PostScript flavor OpenType* (`.otf`) that supercedes Adobe's PostScript Type 1
- *TrueType flavor OpenType* (also `.ttf`!) that supercedes Apple's TrueType
  - Distinguishing them from the old TrueType format is non trivial, since they have the same file extension.



Font formats compatibility

The new *OpenType* fonts have several new features over the old formats. They are however, supported natively only by XeLaTeX and LuaLaTeX compilers. The most popular pdfLaTeX compiler has limited support for them:

- *OpenType* fonts are not pdfLaTeX-ready by default.
- They require the production of TeX font metrics and other ancillary files in order to be used with LaTeX.
- The production of these files is not easy for the average user.

Consequently, if using pdfLaTeX, the most convenient option is to restrict yourself to the pdfLaTeX-ready font packages in your TeX distribution (exhaustive list provided in the LaTeX Font Catalog).



Font formats compatibility

If you want to use fonts beyond those in the LaTeX Font Catalog (for example, fonts from Google Fonts), the recommended approach is to use XeLaTeX.

- XeLaTeX is set as the default compiler on the Overleaf version of this template.
- The fonts included in the `fonts` folder are *TrueType flavor OpenType* fonts.
- If pdfLaTeX is used, font packages from your TeX distribution will be used instead.

UIC signature font is **Theinhardt**, which is a proprietary font.

- UIC provides Theinhardt, but proprietary fonts sometimes do not permit PDF embedding which might cause substitutions or compatibility issues.
  - This template uses **Open Sans** which has wider spacing and offers more clarity in a scientific setting than Theinhardt. Additionally, it's OFL licensed.



Chapter slides

- Allow you to partition your presentation into multiple chapters
- Also `frames`, but with a few more options
- Created with `\begin{chapter}[<image>]{<color>}{<title>}`
  - Image is optional, color and title are mandatory.
- Multiple background images have been provided in this template which may be used for chapter slides:
  - Some campus photos that I have taken over the years and;
  - some departmental logos.
  - Let's look at a few examples!

Title goes here

Subtitle goes here



Title goes here

Subtitle goes here

UIC ENGINEERING

Electrical and Computer Engineering

Title goes here

Subtitle goes here



Title goes here

Subtitle goes here

UIC LIBERAL ARTS AND SCIENCES

Mathematics, Statistics, and Computer Science



Other departments

- You can use the generic `uic_lockup.png` background, or
- Download your logo from here.
  - What you need is “departmental lockup in EPS format and RGB/inverted RGB color".
- Overlay it on a static background
  - Upload your EPS logo to this Overleaf and use the `overlay_eps.tex` file to superimpose it onto a static background.
    - Copy over the resulting PDF to the `assets` folder.
  - If EPS logo is not available, download an SVG version and use the `overlay_svg.py` Python script provided in the GitHub repository of this template (link on next slide).



Good luck with your presentation!

- This template is hosted on GitHub
  - `https://github.com/usamamuneeb/uic-beamer-template`
- I would appreciate contributions of all sorts (pull requests, identifying issues, etc).
- If you have any suggestions, send them to me!



Thank you!