

L^AT_EX Starter Kit

Mingxi Liu

Energy & Resources Group

University of California, Berkeley

CONTENTS

I	L^AT_EX basics	3
I-A	How to pronounce L ^A T _E X?	3
I-B	What the heck is L ^A T _E X?	3
I-C	WYTIWYG Vs. WYSIWIG	3
I-D	Pros and cons of L ^A T _E X	3
	I-D1 Why should we use L ^A T _E X?	3
	I-D2 What are the drawbacks of L ^A T _E X?	5
II	Prepare to use L^AT_EX	5
II-A	Text editor	5
II-B	Simple start	6
II-C	Important files and arguments	6
	II-C1 <code>.tex</code> file	6
	II-C2 <code>.cls</code> file	6
	II-C3 <code>.bib</code> file	6
	II-C4 <code>.sty</code> file	7
	II-C5 Options	7
	II-C6 Package	7
	II-C7 Document	7

III	Writing your first academic document using \LaTeX	7
III-A	Document structure	7
III-B	Bullets	9
III-C	Font	10
III-C1	Font size	10
III-C2	Font color	10
III-C3	Font families	10
III-C4	Font styles	11
III-D	Math equations	11
III-D1	In-line text math	11
III-D2	Equation environment	11
III-D3	Long equations	12
III-E	Tables	15
III-F	Figures	16
III-G	Reference	18
IV	\LaTeX is more powerful than you think	19
IV-A	Presentation slides	19
IV-B	Drawing	20
IV-C	Others	21
	Appendix	22

I. L^AT_EX BASICS

A. How to pronounce L^AT_EX?

Even today I always call it /'letɛks/, which is absolutely wrong. The correct pronunciation is /'la:tɛx/, /'la:tɛk/, or /'letɛx/.

B. What the heck is L^AT_EX?

L^AT_EX is used to create documents for others to read. In this respect, it is similar to Microsoft Word. But, the similarities end here.

Document preparation with L^AT_EX typically consists of using a text editor to edit a L^AT_EX source file, which has the extension `.tex`, and then running the L^AT_EX program to convert the source file to a document interchange format such as Postscript or PDF. Once the document is in a document interchange format, it can be previewed on the screen, sent to others, printed, etc.

C. WYTIWYG Vs. WYSIWIG

Most of the popular document processing programs, like Microsoft Word or WPS, are What You See Is What You Get (WYSIWYG). In this type of program, the GUI immediately shows what the final printed copy would look like. The GUI further contains a bunch of buttons that can be used to apply formatting, e.g., line space, bold font, etc. Fig. 1 shows a typical editing window of Microsoft Word for Mac.

In contrast, L^AT_EX is the type of program falls into What You Think Is What You Get (WYTIWYG). L^AT_EX won't immediately show the final print until you compile the source file. The usual mode of interaction with L^AT_EX is via a text editor. An example showing text editing with TeXShop can be found in Fig. 2. The left half of Fig. 2 is the text editing window, and the right half is the preview of the final print after compiling.

D. Pros and cons of L^AT_EX

1) Why should we use L^AT_EX?:

- **Superior typographic quality.** The main reason I use L^AT_EX is that it produces output that is, typographically, far better than any of the alternatives. L^AT_EX has excellent built-in fonts, good algorithms for automatic spacing, and the ability to fine-tune the spacing arbitrarily.

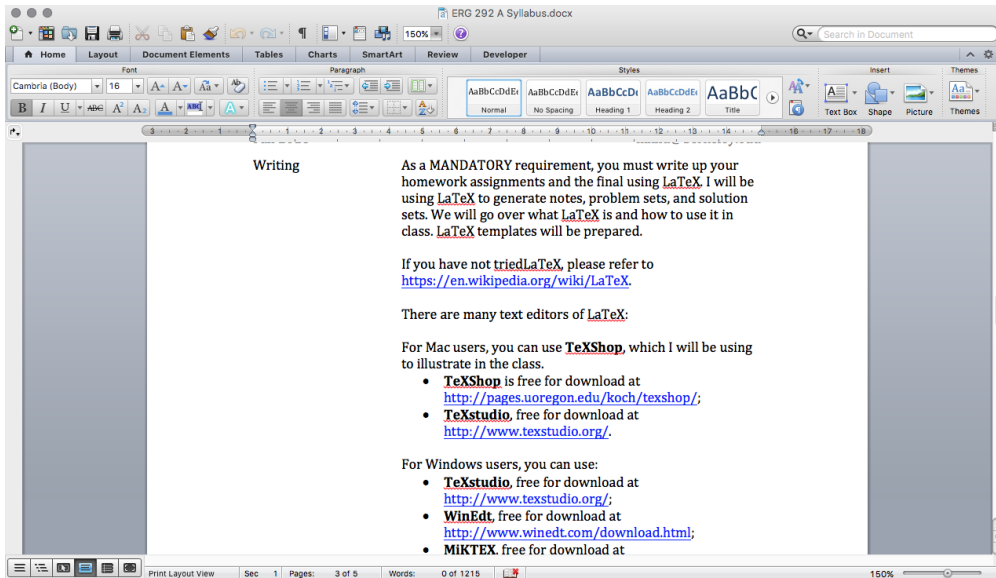


Fig. 1. GUI of Microsoft Word.

