

Inhaltsverzeichnis

1	Vorwort	3
2	Latex Installation	3
2.1	Latex Compiler	3
2.2	Editor	3
3	Bessere Rechtschreibprüfung	4
4	Benutzung dieser Vorlage	7
4.1	Vorbereitung	7
4.2	Einzelarbeit	7
4.3	Gruppenarbeit	7
5	Kapitel, Text, Zeilen- und Seitenumbrüche	9
5.1	Kapitel, Unterkapitel, Laborkapitel	9
5.2	Text, Zeilen- und Seitenumbrüche	10
5.3	Referenzieren und Zitieren	10
6	Inkscape Latex	11
7	Bilder	12
7.1	Bild einfügen	12
7.2	Platzhalter für nachträgliche Bilder	12
8	Einfache Tabellen	14
9	Mathe- Formelumgebung	15
9.1	Formeln	15
9.1.1	Einzelne Formeln	15
9.1.2	Mehrere Formeln gleicher Nummerierung	15
9.1.3	Zeilenumbruch innerhalb einer Formel	16
9.2	Vektoren und Matrizen	16
9.2.1	Vektoren	16
9.2.2	Matrizen	16
9.3	Sonderzeichen	16
10	SI	17
11	Komplexere Tabellen	18
12	Programmiercode	20

13 PDF-Seiten einbinden	21
14 PGFPlots	22
15 Fehlermeldungen	23

1 Vorwort

Diese Vorlage soll vor allem einen Header bieten, der für die meisten Laborberichte und Arbeiten sinnvoll ist und hoffentlich einige nützliche Dinge in LaTeX zeigt.

Bisher ist sie von Steffen Brahtz und Thomas Lübbehüsen gepflegt erstellt worden und wird momentan von Robert Pufall und Jan Breuer gepflegt.

Die Nutzung und Veränderung ist jedem gestattet. Bei Verbesserungsvorschlägen und Anregungen wenden Sie sich bitte an Robert Pufall (R.Pufall@ostfalia.de) oder Jan Breuer (J.Breuer@ostfalia.de). Bitte dies AUSSCHLIEßLICH für eben erwähntes nutzen. Fragen bezüglich der Nutzung von Latex werden ignoriert. Dafür dient diese Bedienungsanleitung!

2 Latex Installation

2.1 Latex Compiler

Als Latex Compiler wurde für diese Vorlage TexLive genutzt. MikTex ist auch möglich, allerdings fehlen diesem Compiler viele Pakete, welche nachträglich installiert werden müssen. Auch wenn die Installation bedeutend länger dauert empfehlen wir ausdrücklich TexLive!

2.2 Editor

Als Editor können viele IDEs genutzt werden. Wir persönlich nutzen TeXstudio. Visual Studio Code ist bspw. auch möglich. Google zeigt einem diverse Editor. Möglicherweise bezieht sich allerdings diese Anleitung auf die Nutzung von TeXstudio.

3 Bessere Rechtschreibprüfung

a) Herunterladen der Datei <https://mega.nz/file/IwoyTYIZ#oqr5qIaVe3RZAFHnLinVUpE2sULYTURT>

1 Java Installieren

2 Language tool zip entpacken und dateien in einem geeigneten ord verschieben (z.B. Programme)

3 languagetool.jar mit doppelklick starten

4-7 Textprüfung → Optionen → Allgemein → Haken bei „Als Server laufen auf Port“ 8081

