

教材作成における MePoTeX の利用

元一関工業高等専門学校 梅野善雄

March 6, 2021

(注) この資料は、当日の発表資料を微修正し、さらに口頭説明の部分を追加したものである。

(2021.03.18 公開, 2021.06.18 微修正)

■はじめに：自己紹介

- 一関高専で数学教員を定年退職し、今年度で非常勤講師も終了した。
- TeX との出会いは MS-DOS 時代から。その頃は、一晩かけてファイルをダウンロードして個別に解凍し、環境変数も自分で設定していた。
- TeX に関する情報はニフティの TeX フォーラムを利用した。
- 教材作成において、図形は主に「emath」を利用し、空間図形は「gnuplot」で作成した。
- 空間図形などは`\includegraphics` で取り込んで Ghostscript で処理するという生活を長年続けてきて、特に不自由はなかったが…
- 令和2年10月にPCのSSDがクラッシュして、TeX環境を再構築するはめになった。TeXインストーラで問題なくインストールはできたが、**【注】2021.6以降、このインストーラは利用不可となる。**
- DVIOUT で画像が表示されない！というトラブルがあった。
- 解決はできたが、その回復を図る中で、
 - もはや、時代はPDF出力で、いちいちEPSに直す必要はない。
 - Ghostscriptを経由しなくてもTeXが処理できるEPSが存在する。
- というようなことを知り、その流れの中で Metafont から派生した MetaPost や MePoTeX を知るに至る。

■ MePoTeX とは？

- Metafont では、フォントの形状を3次のベジエ曲線を利用して描画する。
- MetaPost は、その処理を図版生成で利用するために、J.D.Hobby氏により作成された。EPS, PNG, SVG形式のファイルを出力できる。
- MetaPost の出力する EPS ファイルは、Ghostscript を経由しなくても TeX で処理可能である。「Purified EPS」と呼ばれ、そのことを示すために拡張子は「MPS」が使用される。
- MetaPost での実際の処理は、下記の3段階構成になる。
 - MetaPost のソースコードを記述したテキストファイルを作成する。
 - DOS モードで、MetaPost により MPS ファイルを出力させる。
 - 生成された MPS ファイルを`\includegraphics` で読み込んでグラフを表示する。
- MePoTeX は「みなも」氏により作成されたマクロパッケージで、TeX のソースファイルに MePoTeX のコードを書き込んで、「MetaPost の呼び出し」「MPS ファイルの生成」「図版の組み込み」を行うマクロパッケージである。

■ MePoTeX の利用

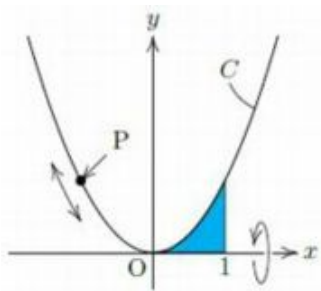
このパッケージは、次のような形で利用することになる。

- 最初に、`\usepackage{mepotex}`により MePoTeX を組み込む。

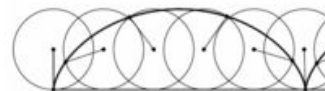
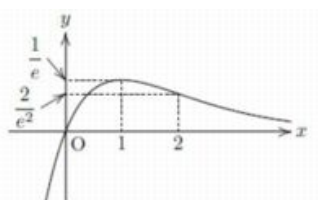
```
\documentclass{jsarticle}
\usepackage[dvipdfmx]{graphicx}
\usepackage{mepotex}
\begin{document}
...
\end{document}
```

- MePoTeX のマクロは「MPpic 環境」の中で利用する。具体的には、`\begin{MPpic}・・・\end{MPpic}`として MePoTeX のマクロを利用する。MPpic 環境の内部では、`\sendMP{・・・}`として、中括弧内に MetaPost のコードを記述することができる。
- MetaPost のコードをそのまま記述することもできるが、簡潔に利用できるようにマクロ化されている。

