

ワードによる教材作成と $\text{T}_{\text{E}}\text{X}+\text{K}_{\text{E}}\text{Tpic}$ による教材作成

長野工業高等専門学校・一般科 前田 善文

Yoshifumi Maeda

Faculty of General Education,

Nagano National College of Technology

東邦大学・薬学部 高遠 節夫

Setsuo Takato

Faculty of Pharmaceutical Science,

Toho University

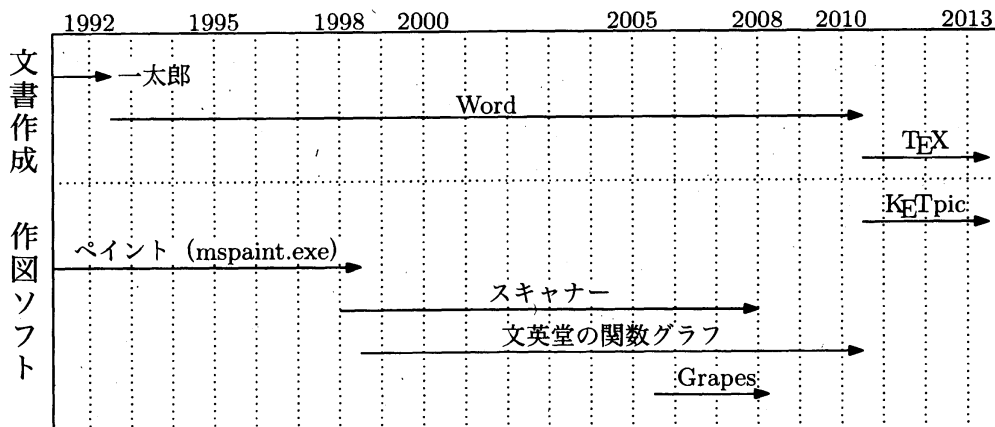
1 はじめに

配付(印刷)教材, 試験の模範解答などのプリントを作成するとき, 図やグラフ, 表を挿入することが必要になる場合が多い. ここでは, 図入り配付教材作成を主目的に考え, さらに, これに伴って付随的に作成された導入教材についても考える.

これまで, 個人的に配付教材のために使用してきたソフト等を紹介し, Word と Grapes 等のグラフ作成ツールを用いた配付教材と $\text{T}_{\text{E}}\text{X}$ と $\text{K}_{\text{E}}\text{Tpic}$ を用いて作成した配付教材との違いについて具体例を示し, 教育教材開発における $\text{K}_{\text{E}}\text{Tpic}$ の有用性について考察する.

2 配付教材作成に使用したソフト等

これまで, 個人的に配付教材作成のために使用したソフト等を表にまとめると下図になる. 初めの頃は, 数式作成は Word に付属したソフトを利用し, 図を使用することは少なかった. しかし, 図を使用しないと説明が困難な教材もあり, 学生が理解し易い効果的な教材作成のためにも適切な挿図教材が必要となった.

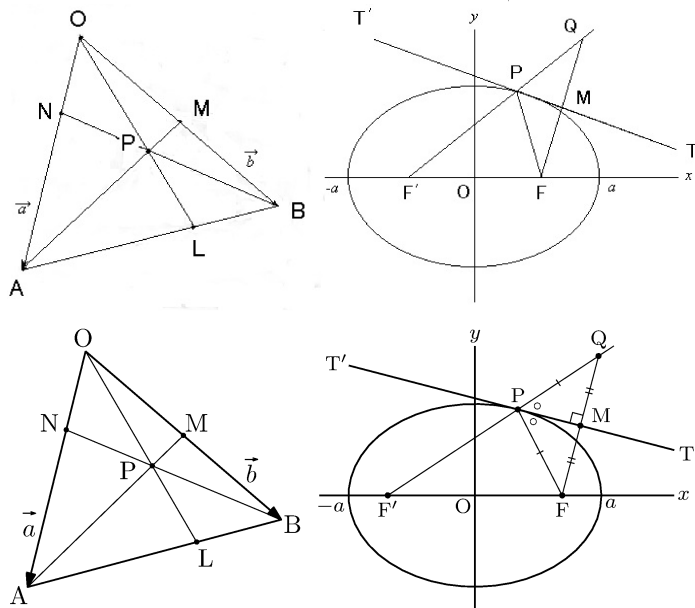


作図ツールとしては、フリーソフトまたは安価なソフトしか使用できなかったため、これまで Windows の mspaint や文英堂の関数グラフのようなソフトを使用することになった。複雑な平面図形や立体図については、これらのソフトでは作成することができなかつたため、教科書等の図をスキャナーで読み取り、それを加工して使用した。

