

TEXを使った論文執筆 ~2024~



my recommendations on research & development tools.

正会員 小川一人†

キーワード: TEX, Texworks, ite, word, bibtex

1. まえがき

1.1 TEXとワード

「論文の投稿においてはPDF形式もしくはPS形式が一般的になっています。この原稿を書く際、何を使いますか？ワードですか、TEXですか。」という問いかけで始まりWindowsにおける論文執筆においてもTEXの利用を勧める“TEX, Winshellを使った論文執筆”¹⁾が、2013年に映情学誌に掲載されました。そして、その補足バージョンである“TeX, Winshellを使った論文執筆その2”²⁾が、2014年に掲載されました。

文献¹⁾の掲載から11年が経ちましたが、Microsoft Word等のワードソフトウェアとTEXで書かれた論文を比較した場合、完成された論文の美しさについては、相変わらずTEXのほうが優勢に感じられます。私がワードを熟知していないためかもしれませんが、自分で論文を書いた場合は間違いなくTEXにより執筆した原稿のほうが美しく見えます。他の著者が書かれた論文を読む際には、行間が整っていない、特に数式の行間に違和感がある原稿を見た瞬間に“これはワードで書かれた論文だな”と思ってしまう、というのが本当のところです。印刷・出版関係の言葉で文字や図を配置する作業のことを組版(くみはん)というそうですが、TEXは、素人でも簡単に綺麗に組版することを目的として作られています。特に、数式表現においてはワードも進歩してきていますが、TEXのほうが利点が大きいです。

当然、ワードにも大きな利点があります。TEXでは文章を書きながら印刷された状態を知ることは不可能です。TEXファイルをコンパイルし、プレビューなどで確認する手間はかかります。ワードでは、コンパイルなどの処理は不要となります。さらに、図の挿入においては、ワードのほうが優れています。TEXで図を描画することは非常に手間がかかります。および、TEXで図を挿入する場合、挿入する図のフォーマットは限定されております。

TEXもワードも良い点はそれぞれです。一概にどちらが

優れている、というべきではありませんが、個人的な感想として、“論文執筆においては、論文の綺麗さが査読者の心に何らかの影響を及ぼす”ことを考えれば、論文執筆においてはTEXの利点が勝っていると感じます。

さて、文献¹⁾の掲載から11年が経ち、Winshellはすでに使われなくなっています。正確には、WinshellがWindowsのバージョンアップに対応できていないため、使用できなくなっています。このため、本稿では、現行のWindowsで対応可能なTex用の環境を紹介するとともに、文献¹⁾および文献²⁾では紹介していないTexの機能の一部を紹介し、TEXによる論文執筆を推奨させていただきます。

1.2 TEX用エディタ

直前に述べましたように、Winshellは使用できなくなっています。このため、別のTEXエディタを用いることとなります。私の近辺で尋ねただけですが、Windows上でTEXを利用して論文を書かれている著者の中ではTex Liveを使用することが多いようです。実際、私もTex Liveを使用しています。Tex LiveはTEX用のディストリビューションと呼ばれるものです。ディストリビューションとは、あるソフトウェアとそれを動かすために必要となる他のソフトウェアやOS処理などを統合したものです。そして、Tex Liveの中には、TEX本体だけではなく、TEX用エディタTexworks等も入っています。Tex Liveの利用者が多いということは、Texworksの利用者が多いとなりますので、本稿では、Texworksを用いた論文の書き方を紹介します。

2. TEX利用の準備をする

WindowsでTEXを利用するための準備として、TEXおよびその関連ソフトウェアをインストールする必要があります。そこで、本章では、それらの必要なソフトウェアが統合されているTex Liveのインストール、および、TEX用エディタTexworksの設定方法を紹介します。

2.1 映像情報メディア学会用スタイルファイルの準備

映像情報メディア学会でも、TEX用のテンプレートなど必要なファイルが準備されています。https://www.ite.or.jp/content/paper/の中に“執筆の手引き”があり、その中の“投稿論文用スタイルファイル (TeX/Word)”をクリックし

† 国立研究開発法人情報通信研究機構 サイバーセキュリティ研究所
"How to Write a Paper Using TEX: 2024" by Kazuto Ogawa (National Institute of Information and Communications Technology, Tokyo)



