

TeX におけるプレゼンテーションツール

小原功任
金沢大学理学部

平成 18 年 1 月 20 日

最近の技術の進歩により、数学の講演でもノートパソコンとデータプロジェクターを使用するものが増えてきた。Microsoft の PowerPoint のような優れたツールもありフリーソフトウェアもあるが、原稿を作成するのに TeX を使いたいという人はかなり多いと思う。TeX で講演用のスライドを作成するにはいくつかの方法がある。単純に、横長になるように用紙サイズを指定して slides クラスを使うというのでもよいのだが、さらにスライドにおいてアニメーションを使いたい、背景画像を使いたいといったことになると、専用に設計されたクラスファイルを利用することになる。このようなクラスファイルには beamer や prosper がある ([1, 2])。本稿では prosper について説明する。作業環境としては UNIX、文字エンコーディングとして EUC-JP を想定しているが、Windows でもほとんどかわりはないはずである。

prosper は、講演用のスライドをつくるための LaTeX クラスである。講演用であるから、まず見栄えに注意が払われている。アニメーションもできるし、PStrick を使えば、図の中に TeX で書かれた数式を簡単に入れられるのも有難い。出力形式は pdf であり、電子文書の世界標準であるから、講演の現場には、特別な環境は必要ない。Adobe Reader の載った Windows 機とデータプロジェクタがあればよい。platex を使うことで当然日本語の利用もできる。

スライド (example.tex) を pdf ファイル (example.pdf) に変換するためには、TeX ソースから dvi ファイル、dvi ファイルから PostScript ファイル、PostScript ファイルから pdf ファイルと三回にわけて変換を行う。TeX から dvi ファイルに変換するにはいつも通りに platex コマンドを利用する。dvi ファイルから PostScript ファイルへは、dvips を、PostScript から pdf ファイルへは ps2pdf を利用する。ps2pdf は Ghostscript に付属するツールである。まとめると、コマンドラインから次のように打てばよい。

```
$ platex example.tex; platex example.tex  
$ dvips example.dvi  
$ ps2pdf example.ps
```

$$\text{example.tex} \xrightarrow{\text{platex}} \text{example.dvi} \xrightarrow{\text{dvips}} \text{example.ps} \xrightarrow{\text{ps2pdf}} \text{example.pdf}$$

dvi ファイルから pdf ファイルに直接に変換するプログラムとしては dvipdfmx が有名であるが、prosper を使用している場合には、非常に残念なことに dvipdfmx は使えない。これは prosper の実装方法に起因する問題である。

prosper の生成する pdf ファイルでは各ページの題目をキーとするブックマークが自動的に作成される。日本語を含む TeX ソースを用いて pdf ファイルを作成する場合には、日本語のブックマークが文字化けするという問題がある。これはブックマークをつくるためには特別な形式の文字列を Unicode¹ で埋め込まなければならないためである。この問題に対応するには、<http://www.rmatsumoto.org/tex-ps-pdf/hyperref.ja.html> から <http://www.rmatsumoto.org/tex-ps-pdf/convert-euc.txt> という perl スクリ

¹文字コードの国際規格の一つ。

プトをダウンロードして, tex のソースと同じディレクトリに置く. そして, 次の手順で pdf ファイルを作成すればよい.

```
$ platex example.tex; platex example.tex
$ dvips -f example.dvi| perl ./convert-euc.txt > example.ps
$ ps2pdf example.ps
```

prosper を用いたスライドの簡単な実例を示そう.

```
\documentclass[pdf,slideColor,colorBG,darkblue]{prosper}
\title{1 変数多項式の GCD}
\author{おはらかつよし}
\email{ohara@kanazawa-u.ac.jp}
\institution{金沢大学理学部}
\begin{document}
\maketitle

\begin{slide}{1 変数多項式の GCD}
\begin{itemize}
\item 1 変数多項式環  $\mathbf{k}[x]$  では割算ができる.
\item  $\color{yellow}\bf$  割算定理.  $0 \neq 0$  でない任意の
 $f, g \in \mathbf{k}[x]$  に対して,
\[\[
f = \color{red}qg + \color{red}r, \quad \deg(r) < \deg(g)
\]
\]
を満たす  $q, r \in \mathbf{k}[x]$  が存在する.
\item したがって,  $\mathbf{Z}$  の場合と同様にユークリッドアルゴリズムに
よって GCD が計算できる.
\end{itemize}
% ここに空行を入れておかないと表示が乱れる場合がある.
\end{slide}
\end{document}
```