2.1 Windows のペイント (mspaint) による教材作成について

右図の上段が mspaint で作成した図形で、下段が Scilab 版の K_E Tpic で作成した図形である。

mspaint は bitmap であるため、直線がスムーズではない。このため、図を大きく描いて縮小して張り付ける必要がある。また、mspaint では一度作図してしまふと図を変更することは困難である。左側の図について、比を変えた問題を作成すると図を新たに作り直さなければならない。 K_E Tpic の場合は初期値を変更し、表示する数値の位置を微調整するだけで同種の問題に

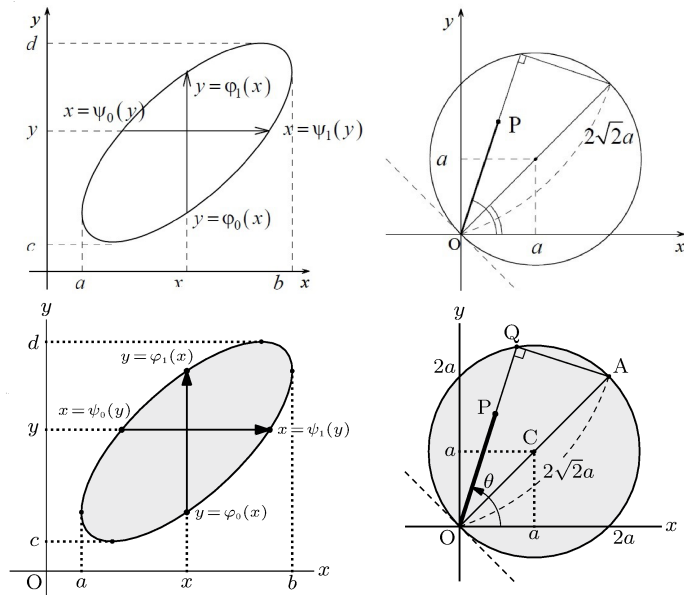


対応することができる。右側の図を比較すると、 K_E Tpic の方が正確で補助的な線や図形を簡単に入れることができるため、理解し易い図となっている。

2.2 文英堂の関数グラフによる教材作成について

上段が文英堂の関数グラフで作成した図形で、下段が K_E Tpic で作成した図形である。

文英堂の関数グラフは Word の中で簡単な操作で利用することができる、とても便利なツールである。しかし、適切な位置に数値や数式を書き入れることができない。図の数値や数式は、Word の Textbox を用いて張り付けてある。Word では位置の指定が難しく、適切な位置に微調整して数式等を張り付けることが困難である。 K_E Tpic では