ますと、“itestyle.zip”がダウンロードされます。このzipファイルには、いろいろなファイルがまとめられています。今回必要となるのが、itestyle\TeX_style\Winの中にある“ite.cls”と“readme_japanese.tex”となります。

2.2 TEXのインストール

文献¹⁾が書かれた時期と比較すると、TEXのインストールは非常に楽になっています。Tex Liveをインストールすれば、TEXがインストールされるだけではなく、その関連ソフトウェアも一度にインストールされるためです。かつてのように別々にいろいろなソフトウェアをインストールする必要がなくなっています。実際、Tex Liveでは

- ・TEX本体
- ・プレビューアー (dviout等)
- ・フォント関連ファイル (ghostscript等)
- ・エディタ (Texworks)

等がインストールされます。

ここで、2通りのTex Liveのインストール方法を紹介します。どちらの場合も現在のTex Liveの最新バージョンであるTex Live 2023がインストールされます。

(a) ネットワークインストーラでのインストール

インストールのためにはネットワークインストーラの使用が便利です。http://mirror.ctan.org/systems/texlive/tlnetなど、複数のWebサイトにTex Liveが準備されています。まず、そのどれか一つのサイトにアクセスし、“install-tl-windows.exe”をダウンロードします。これがWindows用のインストーラとなります。install-tl-windows.exeをダウンロードした後、install-tl-windows.exeをダブルクリックすれば、自動的にソフトウェア一式をインストールしてもらえます。ただし、最初にWindowsから“WindowsによってPCが保護されましたWindows Defender SmartScreenは…”として、Windowsの保護機能によりインストールが中断されます。この画面が出ましたら“詳細情報”と書かれている部分をクリックします。クリック後に“[実行]”が表示されますので、“[実行]”をクリックしますとインストールが開始されます。途中で、インストール先のディレクトリ等が変更できますが、気にせずにそのままの設定でインストールされて良いと思います。このインストールには結構な時間が必要となります。ネットワークの環境等に依存しますが、半日作業の場合もあるようです。短い場合であっても1時間程度は必要だと思いますので、時間に余裕のある際にインストールすることをお勧めします。そして、“Tex Liveへようこそ!”のメッセージが表示されましたら、インストール完了です。

(b) ディスクイメージファイルからのインストール

ネットワークインストーラの場合、企業や学校によっては、プロキシの設定、ファイアウォールの設定、ワクチンソフト、ウイルスチェックの設定などにより、ネットワークインストーラが正しく動作しないことがあるようです。このため、ディスクイメージを使用したインストール方法も準備

されています。まず、https://jp.mirrors.cicku.me/ctan/systems/texlive/Images/などにアクセスし、“texlive2023.iso”をダウンロードします。ネットワークインストーラでダウンロードしたinstall-tl-windows.exeは20 MB程度のファイルですが、texlive2023.isoは4.7 GBと大きなファイルです。ご自身のPCの空き容量に注意が必要です。texlive2023.isoが入手できましたら、このファイルをWindowsに“ディスクである”ことを認識させなければなりません。この操作が“マウント”となります。Windows11の場合ですと、texlive2023.isoを左クリックするとツールバーに“マウント”と表示され、ここをクリックするとtexlive2023.isoがディスクとして認識されます。Windowsの初期画面に“PCのアイコンがあると思いますが、ここをダブルクリックしていただくと、“DVDドライブ (D:) texlive2023”が出現していると思います。DVDドライブ (D:) texlive2023の中にautorunがありますので、DVDドライブ (D:) texlive2023をクリックすると自動でインストーラが動作を始めます。ワクチンソフトなどの影響でautorunが拒否される場合は、DVDドライブ (D:) texlive2023の中に“install-tl-windows.bat”がありますので、これをダブルクリックしtexliveをインストールします。この方法でも、“Tex Liveへようこそ!”のメッセージが表示されましたら、インストール完了です。インストール後、DVDドライブ (D:) texlive2023はアンマウントすることをお勧めします。アンマウント方法は、DVDドライブ (D:) texlive2023を左クリックすると、ツールバーに“取り出し”マークがあらわれますので、これをクリックすれば実行されます。

2.3 Texworksの準備

Tex Liveをインストールする際に、言語を日本語に設定してあれば、ほぼ何の設定もなくTexworksは使えるはずですが、その前にTexworksのショートカットをWindowsのデスクトップ画面に表示したり、“.tex”や“.bib”ファイル (bibファイルについては後述) をTexworksに関連付けておくとう便利です。