8 OK drücken und Language-tools schließen

9 TexStudio Starten

10-12 Optionen → TeXstudio konfigurieren... → Sprache Prüfen

13 als server URL <http://localhost:8081/> einfügen

14 prüfen ob der haken aktiviert ist

15 pfad zur java.exe reinschreiben oder suchen mit 15.)a

16 pfad zu Language-tools +languagetool-server.jar eingeben

17 textsudio neutsarten

18 max 2min warten bis das zeichen unten links „leuchtet“

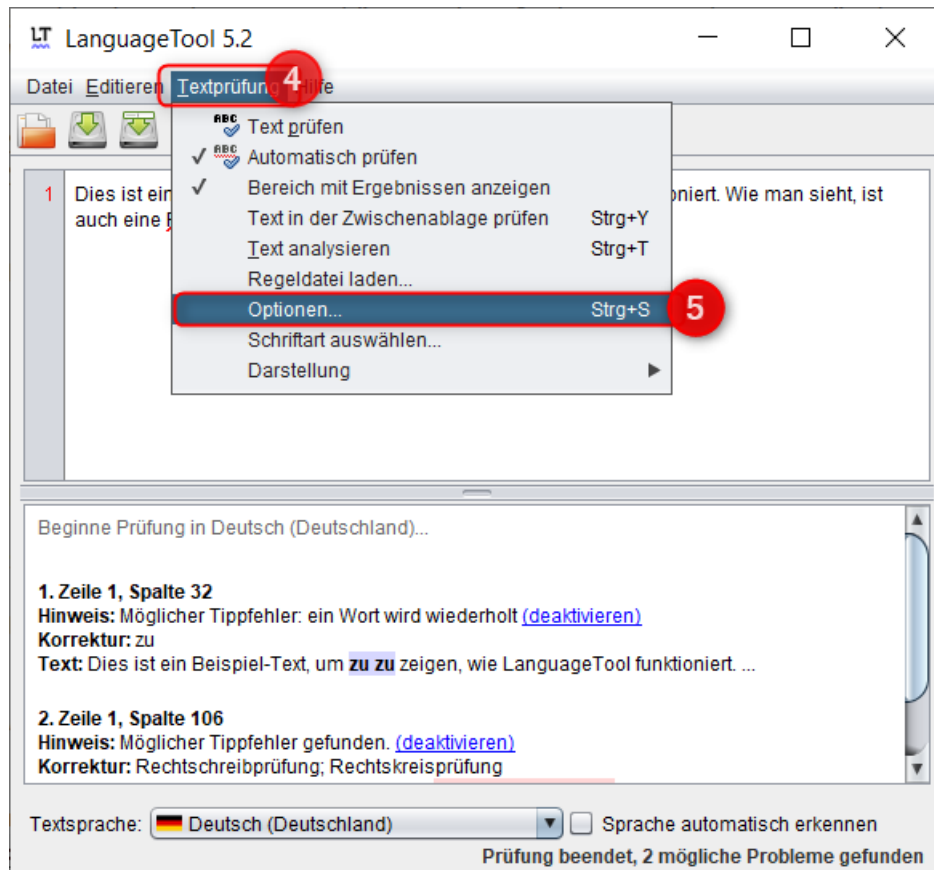