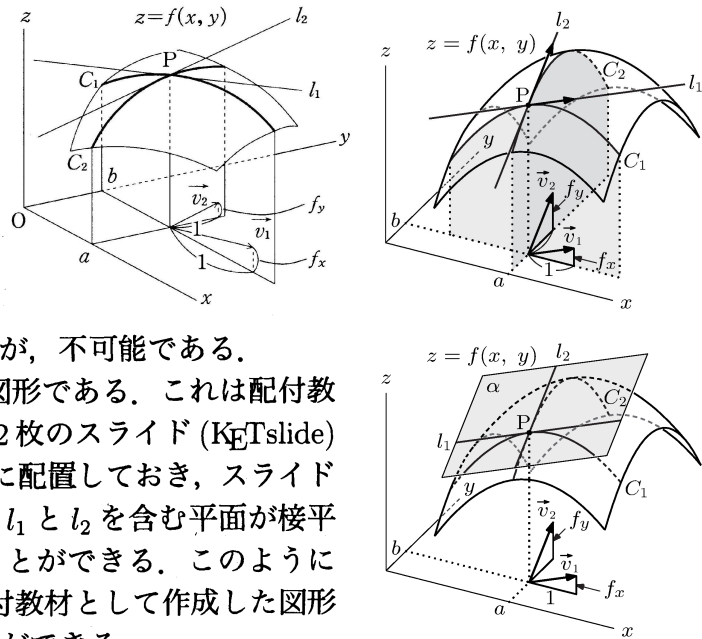
図作成プログラムの中で簡単に位置や文字の大きさ等を指定することができる。

文英堂のソフトは基本的に関数のグラフを作成するツールであるから、複雑な図形を描くことはできない。矢印でさえ描き入れることは難しい。また、使える関数の数に制限があるため、図は2枚の画面を重ね合わせて作成されている。

2.3 スキャナーによる教材作成について

上段左図が教科書の図をスキャナーで読み込み mspaint で修正した図である。スキャナーで読み込むと図の一部がかすれ、また、汚れが読み込まれることもあるため、修正が必要となる。この作業は時間がかかり大変である。説明する内容に合わせて図の一部を修正したいこともあるが、不可能である。

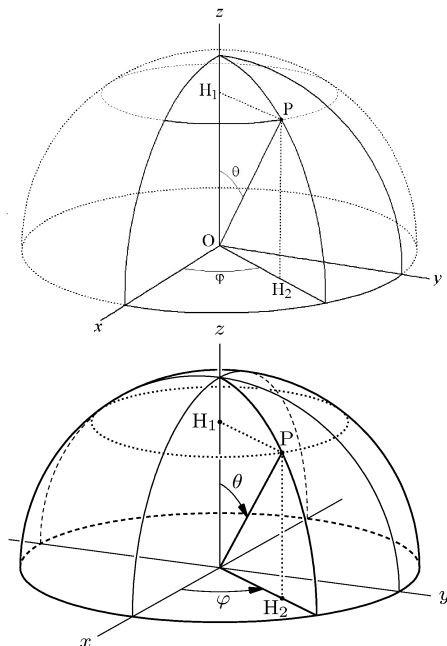
右図2枚は K_FTpic で作成した図形である。これは配付教材として使用するだけでなく、2枚のスライド (K_FTslide) を作成し、図の位置を正確に同じに配置しておき、スライドの画面を切り替えると2つの接線 l_1 と l_2 を含む平面が接平面であることを動的に理解することができる。このように K_FTpic と K_FTslide を用いると配付教材として作成した図形を導入教材としても使用することができる。



2.4 Grapes による教材作成について

図の上段が Grapes で作成した立体図 (3次元極座標の説明図) である。Grapes は様々なグラフや図を作成することができ、それを動的に扱うことができる。関数を定義すれば立体図も作成することができる優れたソフトである。しかし、配付教材として利用するためには画面を切り取った図を張り付けて使用することになる。したがって、大きく作成した図を切り取り、縮小して印刷しなければ図に荒さが目立ってしまう。

下段の K_FTpic で作成した立体図は、細かな点にまで配慮をしている。配付教材の立体図に欠かすことができない陰線処理やスケルトン処理を K_FTpic では簡単に実現することができる。これは Grapes 等のソフトでは難しい処理である。



3 K_FTpicによる教材への応用例

3.1 配付教材の例

右図は複素関数 $w = e^z$ について、 z と w の対応関係を z 平面と w 平面で表した図である。当初はスキャナーで読み取った図を使用していたが、K_FTpicを使うようになって、描き直したものである。

この図を宿題や試験で利用している。上段が問題で下段が解答に使用する図である。上段左図と下段の図は z 平面

の領域と w 平面の領域が対応している。スキャナーで読み込んだ図を使用しているときは、領域の区別を手で描き込んでいたが、K_FTpicでは、領域の位置指定に対して、自動的にその指定が z 平面と w 平面の領域に反映された図を作成するプログラムが可能である。これによって、問題と解答の作成が簡単になった。

