

L^AT_EX Document Package

Heng-Sheng Chang

December 21, 2025

Abstract

This document serves as comprehensive documentation for the `document.sty` package, a versatile L^AT_EX 2_E style file designed to streamline document creation with professional formatting and advanced features. The package integrates over 40 carefully selected L^AT_EX packages, providing a unified interface for mathematics typesetting, theorem environments, code listings, bibliography management, and document structuring.

Key features include: customizable headers and footers with formal and default styles; rich theorem-like environments (definition, theorem, lemma, corollary, proof, example, and more) with elegant colored boxes; comprehensive mathematics support through AMS packages, special symbols, and SI units; syntax-highlighted code listings for LaTeX and Python; professional table and figure formatting with captions and subcaptions; algorithm typesetting with pseudocode; flexible bibliography management via biblatex; and modular content organization with directory-based inclusion.

The package offers several configurable options including theorem environment toggle, color scheme control, code listing support, draft mode, font selection, header styles (default or formal), and bibliography styles. This documentation demonstrates all features through practical examples, including algorithm implementations, publication-quality tables, figure inclusion with TikZ graphics, and all available theorem-like environments.

Contents

1 Format	1
1.1 Package Usage and Options	1
1.2 Imported Packages	1
1.3 Metadata Commands	4
1.4 Header & Footer	4
1.5 Document Structure	5
1.6 Hyperlink & Reference	5
1.7 Citation & Bibliography	6
2 Environment	7
2.1 Lists and Quotes	7
2.2 Script	9
2.3 Mathematics	9
2.4 Algorithm	10
2.5 Table	10
2.6 Figure	10
2.7 Frame	11



1 Format

The style file `document.sty`, written in L^AT_EX 2_E, provides a comprehensive document formatting solution. The theorem-like environments are inspired by the template [1]. For detailed implementation, refer to the GitHub repository [2].

1.1 Package Usage and Options

To use this package, add `\usepackage{document}` after the `\documentclass` command. The package accepts several key-value options:

theorems=true/false (default: `true`) Enable or disable theorem-like environments (definition, theorem, lemma, corollary, proof, example, etc.). When `false`, the `thmtools` and `tcolorbox` packages are not loaded, and all theorem-like environments become unavailable.

colors=true/false (default: `true`) Enable extended color support. When `true`, loads `xcolor` with `dvipsnames` option for additional color names.

code=true/false (default: `true`) Enable syntax-highlighted code listings. When `true`, loads the `minted` package with pre-configured environments for L^AT_EX and Python code.

draft=true/false (default: `false`) Enable draft mode for faster compilation (currently reserved for future use).

timesfont=true/false (default: `true`) Use Times font family for the document text.

header=default/formal (default: `default`) Select header style. The `default` style shows section titles at the center with page numbers in the footer. The `formal` style displays purpose, title, author, institute, and page count.

bibstyle=style (default: `ieee`) Set bibliography style for `biblatex`. Common options include `ieee`, `apa`, `nature`, `numeric`, or `alphabetic`.

Example usage:

```
1 \usepackage[  
2   theorems=true,  
3   colors=true,  
4   code=true,  
5   header=formal,  
6   bibstyle=ieee  
7 ] {document}
```

1.2 Imported Packages

The `document.sty` package imports and configures numerous packages organized by functionality:

Parsing and Options

- `xparse` — Advanced command definition with flexible argument parsing
- `ifthen` — Conditional statements and logic
- `etoolbox` — Programming facilities and command patching
- `kvoptions` — Key-value option handling for package options

Page Layout

- `geometry` — Page dimensions and margins (0.5in sides, 1.5in bottom)
- `emptypage` — Clean formatting for empty pages
- `fancyhdr` — Customizable headers and footers
- `lastpage` — Reference to the last page number (for formal header)

Text Encoding and Fonts

- `inputenc` (`utf8`) — Unicode character input support
- `fontenc` (`T1`) — Font encoding for proper character rendering
- `times` — Times Roman font family (when `timesfont=true`)
- `textcomp` — Additional text symbols and companion fonts
- `fontawesome` — Icon font support for symbols like ☕