Abbildung 3.1

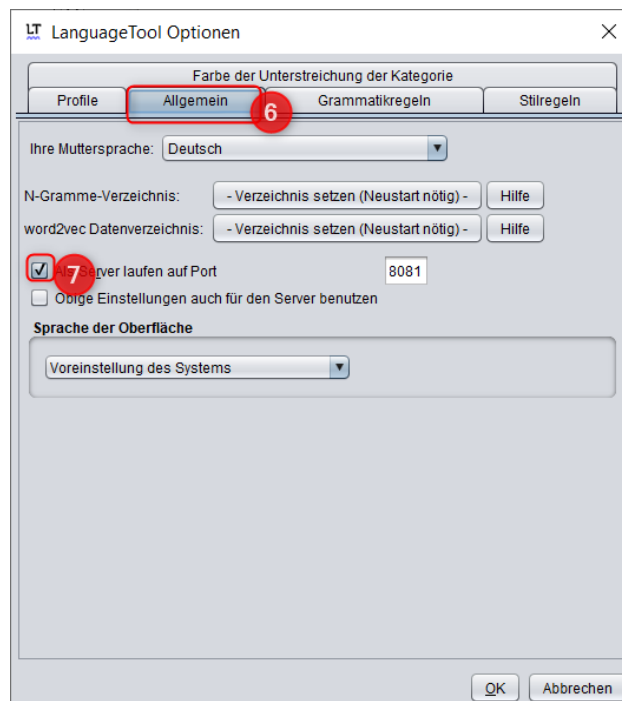


Abbildung 3.2

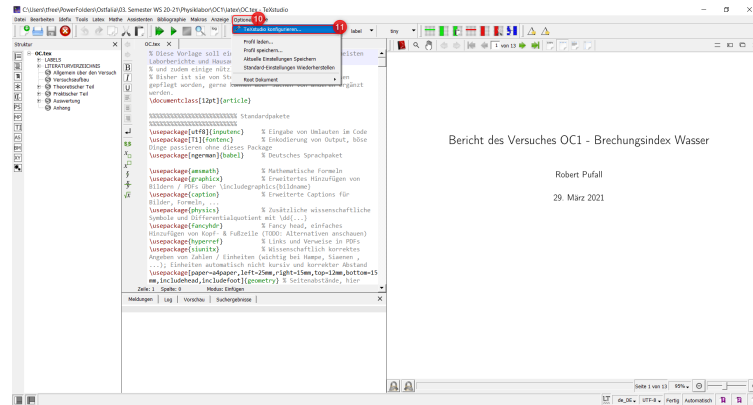


Abbildung 3.3

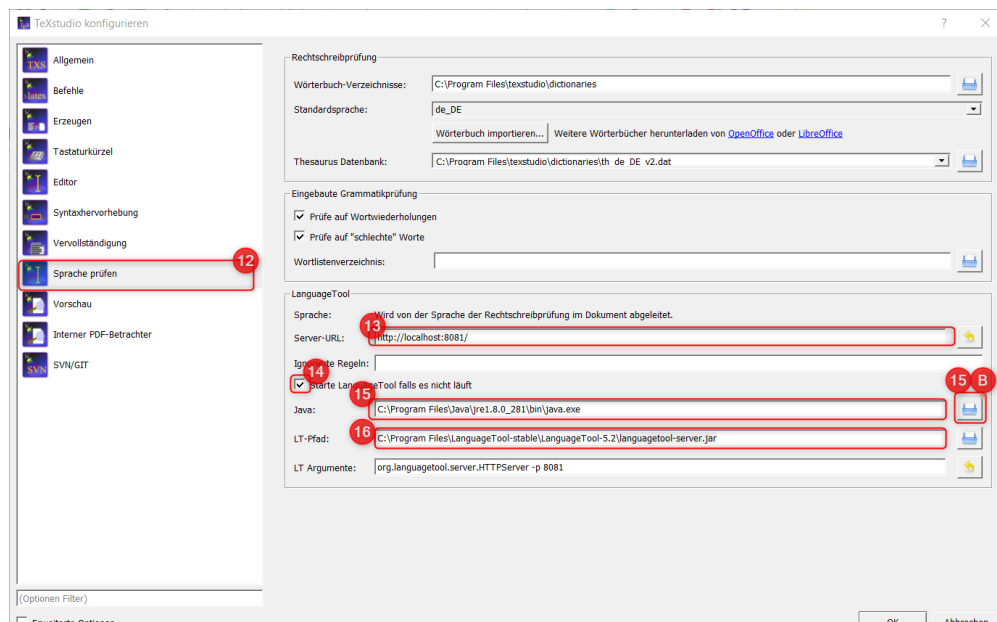


Abbildung 3.4

4 Benutzung dieser Vorlage

4.1 Vorbereitung

Zu Beginn muss die Datei „bitte_ausfuellen.tex“ mit Name(n), Matrikelnummer(n) etc. versehen werden. Wenn bspw. Matrikelnummer(n) nicht erwünscht sind, wird das entsprechende Feld einfach leer gelassen.