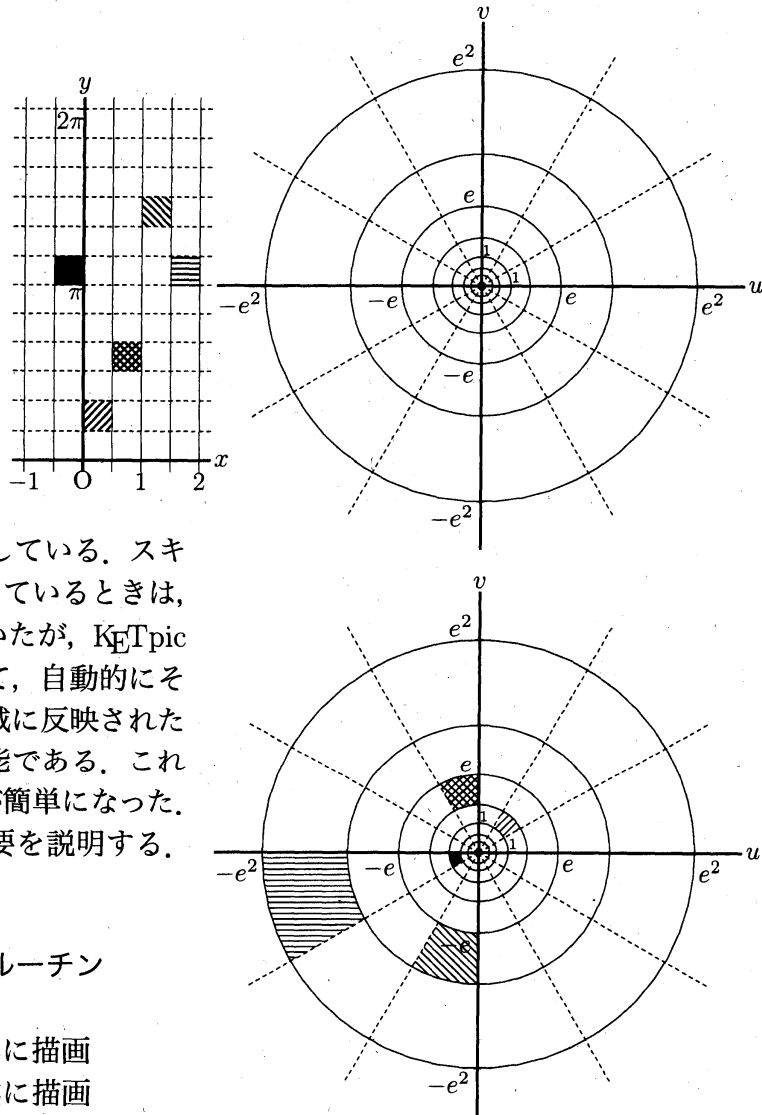
次に、このプログラムの概要を説明する。

プログラムの概要

※ハッチデータの作成と描画ルーチン

```
function Hatch(k, Lst)
//k=0 のとき z 平面の領域に描画
//k=1 のとき w 平面の領域に描画
//Lst は位置とハッチデータの種類のリスト
//Domz(i, j) は z 平面の領域, Domw(i, j) は w 平面の領域を表す。
i=Lst(1); j=Lst(2); st=Lst(3);
if k==0 then
    TmpD=Domz(i, j); TmpL=0.75; //TmpL は斜線の間幅: z 平面は狭く
else
    TmpD=Domw(i, j); TmpL=1.5; //TmpL は斜線の間幅: w 平面は広く
end

//st はハッチデータの種類
select st,
case "p" then //正の向きの斜線によるハッチデータの描画
    Drwline(Hatchdata(list('i'), list(TmpD), 45, TmpL));
```



```

case "m" then //負の向きの斜線によるハッチデータの描画
  Drwline(Hatchdata(list('i'), list(TmpD), 135, TmpL));
case "h" then //水平線によるハッチデータの描画
  Drwline(Hatchdata(list('i'), list(TmpD), 0, TmpL));
case "v" then //垂直線によるハッチデータの描画
  Drwline(Hatchdata(list('i'), list(TmpD), 90, TmpL));
case "w" then //正, 負の向きの斜線によるハッチデータの描画
  Drwline(Hatchdata(list('i'), list(TmpD), 45, TmpL));
  Drwline(Hatchdata(list('i'), list(TmpD), 135, TmpL));
case "s" then //Shade をかける
  Shade(TmpD, 0.7);
end
endfunction