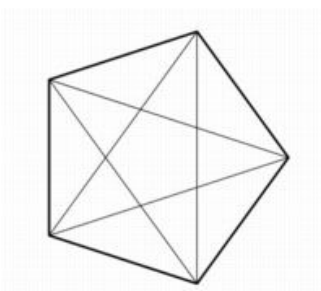
■ MePoTeX の出力サンプル (1) —マニュアルからの転載—



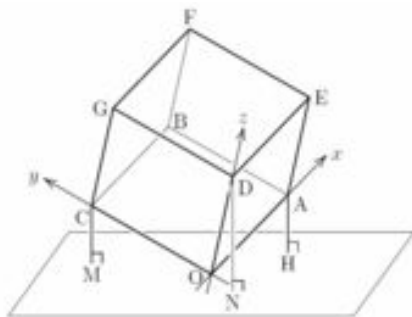
多彩な矢印



任意曲線を指定したマクロ



多角形マクロ



空間座標を射影

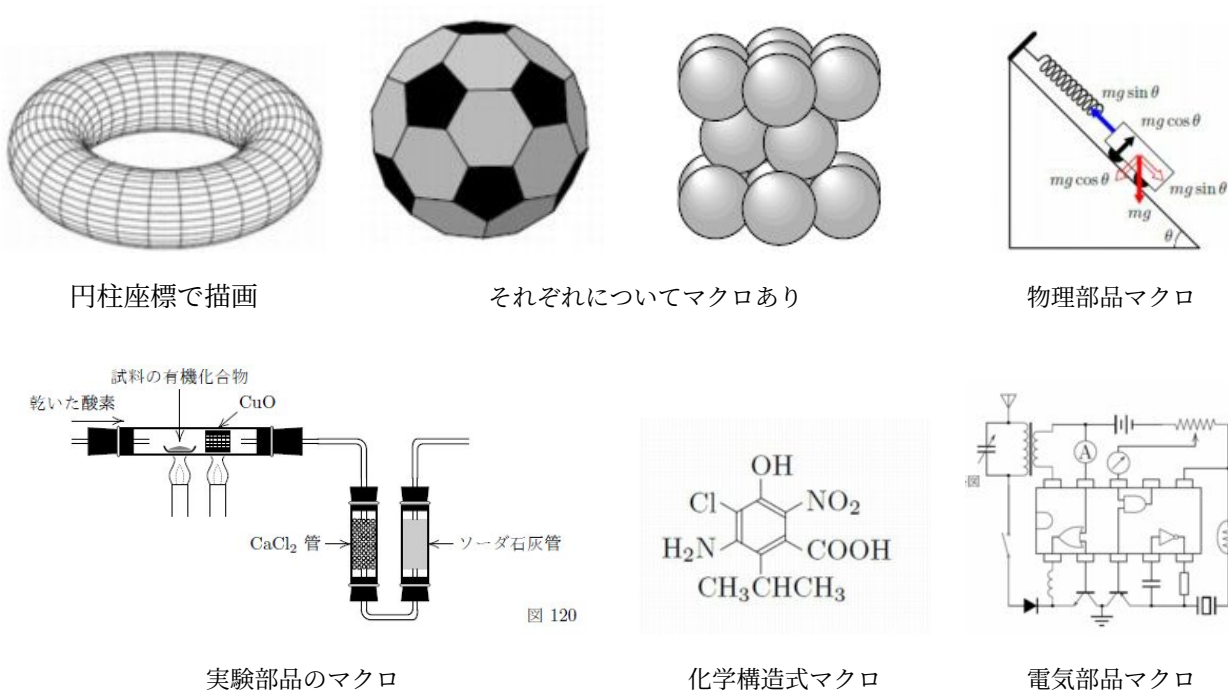


フラクタルマクロ

(解説)

- 指定点へのラベルは、矢印の有無、形状、伸ばす線の長さ、色等を一つのコマンドで指定できる。
- サイクロイドはマクロ化されており、任意曲線に沿って描画でき、途中点での円を描画することもできる。
- 多角形は角数と2点を指定すると描画され、頂点の座標は自動的に変数に割り振られて利用できる。
- 空間図形は、直交座標や円柱座標で指定して、それを平面に投影することで描画される。視点も自由に変更することができる。各種のフラクタルのマクロもある。

■ MePoTeX の出力サンプル (2) —マニュアルからの転載—



(解説)

- トーラスは円柱座標で定義されている。
- サッカーや結晶格子を描画するマクロも作成済みであり、形状や色等を指定するだけですむ。
- 物理、化学、電気の実験系の部品を描画するマクロが多数作成済みであり、その大きさや配置の仕方を指定するだけですむ。

■ MePoTeX のマクロ

MePoTeX を組み込むと、下記のマクロが自動的に読み込まれる。これらの大部分は、すでに 2003 年には作成済みである。他に、JIS 指定の和文の色名を定義した mp ファイルもある。

- MePoTeX.mp：描画・塗りつぶし等に関するマクロ 57 個。
- mptMath.mp：関数・図形・統計等の数学マクロ 71 個。
- mptPhys.mp：物理関係のマクロ 29 個。
- mptChem.mp：化学関係のマクロ 17 個。
- mpt3D.mp：立体表現に関する基礎マクロ 48 個。
- mptParts.mp：模式図の部品に関するマクロ 17 個。
- mpt3Dex.mp：立体図の部品に関するマクロ 59 個。
- elementary.mp：LOGO 風の動作を実現するマクロ 34 個。

■ MetaPost と MePoTeX のソースコード

MePoTeX は, MetaPost を $\text{T}_\text{E}\text{X}$ の中で利用しやすくしたマクロパッケージである. したがって, MePoTeX のマクロパッケージには, MetaPost を利用しやすくした MetaPost のマクロと, $\text{T}_\text{E}\text{X}$ 側のマクロとが混在している. 先頭に「\mp」がついているのは $\text{T}_\text{E}\text{X}$ 側のマクロである.