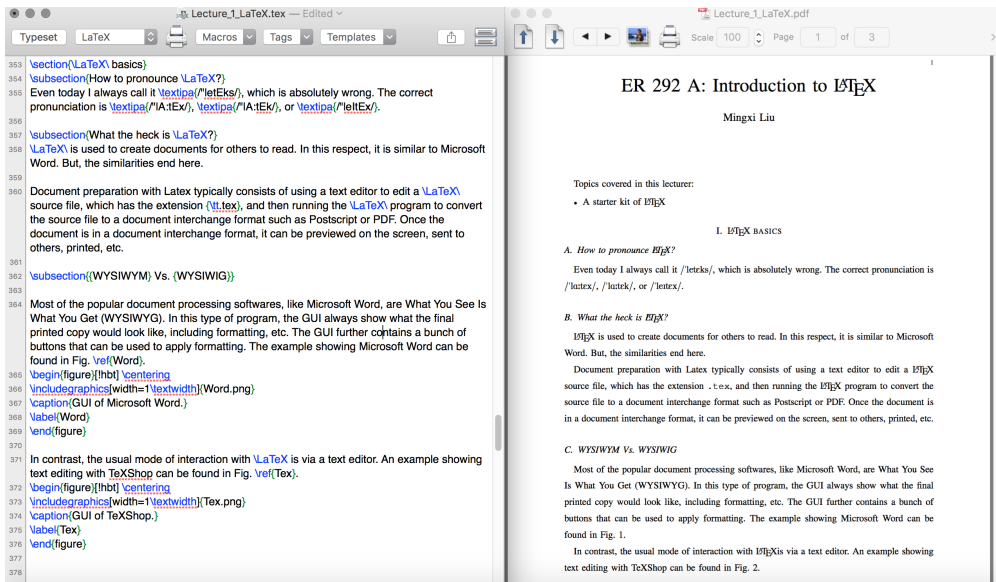


Fig. 2. GUI of TeXShop.

- **Output device independence.** Program like Microsoft Word has a “bug”, where the formatting applied to a document depends on the software version or such.
- **Portability.** L^AT_EX runs on virtually any operating system.
- **Macros and other programmatic features.** L^AT_EX allows us define macros, canned sequences of text and/or markup, which can be used repeatedly.

- **Mathematical typesetting.** Typesetting mathematics is just gorgeous in \LaTeX .
- **Bibliography.** \LaTeX has an extremely neat form of references.

2) *What are the drawbacks of \LaTeX ?:*

- **More moving parts.** Using \LaTeX means using an editor, \LaTeX itself, a document previewer, and usually a few other assorted programs. In contrast, WYSIWYG word processors are self-contained.
- **Difficulty knowing/remembering markup commands.** Learning the markup commands takes time, and it can initially be quite frustrating. That is the reason why provide a template for you. This template contains almost every situation you will meet.
- **Previewing delay.** There is a delay between typing something in the editor and seeing the result in the document previewer.
- **Possibility of syntax errors.** Unlike in a WYSIWYG word processor, it is possible to create a `.tex` file, i.e., the source file, that \LaTeX will reject, complaining of a syntax error. Worse, the actual error report is needlessly cryptic. It takes a bit of experience to learn how to deal with these errors.

II. PREPARE TO USE \LaTeX

A. Text editor

As mentioned before, you need a text editor to enter all the plain text to generate a `.tex` file. Even simple editors, like NotePad, can do this job. However, unless embedded the \LaTeX program, not all text editors can typeset/compile the `.tex` file. There exist many \LaTeX text editors for both Windows and Mac. TABLE I lists some free popular editors and the download link to them. TeXShop is the one I'm going to use for illustration.

TABLE I
LIST OF \LaTeX TEXT EDITORS.