```

※ハッチの場所と種類のリスト

```

Dlst=list();
// list(a, b, "*" ) a, b は位置の指定, * はハッチの種類
//  $\frac{a}{2} \leq x \leq \frac{a+1}{2}, \frac{\pi}{6}b \leq y \leq \frac{\pi}{6}(b+1)$ 
Dlst($+1)=list(0, 1, "p"); Dlst($+1)=list(-1, 6, "s"); Dlst($+1)=list(1, 3, "w");
Dlst($+1)=list(2, 8, "m"); Dlst($+1)=list(3, 6, "h");
☆このリストを変更することによって, 自由に領域とハッチデータの種類を指定できる.

```

※ for 文で z 平面に描き込み

```

for m=1:length(Dlst),
  Hatch(0, Dlst(m));
end

```

※ for 文で w 平面に描き込み

```

for m=1:length(Dlst),
  Hatch(1, Dlst(m));
end

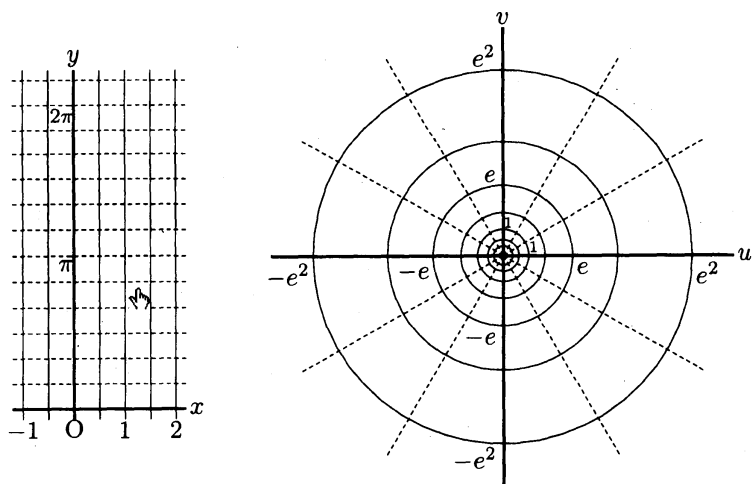
```

3.2 導入教材の例

サブセクション 3.1 で説明した z 平面と w 平面の対応関係を表した図を導入教材として利用するために, \TeX 文書に `hyperref` パッケージを利用して, 多数のリンクを張って使うことにした.

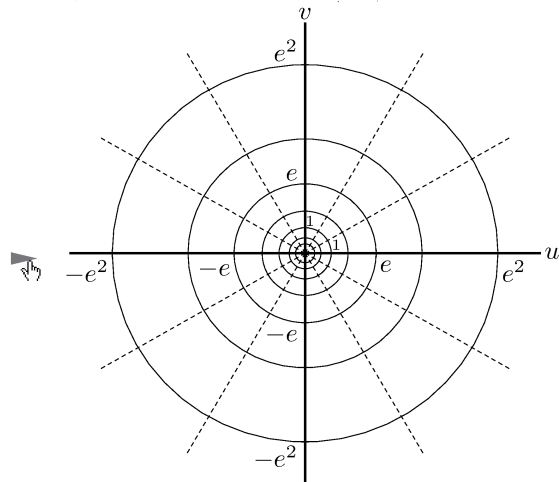
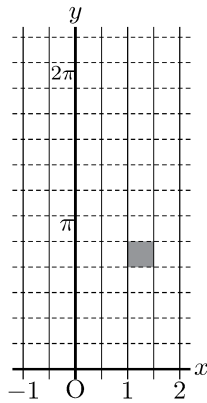
図は \KpTslide で作成した PDF スライドの 1 ページである. \KpTpic を使って \TeX 文書中に多数のページを作り, ページごとに多数の領域

に `hyperlink` と対応するページに `hypertarget` を張ったスライドを作成した.

