

LaTeX Workshop

Joshua Pribe

13 February, 2020

For these slides and some practice material, go to

<https://github.com/jpribe/latex-course>

You should be able to log in to www.overleaf.com with your Purdue login info to do the exercises

Outline

- What is LaTeX?
 - LaTeX vs. Word (and other “WYSIWYG” word processors)
 - Situations where LaTeX is more/less efficient
- Basics of LaTeX coding
 - Outline of a document
- LaTeX for academic papers
 - Equations
 - Figures
 - Bibliography/citations

LaTeX background

- Essentially a programming language
- You write code, and a compiler turns it into a pretty-looking document

```
28 \title{Plastic strain gradients and transient fatigue crack growth: a computational
study}
29
30 %% Group authors per affiliation:
31 \author[mymainaddress]{Joshua D. Pribe}
32 \author[mymainaddress]{Thomas Siegmund\corref{mycorrespondingauthor}}
33 \author[mysecondaddress]{Vikas Tomar}
34 \author[mythirdaddress]{Jamie J. Kruzic}
35
36 %% or include affiliations in footnotes:
37 \cortext[mycorrespondingauthor]{Corresponding author}
38
39 \address[mymainaddress]{School of Mechanical Engineering, Purdue University, West
Lafayette, IN 47907, USA}
40 \address[mysecondaddress]{School of Aeronautics and Astronautics, Purdue University,
West Lafayette, IN 47907, USA}
41 \address[mythirdaddress]{School of Mechanical and Manufacturing Engineering,
University of New South Wales, Sydney, NSW 2052, Australia}
```

Plastic strain gradients and transient fatigue crack growth: a
computational study

Joshua D. Pribe^a, Thomas Siegmund^{a,*}, Vikas Tomar^b, Jamie J. Kruzic^c

^a*School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907, USA*

^b*School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907, USA*

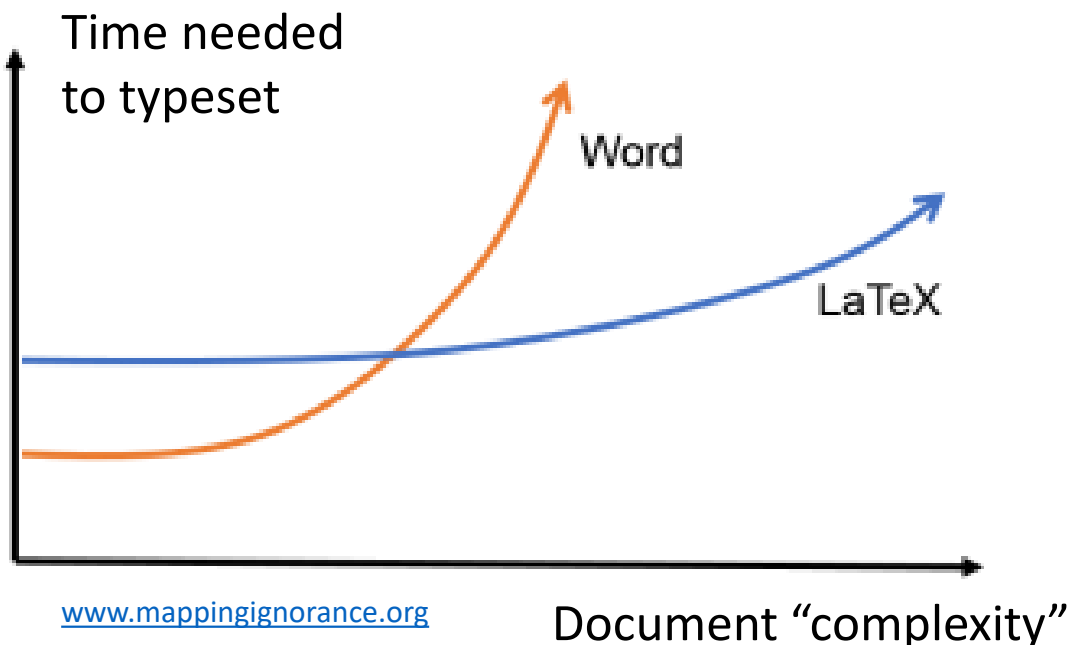
^c*School of Mechanical and Manufacturing Engineering, University of New South Wales, Sydney, NSW 2052, Australia*

LaTeX advantages

- Focus on content instead of appearance
 - Style files take care of appearance when you compile your code
- Write complex equations and reference them in the text
- Easily insert and reference figures
- Copy and paste code snippets
 - Once you learn something once, you never have to do it again!