Editor	Link
TeXShop (Mac)	http://pages.uoregon.edu/koch/texshop/index.html
Texmaker (Mac & Windows)	http://www.xmlmath.net/texmaker/
Texstudio (Mac & Windows)	http://www.texstudio.org/
WinEdt (Windows)	http://www.winedt.com/download.html
MiKTeX (Windows)	http://miktex.org/howto/install-miktex

B. Simple start

Ready to \LaTeX ? That's so easy. A simple working \LaTeX document may start with

```
\documentclass[journal,12pt,onecolumn]{IEEEtran}
\usepackage{mathrsfs}
\begin{document}
  Ooh look! It's a \LaTeX\ document.

  $\mathscr{E}=\mathscr{M}c^2$.
\end{document}
```

In this case, the output would be

Ooh look! It's a \LaTeX document.

$$\mathcal{E} = \mathcal{M}c^2.$$

C. Important files and arguments

In contrast to Microsoft Word, a \LaTeX document contains more than one source files. These files include the source file, auxiliary file, class file, log file, image files, etc. In the following, some import files in the document folder and several arguments to make the \LaTeX work will be briefly introduced.

1) *.tex file*: A file ending with *.tex* is the main source file of your document. All characters in the example in Section II-B need to be typed into a *.tex* file.

2) *.cls file*: In the example in Section II-B, inside the $\{\}$ of `\documentclass[]{\}` is the document class file. The class file defines the global format of your document. \LaTeX has many default class files, such as *article*, *book*, *paper*, etc. However, in most cases, when we are preparing documents for a specific journal, conference, or book, a special class file ending with *.cls* is required. Some of the examples include *IEEEtran.cls*, *elsarticle.cls*, *asme2ej.cls*, *wileysix.cls*, etc. This class file serves as the templates of your work. Any non-default class file needs to be put into the root directory of the *.tex* file.

3) *.bib file*: This file contains the metadata of all possible references to be used in your document. This will be introduced in later sections.

4) *.sty file*: This is an optional file for your document. Some authors would like define some new commands or load some commonly-used packages for their own convenience.

5) *Options*: Inside the `[]` of `\documentclass[]{}{}` are the options for formatting the entire document. In the example in Section II-B, `journal` means the format is for IEEE journal papers; `12pt` indicates the base font size; and `onecolumn` puts the document in a single column fashion.

6) *Package*: In the permeable, `\usepackage{}{}` provides us with numerous options for editing. The package, in general, is some packaged L^AT_EX code with an interface. It can add and modify the style of the document but also can just add more functionality. There is no need and it is impossible to include all packages in one document. Some packages also have conflicts, thus only necessary packages need to be loaded in the preamble before `\begin{document}`. Some sample/popular packages include `amsmath`, `cite`, `color`, `multicol`, `graphicx`, etc.

7) *Document*: The main body of your document is embraced by `\begin{document}` and `\end{document}`.

III. WRITING YOUR FIRST ACADEMIC DOCUMENT USING L^AT_EX

A. Document structure

Let's first see an example showing the document structure. The input below

```
\documentclass[journal,12pt,onecolumn]{IEEEtran}
\usepackage{mathrsfs}
\begin{document}
\title{A \LaTeX\ Example}
\author{Mingxi Liu}
\maketitle
\section{Introduction}
\LaTeX\ \cite{Latex_book} is a powerful tool.
  \subsection{Basics}
  Papers \cite{Kittner_RE_2016, Callaway_PRL_2000} are generated
  by \LaTeX.
  \subsubsection{How to download}
\section{Start using}
```

```

\subsection{Text editor}
\section{Conclusion}
\LaTeX\ is indeed a powerful tool.
\bibliographystyle{ieeetr}
\bibliography{ERG_bib}
\end{document}

```

will generate a document shown in Fig. 3. From this simple example, we have the following