(a) Texworksのショートカット生成

こちらはTEXの機能ではなく、Windowsの機能ですが、ここに記しておきます。

Texworksは、Tex Liveをインストールしたディレクトリを<install_directory>とすれば、“C:\<installdirectory>\w32tex\share\texworks”にインストールされます。このtexworksのアプリケーションに対して、「Alt」キーを押しながらドラッグしてデスクトップにドロップすればデスクトップにTexworksのショートカットが作成されます。

(b) “.tex”、“.bib”ファイルとTexworksの関連付け

こちらもWindowsの機能ですが、ここに記しておきます。先ほど作成したTexworksのショートカットをダブルクリックしてtexworksを立ち上げます。何でも良いので、例えば“abcdefg”などと書き、ファイル→名前を付けて保存などにより、ファイルを保存します。その後、Windows

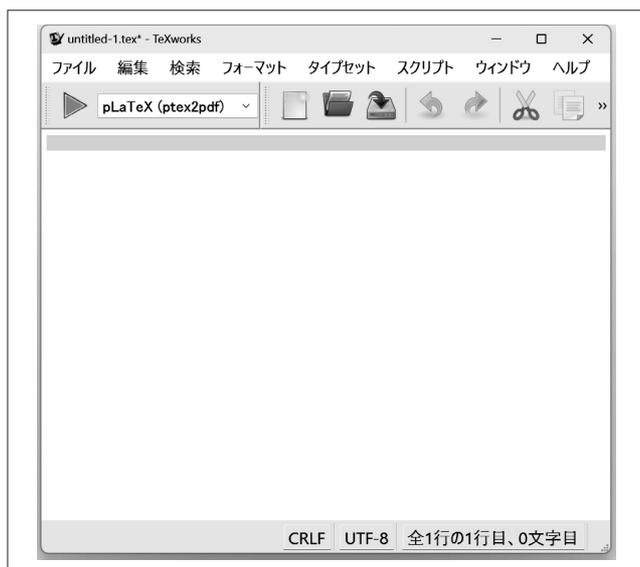


図1 TeXworks 初期画面

の中で

設定→アプリ→規定のアプリ→ファイルの種類で規定値を選択する→.tex→PCでアプリを選択→texworks→規定値を設定する

を行うことで、関連付けができます。“bib”ファイルの場合はこの手順の“.tex”を“.bib”に置き換えて、同じ操作を行います。

この設定により、以後は“.tex”，“.bib”ファイルをダブルクリックすれば、TeXworksが立ち上がります。

(c) エディタでの設定

texworksを立ち上げてみます。Tex Liveが正常にインストールされていれば，“すべてのアプリ”の中に“texworkseditor”が入っています。これをたちあげると図1のようになります。

ほとんど設定する項目はありませんが、使用するフォントが日本語の場合、エンコーディングの方法が異なるファイルを開きますと文字化けします。文字化けを避けるために、編集→設定→エディタ→エンコーディングの中から適切なものを選び設定することをお勧めします。

3. 使ってみる：論文執筆

TEXには多くのコマンドがあり、それらをすべて紹介することはできません。そこで、本章においても、文献¹⁾と同じように、ワードからTEXに移行することを考えます。そして、文献¹⁾²⁾では紹介していないが有用で、ワードより優位と考えられるコマンドを紹介します。ワードからTEXへの移行方法の基礎的な部分は文献²⁾を参照いただきたいと思います。

3.1 texworksの使い方

texworksを立ち上げていただき、図2のように記載してTEXファイルを作成します。TEXファイルの保存は、“ファイル→保存”でできますが最初はファイル名を尋ねら