例の最初の行では文書クラスに prosper を指定している. prosper を使う以上, これは必須である. 次に文書クラスに適切なオプションをつける必要がある. この例では, pdf, slideColor, colorBG, darkblue がオプションである. オプションのうち darkblue がスライドのスタイルである. スタイルは主に背景画像や各ページの余白の大きさなどに関係する. どのスタイルがインストールされているかはシステムによって異なるが, 例えば Knoppix/Math に含まれているスタイルは次の通りである. alcatel, alienglow, autumn, azure, blends, capsules, contemporain, corners, darkblue, default, frames, fyoma, gyoma, lignesbleues, mancini, nuancegris, prettybox, rico, serpaggi, thomasd, troispoints, whitecross, winter, wj.

スライドの背景に正しく画像を表示するには, 残りのオプションも指定しなければならないので注意すること. Knoppix/Math の場合, スタイル darkblue の実体は, /usr/share/texmf/tex/latex/prosper/PPRdarkblue.sty というファイルであり, 単にこのファイルが参照されているだけである. よって簡単にスタイルが作成できることがわかる. 中にはフリーでスタイルを配布しているサイトもある.

次に通常の TeX ソースと同様に必要ならばプリアンブルで, パッケージを指定する. maketitle は 1 ページ目にスライドの表紙を出力する. maketitle がない場合は本文からはじまる.

スライドの各ページは、slide 環境として実現される。slide 環境は 1 つの引数を取り、そのページの副題になる。また副題はそのまま、ブックマークとして扱われる。箇条書きにしたい場合は、itemize 環境が使える。

このソースを変換したものが、次の図である。



図 1: 表紙

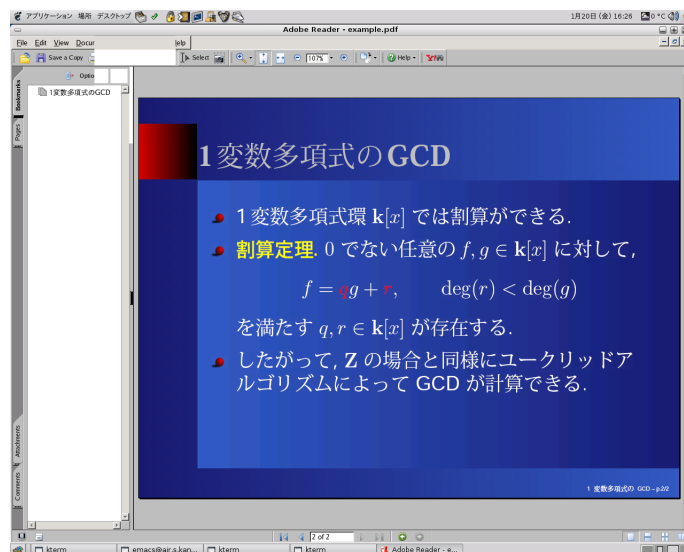


図 2: 本文

次にアニメーションを用いたスライドを作ってみよう。

講演では、話の進展にあわせてスライドを変化させたいときがある。これがアニメーションであるが、prosper では、スライドの overlays という概念で対応している。例えば、slide 環境が overlays で

```
\overlays{3}{
\begin{slide}{1 変数多項式の GCD}
\begin{itemize}
\item 1 変数多項式環  $\mathbf{k}[x]$  では割算ができる。
\fromSlide*{2}{
\item {\color{yellow}\bf 割算定理.}  $0 \neq 0$  でない任意の
 $f, g \in \mathbf{k}[x]$  に対して,
\[
f = \color{red}{q}g + \color{red}{r}, \quad \deg(r) < \deg(g)
\]
を満たす  $q, r \in \mathbf{k}[x]$  が存在する。
}
\fromSlide*{3}{
\item したがって,  $\mathbf{Z}$  の場合と同様にユークリッドアルゴリズムに
よって GCD が計算できる。
}
\end{itemize}
% ここに空行を入れておかないと表示が乱れる場合がある。
\end{slide}
}
```