以下では, TeXWiki の「MetaPost」の解説の冒頭で紹介されているハートマークを描画するコードを, MetaPost と MePoTeX で比較する.



- MetaPost のコード

```
1: prologues := 3;
2: beginfig(1);
3: pickup pencircle scaled 2;
4: draw (0,40)..(10,50)..(20,40)..(8,20)..(0,0);
5: draw currentpicture xscaled -1;
6: endfig; end
```

(解説) ここでは, 次のような指定がなされている.

- 3: 「pickup」で描画する線を直径 2pt の黒丸で描くことを指定している.
- 4: 「draw」で, 各点をベジェー曲線で描く. ハートマークの右側が描画される.
- 5: 次の「draw」では, 描画された右側図形「currentpicture」の x 方向を -1 倍することで, 要するに縦軸に関して反転させた左側が描画される.

- MePoTeX のコード.

```
1: \begin{MPpic}<1cm>(1|1,2)
2: \sendMP{path hrt; z0=origin; z1=(0,h);
3: hrt=(0,40)..(10,50)..(20,40)..(8,20)..(0,0);
4: xdraw(2pt) hrt;
5: xdraw(2pt) hrt reflectedabout (z0,z1);}
6: \end{MPpic}
```

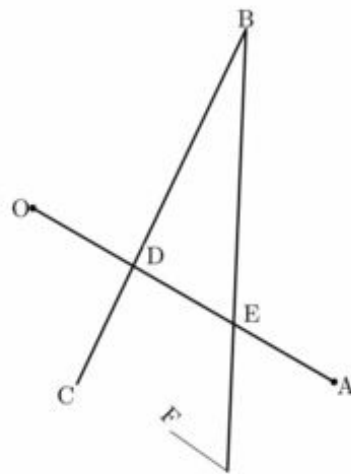
(解説)

- 1: 基本単位を 1cm に指定し, 範囲を $-1 \leq x \leq 1, 0 \leq y \leq 2$ に指定する.
- 2: 原点を z_0 , 点 $(0, 1)$ を z_1 として定義する. 「h」は, 縦方向の 1 単位 の長さのことである. 「path」では, 点を結んで得られる経路 (パス) を記録する変数として「hrt」を使用することを宣言している.
- 3: ハートマークの右側の点を結んで得られる経路を hrt とする. 「..」で結ぶと MetaPost と同じベジェー曲線で結ばれる. 「-」で結ぶと線分で結ばれる.
- 4: 「xdraw(2pt)」では, 太さ 2pt の線で定義した hrt を描画する.
- 5: 「hrt」を z_0 と z_1 を結ぶ直線に関して反転 (reflectedabout) させることで, 左側を描画する.

■連立 1 次方程式の解法

平面上の点の間に成り立つ連立方程式を与えると、その解として表される点を図形描画に利用できる。

```
1: \begin{MPpic}<1cm>(4,2|2)
2: \mptPoint{z.O}[O][r](3pt){origin}
3: \mptPoint{z.A}[A][tl]<0mm,1mm>(3pt){5w*dir-30}
4: \mptPoint{z.B}[B][b]{4w*dir40}
5: \sendMP{2*z.O+z.A=2*z.C+z.B=3*z.D;
6:   z.B+2*z.F=z.O+2*z.A=3*z.E;
7:   xdraw(1pt) z.O--z.A;xdraw(1pt) z.B--z.C;
8:   xdraw(1pt) z.B--z.F;}
9:   (中略)
10: \mptLabel{z.E}[bl]<1mm,0mm>{E}
11: \mptLabel{z.F}[br][dist=10mm*dir145,angle=-45]{F}
12: \end{MPpic}
```



(解説) この図は単に 3 本の線が交差しているだけであるが、それぞれ 1 : 2 の関係をもって交差している。具体的な点として座標を定めているのは点 O, A, B の 3 点だけであり、他の点 C, D, E, F はすべて連立 1 次方程式を解くことで得られている。MetaPost には、このような連立 1 次方程式を解く機能がある。