LaTeX vs. Word

- LaTeX is more efficient as you add more “stuff” to a document
 - Equations
 - Figures
 - References/special characters in the text
- Word processors work better for large blocks of text



- Steep, but relatively short learning curve
- Google is your friend!

LaTeX Basics

- Write in plain text
- Commands and environments specify the structure
 - Commands start with a backslash `\` and typically include an argument in braces `{ }`

```
Here is some      boring normal text.
```

```
Here is some emphasized text: \emph{This text is important}.
```

Here is some boring normal text.

Here is some emphasized text: *This text is important.*

- Words are separated by one or more spaces
- Paragraphs are separated by one or more blank lines
- LaTeX automatically handles spacing, indentation, etc. based on a style file

LaTeX Basics

- Write in plain text
- Commands and environments specify the structure
 - Commands start with a backslash `\` and typically include an argument in braces `{ }`
 - Environments tell the interpreter how to interpret a block of code

```
Here is a bulleted list:  
\begin{itemize}  
  \item Item 1  
  \item Item 2  
  \item Item 3  
\end{itemize}
```

Here is a bulleted list:

- Item 1
- Item 2
- Item 3

- The `\begin{ }` and `\end{ }` commands mark the start and end of an environment
- The `itemize` environment makes a bulleted list
- The `enumerate` environment makes a numbered list

LaTeX Basics

You can also include math in line (this is where LaTeX really shines!)

Use dollar signs (`\$`) when you want to refer to a variable like `\sigma` or `a` or a simple equation like `\sigma = E \epsilon` in the text.

Use dollar signs (\$) when you want to refer to a variable like σ or a or a simple equation like $\sigma = E\epsilon$ in the text.

- Text in between the \$ signs is interpreted in “math mode”
- Need to “escape” the \$ by writing `\$` in the code
 - Do this for `$&%\#` signs
- More on equations later...

Document structure

- Always start with the `\documentclass` command
 - Many journals and publishers have their own `documentclass` (e.g. `elsarticle` for Elsevier journals)
- `%` sign indicates a comment
- All document text goes between `\begin{document}` and `\end{document}`
- Let's add some organization to our document...

```
\documentclass{article}

\begin{document}

This is my first \LaTeX\ document. % your content goes here...

\end{document}
```

This is my first L^AT_EX document

Document structure

- `\section{ }` and `\subsection{ }` create numbered sections and subsections
- Adding a `*` removes the number (e.g. `\section*{ }`)

```
\documentclass{article}

\begin{document}

\section{Introduction}
This is my first \LaTeX\ document.

\section{Methods}

\section{Results}

\subsection{Experiments}

\subsection{Finite element modeling}

\section{Conclusions}

\section*{References}

\end{document}
```

1 Introduction

This is my first \LaTeX document.

2 Methods

3 Results

3.1 Experiments

3.2 Finite element modeling

4 Conclusions

References

Document structure

- The “preamble”: space between `\documentclass` and `\begin{document}`
- In the preamble you can:
 - Include packages: get access to more commands and environments
 - Define metadata (authors, title, date, ...)

- `amsmath` is useful for equations
- `graphicx` needed for making figures
- `subcaption` lets you make multi-part figures

```
\documentclass{article}

\usepackage{amsmath}
\usepackage{subcaption}

\title{\LaTeX\ Tutorial}
\author{Joshua Pribe}
\date{13 February, 2020}

\begin{document}
\maketitle

\section{Introduction}
This is my first \LaTeX\ document.

\section{Methods}

\section{Results}
```

LaTeX Introduction

Joshua Pribe

13 February, 2020

1 Introduction

This is my first LaTeX document.