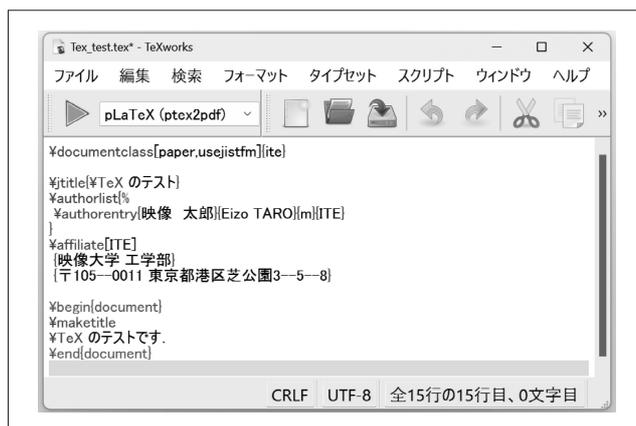


図2 TeXworksのテスト

表1 図2の記載内容説明

<code>\documentclass [paper, usejstfm]{ite}</code>	映像情報メディア学会誌への論文投稿の場合は、最初にこの1行を書く、とってください。
<code>\jtitle</code>	タイトルを記載します。
<code>\authorlist{</code> <code>\authorentry</code> <code>}</code>	著者名等を記載します。
<code>\affiliate</code>	著者の所属などを記載します。映像情報メディア学会誌の場合1ページ目の左下にこの記載がされます。
<code>\begin{document}</code> ...	本文の始まり ここに本文を書きます。
<code>\end{document}</code>	本文の終わり

れますので、ファイル名を入力してください。図2では“Tex_test.tex”をファイル名としてあります。

参考までに、図2の記載内容の説明を表1に記載しておきます。1行目に“{ite}”が書かれていますので、この書式は映像情報メディア学会の論文用の書式になります。他の論文誌ではその論文誌の指示に従うことになります。

さらに、映像情報メディア学会用の論文執筆を行う著者の皆様に補足させていただきますが、表1の1行目に“\documentclass [paper, usejstfm]{ite}”，“映像情報メディア学会誌への論文投稿の場合は、最初にこの1行を書く、とってください。”と記載しました。実は、2.1節で紹介した映像情報メディア学会のzipファイルの中には、“readme_japanese.tex”が入っており、そのファイルの中から、5行目の%を外したコマンドがこのコマンドと一致します。論文執筆の場合は、この1行に加えて、9行目“\usepackage {graphicx}”から28行目“\No {1}”までの内容をコピー・ペーストして使用することをお勧めします。申し訳ございませんが、この部分については説明いたしません。内容を知りたい場合は、Web等で調べていただきたいと思います。

話を戻しますが、TEXファイルを作成した後、TEXファイルのコンパイルを行います。この際、ファイル“ite.cls”がTEXファイルと同じディレクトリにあることが必要となりますので、コピーしておいてください。そして、左上にある再生ボタンをクリックします(図3)。この再生