- 1: 単位と範囲 $0 \leq x \leq 4, -2 \leq y \leq 2$ を指定している。
- 2: 原点 (origin) を変数 $z.O$ に保存して、その箇所にラベル「O」を配置することを指定している。
- 3: 点 A の座標を $z.A$ に保存している。座標は (,) の形で指定することもできるが、ここでは、原点から伸びる長さ $5w$ の線分を -30° 回転させた点を A とし、その点に直径 3pt の黒丸を配置することを指定している。「w」は最初に指定した単位である 1cm をもとにした横幅の単位で、 $5w$ は 5cm のことである。
- 4: 点 B は、長さ $4w$ の線分を 40° 回転した点として定義する。
- 5: 線分 OA と線分 CD に対して、点 D は 1 : 2 の点であることを指定している。
- 6: 線分 AO と線分 FB に対して、点 E は 1 : 2 の点であることを指定している。
- 7: 方程式を記述すると、MetaPost はそれを解いて、それにより解として点 C, D, E, F の座標が定まる。それらの各点を $xdraw()$ で繋いでいる。
- 8: BF を描画する。
- 9: 各点のラベルを 10~11 行目と同様にして配置する。
- 10: $z.E$ に保存されている座標に「E」を配置する。「bl」は、「E」の下辺の左端が配置点にくるように指定している。ただ、線分に近すぎるので、さらに x 方向に 1mm ずらした点に配置している。
- 11: 同様にして $z.F$ の座標に「F」を配置している。特に意味はないが、ここでは水平方向から 145° の方向の、配置点 ($z.F$) から 10mm 離れた箇所に、「F」を -45° 傾けて配置するように指定している

■多彩なオプション指定

MetaPost は線分で結ぶにしてもいろいろな指定をすることができるが、その指定は一つずつコマンドで記述する必要がある。MePoTeX では、関連する指定は一つのコマンドにまとめて、その中のオプションで指定できるようにマクロ化されている。

- `xdraw()` の括弧内では多彩な指定をすることができる。
 - 「2pt」などとすると、線の太さを指定できる。
 - 「hasen(), tensen()」などで破線や点線の指定ができる。
 - 「red, blue」などで、色の指定ができる。
 - 「"-->」により、先端に矢印をつけることができる。
" <-->", " |-->", " |--*" など多数の記号がある。
- `\mptLabel{ 配置点 }[位置][key=値] <補正> { 文字列 }`

指定した点にどのような文字を配置するかを指定でき、「key」を利用すると多数のオプション指定ができる。下記はその一部である。

- 「dist」で配置点からの距離、「angle」では配置文字列の角度を指定することができる。
- 「arrowtype」「arrowlength」「arrowangle」などで、先端の矢印のタイプや、やじりの長さ、角度などを指定できる。
- 「linecolor」「linetype」で、線の色や線種を指定できる。
- 他にも多数のことを指定することができる。

■平面の基本図形

3 辺の比や 2 辺挟角を指定した三角形。4 辺の比を指定した内接四角形や正多角形マクロもある。

```
\begin{MPpic}
```

```
% 3 辺が 3:4:5 の三角形を定義
```

```
1: \sendMP{
```

```
2: \SetTriangle(z.A, z.B, z.C)(3,4,5);}
```

```
% 2 点を定義。もう 1 点は自動決定。
```

```
3: \mptPoint{z.A}[A][t]<0mm,-1mm>{origin}
```

```
4: \mptPoint{z.B}[B][t]<0mm,-1mm>{(4w,0)}
```

```
5: \mptPoint{z.C}[C][b]<0mm,1mm>{z.C}
```

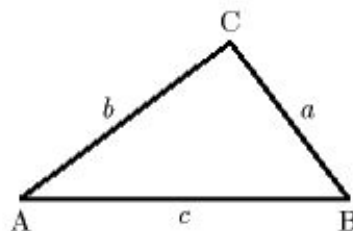
```
% 点のラベル (省略)
```

```
% 三角形を描画
```

```
\sendMP{
```

```
6: xdraw(1pt) z.A--z.B--z.C--cycle;}
```

```
\end{MPpic}}
```



2 行目で、3 辺の比を 3 : 4 : 5 とすることと頂点の座標を保存する変数を指定している。3 行目と 4 行目では 2 つの頂点を指定し、それによりもう一つの頂点が決まる。以上をもとにして、6 行目では各頂点を線で結んでいる。

■平面図形の変形

```
1: \begin{MPpic}<1cm>(4|1,3|2)
```

```
% 格子と軸を描画
```

```
2: (中略)
```

```
3: \sendMP{
```

```
4: path fig;
```

```
5: z0=origin; z1=(w,h);
```

```
6: fig=(0,0)--(w,0)
```

```
7: --(w,h)--cycle;
```

```
8: xfill(red) fig;
```

```
% いろいろな変形
```

```
9: xfill(green)
```

```
10: fig reflectedabout (z0,z1);
```