Namen stehen dabei untereinander in der Kopfzeile links, die Gruppennummer in der Mitte und der Titel rechts. Das Datum ist immer automatisch das, was zur Kompilierzeit gilt.

Quellen liegen in der „sources.bib“. Dort sind bereits standard Bücher für Labore für E-Technik Studenten eingetragen, eigene Quellen können aber nach Belieben eingefügt werden. In dem Quellverzeichnis des Dokuments stehen am Ende nur Quellen, welche auch im Text benutzt und zitiert wurden, nicht automatisch alle, welche in dieser Datei stehen!

4.2 Einzelarbeit

Wenn alleine gearbeitet wird befindet sich im Ordner „Einzelarbeit“ die Datei „Nachname_Name.tex“. Dort kann unter folgendes mit dem eigenen Text begonnen werden.

```
%%%%%%%%%%%%%%  
%%%%%%%%%%%%%% Ab hier Text %%%%%%%%%%%%%%%  
%%%%%%%%%%%%%%
```

Die Kommentare können, wenn gewünscht, gelöscht werden, diese zeigen nur meist wichtige Befehle für LaTeX.

Über dem „Ab hier Text“ sind noch Befehle auskommentiert. Wenn Sachen wie Deckblatt oder Sperrvermerk inkludiert werden sollen, muss einfach das Prozentzeichen entfernt werden (und der Name angepasst werden).

4.3 Gruppenarbeit

Für die Gruppenarbeit liegen im Ordner „Gruppenarbeit“ 3 weitere Ordner bereit. In Inhalt 1 kann einer arbeiten und in Inhalt 2 kann einer arbeiten. Die entsprechenden Dateien kompilieren auch nur diesen speziellen Inhalt. .Inhalt1_2 fügt beides zusammen (zuerst Inhalt1 und darunter Inhalt2).

Ganz oben im Header gibt es 2 !Tex root. Das mit einem Prozentzeichen ist aktiv, das mit 2 auskommentiert. Kompiliert man einmal alles, also Inhalt1_und_Inhalt2 erhält man sämtliche Labels, welcher der Gruppenpartner verwendet hat. Danach kann man wieder nur den eigenen Inhalt kompilieren und mit den Labels weiter arbeiten. Zum Aktualisieren muss jedes mal einmal der gesamte Inhalt kompiliert werden.

5 Kapitel, Text, Zeilen- und Seitenumbrüche

5.1 Kapitel, Unterkapitel, Laborkapitel

`\section{KaptitelName}`

Neue Kapitel werden mit diesem Befehl eingeleitet. Diese sind Fettgedruckt und erhalten eine einzelne Ziffer x

`\subsection{UnterKapitelName}`

Unterkapitel werden mit damit eingeleitet. Diese sind nicht mehr fettgedruckt, leicht eingerückt und erhalten die Nummerierung $x.y$

`\subsubsection{UnterUnterKapitelName}`

Soll das ganze weiter unterteilt werden funktioniert dies mit diesem Befehl. Diese sind gleich zu Unterkapiteln, allerdings weiter eingerückt und mit $x.y.z$ nummeriert.

`\laborsubsection{arg1}{arg2}`

Für Labore wurde im Header die neue Funktion erstellt. In $arg1$ kommt der Buchstabe des Versuchsteils, also V,D oder A. $arg2$ ist der Kapitelname des Kapitels.