Mathematics

- `amsmath` — Enhanced mathematics environments and commands
- `amsfonts` — Mathematical fonts including blackboard bold
- `amsthm` — Theorem environment support
- `amssymb` — Extended mathematical symbols
- `mathtools` — Enhancements and fixes for `amsmath`
- `mathrsfs` — Ralph Smith's Formal Script font for math
- `mathdots` — Commands for improving the appearance of dots in math
- `nicefrac` — Typeset nice diagonal fractions
- `cancel` — Cross out mathematical expressions
- `MnSymbol` — Additional mathematical symbols
- `dsfont` — Doublestroke font for mathematical alphabets
- `upgreek` — Upright Greek letters in mathematics
- `systeme` — Systems of linear equations formatting
- `siunitx` — Consistent SI units and number formatting
- `pifont` — Access to Pi fonts for special symbols

Colors and Hyperlinks

- `xcolor` — Color support with optional `dvipsnames` palette
- `hyperref` — Hyperlinks and PDF metadata with color configuration

Text Formatting

- `soul` — Letter spacing and highlighting (`\spaceout` command)
- `fancyvrb` — Enhanced verbatim text for inline code
- `csquotes` — Context-sensitive quotation marks
- `enumitem` — Enhanced control over list environments

Code Listings

- `minted` — Syntax highlighting using Pygments (when `code=true`)

Tables

- `booktabs` — Publication-quality table rules
- `arydshln` — Dashed lines in tables
- `multirow` — Cells spanning multiple rows
- `multicol` — Multiple column environments

Floats and Captions

- `float` — Improved float control with H placement specifier
- `caption` — Customizable caption formatting
- `subcaption` — Support for subfigures and subtables

Algorithms

- `algorithm` — Floating environment for algorithms
- `algpseudocode` — Pseudocode formatting commands

Graphics and Plotting

- `graphicx` — Enhanced graphics inclusion
- `tikz` — Powerful graphics and diagram creation
- `pgfplots` — High-quality function and data plots
- `multimedia` — Multimedia content embedding

Additional Utilities

- `pbox` — Fixed-width paragraph boxes
- `nameref` — Reference by section names

Document Structure

- `appendix` — Appendix formatting and management

Bibliography

- `biblatex` — Advanced bibliography management with Biber backend

Theorem Environments

- `thmtools` — Enhanced theorem configuration (when `theorems=true`)
- `tcolorbox` — Colored and framed boxes for theorems (when `theorems=true`)

1.3 Metadata Commands

The package provides enhanced metadata commands with optional short forms:

`\author[short]{full}` Define document author. The optional `short` form is used in headers.

`\title[short]{full}` Define document title. The optional `short` form is used in headers.

`\purpose{text}` Specify the document's purpose (displayed in formal header).

`\institute{text}` Specify the institution or affiliation (displayed in formal header).

Example:

```
1 \title[Short Title]{Full Document Title}
2 \author[J. Doe]{John Doe}
3 \purpose{Technical Report}
4 \institute{Department of Computer Science}
```

1.4 Header & Footer

The package uses the `fancyhdr` package to create professional headers and footers. Two header styles are available, selectable via the `header` option:

Default Header Style The default style is simple and clean:

- Header: Section title centered
- Footer: Page number centered
- Header height: 0.75 inches

Formal Header Style The formal style provides comprehensive document information:

- Left header (top line): Purpose and short title with author
- Left header (bottom line): Institute and page count (“page X of Y”)
- No header rule
- Header height: 0.75 inches

Page margins are configured as 0.5 inches on left and right sides, with 1.5 inches at the bottom. Empty pages (in two-sided printing) are automatically cleaned using the `emptypage` package.

Footnotes should be used sparingly and are created using the `\footnote{text}` command. The footnote text is automatically numbered, formatted and placed at the bottom of the page.¹

1.5 Document Structure

The package provides commands for modular document organization:

`\setdirectory[path]` Set the base directory for content files (default: current directory).
`\setappendix[path]` Switch to appendix mode and set the directory for appendix files.
`\addcontent{filename}` Include content from the specified file in the current directory.

Example usage:

```
1 % Include main content
2 \setdirectory[content]
3 \addcontent{1_introduction}
4 \addcontent{2_methodology}
5
6 % Include appendices
7 \setappendix[appendix]
8 \addcontent{1_proofs}
9 \addcontent{2_data}
```

The default document class is `article` with 11pt font size and two-sided printing. Sections are numbered automatically with hierarchical structure: section, subsection, and subsubsection.