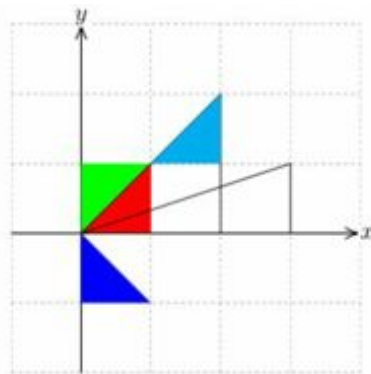
```
11: xfill(blue) fig rotated -90;
```

```
12: xdraw() fig xscaled 3;
```

```
13: xdraw() fig scaled 2;
```

```
14: xfill(Cyan) fig shifted z1;}
```

```
15: \end{MPpic}
```



MetaPost には、平面図形をアフィン変換する機能がある。ここでは、赤色の三角形に対していろいろな変換を行っている。

(解説)

1: 範囲を $-1 \leq x \leq 3, -2 \leq y \leq 3$ に指定する。

2: 格子や座標軸を描画する。

3: MetaPost に送り込むコマンドであることを宣言する。

4: 点を結んでできる経路をパスという。そのパスを保存する変数として「fig」を宣言する。

5: 原点を z_0 、点 $(1,1)$ を z_1 とする。

6: 赤色の三角形の周の経路 (パス) を「fig」として定義する。

8: 定義した「fig」の内部を赤 (red) で塗りつぶす。

9: 2点 z_0, z_1 を通る直線に関して反転した三角形を緑で塗りつぶす。

11: 「fig」を -90° 回転して青で塗りつぶす。

12: 「fig」を x 方向に 3 倍する (塗りつぶしはしていない)。

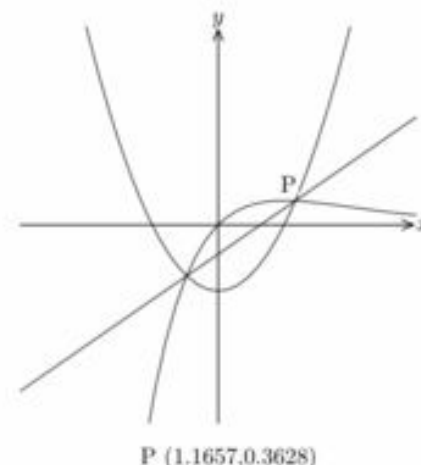
13: 「fig」を原点を中心に 3 倍する (塗りつぶしはしていない)。

14: 「fig」を点 $z_1 = (w, h)$ の箇所に平行移動してシアン色で塗りつぶす。なお、ここでは、 $w=h=1\text{cm}$ である。

1 行目で $\langle 1\text{cm}, 0.5\text{cm} \rangle$ と指定すると、 $w=1\text{cm}, h=0.5\text{cm}$ になる。

■平面グラフ

下記では、 $y = xe^{-x}$, $y = x^2 - 1$ のグラフの交点を通る直線を描いている。超越方程式の解が求められている。



```
1: \begin{MPpic}<1cm>(3|3,3|3)
% グラフの描画
2: \sendMP{path gra,grb;
3:   gra=kansu(t*exp(-t))(-2,4,40);
4:   grb=kansu(t*t-1)(-3,3,40);
5:   xdraw() ClipPATH(gra);
6:   xdraw() ClipPATH(grb);
7:   z.A=InSecPoint(t*t-1)(t*exp(-t))(-3,0,20);
8:   z.B=InSecPoint(t*t-1)(t*exp(-t))(0,3,20);
9:   xdraw() ExtLine(z.A,z.B);}
10: SavePointName(z.B);}
11: \mptLabel{z.B}[b]<-1mm,1mm>{P}
12: \mptLabel{(-w,-3.5h)}{P \UseNumeric{z.B}}
13: \end{MPpic}
```

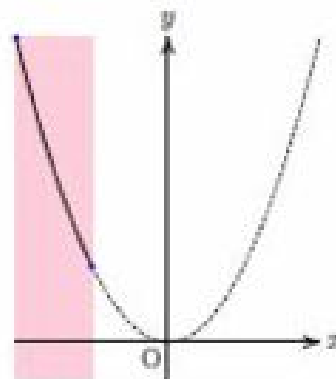
(解説)