Um den Counter zurückzusetzen, also beispielsweise von V1.3 zu D1.1 gibt es folgenden Befehl:

`\resetlaborsectioncounter`

5.2 Text, Zeilen- und Seitenumbrüche

5.3 Referenzieren und Zitieren

Um auf Gleichungen, Kapitel, Bilder, Tabellen usw. zu verweisen sollte der `autoref` Befehl verwendet werden. Dazu muss das Objekt, auf das verwiesen werden soll mit einem `label` benannt werden.

```
\label{key}      \autoref{label}
```

Das Bild Abbildung 5.1 zeigt genau dieses Beispiel zur Nutzung von Labeln und Referenzierungen:

```
Das Bild \autoref{fig:beispielautoref} zeigt ein Beispiel zur Nutzung von Labeln und Referenzierungen:  
\begin{figure}[H]  
  \centering  
  \includegraphics[width=0.9\linewidth]{figs/Beispiel_autoref.png}  
  \caption{Beispiel für Label und \autoref}  
  \label{fig:beispielautoref}  
\end{figure}
```

Abbildung 5.1: Beispiel für Label und Autoref

Autoref erkennt automatisch in welcher Umgebung sich das Label befindet und kennt seine Nummerierung. Zudem ist diese Referenzierung im PDF anklickbar und man gelangt direkt zu dem Teil, wo das Label gesetzt wurde.

Für Quellen wird der folgende Befehl verwendet:

```
\cite[postnote]{bibid}
```

Bibid ist der Name der Quelle aus der „sources.bib“, also die verwendete Quelle. Als postnote kann zum Beispiel S22-25 eingetragen werden.

6 Inkscape Latex

7 Bilder

7.1 Bild einfügen

Um ein Bild einzufügen kann dies händisch mit nachfolgendem geschehen oder in TeXstudio → „Assistenten“ → „Grafik Einfügen“ als Hilfe genutzt werden.

```
\begin{figure}[H]
\centering
\includegraphics[width=0.7\linewidth]{ORDNER/BILDNAME.DATEIFORMAT}
\caption{BILDUNTERSCHRIFT}
\label{fig:LabelName}
\end{figure}
```

Das [H] steht für „here“, das Bild wird also genau an der Stelle eingefügt, wo auch der Code in Latex steht. Ansonsten erscheint das Bild immer ganz oben im PDF.

„Centering“ ist um das Bild zu zentrieren.

„includegraphics“ importiert das Bild. Dabei ist darauf zu achten, dass der Pfad, Dateiname und das Dateiformat richtig eingetragen sind. In dieser Vorlage sind Ordner wie „figs“ und „Bilder“ bereits Standardpfade und müssen nicht angegeben werden.

Die Größe kann mit „width=“ verändert werden.

„Caption“ ist die Bildunterschrift, also der Text unter dem Bild, welcher im PDF erscheint.

„Label“ sollte bereits bekannt sein. Mit diesem Namen kann auf das Bild referenziert werden.

Zur Kennzeichnung ist es üblich, dass Labels für Bilder mit „fig:“ beginnen und dahinter der Name kommt.

7.2 Platzhalter für nachträgliche Bilder

Platzhalter dienen dazu, noch nicht fertige/vorhandene Bilder mit einem Achtungszeichen zu integrieren. So hat man das Label, kann damit weiterarbeiten und denkt später daran, das Bild noch hinzuzufügen und muss nur eine Kleinigkeit ändern.

```
\begin{figure}[H]
\centering
\missingfigure{TEXT ZUR BESCHREIBUNG WAS FEHLT}
\captionof{figure}{BILDUNTERSCHRIFT}
\label{fig:LABEL}
\end{figure}
```

In „missingfigure“ kann ein kleiner Text zur Beschreibung was genau Fehlt. Dieser erscheint auch im PDF.

8 Einfache Tabellen

Um simple Tabellen zu erstellen hilft die Website <https://tableconvert.com/>
Dort können die Spalten manuell ausgefüllt werden oder Excel, CSV, uvm. Tabellen importiert werden. Die Website wandelt die Tabelle dann in Latex Code um.