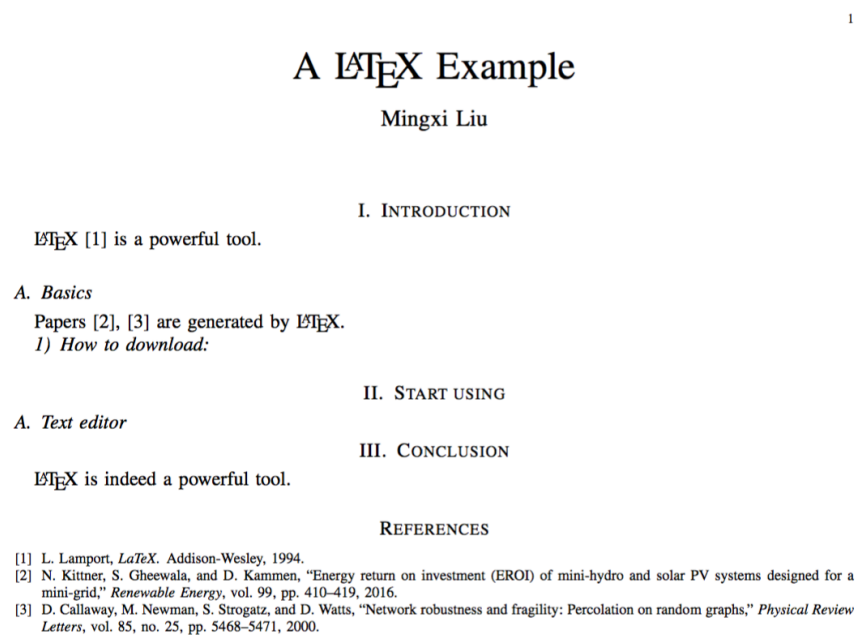


Fig. 3. An example of L^AT_EX.

remarks:

- `\title{}` gives the title of this document.
- `\maketitle` generates the title.
- `\author{}` gives the authors.
- Three or more levels of sections can exist in one document. `\section{}`, `\subsection{}`, and `\subsubsection{}` give the first, second, and third level of sections, respectively.
- `\bibliographystyle{}` gives the reference citing style, e.g., IEEE, APA, etc.

- `\bibliography{}` generates the references contained in the `.bib` file and cited in the document.

B. Bullets

Convenient and predictable list formatting is one of the many advantages of using \LaTeX . The environment for constructing a bullet list is `itemize`. Please see the following example for details.

Example:

- Energy
 - Renewable energy
 - * Solar
 - * Tidal
 - * Wind
 - Conventional energy
 - * Fossil-based
 - * Hydro
 - Nuclear
- Resources

The code for generating this list is as follows.

```
\begin{itemize}
  \item Energy
    \begin{itemize}
      \item Renewable energy
        \begin{itemize}
          \item Solar
          \item Tidal
          \item Wind
        \end{itemize}
      \item Conventional energy
        \begin{itemize}
```

```

        \item Fossil-based
        \item Hydro
    \end{itemize}
    \item Nuclear
\end{itemize}
\item Resources
\end{itemize}

```

C. Font

1) *Font size*: In \LaTeX , there are many levels of font sizes. The following gives an example of font size adjustment.

XXXXXXXXXX

The commands for generating the above line are as follows.

```

{\Huge X}{\huge X}{\LARGE X}{\Large X}{\large X}{\normalsize X}
{\small X}{\footnotesize X}{\scriptsize X}{\tiny X}

```

2) *Font color*: To enable the color feature, we need to load `\usepackage{color}` in the preamble. Then in the text, simply type `{\color{colorname}text}` to generate colored text. An example below shows some default colors.

ERGie

The commands for generating the above line are as follows.

```

{\color{blue}E}{\color{red}R}{\color{yellow}G}
{\color{green}i}{\color{cyan}e}

```

3) *Font families*: By default, in standard \LaTeX classes the serif typeface i.e., roman, font is used. The other font typefaces (sans serif and typewriter, i.e., monospace) can be used by entering some specific commands. We use the following example to illustrate.

ERGie **ERGie** **ERGie**

Three font families are used in the above line, and the commands for generating it is

```

\textrm{ERGie}          \textsf{ERGie}          \texttt{ERGie}

```

4) *Font styles*: Some most commonly used font styles include normal, italic, bold, uppercase, small capitals, and underline. The following example will illustrate those popular styles.

normal *italic* **bold** UPPERCASE SMALL CAPITALS underline

The commands for generating the above line are

```
normal      \emph{italic}      \textbf{bold}      \uppercase{uppercase}
\textsc{small capitals}      \underline{underline}
```

D. Math equations

Mathematical expression is one of the most powerful features of \LaTeX . Different from other WYSIWYG programs, math equations in \LaTeX are realized by using command-like expressions. In addition, in order to type math equations, we need to enter the “math mode”. Briefly, there are two environments where mathematical expressions are typed, i.e., in-line text environment and equation environment.

1) *In-line text math*: To enter the in-line text math typing, we need to use two $\$$ s to embrace the math expression. For example, the following in-line text

“By following a similar logic, we have $\sum_{i=1}^{\infty} x_i = 1.$ ”

is generated by

```
By following a similar logic, we have  $\sum_{i=1}^{\infty} x_i = 1.$ 
```

2) *Equation environment*: Sometimes the equation needs to be written in a new line and given a citing label. In this case, we need to enter the equation environment. For example, to have

“By following a similar logic, we have

$$\sum_{i=1}^{\infty} x_i = 1. \tag{1}$$

In (1), x_i is randomly generated.”

the input would be

```
By following a similar logic, we have
\begin{equation} \label{eqn_exp}
\sum_{i=1}^{\infty} x_i = 1.
\end{equation} In \eqref{eqn_exp},  $x_i$  is randomly generated.
```

Several remarks regarding equations are as follows.