- 1: 単位を 1cm として x, y の範囲を $-3 \sim 3$ として定義する。
- 2: 点を繋いでできる経路 (パス) を保存する変数として, gra, grb を宣言する。
- 3: 「 gra 」には xe^{-x} のグラフを保存する。それは, $-2 \leq x \leq 4$ の範囲を 40 分割して得られるグラフである。「 $kansu(\text{式})(\text{下限}, \text{上限}, \text{分割数})$ 」でグラフの経路が生成される。「 gra 」に保存されるだけでグラフは描画されない。「 $xdraw() kansu \cdot \cdot \cdot$ 」とするとグラフが描画される。なお, 1 変数関数のグラフ描画では独立変数は t を使用する。
- 4: 「 gra 」に $x^2 - 1$ のグラフを保存する。
- 5: 「 $xdraw$ 」により保存したグラフ「 gra 」を描画する。「 $ClipPATH$ 」により, 最初に指定した範囲から外れている部分をカットする。
- 6: 「 grb 」についても同様にする。
- 7: 2つのグラフの交点のうち, $3 \leq x \leq 0$ の範囲にある x 座標を $z.A$ とする。
- 8: 2つのグラフの交点のうち, $0 \leq x \leq 3$ の範囲にある x 座標を $z.B$ とする。
- 9: 2点 $z.A$ と $z.B$ を結ぶ直線を, 指定された範囲内に描く。「 $ExtLine$ 」は範囲内全体に, 「 $extLine$ 」は指定した端点よりちょっと伸ばして描画する。
- 10: 点 $z.B$ の座標を保存する。
- 11: 右側の交点に「 P 」を割り振る。
- 12: 保存した点 P の座標 $z.B$ を, 点 $(-w, -3.5h)$ の箇所に表示する。座標は単位付きで指定する必要がある。単位は 1 行目で定義し, 横幅「 w 」や縦幅「 h 」は, この例ではいずれも 1cm である。

■パラパラ漫画

「MPpara」環境を利用すると、「パラパラ漫画」風のPDFを生成することができる。下記は、MePoTeXにあるサンプルである（「parapara」フォルダにある）。放物線の左端の桃色の長方形が右側に動いていく。

```
1: \mptparapara{20}
2: \begin{MPpic}<10mm>(2|2,4.4|0.5)(0,0)
3: \mptgetkoma{n}
4: \sendMP{numeric a; a:=(n/20)[-2,1];}
5: \sendMP{
6:   xfill(hsb(0.95,0.2,1))
7:   (a*w,-0.5h)--(a*w,4h)--((a+1)*w,4h)
8:     --((a+1)*w,-0.5h)--cycle;}
9: \sendMP{
10:  xdraw(0.4pt,hasen()) kansu(t*t)(-2,2,100);
11:  xdraw(1pt) kansu(t*t)(a,a+1,25);
12:  for xx=a,a+1:
13:    xtanten((xx*w,xx*xx*h))(1.5pt,blue);
14:  endfor}
15: \end{MPpic}
16: \repeat
```



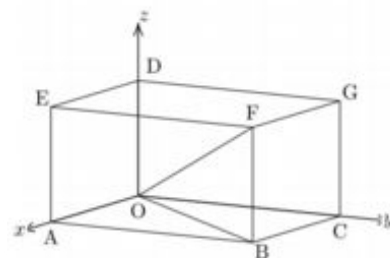
(解説)

「パラパラ漫画」は、画像を次々に捲っていくことで、あたかもアニメーションのように見える機能である。MePoTeXでは、そのように見えるPDFを簡単に作成することができる。上記のコードで生成されるPDFで「→」を押し続けていると、ピンクの長方形が右側に移っていく。作成方法は2種類用意されており、上記はMPpic環境を「\mptparapara」と「\repeat」で囲う場合である。

- 1: MPpic環境を、\mptparaparaと\repeatで囲って、パラパラ漫画を作成することを宣言する。 $\{n\}$ の n はコマ数であるが、 n は0～20と変化するので、実際には21ページになる。
- 3: 最初に、\mptgetkomaにより、コマ数の値を受け取る。
- 4: 「numeric a」では、数値変数 a を宣言し、区間 $[-2, 1]$ を20等分した n 番目の値として a を定義する。
- 6: 6行目では、閉曲線の内部をHSB形式で指定したピンク色のコードで塗りつぶすことを宣言する。7～8行では、前の行で指定した a を元に長方形の左下から右回りに長方形を描画する。6行目の指定により、内部がピンク色に塗りつぶされる。
- 10: 放物線の $[-2, 2]$ の範囲を、太さ0.4ptの破線で描画する。
- 11: 任意コマ目の、長方形の内部にある放物線を実線で描画する。
- 12: 繰り返し変数「xx」が a と $a+1$ の場合に13行目を繰り返す。ピンク色の内部にある放物線の端点が直径1.5ptの青丸になる。