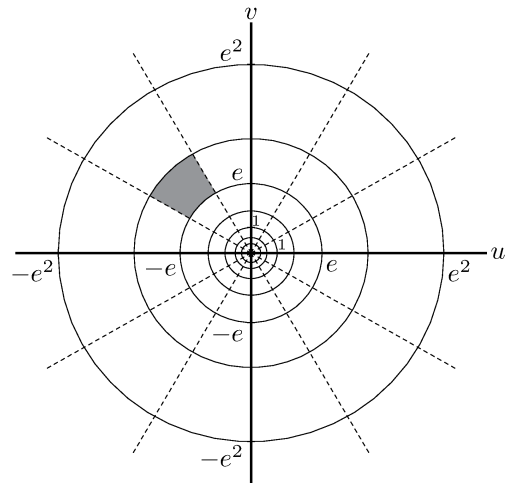
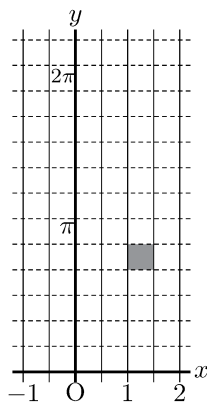


前ページの図が作成された PDF スライドの初期画面である。このスライドをどのように使用するかを手順を追って紹介する。

(1) PDF スライドの画面で z 平面上の領域にマウскарソルを持って行き、左クリックすると、その領域が選択されたことを識別できるように領域が色付けされ、 z 平面と w 平面の間に、対応関係の変換を促すための三角マークが表示される。



(2) この三角マークをクリックすると、 z 平面上の領域に対応する w 平面上の領域が色付けされる。

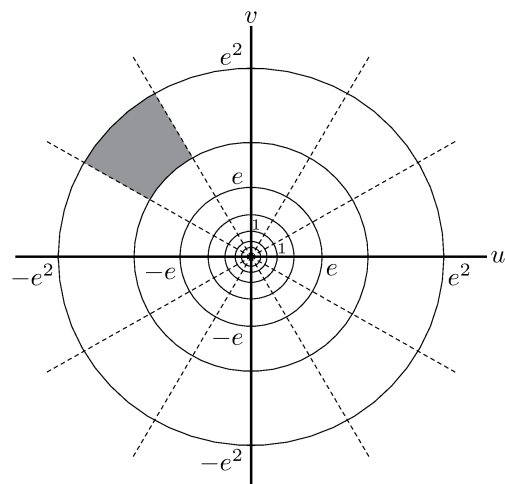
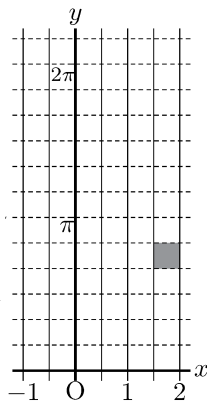


(3) z 平面上の領域にマウскарソルを持って行き、もう一度クリックすると、(1)(2)の操作を繰り返すことができるようになる。

説明をしながら、この操作を行うことによって、複素関数 $w = e^z$ の対応関係を学生に理解させることができる。

(4) z 平面の下に表示された4種類の三角マークをクリックすると、マークに対応して z 平面上の領域が動き、同時に対応する w 平面上の領域も動くように作成した。

この操作によって、 z 平面と w 平面の対応関係を、学生にさらに深く理解させることができると思われる。



この導入教材は157枚のスライドページで構成されている。 z 平面の78箇所にリンク (hyperlink) を張ってある。さらに、三角マークにもリンクが張られている。多いページでは82箇所になる。また、各ページにはリンク先のラベル (hypertarget) も張られている。このため、リンクを直接手入力して TeX 文書中に書き込むことは不可能である。

3.3 プログラムの概要

ここでは、 $\text{K}\epsilon\text{Tpic}$ と $\text{K}\epsilon\text{Tslide}$ を用いてリンクを多数使用した導入教材作成の Scilab 版 $\text{K}\epsilon\text{Tpic}$ のプログラムを紹介する。