- `\begin{equation}` and `\end{equation}` set up the equation environment. As long as within the equation environment, the `$` is no longer necessary.
- `\label{}` equips the equation with a label for the purpose of referring to in the document. To refer to this equation, calling `\eqref{labelname}` in the text gives the bracketed equation number.

The mathematical expression typing in \LaTeX might be torturing in the very beginning, however, the output is extremely beautiful and neat. For a full list of \LaTeX math symbols, please refer to <https://en.wikibooks.org/wiki/LaTeX/Mathematics>. You can always Google “how to type *** in latex” for instant help.

3) *Long equations*: When dealing with some complicated math derivations, we sometimes may encounter the situation that the length of the equation is longer than the page width. In some other cases, we may need several lines to show the derivation procedure. To this end, it is necessary to adopt other math environments. Several commonly-used environments are `subequations`, `eqnarray`, `align`, `aligned`. This will be revealed by the following examples.

Examples:

1.

$$z = \overbrace{\underbrace{x}_{\text{real}} + i \underbrace{y}_{\text{imaginary}}}^{\text{complex number}} \quad (2)$$

```
z = \overbrace{
  \underbrace{x}_{\text{real}} + i
  \underbrace{y}_{\text{imaginary}}
}^{\text{complex number}}
```

2.

$$\lim_{x \rightarrow 0} \frac{e^x - 1}{2x} \overset{\left[\frac{0}{0}\right]}{\underset{H}{=}} \lim_{x \rightarrow 0} \frac{e^x}{2} = \frac{1}{2}$$

```
\lim_{x \to 0} {\frac{e^x-1}{2x}}
\overset{\left[\frac{0}{0}\right]}
```

```
{\underset{\mathrm{H}}{=}}
\lim_{x\to 0}{\frac{e^x}{2}}={\frac{1}{2}}
\nonumber
```

3. Maxwell's equations:

$$B' = -\nabla \times E, \quad (3a)$$

$$E' = \nabla \times B - 4\pi j, \quad (3b)$$

```
\begin{subequations}
Maxwell's equations:
\begin{align}
B' &= -\nabla \times E, \\
E' &= \nabla \times B - 4\pi j,
\end{align}
\end{subequations}
```

4.

$$A = \begin{bmatrix} 1 & 2 & \cdots & n \\ E & R & \cdots & G \\ \vdots & \vdots & \ddots & \vdots \\ U & C & \cdots & B \end{bmatrix} \quad (4)$$

```
\begin{equation}
A=\left[ \begin{array}{cccc}
1 & 2 & \cdots & n \\
E & R & \cdots & G \\
\vdots & \vdots & \ddots & \vdots \\
U & C & \cdots & B
\end{array} \right]
\end{equation}
```

5.

$$\mathcal{X}(k) = \begin{bmatrix} x_{\ell_1}(k)^T \cdots x_{\ell_n}(k)^T \\ x_{\ell_p}(k)^T \cdots x_{\ell_q}(k)^T \end{bmatrix}^T, \quad (5)$$

```
\begin{equation}
\begin{aligned}
\mathscr{X}(k) &= \left[ \begin{array}{c} x_{\ell_1}(k)^{\mathsf{T}} \cdots \\ x_{\ell_n}(k)^{\mathsf{T}} \end{array} \right. \\
&\quad \left. \left. \begin{array}{c} \left. x_{\ell_p}(k)^{\mathsf{T}} \cdots \\ x_{\ell_q}(k)^{\mathsf{T}} \end{array} \right] \right]^{\mathsf{T}}, \\
\end{aligned}
\end{equation}
```

6.

$$\begin{aligned} x_{i+1} &= Ax_i + Bu_i + D\omega_i, \\ Cx_i + Du_i + E\omega_i &= y_i. \end{aligned} \quad (6)$$

```
\begin{equation}
\begin{aligned}
x_{i+1} &= Ax_i + Bu_i + D\omega_i, \\
Cx_i + Du_i + E\omega_i &= y_i.
\end{aligned}
\end{equation}
```

7.

$$x_{i+1} = Ax_i + Bu_i + D\omega_i, \quad (7)$$

$$Cx_i + Du_i + E\omega_i = y_i. \quad (8)$$

```
\begin{eqnarray}
x_{i+1} &=& Ax_i + Bu_i + D\omega_i, \\
Cx_i + Du_i + E\omega_i &=& y_i.
\end{eqnarray}
```

```
\end{eqnarray}
```

E. Tables

Different from the math equations, tables in L^AT_EX must be put in the table environment. This will be illustrated by the following example.