9 Mathe- Formelumgebung

Einen Überblick über die Umgebung und Sonderzeichen, welche in Formeln häufig gebraucht werden lässt sich unter https://www-astro.physik.tu-berlin.de/files/Uebung/Dokumentationen/mathe_in_latex2e.pdf finden

9.1 Formeln

9.1.1 Einzelne Formeln

Um einfache Mathematische Formeln in den Text einzubinden gibt es folgende Umgebung:

```
\begin{equation}  
Inhalt...  
\end{equation}
```

Bei dieser wird die Nummerierung automatisch normgerecht angepasst. Die Gleichung kann zusätzlich mit einem Label versehen werden und auf diese mittels autoref im Text verwiesen werden. Für einfache Unterscheidung sollten Label für Gleichungen immer mit „eq:“ beginnen.

9.1.2 Mehrere Formeln gleicher Nummerierung

Hat man mehrere Gleichungen, welche die gleiche Nummer erhalten sollen und nur durch a) b) c)... unterschieden werden funktioniert dies mit folgender Umgebung:

```
\begin{subequations}  
\begin{align}  
&Gleichung 1\\  
&Gleichung 2\\  
&Gleichung 3  
\end{align}  
\end{subequations}
```

Jede neue Gleichung beginnt mit einem & und endet mit

für einen Zeilenumbruch zur nächsten Formel. Die letzte erhält kein

Ansonsten entsteht eine weitere Formel ohne Inhalt!

Jede einzelne der Subequations kann mit einem Label versehen werden.

9.1.3 Zeilenumbruch innerhalb einer Formel

Für längere Formeln, welche nicht in eine Zeile passen gibt es folgende Möglichkeit:

```
\begin{equation}
\begin{split}
Zeile1 &= Zeile1 \\
Zeile2 &= Zeile2
\end{split}
\end{equation}
```

Der Vorteil des &= ist, dass die Zeilen nach dem „=“ ausgerichtet sind untereinander. Es gibt auch die „multiline“ Umgebung, allerdings ist dort die zweite Zeile Linksbündig, was nicht gut aussieht.

9.2 Vektoren und Matrizen

9.2.1 Vektoren

9.2.2 Matrizen

9.3 Sonderzeichen

10 SI

Ausführliche Dokumentation unter <https://ctan.space-pro.be/tex-archive/macros/latex/contrib/siunitx/siunitx.pdf> zu finden.

11 Komplexere Tabellen

```
\begin{table}[H]
\centering
\captionof{table}{Tabellenüberschrift}
\renewcommand{\arraystretch}{2} % Senkrechten Abstand für diese Tabelle erhöhen
\setlength{\tabcolsep}{0.3em} % Horizontalen Abstand für diese Tabelle erhöhen
% S in table richtet diese nach kommata aus
% dies funktioniert bei starkunsymmetrischen zahlen nicht.
% dazu kann man mit table-format = A.B die anzahl der erwarteten stellen eingeben
dabei steht A für die Anzahl der zahlen vor dem komma und B für die Anzahl hinter
dem komma
% test in der spalte mit {} umklammern bsp {text $P_A$}

% \makecell{ 1. zeile \\ 2. zeile} ermöglicht einfache zeilenumbrüche in einer zelle
\begin{tabular}{|c|c|S[table-format = 2.3]|S[table-format = 2.3]|
S[table-format = 1.4]|S[table-format = 1.4]|}
\hline
\multicolumn{2}{|c|}{& {\makecell{Ergebnis\\ der Wirkleistung \\
aus der Simulation \\ in Watt}} & {\makecell{berechnete \\
Wirkleistung\\ in Watt}} & {\makecell{Abweichungen}}
& {\makecell{Abweichung\\ in \%}} \\ \hline
\multirrow{3}{*}{\makecell{$L_1$\\ als Bezug}}
& $P_A$ & 15,500 & 15,4954 & 0,0046 & 0,0297 \\ \cline{2-6}
& $P_B$ & 47,761 & 47,7607 & 0,0003 & 0,0006 \\ \cline{2-6}
& $P_{ges}$ & 63,261 & 63,2561 & 0,0049 & 0,0077 \\ \hline
\multirrow{3}{*}{\makecell{$L_2$\\ als Bezug}}
& $P_A$ & 38,398 & 38,3972 & 0,0008 & 0,0021 \\ \cline{2-6}
& $P_B$ & 24,863 & 24,8624 & 0,0006 & 0,0024 \\ \cline{2-6}
& $P_{ges}$ & 63,261 & 63,2596 & 0,0014 & 0,0022 \\ \hline
\end{tabular}
\end{table}
```