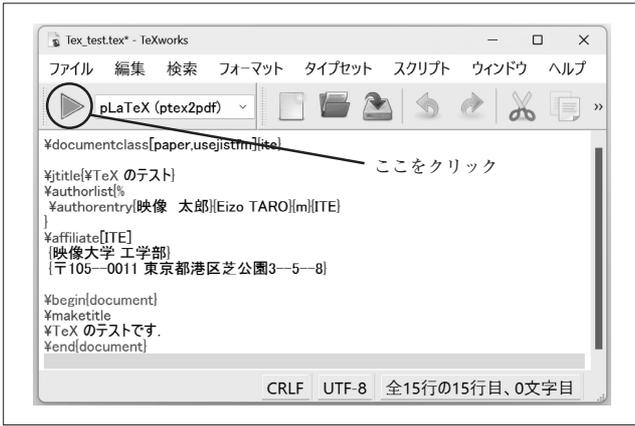


図3 TeXworksでのコンパイル

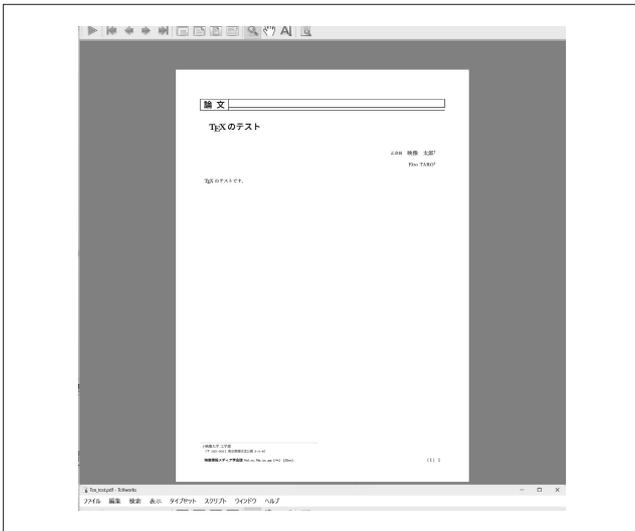


図4 作成されたPDF

ボタンではそのすぐ右にある項目“pLaTeX (ptex2pdf)”によりTEXファイルがコンパイルされます。

この後、ウィンドウの下側に、どのような処理が行われたかが表示され、最後にPDFファイルが作成され、図4のウィンドウが開かれると思います。

この“TEXファイル生成→コンパイル→PDF作成”がTEXにおける基本的な一連の作業となります。

ちなみに、ワードで書かれた論文は印刷された状態を知ることができますが、PDF化する場合は“PDFへの印刷”などを選択して行うことになります。

3.2 数式

次に、TEXでの数式の記載について説明します。文献¹⁾では“数式についてはTEXがワードよりも優位”と書かれていますが、優位な部分についての説明がありませんでした。本節では、TEXがワードより優位な部分を少し紹介したいと思います。

数式を比較した場合、数式が複雑になるほどTEXのほうが綺麗に記述される気がしています。例えば、指数と分数と総和計算を組み合わせた数式 $f(x) = \sum_{i=1}^{\infty} \frac{1}{i^x}$ をワードと

$f(x) = (x + 1)(x + 2)$ $= x^2 + 3x + 2$ $= \frac{k}{z}(x + \alpha)(x + \beta)$ $= ax^2 + bx + c$ <p>(a) ワード</p>	$f(x) = (x + 1)(x + 2)$ $= x^2 + 3x + 2$ $= \frac{k}{z}(x + \alpha)(x + \beta)$ $= ax^2 + bx + c$ <p>(B) TEX</p>
--	--

図5 数式比較 (注：数式には何の意味もありません)

TEXで比較した場合、なぜかワードでは行間が広がってしましますが、TEXではそのような感じは受けません。ちなみに、この数式のTEXのソースコードは $f(x) = \sum_{i=1}^{\infty} \frac{1}{i^x}$ なので、ご自身の環境で、ワードとTEXの違いを感じてみてください。

また、数式を並べて記述することは良くありますが、ここでも同じことが生じます。例えば、

$$y = (x + 1)(x + 2)$$

$$= x^2 + 3x + 2$$

であれば、TEXもワードも見た目は変わりません。この数式にさらに分数を含む数式を加えた図5の例では、TEXとワードで差が出ます。

デフォルトで使用されているフォントの違いはさておき、ワードでは3行目の縦方向の幅だけが広くとられており、TEXの場合はすべて等間隔です。人により好みは分かれるのかもしれませんが、私はTEXのほうが綺麗に見えます。特に、このような状況が何行にも続き、広い間隔と普通の間隔の行の混在は、見た目が良くないと感じます。

同じような現象は数式を使う際にはしばしば起こります。いろいろ試してみられると面白いと思います。

3.3 ファイルの結合

ほぼすべての論文は複数の章に分かれています。論文が長編となった場合は、その編集を章ごと、さらには節ごとに行えるほうが効率的になります。そこで、本節では、章(節)毎に作成されたファイルの結合方法について説明します。

TEXのファイルの結合は \input コマンドを使います。例えば、論文体をmain.texとし、第1章をsection1.tex、第2章をsection2.texとして別々のファイルとして執筆します。具体的なmain.tex、section1.tex、section2.texの内容は表2のようになります。section1.texやsection2.texを挿入したい場所に $\input{section1.tex}$ や $\input{section2.tex}$ と記載することで、ファイルの挿入ができます。

TEXの場合は、どこを編集した場合でも、PDFファイルに反映させるためには、最低一度はコンパイルを行います。ファイルを統合する場合、各ファイルを保存しておき、main.texのコンパイルを行うことで、section1.tex、section2.texが統合されて、PDF化されます。

ワードにおけるファイルの結合は、ファイルを挿入、ファイルへのリンク表示、の2種類が主な方法となります。

表2 論文を分割して作成する際のT_EXソース
(抜粋)

main.tex ソース
<code>¥documentclass{paper}{ite}</code>
...
<code>¥begin{document}</code>
...
<code>¥input{section1.tex}</code>
<code>¥input{section2.tex}</code>
...
<code>¥end{document}</code>
section1.tex ソース
<code>¥section{まえがき}</code>
<code>¥subsection{T_EX とワード}</code>
“論文の投稿...”
section2.tex ソース
<code>¥section{T_EX 利用の準備をする}</code>
Windows で T _E X を利用する ...

