

2025 年度

福井大学 工学研究科

学士論文

## 修論・卒論本旨の L<sup>A</sup>T<sub>E</sub>X テンプレート

---

LaTeX template for master's thesis and graduation thesis  
abstracts

2026年2月7日

福井大学 工学部  
機械・システム工学科 機械工学コース

氏名 書く

学籍番号 書く

指導教員 梅本和希 准教授

# 目次

<b>第 1 章 <math>\text{\LaTeX}</math> の概要</b>	1
1.1 Overleaf と CloudLatex . . . . .	1
1.2 ローカル $\text{\LaTeX}$ 環境の構築 . . . . .	1
1.3 章と節、その他 . . . . .	2
<b>第 2 章 表と図</b>	3
2.1 図表の配置オプション . . . . .	3
<b>第 3 章 数式</b>	5
3.1 数式環境 . . . . .	5
3.2 数式コマンド . . . . .	5
3.3 よく使う数式コマンド一覧 . . . . .	5
<b>第 4 章 参考文献</b>	6
<b>第 5 章 差分管理</b>	7
<b>第 6 章 報告書の章立て</b>	8
<b>参考文献</b>	8
<b>謝辞</b>	10

# 第1章 LATEX の概要

Microsoft Word みたいなものです。テキストファイルでいろいろ書きます。習うより慣れよ。

- 図や式の番号を自動管理してくれる (参照)
- 参考文献リストが扱いやすい
- 長い文章でも重くならない (Word よりは)
- ソフトが落ちない (ただの平文テキストファイルなので)
- きれい
- 図が画面外に消えていなくならない

などのメリットがありますが覚えることが多いのが難点。ある程度網羅的なサンプルがあればコピペで作れるのでこのテンプレを全員で育てましょう。

## 1.1 Overleaf と CloudLatex

どちらもオンライン Latex 環境ですが、Overleaf は無料プランではコンパイル時間に制限があります。使い勝手は Overleaf のほうが良い印象ですが、Cloud Latex はコンパイル時間に制限がない、VS Code と連携できたりとメリットがあります。報告書であれば Overleaf が勝手が良いとは思いますが、どちらか一つしか使わないのであれば Cloud Latex 推奨です。Overleaf で使う場合には latexmkrc ファイルと、左上メニューからコンパイラ: Latex の設定が必要です。どちらにしても新規プロジェクトから zip ファイルのアップロードができます。

## 1.2 ローカル LATEX 環境の構築

卒論などの長文では、Git でバージョン管理したほうがよく、また latexdiff-vc で差分ファイルを作成できると便利です。Cloud Latex はローカルファイルと同期させることもできるので実現可能ですが、latexdiff-vc は使えないでのローカル環境を整えたほうが良い場合も多いです。最も手軽な VS Code-Docker 環境のセットアップを紹介します。Windows 環境でインストールすべきは

- VS Code
- Docker Desktop

です。また、VS Code の Extensions として、

- Remote Development (4つの extensions のセット、必須なのは Dev Containers)
- LaTeX Workshop
- Remote Explorer (必須ではない)

をインストールしましょう。これで準備完了です。使い方は

1. VS Code を起動し、フォルダを D&D してワークスペースを開く
2. 左下の “><” から、Reopen in container を押して Docker コンテナのシェルに入る
3. main.tex を開いて TEX のところから Build LaTeX project - Recipe: latexmk かまたは、Terminal から latexmk main.tex と打つ。どのファイルを選択していてもよいので、コマンド打つ方が早い。

で、PDF 出力までされると思います。SyncTeX に対応しており (LaTeX Workshop の設定が必要かも)、PDF を ctrl-クリックで該当のソースへ移動、ソースからは Tex タブの View LaTeX PDF - Refresh... で該当の PDF の場所に移動ができます。ちなみに、.devcontainer フォルダには docker の設定ファイル、.vscode フォルダには LaTeX Workshop