2 Methods

3 Results

3.1 Experiments

3.2 Finite element modeling

4 Conclusions

References

Equations

Equations: Math mode basics

- Within the text, use `$` signs to signify math mode
- LaTeX ignores spaces in math mode
 - Use spaces freely to make your code look nice

Use caret `^` for superscripts and underscore `_` for subscripts.

```
$y = c_2 x^2 + c_1 x + c_0$
```

$$y = c_2 x^2 + c_1 x + c_0$$

Use curly braces `{}` `}` to group superscripts and subscripts.

```
$F_n = F_{n-1} + F_{n-2}$ % oops!
```

$$F_n = F_n - 1 + F_n - 2$$

```
$F_n = F_{n-1} + F_{n-2}$ % ok!
```

$$F_n = F_{n-1} + F_{n-2}$$

There are commands for Greek letters and common notation.

```
$\mu = A e^{Q/RT}$
```

$$\mu = A e^{Q/RT}$$

```
$\Omega = \sum_{k=1}^n \omega_k$
```

$$\Omega = \sum_{k=1}^n \omega_k$$

Capital first letter in the code → capital Greek letter

Equations: The `equation` environment

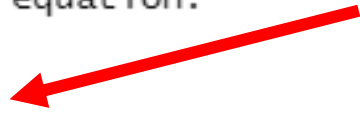
For numbered equations, use the `equation` environment

This is my first `\LaTeX` document.

Now I will write an equation:

```
\begin{equation}
  \label{eq:myEq}
  \bar{\sigma} = \sigma_Y \sqrt{f^2 ( \epsilon^p ) + l\eta^p}.
\end{equation}
```

Labeling your equation makes
it easy to reference later



In Eq. \ `\eqref{eq:myEq}`, `\sigma_Y` is the yield strength, `\ldots`

This is my first \LaTeX document. Now I will write an equation:

$$\bar{\sigma} = \sigma_Y \sqrt{f^2(\epsilon^p) + l\eta^p}. \quad (1)$$

In Eq. (1), σ_Y is the yield strength, ...

Equations: Other considerations

Sometimes equations look ugly without some additional commands...

```
\begin{equation}  
T_n = \sigma_{\max} \exp (1 - \frac{\Delta_n}{\delta_0} ) \frac{\Delta_n}{\delta_0}  
\end{equation}
```

$$T_n = \sigma_{max} \exp(1 - \frac{\Delta_n}{\delta_0}) \frac{\Delta_n}{\delta_0} \quad (2)$$

Equations: Other considerations

Sometimes equations look ugly without some additional commands...

```
\begin{equation}
T_n = \sigma_{\max} \exp (1 - \frac{\Delta_n}{\delta_0} ) \frac{\Delta_n}{\delta_0}
\end{equation}
```

$$T_n = \sigma_{\max} \exp(1 - \frac{\Delta_n}{\delta_0}) \frac{\Delta_n}{\delta_0} \quad (2)$$

```
\begin{equation}
T_n = \sigma_{\text{max}} \operatorname{exp} \left( 1 - \frac{\Delta_n}{\delta_0} \right)
\frac{\Delta_n}{\delta_0}
\end{equation}
```

$$T_n = \sigma_{\max} \exp \left(1 - \frac{\Delta_n}{\delta_0} \right) \frac{\Delta_n}{\delta_0} \quad (2)$$

- Use `\text{ }` if you want plaintext in an equation
- Use `\operatorname{ }` for functions like `exp`
- `\left(` and `\right)` make sure the `()` look nice