Section Create a new section with `\section{Section title}`. Within sections, use `\subsection{Subsection title}` for subsections and `\subsubsection{Subsubsection title}` for deeper nesting.

Paragraph Create a titled paragraph with `\paragraph{Paragraph title}`. The paragraph title appears in bold-face, with text following on the same line. To start a new paragraph without a title, simply leave a blank line.

1.6 Hyperlink & Reference

The `hyperref` package creates clickable hyperlinks in the PDF output. All links use the default text color (no colored boxes) for print-friendly documents.

¹The footnote text is automatically numbered, formatted and placed at the bottom of the page like this.

External Links Create hyperlinks with `\href{url}{text}`, where `url` is the web address and `text` is the displayed link text.

Internal References The package provides convenient reference commands for different element types:

Standard references:

- `\ref{label}` — Basic reference to labeled item
- `\nameref{label}` — Reference by section name
- `\pageref{label}` — Reference to page number

Specialized reference commands:

- `\refalg{name}` — Reference to Algorithm `\ref{alg:name}`
- `\reffig{name}` — Reference to Figure `\ref{fig:name}`
- `\reftab{name}` — Reference to Table `\ref{tab:name}`

Corresponding label commands:

- `\labelalg{name}` — Equivalent to `\label{alg:name}`
- `\labelfig{name}` — Equivalent to `\label{fig:name}`
- `\labeltab{name}` — Equivalent to `\label{tab:name}`

When the `theorems` option is enabled, additional reference commands are available for theorem-like environments (see Section 2.7).

1.7 Citation & Bibliography

The `biblatex` package manages citations and bibliographies with the Biber backend. Biber provides superior Unicode support, advanced sorting, and better handling of complex bibliography requirements compared to BibTeX.

Setup Specify the bibliography file in the preamble:

```
1 \addbibresource{references.bib}
```

Citations Create citations using:

- `\autocite{key}` — Context-sensitive citation (recommended)
- `\cite{key}` — Standard citation
- `\textcite{key}` — Textual citation (e.g., “Author (2023) shows...”)
- `\parencite{key}` — Parenthetical citation

Citations appear with square brackets (e.g., [1]) regardless of the bibliography style.

Bibliography Print the bibliography at the document end:

```
1 \printbibliography
```

The bibliography style is set via the `bibstyle` package option (default: `ieee`). Common styles include numeric, alphabetic, `ieee`, `apa`, and `nature`.

2 Environment

An *environment* in L^AT_EX is a section of the document formatted in a specific way. This document demonstrates various environments, including lists, quotes, code blocks, scripts, tables, figures, algorithms, and framed theorem-like environments. The basic syntax of an environment is:

```
1 \begin{environment}  
2   content included in the environment  
3 \end{environment}
```

2.1 Lists and Quotes

Common text environments include `enumerate` (numbered lists), `itemize` (bulleted lists), and `displayquote` (block quotations). These environments can be customized using the `enumitem` package for lists and `csquotes` package for quotations.

1. First item in an enumerated list.

```
1 \begin{enumerate}  
2   \item First item in an enumerated list.  
3   \item Second item in the same list.  
4   \item Third item and more ...  
5 \end{enumerate}
```

2. Second item in the same list.

3. Third item and more ...

• First item in an unordered list.

```
1 \begin{itemize}  
2   \item First item in an unordered list.  
3   \item Second item in the same list.  
4   \item Third item and more ...  
5 \end{itemize}
```

• Second item in the same list.

• Third item and more ...

This is a quote, which uses the `csquotes` package with customization set inside the `document.sty` file.

```
1 \begin{displayquote}  
2   This is a quote, which uses the \verb|csquotes|  
   → package with customization set inside the  
   → \verb|document.sty| file.  
3 \end{displayquote}
```

2.2 Script

Inline Code Use the `fancyvrb` package's `\Verb|print ("Hello, World!")|` command for inline code. For example, `\Verb|print ("Hello, World!")|` produces `print ("Hello, World!")`. The delimiter (typically `|`) can be any character not appearing in the code.

Code Blocks The `minted` package provides syntax-highlighted code blocks for various programming languages. When the `code=true` option is enabled, the package pre-configures `latexcode` and `pythoncode` environments.