印刷された状態を知ることができることがワードの利点であるため、ワードとしては、ファイルの挿入のほうが利点があると言えます。ただし、挿入するファイルを編集した場合、以前に挿入した部分を削除し、再度、ファイルの挿入の操作をしなければなりません。したがって、挿入されるファイルを編集する場合は、結構手間がかかります。

3.4 bib ファイル：参考文献の書き方

bib ファイルの利用はファイルの結合の一種となりますが、参考文献の参照・記載で有用となる bibtex について紹

介します。

文献¹⁾では引用した文献を論文内に bibitem コマンドを用いて、直接記載する方法が記載されています。bibitem の利便性を良くしたものが bibtex です。bibtex では本体の “.tex” ファイルとは異なる “.bib” ファイル内に文献を記載しておきます。そして、コンパイル時に、 “.bib” ファイルを読み込み、統合します。

bibtex を用いた参考文献の書き方を表3に記載します。また、比較のために bibitem、ワードでの参考文献の書き方を併記します。

bibtex を使う場合、以下のように行います。

- ・ T_EX 本文 (main.tex) : `¥bibliographystyle {xxxx}`, `¥bibliography {bib_filename}` を `¥end {document}` の前に記載
- ・ bib ファイル (bib_filename.bib) : 著者 (author), タイトル (title), ..., 発行年 (year) などを記載

参考文献の書式は、`¥bibliographystyle {xxxx}` の xxxx により決まります。表3の例では、xxxx が plain となっていますが、スタイル xxxx により、参考文献の並べ方(アルファベット順なのか、出現順なのか等)、どの項目を参考文献に記載するかなどが指定されます。Web 上で検索すれば、さまざまなスタイルファイルがありますので、ご自身の意図に合うスタイルを探してみるのも良いかもしれません。

そして、bib ファイルの書き方は、参考文献毎に、引用

表3 bibtex, bibitem, ワードの比較

bibtex
main.tex の最後 <code>¥{document}</code> の前に以下の 2 行を記載 <code>¥bibliographystyle{plain}</code> <code>¥bibliography{bib_filename}</code>
bib_filename.bib ファイルに参考文献情報を記載 <code>@article{Ogawa2013,</code> <code>author = { 小川一人 },</code> <code>journal = { 映像情報メディア学会誌 },</code> <code>title = {{T_EX, Winshell を使った論文執筆}},</code> <code>volume = {67},</code> <code>number = {6},</code> <code>pages = {498-499},</code> <code>year = {2013}</code> <code>}</code> <code>@article{Ogawa2014,</code> <code>author = { 小川一人 },</code> <code>journal = { 映像情報メディア学会誌 },</code> <code>title = {{T_EX, Winshell を使った論文執筆 その2}},</code> <code>volume = {{68},</code> <code>number = {4},</code> <code>pages = {327-329},</code> <code>year = {2014}</code> <code>}</code>
bibitem
main.tex の最後に以下を記載 <code>¥begin{thebibliography}{99}</code> <code>¥bibitem{Ogawa2013}, 小川一人, “TEX, Winshell を使った論文執筆”, 映像情報メディア学会誌, 67(6):498-499, 2013.</code> <code>¥bibitem{Ogawa2014}, 小川一人, “TEX, Winshell を使った論文執筆 その2”. 映像情報メディア学会誌, 68(4):327-329, 2014.</code> <code>¥end{thebibliography}</code>
ワード
1) 小川一人. TEX, Winshell を使った論文執筆. 映像情報メディア学会誌, 67(6):498-499, 2013.
2) 小川一人. TEX, Winshell を使った論文執筆 その2. 映像情報メディア学会誌, 68(4):327-329, 2014.