Example:

A table describing how much I like different animals is shown in Table II. The input to

TABLE II
LIST OF ANIMALS.

Number	Name	Location	Like?
1	Pig	All around the world	The most
2	Eagle	US	OK
3	Snake	Canada	No way

generating this table is

```
\begin{table}[!hbt]
\caption{List of animals.}
\label{Table_exp}
\centering
\begin{tabular}{|c|c|c|c|}
\hline
Number & Name & Location & Like? \\
\hline
1 & Pig & All around the world & The most \\
\hline
2 & Eagle & US & OK \\
\hline
3 & Snake & Canada & No way \\
\hline
\end{tabular}
\end{table}
```

To illustrate how this works, we have the following statements.

- `\begin{table}` and `\end{table}` construct the table environment.
- `!hbt` sets the position of this table. Priority: here, bottom, and top. You can switch the priorities according to your own preference.
- `\label{}` gives the citing label of this table. To refer to this table in the document, simply call `Table \ref{lablename}`, `TABLE \ref{lablename}`, or `Tab. \ref{lablename}` in the text. `Table`, `TABLE`, or `Tab.` is decided by the document class.
- `\caption{}` gives the caption of this table.
- `\centering` aligns the table to the center. You can also use `\raggedright` and `\raggedleft` to align the table to the right and left, respectively.
- `\begin{tabular}` and `\end{tabular}` construct the table environment, which is normally text based. Thus, within `tabular`, any math expressions need to be encased by `$`.
- `{|c|c|c|c|}` defines the horizontal alignments of each column of the table. `c` represents aligning to column center. Whereas, `l` and `r` align the column elements to left and right, respectively. The number of columns are controlled by the total number of `c`, `l`, and `r`. The `|` draws the vertical line. For three-line tables, no vertical lines are needed.
- `\hline` draws the horizontal line. For three-line tables, only the first two and the last `\hline` are needed.
- `\\` changes the row. This must be added to the end of each row.
- `&` separates elements in different columns.
- Methods to composing more advanced table formats including multicolumn tables can be readily obtained by Googling.

F. Figures

Inserting figures to the document using \LaTeX might be a little bit complicated. However, as long as necessary packages are included, there won't be any problem. \LaTeX cannot manage pictures directly, so we will need some extra help: we have to load the `graphicx` package in the preamble of our document. Having this package loaded, the following figure formats are supported by \LaTeX .

- `.eps`: Strongly recommended. This type of vector image won't get blurred when you enlarge/print the document. Windows users may need to load `epstopdf` package right after the `graphics` package to get the `.eps` inserted.
- `.png`: Recommended. It is a very common and lossless format and it's the best choice if `.eps` cannot be obtained.
- `.pdf`: Not quite recommended.
- `.jpg`: Strongly not recommended. Easily getting blurred when it is enlarged.

Let's see an example to illustrate how to insert a figure in your document.

Example: The seal of UC Berkeley is shown in Fig. 4. To insert this figure, we use the



Fig. 4. Seal of UC Berkeley.

following commands.

```
\begin{figure}[!hbt] \centering
\includegraphics[width=0.4\textwidth]{Berkeley.eps}
\caption{Seal of UC Berkeley.}
\label{Berkeley_seal}
\end{figure}
```

To illustrate how this works, we have the following remarks.

- `\begin{figure}` and `\end{figure}` construct the figure environment.

- `[!hbt]` sets the position of this figure. Priority: here, bottom, and top. You can switch the priorities according to your own preference.
- `\label{}` gives the caption of this figure. To refer to this figure in the document, simply call `Figure \ref{lablename}` or `Fig. \ref{lablename}` in the text. `Figure` or `Fig.` is decided by the document class.
- `\includegraphics[]{}` loads the figure files to \LaTeX .
- `[width=0.4\textwidth]` controls the size of this figure. `\textwidth` is the width of the document body.
- `{Berkeley.eps}` is the figure file which should be put in the same directory of the `.tex` file.
- You need to explore more advanced functions by yourselves. Some of them include `clip`, `minipage`, etc.