の設定ファイル(ワークスペース限定)が入っています。gitignoreはgitの無視するファイルを設定。latexmkrcはlatexmkコマンドの設定。

Gitを使うなら、

- Git
- GitHub Desktop

くらい入れれば良い。Clone, Commit, Fetch (Pull), Push, Checkout くらいを使えれば便利さを感じることができると思います。

## 1.3 章と節、その他

章は \section{ 章タイトル }、節は \subsection{ 節タイトル } と書く。subsubsection もある。基本的にテキストで書いた順番に並べられるが、図表は自動で配置を決定する。空白行を入れると段落が変わる。数式や図を入れるときに空白行を無駄に入れてしまうのに注意。逆に空白行さえ入れなければ段落は変わらず、改行もなく続けて表示される。バックスラッシュ 2 個で改行できるが、基本的に使わないほうがよい。% でコメントアウトできる。

## 第2章 表と図

表は表 1 のように、図は図 1 のように書く。並べたい場合は `\begin{minipage}` を使う (図 2, 3)。バックスラッシュ 2 つで改行を入れて縦に並べることもできる。

### 2.1 図表の配置オプション

図表の配置は `\begin{figure}[配置オプション]` や `\begin{table}[配置オプション]` で指定する。主な配置オプションは以下の通り:

- `h: here` (可能ならその場所に配置)
- `t: top` (ページの上部に配置)
- `b: bottom` (ページの下部に配置)

表 1: Captions should be in English.

---

1	2
abc	xyz

---



図 1: Captions should be in English.

- p: page (図表だけを集めた別ページに配置)
- H: Here (強制的にその場所に配置、float パッケージが必要)
- !: 制約を緩和 (例: !t で強制的に上部)

複数組み合わせることも可能で、[tb] とすればページ上部または下部に配置される。基本的には **t**、**b**、**tb** を推奨。配置にこだわってうまくいかないことが多いので気にしないということも重要。H オプションで tex ファイルで書いた位置に画像を入れることができるが、ページ内に収まらない場合はレイアウトが崩れるため注意。

キャプションをまとめたいときは `subcaption` (図 4, 4(a))。2 段組み中の 1 段組みは図 5 のように `figure*` 環境を使う。

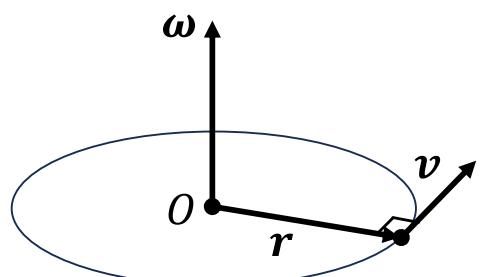


図 2: Caption

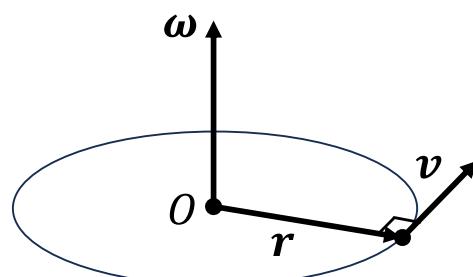
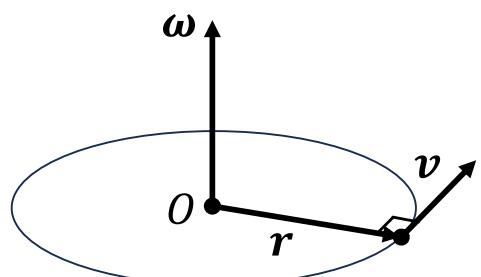
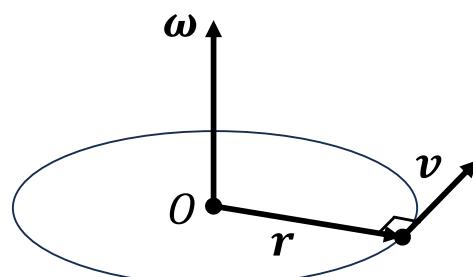


図 3: Caption



(a) Caption



(b) Caption

図 4: Captions should be in English.