Python example:

```

1 def fibonacci(n):
2     """Calculate nth Fibonacci number."""
3     if n <= 1:
4         return n
5     return fibonacci(n-1) + fibonacci(n-2)
6
7 # Compute first 10 Fibonacci numbers
8 for i in range(10):
9     print(f"F({i}) = {fibonacci(i)}")

```

```

1 \begin{pythoncode}
2 def fibonacci(n):
3     """Calculate nth Fibonacci number."""
4     if n <= 1:
5         return n
6     return fibonacci(n-1) + fibonacci(n-2)
7
8 # Compute first 10 Fibonacci numbers
9 for i in range(10):
10    print(f"F({i}) = {fibonacci(i)}")
11 \end{pythoncode}

```

The `minted` package supports many languages including Python, C, C++, Java, JavaScript, and more. Custom language environments can be added using the `\newminted` command (see lines 211-235 in `document.sty` for examples of `latexcode` and `pythoncode` configurations).

2.3 Mathematics

The package includes comprehensive mathematics support through AMS packages and specialized math fonts.

Inline and Display Math Inline math: $E = mc^2$ is typeset with `$E = mc^2$`. Display math uses `\[... \]` or `equation` environment:

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

Special Symbols and Fonts The package provides extensive symbol support:

- Blackboard bold: $\mathbb{R}, \mathbb{C}, \mathbb{N}, \mathbb{Z}, \mathbb{Q}$
- Doublestroke: $\mathbb{1}, \mathbb{R}$
- Calligraphic: $\mathcal{L}, \mathcal{F}, \mathcal{H}$
- Script: $\mathcal{L}, \mathcal{F}, \mathcal{H}$
- Fraktur: $\mathfrak{g}, \mathfrak{h}, \mathfrak{su}$
- Upright Greek: π, μ, σ

Fractions and Units Nice diagonal fractions: $1/2, a/b$ using `\nicefrac{1}{2}`. SI units with `siunitx`: $299\,792\,458\,\text{m}\,\text{s}^{-1}$ (speed of light), $1.602 \times 10^{-19}\,\text{C}$ (elementary charge), $6.626 \times 10^{-34}\,\text{J}\,\text{s}$ (Planck constant).

Algorithm 1 Quicksort algorithm

Require: Array $A[1..n]$ of comparable elements
Ensure: Array A sorted in non-decreasing order

```

1: function QUICKSORT( $A$ , left, right)
2:   if left < right then
3:     pivotIndex  $\leftarrow$  PARTITION( $A$ , left, right)
4:     QUICKSORT( $A$ , left, pivotIndex – 1)                                ▷ Sort left subarray
5:     QUICKSORT( $A$ , pivotIndex + 1, right)                               ▷ Sort right subarray
6:   end if
7: end function
8:
9: function PARTITION( $A$ , left, right)
10:   pivot  $\leftarrow A[\text{right}]$                                          ▷ Choose last element as pivot
11:    $i \leftarrow \text{left} - 1$ 
12:   for  $j \leftarrow \text{left}$  to right – 1 do
13:     if  $A[j] \leq \text{pivot}$  then
14:        $i \leftarrow i + 1$ 
15:       swap  $A[i]$  and  $A[j]$                                          ▷ Move smaller elements to left
16:     end if
17:   end for
18:   swap  $A[i + 1]$  and  $A[\text{right}]$                                      ▷ Place pivot in correct position
19:   return  $i + 1$                                                  ▷ Return pivot index
20: end function
```

Systems of Equations The `systeme` package formats systems elegantly:

$$\begin{cases} x + y + z = 6 \\ 2x - y + z = 3 \\ x + 2y - z = 2 \end{cases}$$

2.4 Algorithm

Algorithms should be presented clearly and concisely, typically placed at the top of a page using the `[t]` placement specifier. The `algorithm` package provides the floating environment, while `algpseudocode` provides pseudocode formatting commands. See Algorithm 1 for an example.