■空間座標

MePoTeX は空間図形を描画することができる。それは、空間図形を空間座標 (x, y, z) を利用して定義し、それを平面に射影 (proj) して、射影された点を「xdraw」で結ぶことで描画される。座標系は直交座標と円柱座標が利用できる。円柱座標は度数法では「rdh(r,d,h)」、弧度法では「rrh(r,r,h)」とする。媒介変数が利用できるので、球座標も利用することができる。



```
\begin{MPpic}<1cm>(4,4)
% 座標軸と頂点
1: \mptXTaxis|0w==4w>[r]<1mm,0mm>{\$x\$}
2: \mptYTaxis|0w==5w>[r]<1mm,0mm>{\$y\$}
3: \mptZTaxis|0w==3w>[b]<1mm,0mm>{\$z\$}
4: \mptPoint{z.O}[O][t]<0mm,-1mm>{\proj(0,0,0)}
5: \mptPoint{z.P}[P][b]<0mm,1mm>{\proj(3w,4w,2w)}
  (中略)
% 直方体の描画
6: \sendMP{xdraw() z.O--z.F--z.B--cycle;
7:   xdraw() z.D--z.E--z.F--z.G--cycle;
8:   xdraw() z.A--z.B--z.C;
10  xdraw() z.A--z.E; xdraw() z.C--z.G;}
\end{MPpic}
```

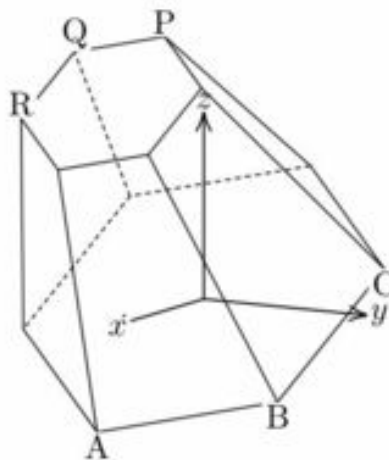
(解説)

- 1: 空間内で x 軸を描画する。
- 2: 空間内で y 軸を描画する。
- 3: 空間内で z 軸を描画する。
- 4: 点 $(0,0,0)$ を投影した平面上の点を $z.O$ として、その点に「O」を置く。
- 5: 点 $(3,4,2)$ を投影した平面上の点を $z.F$ として、その点に「F」を置く。
- 6: 10 行目までにかけて、投影された平面上の点を「xdraw()」を利用して繋いで直方体と対角線を描画する。
 - 空間の中で直交座標で定義した点を「proj」で平面に射影し、射影された点を「xdraw()」で結んでいくだけで立体図形が描画できる。
 - デフォルトの視点は「SetProjDir(10,30)(blue);」で設定されている、「blue」(青)は色の指定ではない。「blue」のRGBコードを0~1の数にすると $(0,0,1)$ であり、「SetProjDir(10,30)(blue);」はベクトル $(0,0,1)$ で定める方向を上向きとすることを指定している。 $(10, 30)$ は、投影する平面から観測者に向かう向きを(緯度, 経度)で指定している。「SetProjDir」の指定を\sendMP内に置けば、観測者に向かう向き(視点)を自由に変更することができる。

■空間の基本図形

円錐・角錐や円錐台・角錐台を描画して自由に回転することができる。下図は、角錐台を y 軸の周りに -30 度回転したものである。

```
1: \begin{MPpic}<5mm>(4|4,4|1)
2: \sendMP{
3:   SetProjDir(20,40)(blue);
4:   SetPolyFrame;
5:   z0=(0,0);
6:   mptColumnCone(z0)
7:   ((4w,2w),6h,6)(-30,0);}
8: \mptLabel{z0b0}[t]{A}
9: \mptLabel{z0b1}[t]{B}
10: \mptLabel{z0b2}[t]{C}
11: \mptLabel{z0t3}[b]{P}
12:   (中略)
13: \mptZTaxis|==4w>{\$z\$}
14: \end{MPpic}
```



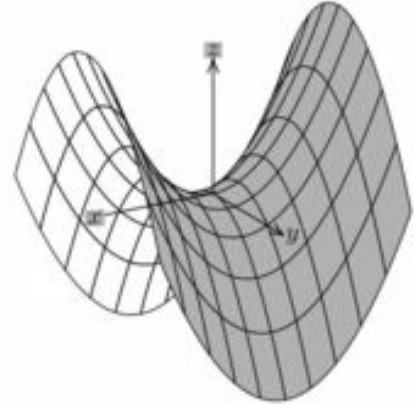
(解説)

- 3: 視点を変更している。
- 4: 枠線で描画することを指定する。表面や裏面だけや、両面の指定をすることもできる。
- 6: 「mptColumnCone」が、円錐・角錐や、それらの台を描画するマクロである。「((4w, 2w), 6h, 6)」では、底面の半径を 4、上面の半径を 2、そして高さを 6 とした六角錐台であることを指定している。最後の角数に負数を指定すると、円錐や円錐台になる。後半の「(-30,0)」では、 y 軸や z 軸の回りの回転角を指定する。ここでは、 y 軸の回りに -30 度回転することを指定している。
- 8: 角錐や角錐台を描画すると、その頂点の座標が、自動的に変数に割り振られる。 $z0$ を中心とした場合は、底面の頂点は $z0b0, z0b1, \dots$ に、上面の頂点は $z0t0, z0t1, \dots$ に割り振られている。ここでは、その頂点を表す変数を利用して、底面や上面の 3 つの頂点にラベルを割り振っている。