G. Reference

As mentioned above, \LaTeX has a very powerful reference management system. The only thing you need to do when writing a document is focusing on the writing itself, and \LaTeX will take care of your references. In \LaTeX , all possible references (whether or not you cite them) are stored in a `.bib` file. As a plain text, one `.bib` file contains many items defining all the included references. For example, a journal paper by Kittner *et al.* is represented by

```
@article{Kittner_RE_2016,
  Author = {N. Kittner and S. Gheewala and D. Kammen},
  Date-Added = {2016-08-10 21:03:29 +0000},
  Date-Modified = {2016-08-10 21:04:49 +0000},
  Journal = {Renewable Energy},
  Pages = {410-419},
  Title = {Energy return on investment ({EROI}) of
           mini-hydro and solar {PV} systems designed
           for a mini-grid},
  Volume = {99},
  Year = {2016}}
```

Several comments regarding the above input:

- `@article` gives that the type of this reference is `article` or `journal`. There are also many other choices, like `book`, `conference`, `webpage`, etc.
- `Kittner_RE_2016` is the cite key, which will be called in the document to cite this reference. To cite this paper in the document, we call `\cite{Kittner_RE_2016}`. Use comma to separate if you have more than one to cite together.
- `Author={}` contains names of all authors. Format should follow specific requirements, but putting full names there will never be a problem.
- `Date-Added` and `Date-Modified` are not necessary.
- `Journal={}` gives the journal name. If other types of reference are selected, this `Journal` is subject to change.
- `Title={}` contains the title of this reference. Note that, in journal papers, only the first character of the first word is capitalized. Thus, words like EROI must be encased to ensure it is capitalized.

The easiest way to have the above entry is not typing by yourselves. Almost all publishers provide the citation export, where you can choose `bibtex` to get entries similar to the above example. Some adjustments are necessary if you want to keep all references in consistency. However, it is also not smart to copy and paste those entries every time. To this end, some softwares provide GUI and generate very neat `.bib` files. Some of the examples are BibDesk, Mendeley, EndNote, etc.

IV. L^AT_EX IS MORE POWERFUL THAN YOU THINK

A. Presentation slides

L^AT_EX can also be used in preparing professional presentation slides. One popular template for this purpose is the `beamer` template, which requires the class file `beamer`. An example can be shown in Fig. 5. Comparing with slides generated by Microsoft PowerPoint and Prezi, the L^AT_EX slides are not quite dynamic. Since the output file is always `.pdf`, limited animations can be realized. But again, presenting mathematical equations is still the pros of L^AT_EX slides.

Modeling (cont'd)



- Index-based aggregation model using the state bin idea:

$$\begin{aligned}\zeta^+ &= f(\zeta, u), \\ y &= \zeta^T u m P_{ave}.\end{aligned}$$