2.5 Table

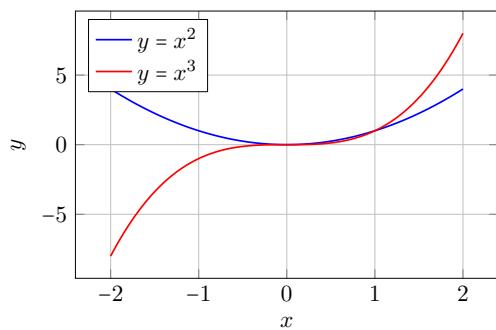
Tables should be centered, legible, and positioned at the top of the page. Table captions appear *before* the table content and use sentence case (capitalize only the first word and proper nouns). Publication-quality tables avoid vertical rules [3]. The `booktabs` package provides professional horizontal rules: `\toprule`, `\midrule`, `\cmidrule`, and `\bottomrule`. Additional packages support dashed lines (`arydshln`), multi-row cells (`multirow`), and multi-column layouts (`multicol`). See Table 1 for an example.

2.6 Figure

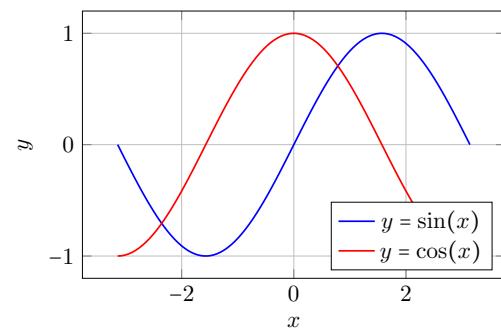
Figures must be centered, legible, and positioned at the top of the page. Figure captions appear *after* the figure content, using sentence case and a smaller font than the main text. The `caption` and `subcaption` packages enable sophisticated

Table 1. Performance comparison of sorting algorithms

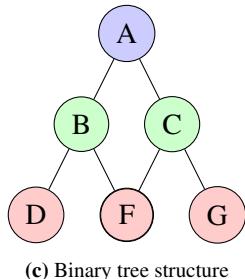
Algorithm	Best Case	Average Case	Worst Case
Bubble Sort	$O(n)$	$O(n^2)$	$O(n^2)$
Insertion Sort	$O(n)$	$O(n^2)$	$O(n^2)$
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$
Merge Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$
Quick Sort	$O(n \log n)$	$O(n \log n)$	$O(n^2)$
Heap Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$



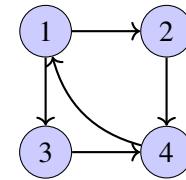
(a) Polynomial functions



(b) Trigonometric functions



(c) Binary tree structure



(d) Directed graph

Figure 1. Examples of figures created with TikZ and PGFPlots. The package supports mathematical plots, tree structures, and graph visualizations.

caption formatting including subfigures. The package supports multiple graphics formats via `graphicx` and provides TikZ/PGFPlots for creating publication-quality diagrams and plots. Colored figures are acceptable when they remain legible in both color and grayscale printing. See Figure 1 for examples.

2.7 Frame

When the `theorems=true` option is enabled, the package provides elegant theorem-like environments using `tcolorbox` with color-coded frames. All environments share the same counter (numbered within subsections) and support three arguments: title, label, and content.

Definition 2.7.1 (Continuity) A function $f : \mathbb{R} \rightarrow \mathbb{R}$ is *continuous* at point x_0 if for every $\epsilon > 0$, there exists $\delta > 0$ such that $|x - x_0| < \delta$ implies $|f(x) - f(x_0)| < \epsilon$.

Assumption 2.7.2 (Bounded Domain) We assume the domain $\Omega \subset \mathbb{R}^n$ is bounded, connected, and has a piecewise smooth boundary $\partial\Omega$.

Conjecture 2.7.3 (Collatz Conjecture) For any positive integer n , the iterative sequence defined by $a_{i+1} = a_i/2$ if a_i is even, or $a_{i+1} = 3a_i + 1$ if a_i is odd, eventually reaches 1.

Lemma 2.7.4 (Triangle Inequality) Based on Definition 2.7.1, for any $x, y \in \mathbb{R}^n$, we have $\|x + y\| \leq \|x\| + \|y\|$, where $\|\cdot\|$ denotes the Euclidean norm.