図6 bibtexによるコンパイル

名、著者名、発行誌、タイトルなどを項目毎に“項目名={値}”という形式で記載します。引用名とは本文中で参考文献を引用するために使用する名前です。例えば、本稿で“文献¹⁾”と書いた文献は“Ogawa13”と名付けています。引用する際に“\cite{Ogawa13}”と書くことで“¹⁾”のように記載されます。

こうして作られたbibファイルをTeXファイルと統合しPDF化するためにコンパイルを行います。bibtexを利用する場合はTeX本文(main.tex)において、“pLaTeX (ptex2pdf)によるコンパイル→pBibTeXによるコンパイル→pLaTeX (ptex2pdf)によるコンパイル→pLaTeX (ptex2pdf)によるコンパイル”と4回のコンパイルが必要となります。pLaTeX (ptex2pdf)によるコンパイルは図6に示すように再生ボタンの右側のプルダウンメニューからpLaTeX (ptex2pdf)を選択し、再生ボタンをクリック、pbibtexによるコンパイルは、pBibTeXを選択し、再生ボタンをクリックすることで実行できます。pbibtexによるコンパイルとは、作成されたTeXファイルで指定されているbibファイルをTeXファイルと統合する操作となります。これにより、bibファイルの内容がTeXファイルの内容と統合され、PDFファイルに反映されます。

bibtexの場合、bibファイルの中に不要な文献が入っていても問題ありません。コンパイル時に不要な文献はPDF化に使用されないように制御されています。私的にはこの機能はかなり便利だと思っています。何度も論文を書く際に、同じ分野の研究であれば、複数の論文で同じ文献を引用することはしばしば起こります。この際、同じbibファイルが使えることとなります。bibファイル内に不要な論文があったとしても自動的に削除されますので、不足している論文をbibファイルに加筆すれば(引用名は他の文献と重複しないように注意が必要)、次の論文でも同じbibファイルが使用できることとなります。何年にもわたり、同じ研究に携わるのであれば、この機能は必須だと感じます。

さて、比較として併記したbibitem、ワードの参考文献の書き方についても少し触れます。bibitemの場合、ここから参考文献であることを示す\begin{thebibliography} (99)と

参考文献の終了を示す\end{thebibliography}を書き、その間には含まれた部分に\bibitem参考文献名を直接記載します。文献の引用方法はbibtexと同じ方法です。ただし、コンパイルの方法は、“pLaTeX (ptex2pdf)によるコンパイル→pLaTeX (ptex2pdf)によるコンパイル”の2回となります。コンパイルにより自動的に附番され、PDF化されます。

ワードの場合は、文献名を直接記載します。引用する部分に著者が“文献¹⁾”のように文献番号を直接記載します。参考文献を記載するだけであれば、ワードが一番便利です。ただし、ワードではテンプレートやスタイルファイルが提供され、自動的な附番の準備が整えられていることは稀であり(私は知りませんが)、結果として、使い勝手はよくありません。

参考文献の記載方法に関しては、ワード> bibitem > bibtexとなりますが、附番、将来利用を考慮した場合、bibtex > bibitem > ワードではないかと思えます。

4. むすび

TeXでは、引用文献の自動附番以外にも、図表、章や節の附番も自動で行います。特に学会で指定の附番書式などがある場合は便利です。さらに、タイトルの位置調整したり、図・表の位置調整、文字間のスペース調整などを自動で行い、綺麗に整えてくれます。慣れるまでは、このTeXによる勝手の位置調整を腹立たしく思われる人もいますが(TeXを使い始めた頃の私の実感です)、慣れてきますと、組版の便利さ、論文の綺麗さがわかってきて、自動の位置調整などに頼る方が多いようです。文献¹⁾²⁾でも書かれています。ワードで文章、図、表、数式などを組版した論文と、TeXで書いた論文で同じ論文はできます。ただし、ワードでかなり一生懸命組版しなければ、TeXによる自動組版の美しさが実現できません。私見ではありますが、ワードに貼り付ける図を別のアプリで作成するように、図を他のepsエディタで作れば、TeXの組版はかなり綺麗な文章を作ってくれます。綺麗な論文に仕上げたことは採録へのワンステップとも言えます。TeXで論文を執筆してみたいはいかがでしょうか。(2023年11月27日受付)

〔文献〕

- 1) 小川一人：“TeX, Winshellを使った論文執筆”，映情学誌，67, 6, pp.498-499 (2013)
- 2) 小川一人：“TeX, Winshellを使った論文執筆その2”，映情学誌，68, 4, pp.327-329 (2014)



おがわ かずと
小川 一人 1987年、東京大学工学部電気・電子工学科卒業。同年、NHK入局。甲府放送局を経て、1991年～2023年、同放送技術研究所に勤務。2023年より、情報通信研究機構に勤務。コンテンツ流通のセキュリティ技術、暗号・署名技術の研究に従事。現在、情報通信研究機構上席研究員。博士(情報理工学)。正会員。