Further we have

$$\zeta^+ = A(u)\zeta,$$

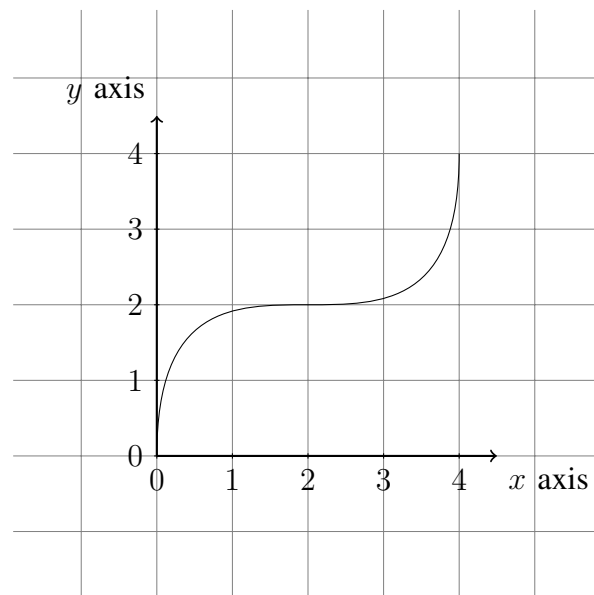
where

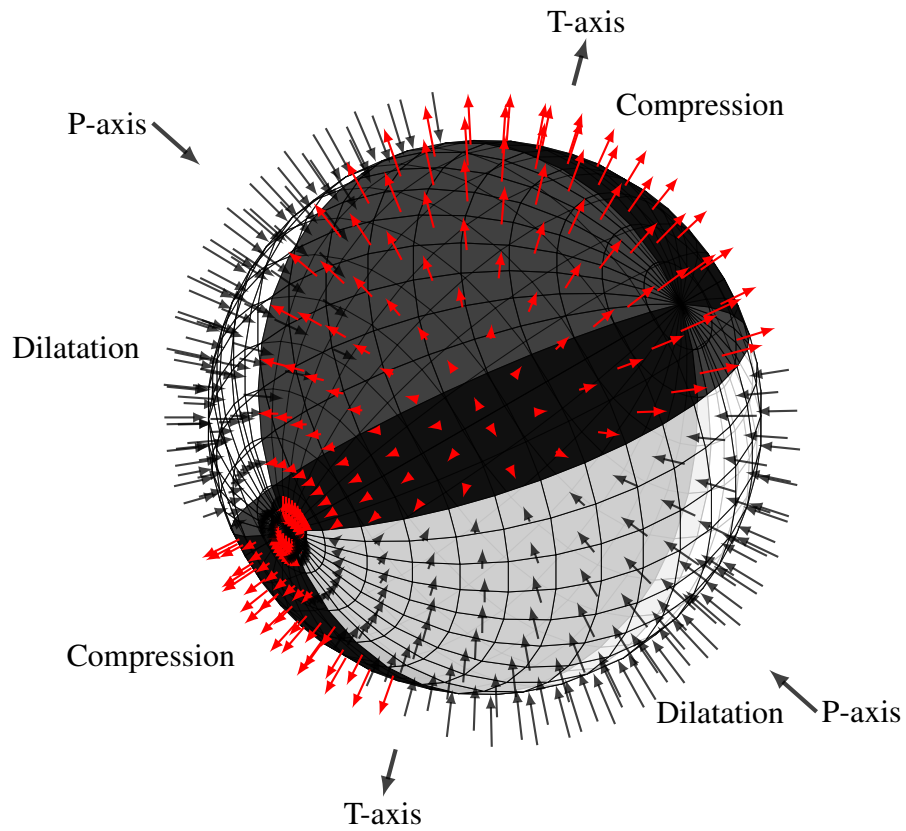
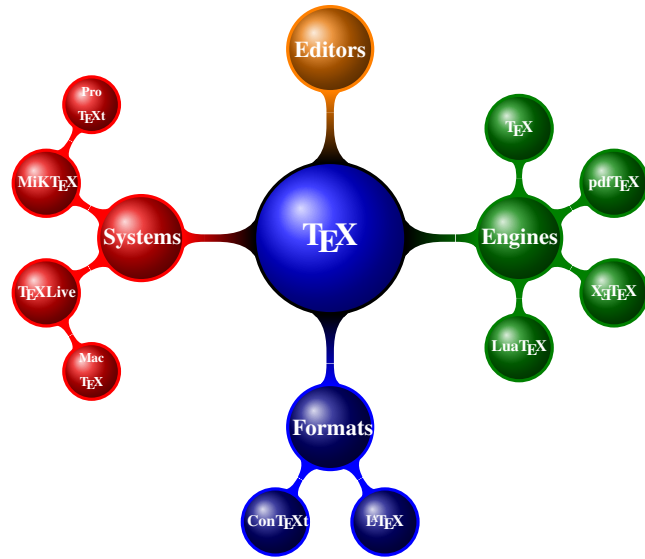
$$A(u) = \begin{bmatrix} a_{11}(u_1) & a_{12}(u_2) & \cdots & a_{1n}(u_n) \\ a_{21}(u_1) & a_{22}(u_2) & \cdots & a_{2n}(u_n) \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1}(u_1) & a_{n2}(u_2) & \cdots & a_{nn}(u_n) \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

Fig. 5. An example of professional slides.

B. Drawing

There is one package called `TikZ` that allows you to draw high quality and often quite complex diagrams directly in `LATEX`. Three simple examples can be shown below.





Seismic focal mechanism and Pression-Tension axis.

C. Others

More advanced features are subject to your own exploration.

APPENDIX

The tips below for successfully compiling \LaTeX files are based on my own experience. They may not cover all problems I met before but I will keep them updated.

- Never forget Google. Whenever you get a syntax error and don't know how to deal with, simply copy and Google it. There are many folks having the same problems. Google always works.
- The workflow of compiling \LaTeX is PDFLaTeX \rightarrow BibTeX \rightarrow PDFLaTeX \rightarrow PDFLaTeX.
- Copy the whole folder when you want to transplant your document to another PC/Mac.
- For special symbols, such as \$, %, and #, make sure you put a \ in front of the symbol to avoid syntax errors.
- Once entered math mode, every character you type is in math format. If text format is needed, use `\text{your text}` to express.
- You can use \LaTeX syntax in both MathType and Matlab.
- In text mode, you need at least one blank line to start a new paragraph.
- Sometimes, though syntax errors are fixed, \LaTeX refuses to work again. To get it back, simply delete all source files, ones with the same name as the `.tex` file, except the `.tex` file. Typeset again and everything will come back.
- % serves as comment.
- ‘ is realized by the key on the left of “1” on your keyboard.
- Sometimes \LaTeX gives you a large blank space in one page. This might be caused by a long equation, a too-large figure, a too-long table, etc. You can always eliminate this space by adjusting the document setup.

Enjoy typing!