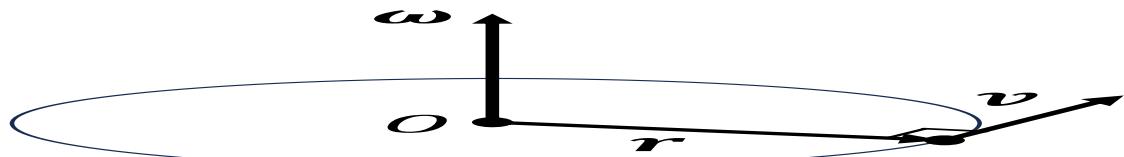


図 5: Captions should be in English. The figure spans both columns in a two-column layout.

# 第3章 数式

## 3.1 数式環境

equation 環境:

$$y = f(x) \quad (1)$$

align 環境:

$$y_1 = f_1(x_1, x_2) \quad (2)$$

$$y_2 = f_2(x_1, x_2) \quad (3)$$

aligned 環境:

$$\begin{aligned} y_1 &= f_1(x_1, x_2) \\ y_2 &= f_2(x_1, x_2) \end{aligned} \quad (4)$$

bmatrix 環境:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (5)$$

長い数式は figure\*環境の中に equation 環境等を入れる。その際キャプションはつけない。他はだれかが追記。[https://qiita.com/t\\_kemmochi/items/a4c390b4967b13f3afb7](https://qiita.com/t_kemmochi/items/a4c390b4967b13f3afb7) にいろいろ載っている。また、[https://www.ieice.org/iss/prmu/jpn/latex\\_templates/tecrep.pdf](https://www.ieice.org/iss/prmu/jpn/latex_templates/tecrep.pdf) には数式を枠に収めるテクニックがいろいろ記載。

## 3.2 数式コマンド

ギリシャ文字  $\alpha, \beta$ 、等号不等号  $\leq, \approx$ 、飾り  $\bar{x}, \tilde{x}, \dot{x}, \ddot{x}$  等かなり数が多い。<https://cns-guide.sfc.keio.ac.jp/2002/13/4/5.html> にいろいろ載っている。

## 3.3 よく使う数式コマンド一覧

表2: ギリシャ文字

コマンド	表示	コマンド	表示
\alpha	$\alpha$	\beta	$\beta$
\gamma	$\gamma$	\delta	$\delta$
\epsilon	$\epsilon$	\theta	$\theta$
\lambda	$\lambda$	\mu	$\mu$
\sigma	$\sigma$	\omega	$\omega$

## 第4章 参考文献

末尾に追加される。bib ファイルで管理がしやすい。参照するときは [1] とする。bib ファイルの作成には zotero や JabRef、mac OS なら BibDesk などを使うとよい。基本的に bib ファイルを論文検索サイトからダウンロードしてきて 1 つのファイルにまとめていく。

## 第5章 差分管理

`latexdiff-vc` を使うと、前回のコミットとの差分を青や赤でマーキングできます。使い方:

1. Git リポジトリを初期化: `git init`
2. 初回コミット: `git add . && git commit -m "Initial commit"`
3. 修正後に再度コミット: `git add . && git commit -m "Update"`
4. 差分 PDF 生成: `bash diff.sh`

生成された `main-diffHEAD-1.pdf` で変更箇所が確認できます。

# 第6章 報告書の章立て

研究報告は以下の構成で書くこと。

## 1. 概要

報告の内容を簡潔にまとめる。

## 2. 研究計画

しっかり研究計画を練ること。

## 3. 導入

以下の項目について書く。

- 動機付け: 研究の背景や意義は?
- 関連研究: 既存研究の紹介と解決すべき課題は?
- 目的: なにに取り組むのか?

## 4. 本題

適宜、章を分ける。

## 5. 結論

## 参考文献

- [1] 藤原邦男, 物理学序論としての力学(基礎物理学1), 東京大学出版会, 1984.

## 謝辞

本研究の遂行にあたり、ご指導いただいた梅本和希准教授に深く感謝いたします。また、実験協力や有益な助言をいただいた関係者の皆様に心より御礼申し上げます。