Proof

By the Cauchy-Schwarz inequality, $\langle x, y \rangle \leq \|x\|\|y\|$. Then:

$$\|x + y\|^2 = \|x\|^2 + 2\langle x, y \rangle + \|y\|^2 \leq \|x\|^2 + 2\|x\|\|y\| + \|y\|^2 = (\|x\| + \|y\|)^2$$

Taking square roots of both sides yields the result. ■

Theorem 2.7.5 (Fundamental Theorem of Calculus) If f is continuous on $[a, b]$ and F is an antiderivative of f on $[a, b]$, then $\int_a^b f(x) dx = F(b) - F(a)$.

Proof sketch

Define $G(x) = \int_a^x f(t) dt$. By continuity of f , we have $G'(x) = f(x)$. Since both F and G are antiderivatives of f , they differ by a constant: $F(x) = G(x) + C$. Evaluating at $x = a$ gives $C = F(a)$, and at $x = b$ gives the result. ■

Corollary 2.7.6 (Integration by Parts) If u and v are continuously differentiable functions, then $\int u dv = uv - \int v du$.

Proof

This follows directly from Theorem 2.7.5 applied to the product rule: $(uv)' = u'v + uv'$. ■

Proposition 2.7.7 (Derivative of Exponential) For the exponential function e^x , we have $\frac{d}{dx} e^x = e^x$.

Proof

Using the limit definition:

$$\frac{d}{dx} e^x = \lim_{h \rightarrow 0} \frac{e^{x+h} - e^x}{h} = e^x \lim_{h \rightarrow 0} \frac{e^h - 1}{h} = e^x$$

Remark 2.7.8 (Uniqueness of Exponential) The exponential function is the only function (up to scaling) that is its own derivative, making it fundamental in differential equations.

Note 2.7.9 (Notation Convention) Throughout this document, we use \mathbb{N} for natural numbers, \mathbb{Z} for integers, \mathbb{Q} for rationals, \mathbb{R} for reals, and \mathbb{C} for complex numbers.

Example 2.7.10 (Computing a Derivative) Find the derivative of $f(x) = x^2 \sin(x)$ using the product rule.

Solution

By the product rule: $f'(x) = (x^2)' \sin(x) + x^2(\sin(x))' = 2x \sin(x) + x^2 \cos(x)$. □

Problem 2.7.11 (Optimization Problem) A rectangular box with square base must have volume 1000 cm³. Find the dimensions that minimize the surface area.

ToDo 2.7.12 (Future Enhancement) Extend the package to support additional theorem styles and custom color schemes for different document themes.

These environments are designed to enhance the readability and organization of mathematical documents. Each environment has a specific purpose and style, making it easier for readers to follow the logical flow of the content.

Reference Commands The package provides convenient reference commands for theorem-like environments:

- `\refdef{label}` — Reference to Definition (e.g., `\ref{def:label}`)
- `\refasm{label}` — Reference to Assumption (e.g., `\ref{asm:label}`)
- `\refconj{label}` — Reference to Conjecture (e.g., `\ref{conj:label}`)
- `\reflem{label}` — Reference to Lemma (e.g., `\ref{lem:label}`)
- `\refthm{label}` — Reference to Theorem (e.g., `\ref{thm:label}`)
- `\refcor{label}` — Reference to Corollary (e.g., `\ref{cor:label}`)
- `\refprop{label}` — Reference to Proposition (e.g., `\ref{prop:label}`)
- `\refremk{label}` — Reference to Remark (e.g., `\ref{remk:label}`)
- `\refnote{label}` — Reference to Note (e.g., `\ref{note:label}`)
- `\refex{label}` — Reference to Example (e.g., `\ref{ex:label}`)
- `\refprob{label}` — Reference to Problem (e.g., `\ref{prob:label}`)
- `\reftodo{label}` — Reference to ToDo (e.g., `\ref{todo:label}`)

These commands automatically include the environment type (e.g., “Definition”) and number, creating clear cross-references throughout the document.

Appendices

References

- [1] sleepymalc, *Latex template*, <https://github.com/sleepymalc/LaTeX-Template/releases/tag/v1.0.2022>.

- [2] H.-S. Chang, *L^AT_EX document template*, <https://github.com/hanson-hschang/LaTeX-Document-Template>, 2025.
- [3] NeurIPS. “Neurips latex template.” [2023], [Online]. Available: <https://neurips.cc/Conferences/2023/PaperInformation/StyleFiles> [visited on 07/24/2025].