Equations: Multiple lines

- Use the `split` environment to handle equations that span multiple lines
 - Nested within an equation
 - `&` indicates where the equations should align
 - `\\` starts a new line
 - Equation number is always centered vertically
 - If you want a label, define it before the `split` environment

```
\begin{equation}
\label{eq:splitEq}
\begin{split}
y &= mx + b \\
&= m(x - x_1) + y_1
\end{split}
\end{equation}
```

$$\begin{aligned} y &= mx + b \\ &= m(x - x_1) + y_1 \end{aligned} \tag{2}$$

Figures

Figure environment

- Include the `graphicx` package
- Upload a JPEG, PNG, or PDF file to your Overleaf project
- Within the `figure` environment, use the `includegraphics` command

```
\documentclass{article}
\usepackage{graphicx}

\begin{document}

    \begin{figure}
        \centering
        \includegraphics{crackGrowthRates.pdf}
        \caption{Normalized crack growth rates for different realizations of the model.}
        \label{fig:crackGrowthRates}
    \end{figure}

\newpage
```

Figure environment

```
Figure \ref{fig:crackGrowthRates} shows \ldots
\begin{figure}
  \centering
  \includegraphics{crackGrowthRates.pdf}
  \caption{Normalized crack growth rates for different realizations of the model.}
  \label{fig:crackGrowthRates}
\end{figure}
\newpage
```

LaTeX automatically sets the caption font size and the spacing before/after the figure based on the style file.

There are commands to change this if you'd like

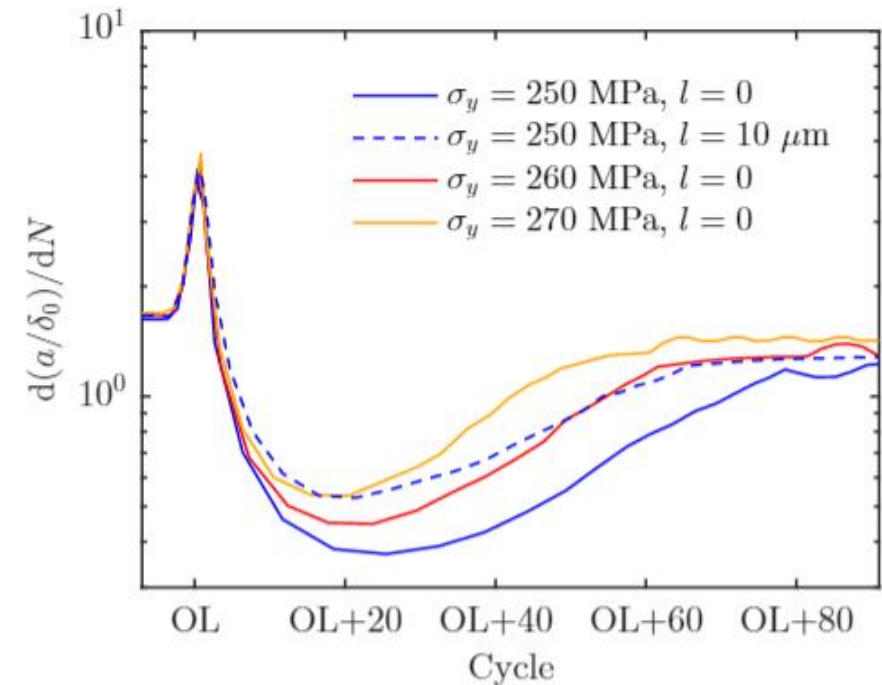


Figure 1: Normalized crack growth rates for different realizations of the model.

Figure 1 shows ...

Aside: floats

- A figure is a float: LaTeX automatically places it in the “best” location
- You can suggest a location by adding an optional argument as input to the figure environment
 - `\begin{figure} [b]` → place the figure at the bottom of the page
 - `\begin{figure} [t]` → ...at the top
 - `\begin{figure} [h]` → place the figure here
- Add an `!` before the letter to *force* LaTeX to do what you want, even if it thinks it's a bad idea (e.g. `[!b]`)

Figure environment: Size

- Optional argument to the `includegraphics` command
- Usually specify size as a fraction of `\textwidth`

```
\begin{figure}
  \centering
  \includegraphics[width=0.8\textwidth]{crackGrowthRates.pdf}
  \caption{Normalized crack growth rates for different realizations}
  \label{fig:crackGrowthRates}
\end{figure}
```