■ 曲面

空間曲線は「kansuQ」を利用して「kansu」と同様に描画できる。曲面は「drawsurface」による。下図は、 $z = x^2 - y^2$ のグラフである。独立変数は s, t を使用する。 x, y, z を媒介変数表示すれば、多彩な曲面を描画できる。「blue=(0,0,1)」は、色の指定ではなく、 z 軸方向を上方向として描画することを指定している。

```
1: \begin{MPpic}<.5cm>(4|4,4|4)
2: \sendMP{
3:   setunitlength(w,w,0.3w);
4:   SetProjDir(20,60)(blue);
5:   SetPolyBoth;
6:   background:=0.7*white;
7:   drawSurface(s,t,s*s-t*t)
8:     (-4,4,10)(-4,4,10);}
9: \mptXTaxis|0w==4w><1mm,0mm>{\$x\$}
10: \mptYTaxis|0w==4w><1mm,0mm>{\$y\$}
11: \mptZTaxis|0w==4w><0mm,1mm>{\$z\$}
12: \end{MPpic}
```



(解説)

- 3: 「setunitlength」により、3つの方向の基本単位を指定できる。
- 4: 視点を、緯度 20 度、経度 60 度の方向から眺めるように指定している。
- 5: 「SetPolyBoth」で、表面と裏面を描画するように指定している。
- 6: 表面の色を、グレー色「0.7*white」にするように指定している。「a*white」で、数 a は 0~1 の値で、0 では黒、1 は白であり、中間の値を指定することでグレーの強弱を指定できる。色は、「white」だけでなく自由に変更できる。「:=」により、右辺の値を左辺に代入することになる。「background」にはすでにデフォルトの値が入っていて、それを変更することになるので「:=」とする必要がある。値を変更する必要のない変数を利用するときは「=」でかまわない。
- 7: 「drawSurface」では、独立変数 s, t を利用した式を記述し、それぞれの範囲や分割数を指定する。媒介変数利用になるので、トーラスなどの多彩な曲面も描画できる。ただし、複数曲面の隠線処理による同時描画にまでは対応していない。
- 9: \mptXTaxis 等では、空間内の座標軸を描画する。

■まとめ：教材作成での「MePoTeX」の利用

- 「MePoTeX」で Web 検索してもほとんど利用例が表示されないので、あまり利用されていないように思えるが、実際に使ってみると以下のことが可能である。
 - MePoTeX パッケージをインストールするだけで MetaPost を簡単に利用でき、Ghostscript を経由しない EPS ファイルを生成できる。
 - 点を、連立方程式の解として定義できる。
 - 基本図形に関する多数のマクロが用意されているので、図形を利用した教材作成が容易である。図形の変形も容易にできる。
 - 整数でなくてもグラフの交点を求めることができる。
 - アフィン変換やパラパラ漫画を作成するマクロもある。
 - 「emath」のノリで、立体図形や媒介変数を利用した曲面まで描画できる。

★ 「emath」 + 「MePoTeX」があれば、通常の教材作成には十分なように思える。

★ MePoTeX は、もっと広く利用されて良いのではないだろうか！

これまで利用されてこなかったのは何故なのだろうか？

- 改善希望点としては、
 - MPpic 環境があるごとに、多数の mps, sft, sav ファイルが生成される。ファイルの整理機能があるとよい。
- 「MePoTeX」の作者である「みなも」氏には、多大の敬意を表します。
今更ですが、できれば在職中に「MePoTeX」を知りたかった…

■補遺：数ナビの部屋 <http://yunavi.la.coocan.jp/>

グラフ電卓絡みで開設した Web サイトであるが、数学関係のページが増えてきて、現在は下記のようなページで構成される。関心を持たれた方はご笑覧いただきたい。QR コードは、新たに作成した「MePoTeX による図形グラフの作成」と題して、MePoTeX の利用法を解説するページである。

- グラフ電卓に関する解説。
- いろいろなリンク集。
(数学学習、ベキ分布、悩みごと等)。
世の中は「ベキ分布」だらけのようです…
- Maxima に関する解説。
- R に関する解説。
- TeX に関する情報。
- emath に関する解説。
- gnuplot に関する解説。
- MePoTeX に関する解説。

<http://yunavi.la.coocan.jp/mepotex.html>