と囲まれている場合、スライドが 3 段階に変化することを意味する。そのうち

```
\fromSlide*{2}{
...
}
```

で囲まれた部分は 2 段階目以降で現れる部分を意味する。

```
\onlySlide*{2}{
...
}
```

と書くと、スライドの 2 段階目のみで現れる部分になる。

overlays で便利な箇条書きの形式として itemstep 環境がある。この環境では、各条がスライドの各段階で一つずつ現れるので、タイプ量が減って便利である。先の例で示したようなスライドでは、次のように書いて非常に便利である。

```
\overlays{3}{
\begin{slide}{1 変数多項式の GCD}
\begin{itemstep}
\item 1 変数多項式環  $\mathbf{k}[x]$  では割算ができる。
\item {\color{yellow}\bf 割算定理.}  $0 \neq 0$  でない任意の
 $f, g \in \mathbf{k}[x]$  に対して,
```

```

\[
f = {\color{red}q}g + {\color{red}r}, \quad \deg(r) < \deg(g)
\]
を満たす  $q, r \in \mathbf{k}[x]$  が存在する.
\item したがって,  $\mathbf{Z}$  の場合と同様にユークリッドアルゴリズムに
よって GCD が計算できる.
\end{itemstep}
%% ここに空行を入れておかないと表示が乱れる場合がある.
\end{slide}
}

```

次にスライドに図を入れることを考えよう. これにはドローイングツールで描いた図を入れる場合と, TeX のソース中に自力で命令を書く場合とがある.

ドローイングツールで描いた図の場合は, TeX の `graphicx` パッケージと `includegraphics` 命令を用いて, 図を取り込めばよい. ただし図のファイル形式は `eps` 形式でなければならないので, `jpeg` や `png` の形式の図の場合には `bmeps` [3] などを用いて変換しておく. 図を取り込むときに注意しなければならないことは, 何も指定しなければ, 反時計回りに 90 度回転されて図が挿入されていることである. これは `prospcr` クラスの実装に起因する問題であるので, ユーザが注意して

```
\includegraphics[scale=0.3,angle=-90]{img.eps}
```

のように `angle` を指定するか, `rotatebox` を使う必要がある.

スライドと連携してアニメーションする図を入れたい場合や数式を含む図を入れる場合には, TeX のソース中に自力で命令を書くことになる. このとき `PStricks` パッケージと連携させることも可能である. `PStricks` の解説は本稿の手に余るので, ここでは三角関数のグラフを描く例を示すにとどめる.

`PStricks` に関する, 日本語のまとまった解説として [6] がある. また [4] から英語またはフランス語で書かれたマニュアルを入手できる.

```

\documentclass[pdf,slideColor,colorBG,darkblue]{prospcr}
\usepackage{pstricks}
\usepackage{pst-plot}
\usepackage{pst-math} %% プリアンプルに入れる

\begin{document}
\overlays{2}{
\begin{slide}{三角関数}
\begin{pspicture}*(-5,-2)(5,2)
\psset{linecolor=white}
\psaxes{->}(0,0)(-5,-2)(5,2)
\psplot[linecolor=yellow]{-5}{5}{x COS}
\onlySlide*{2}{
\psplot[linecolor=red]{-5}{5}{x SIN}
}
\end{pspicture}
\end{slide}
}
\end{document}

```

最後に, psutils [5] に含まれる psselect を用いて, prosper で作成したスライドの中から一部のページを抽出し, 別の pdf ファイルを作成する方法を説明する. 多くのスライドで, アニメーションのためにページ数が水増しされる. そのため, スライドを印刷して配布する場合には, 一部のページを抽出したサブセットの pdf ファイルを作成しておいたほうが有利である. その手順は以下の通りである.

まず完全な pdf ファイルを Adobe Reader などに表示し, 抽出すべきページのページ番号を控えておく. そのページを仮に 1,4-6,8,17-20,31,34-35 ページとしよう.

次に pdf ファイルを作成する手順を振り返ると, 途中で PostScript ファイルを経由したのであった. この PostScript ファイルを加工してから, pdf に変換する.

```
$ psselect -p1,4-6,8,17-20,31,34-35 example.ps subset.ps
$ ps2pdf subset.ps
```

新たにつくられた subset.pdf は抽出された pdf ファイルである. ただし, subset.pdf ではブックマークの情報は壊れているので注意する.

参考文献

- [1] The LaTeX Beamer Class Homepage, <http://latex-beamer.sourceforge.net/>
- [2] Prosper, <http://prosper.sourceforge.net/>
- [3] bmeps, <http://bmeps.sourceforge.net/>
- [4] The PSTricks web site, <http://tug.org/PSTricks/>
- [5] PSUtils, <http://gershwin.ens.fr/vdaniel/Doc-Locale/Outils-Gnu-Linux/PsUtils/>
- [6] M. Goossens 他, LaTeX グラフィックスコンパニオン, アスキー, 2000.

その他, ウェブ上には多数の文献が存在するので google で検索されることをお勧めする.