Dieser Code entspricht folgender Tabelle:

Tabelle 11.1: Tabellenüberschrift

		Ergebnis der Wirkleistung aus der Simulation in Watt	berechnete Wirkleistung in Watt	Abweichungen	Abweichung in %
L_1 als Bezug	P_A	15,500	15,4954	0,0046	0,0297
	P_B	47,761	47,7607	0,0003	0,0006
	P_{ges}	63,261	63,2561	0,0049	0,0077
L_2 als Bezug	P_A	38,398	38,3972	0,0008	0,0021
	P_B	24,863	24,8624	0,0006	0,0024
	P_{ges}	63,261	63,2596	0,0014	0,0022

12 Programmiercode

Um Programmiercode in das Latex Dokument einzubinden dient das Paket listings. Eine ausführlichere Dokumentation findet man unter https://en.wikibooks.org/wiki/LaTeX/Source_Code_Listings

Mittels `\lstset{}` können Einstellungen vorgenommen werden zum Beispiel um Farben für bestimmte Keywords (z.B. „else if“) festgelegt werden, aber auch automatische Zeilenumbrüche und vieles mehr.

Nun gibt es zwei Möglichkeiten den Programmcode einzubinden:

1. Datei importieren:

```
\lstinputlisting[caption=My caption]{sourcefile.cpp}
```

Hierbei ist darauf zu achten, dass die Datei „sourcefile.cpp“ auch in dem Ordner der Datei liegt, die auch kompiliert wird. Ansonsten muss von diesem Ordner aus mittels `../` bzw `/Ordnername` zu dem jeweiligen Ordner der Codedatei navigiert werden.

2. Code händisch einfügen:

```
\begin{lstlisting}[frame=single] % Start your code-block
for i:=maxint to 0 do
begin
{ do nothing }
end;
Write('Case insensitive ');
Write('Pascal keywords. ');
\end{lstlisting}
```

13 PDF-Seiten einbinden

Um PDF-Seiten einzubinden dient der Befehl

```
\includepdf[einstellungen]{Name des PDF}
```

Bei soll nur eine Seiten eingebunden werden, kann [einstellungen] weggelassen werden. Ansonsten kann dort die Seitenzahl eingetragen werden oder einfach - um das gesamte PDF einzubinden.

Bei Name des PDF muss der Name der PDF Datei eingetragen werden OHNE die Endung .pdf Falls das PDF in einem anderen Ordner liegt kann man mittels ../ einen Ordner nach oben gehen oder mittels Ordner_Name_1/Ordner_Name_2/pdfDatei durch die jeweiligen Unterordner zu der Datei navigieren.

Weitere Dokumentation ist unter <https://www.namsu.de/Extra/pakete/Pdfpages.html> zu finden.

14 PGFPlots

Um Graphen mit Latex zu erstellen gibt es das Paket PDFPlots. Eine ausführliche Dokumentation findet man unter <http://pgfplots.sourceforge.net/pgfplots.pdf>.

Wir raten allerdings dazu diese mit MatLab oder Python und Matplotlib zu erstellen und dann als Bild in Latex einzufügen, da diese Variante simpler ist und besser nachträgliche Änderungen zulässt.

PGFPlots kann allerdings auch 3D Plots, wobei die Befehle sehr ähnlich zu denen von MatLab sind.

15 Fehlermeldungen