- (1) z 平面に 78 箇所のリンクコマンドを張る関数

```
function Hlink()
  for K=3:-1:-2,
    for J=12:-1:0,
      Texcom("  $\forall$ putnotec{"+string(X+4*K) +"}{"+string(Y-4*J)...
              +"}{" $\forall$ hyperlink{L"+string(K)+string(J)+"}{8}}");
    end
  end
endfunction
```

【注】 \circ **Texcom** : $\text{T}\epsilon\text{X}$ 文書への書き込みコマンド ... () 内の $\text{T}\epsilon\text{X}$ コマンドの書き込み。

- \circ \forall **putnotec**{ x 座標 }{ y 座標 }{ 書き込み文字列 }
 - \circ (X, Y) : 基準ブロックに対応した座標 (1 ブロック 4mm)
この座標は $\text{T}\epsilon\text{X}$ 文書中の **layer** 環境での座標
 z 平面も 1 ブロック 4mm で作られ, **layer** により z 平面の位置と正確に対応
 - \circ \forall **hyperlink**{ リンク先ラベル }{ 表示テキスト }
- hyperlink** は文字列にリンクを張るため, 表示文字として 8 (白で表示) を使用

- (2) 79 箇所のリンクと 1 箇所にアンカーラベルを張ったスライドページを 78 枚作成する $\text{K}\epsilon\text{Tpic}$ プログラム

```
for k=3:-1:-2,
  for j=12:-1:0,
    Texcom("%*****"+string(k)+string(j));
    Texcom("  $\forall$ sameslide");
    Texcom("  $\forall$ hypertarget{L"+string(k)+string(j)+"}{")
    Texcom("  $\forall$ begin{layer}{130}{0}");
    Hlink(); //  $z$  平面 78 箇所にリンクコマンドを張る関数
    Texcom("  $\forall$ putnotese{"+string(Xz)+"}{"+string(Yz)...
              +"}{"  $\forall$ input{zu/hukusoouz"+string(k)+string(j)+".tex}}");
    Texcom("  $\forall$ putnotec{"+string(Xya)+"}{"+string(Yya)...
              +"}{"  $\forall$ hyperlink{W"+string(k)+string(j)+"}{"  $\forall$ Migi}}");
    Texcom("  $\forall$ putnotec{"+string(Xya)+"}{"+string(Yya)+"}{"  $\forall$ Migi}}");
    Texcom("  $\forall$ putnotese{"+string(Xw)+"}{"+string(Yw)
              +"}{"  $\forall$ input{zu/hukusozouw0.tex}}");
    Texcom("  $\forall$ end{layer}");
  end
end
```

- 【注】
- $\$sameslide$: K_ETslide において同タイトルでのスライドページの作成
 新たなタイトルでスライドページを作成するときは, $\$newslide$
 - $\$hypertarget\{\text{アンカーラベル}\}\{\text{表示テキスト (省略可)}\}$
 このプログラムでは, 表示テキストを省略
 - $\$begin\{layer\}$: K_ETpic の $layer$ の開始
 $\$putnotec$ 等のコマンドで図やテキストを座標を指定して, 正確な位置に張り付けるための $layer$ 環境の開始コマンド. $\$end\{layer\}$ が終了コマンド
 - $hukusozouw*_*_2$: 指定された領域を色付けした z 平面の図
 $*_*_2$ が領域に対応した数値
 - $hukusozouw0$: w 平面 (領域指定なし)
 領域指定した図は, $hukusozouw*_*_2$ であり, $*_*_2$ が z 平面と対応している.
 対応領域を色付けした w 平面を表示するプログラムについて
 - ① $hukusozouw0$ を $hukusozouw*_*_2$ に変更
 - ② $hypertarget$ のアンカーラベルを変更
 - ③ 三角マークなどの変更
 上記プログラムの部分的変更によって簡単に作成できる.
 - $(Xz, Yz), (Xw, Yw)$ はそれぞれ z 平面, w 平面の表示位置
 - ① z 平面の表示座標 (Xz, Yz) は, $Hlink()$ の中の座標 (X, Y) と対応させて定義している.
 - ② 157枚のスライドを作成するため, 位置指定は K_ETslide のページ作成プログラム中で図を表示する座標を定義している. はじめに, 1ページだけ作成してコンパイル後に表示させて微調整をする.
 - ③ この座標は, 図作成の K_ETpic プログラムとは別に定義されている.
 - $\$Migi$: z 平面と w 平面の間の三角マーク
 Xya, Yya は, この三角マークの表示位置