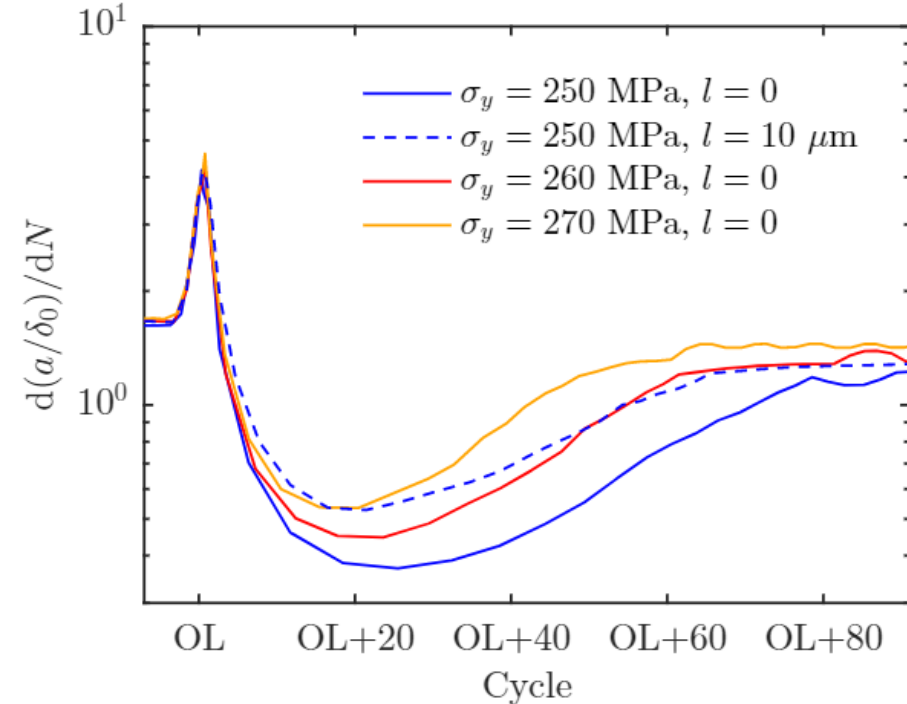


Figure 1: Normalized crack growth rates for different realizations of the model.

Subfigure environment: Multi-part figures

- Include the `subcaption` package in the preamble
- Use the `subfigure` environment within the `figure` environment
 - The width of the subfigure is a *required* argument—put it in `{ }`
 - This basically creates a box on the page with the width you specify; LaTeX puts your graphics into this box

Figure `\ref{fig:tractionProfiles}` shows `\ldots`

```
\begin{figure}
```

```
\centering
```

```
\begin{subfigure}{0.49\textwidth}
```

```
\includegraphics[width=\textwidth]{tracProf1.pdf}
```

```
\caption{Steady-state crack growth.}
```

```
\end{subfigure}
```

```
\begin{subfigure}{0.49\textwidth}
```

```
\includegraphics[width=\textwidth]{tracProf2.pdf}
```

```
\caption{Crack growth after an overload.}
```

```
\end{subfigure}
```

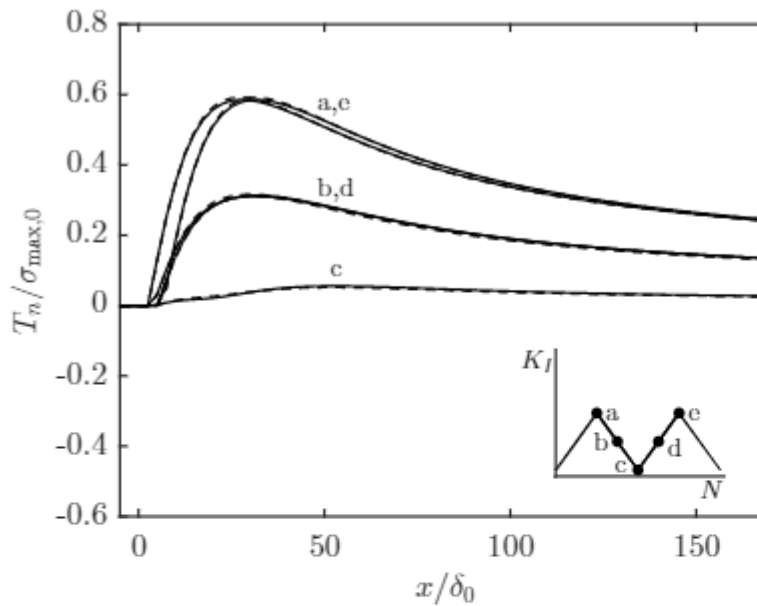
```
\caption{Normal traction ahead of the crack tip at various points in a load cycle.}
```

```
\label{fig:tractionProfiles}
```