この導入教材の作成を可能にしたのは, K_ETpic プログラムと K_ETpic に付属する K_ETlayer , K_ETslide である. 可能にした理由は次の通りである.

- (1) K_ETpic によって, 正確な図を作成することができ, また, for 文を用いることによって, 1部 (指定領域の色付け) が異なる図を多数作成することができる.
- (2) K_ETslide のスライドページ (T_EX 文書) を K_ETpic プログラムで多数作成することができる.
- (3) K_ETlayer によって図やテキストを正確な位置に張り付けることができる.
 - ① 図中の領域ごとに, 図に対応する位置を指定して, T_EX 文書中に $hyperlink$ コマンドを張り付けることができる. これによって, 図にリンクが張られているかのように動作する.
 - ② 表示されているスライドページからリンク先のページにジャンプしたとき, 図の位置が微妙に違っていると, 画面が揺らいでしまう. この点, K_ETlayer ですべてのページで正確に同じ位置に図を置くことができるため, 図全体には変化がなく指定領域だけが色付けされたように動作する.

4 まとめ

長年の間、Wordを用いて配付教材を作成してきたが、1番時間をかけてきたことは、作図も含めて体裁を整えることであった。Wordでは字間や行間を調整することも大変であり、図を適切な位置に張り付けることも難しかった。また、Wordに張り付ける図を作成する描画ソフトに適当なものが見つからなかった。使用した描画ソフトのそれぞれに利点はあるが、配付教材用の図としての体裁など不足している部分があった。

TeXとKpTpicを使い始めて、時間の使い方が一変した。本質的な事柄に対して費やす時間が多くなった。思い通りの作図ができ、この図を発展させるとどのような教材ができるかと考えられるようになった。

次にKpTpicの利点のいくつかを挙げてみる。

- (1) 点や関数の名称などを図に描き込むとき、基準点を中心として適切な位置に書き込むための様々なオプションが用意されている。また、TeX書式をそのまま使うことができるため便利である。文字フォントもTeX本文と同じで印刷時に違和感がない。
- (2) 指定した領域にハッチを付けることも簡単にできる。
- (3) 立体図についても、作成のための様々なコマンドが用意され、立体図形には必須な陰線処理はもちろんのこと、スケルトン処理も簡単なコマンドで行うことができる。これらは配付教材作成において重要なことである。
- (4) KpTpicは指定された通りの正確な図を描くことができ、KpTpicのlayerを使用すれば、指定された通りの正確な位置に図などをTeX文書に張り付けることができる。
- (5) KpTpicのTexcomコマンドを使用すれば、TeX文書をKpTpicのプログラムで作成することもできる。
- (6) 配付教材のための図を1つ作成すれば、作図のプログラムに(4)(5)の利点を用いたプログラムを追加して、動画の作成や今回紹介した「指数関数の対応」のような配付教材と関連した導入のための教育教材を作成することができる。
- (7) 作られた教材がPDFであるため、学生に配付することができる。

上記はKpTpicの利点のほんの一部である。プログラムを工夫すれば、様々なことがKpTpicによって可能となると思われる。今後はさらに、この研究を進めていきたい。

参考文献

- [1] CASTEX 応用研究会：『KpTpicで楽々TeXグラフ』，イーテキスト研究所，2011.
- [2] 山下哲，高遠節夫，「KpTpicによる教材作成とSymbolic Thinking」，数理解析研究所考究録，Vol.1780，pp.72-82，2012年.
- [3] 前田善文，高遠節夫，「陰影を付けた立体図のKpTpicによる描画」，数理解析研究所考究録，Vol.1780，pp.154-159，2012年.