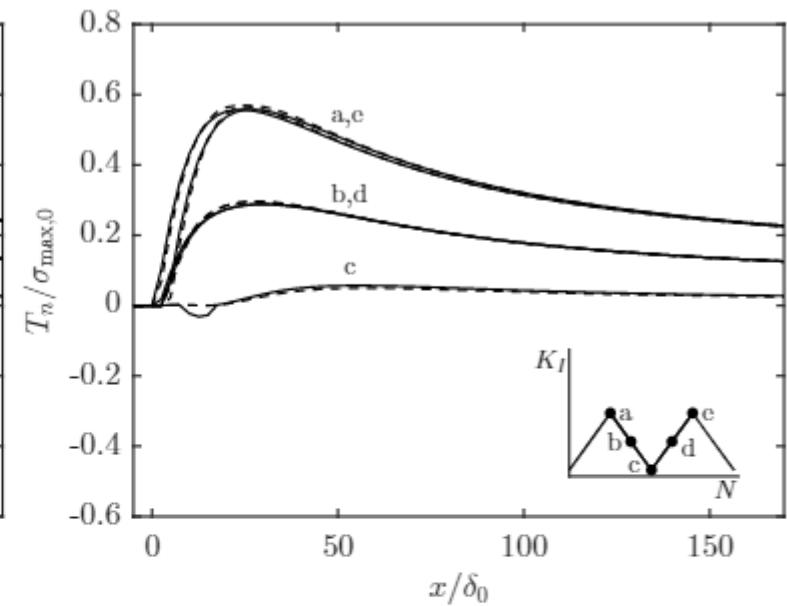
```
\end{figure}
```

Subfigure environment: Multi-part figures

```
Figure \ref{fig:tractionProfiles} shows \lc
\begin{figure}
  \centering
  \begin{subfigure}{0.49\textwidth}
    \includegraphics[width=\textwidth]{...}
    \caption{Steady-state crack growth.}
  \end{subfigure}
  \begin{subfigure}{0.49\textwidth}
    \includegraphics[width=\textwidth]{...}
    \caption{Crack growth after an overload.}
  \end{subfigure}
  \caption{Normal traction ahead of the crack tip}
  \label{fig:tractionProfiles}
\end{figure}
```



(a) Steady-state crack growth.



(b) Crack growth after an overload.

Figure 2: Normal traction ahead of the crack tip at various points in a load cycle.

Figure 2 shows ...

Bibliography

- Most citation management software can output a BibTeX file
- See resources on the next page for how to incorporate a bibliography
 - Typically use the `natbib` package

Other stuff

- Tables
 - Admittedly a bit painful to code by yourself
 - Use a resource like www.tablesgenerator.com/ or Excel2LaTeX to convert WYSIWYG table to LaTeX, and include the packages `tabularx` and `booktabs` to make publication-quality tables
- Beamer: documentclass for making presentations with LaTeX
 - Can also use MathType to write LaTeX-style equations into a PowerPoint file
- TikZ: make beautiful graphics

Other stuff

- Good resources:
 - [Overleaf learn](#) (LaTeX and Overleaf-specific tutorials)
 - [LaTeX wikibook](#) (surprisingly useful!)
 - [TeX StackExchange](#) (probably the first hit if you Google a LaTeX question)
- Overleaf has several autocomplete capabilities that can help speed things up (e.g. it adds the commands within the figure environment after you type `\begin{figure}`)
- Download a LaTeX distribution if you want to work locally on your computer
 - Windows: MikTeX or TeXLive
 - Mac: MacTeX