グラフの描画

金沢光則

平成 19 年 12 月 1 日

概 要

kpic.sty を使うように書き直した schlgraph2.sty ver1.01 を使ってグラフを描く方法をまとめている。昔のものではできたが、この方法でできなかった「領域の斜線」を引くことができるようになった。また、パラメータによる曲線の描画や、空間曲線の描画にも対応した。具体的な例を挙げているので参考にして欲しい。

1　新しいグラフの描き方

kpic.sty で定義されている座標上で、グラフを描く方法を与える。
昔のものとの一番大きな違いは、係数を 100 倍せずにそのまま使うことである。拡大縮小は、unitlength を変更することによる。
古い方法と比べた場合の advantage は次の通りである。

- 計算して点をプロットしているので、ずれない。張り合わせる必要がない。
- ユーザー定義関数が簡単に作れる。
- 作成した関数のグラフを簡単に描くことができる。
- 方程式を解くことができる。

使える要素関数を組み合わせて、ユーザー定義関数を作ることができる機能は優れものである。数行の定義で足りる。新しく作ったこの関数は、要素関数として別の関数定義に使うことができる。
関数要素に対して、描画関数を適用すれば、そのグラフを描くことが容易にできる。その場合、平行移動を指定することができる。

— 最近の更新 —

2007.11.26	`\Fhatch`	領域に斜線を引く
2007.11.30	`\PDraw`	パラメータによる平面曲線の描画
2007.11.30	`\PSDraw`	パラメータによる空間曲線の描画
2007.12.01	`\FDrawP`	極座標による平面曲線の描画

目 次

2　全般について

2.1　線の太さを変える

太くする場合は`\thicklines`を、細くする場合は`\thinlines`を書き込む。

2.2　大きさを変える

格子の大きさは`\unitlength=5mm`などで与える。これを変えればよい。単位は mm や cm , in , pt などが使える。
格子を表示するときは、環境に Dot がついているものを使うが、格子を消したい場合は、Dot を消せ (Dot の無い環境を使う) ばよい。

2.3　原点の位置を変える

原点を中央に置くときは、環境に C がついているものを使う。ずらしたい場合は、`\begin{AxesDot}(8,8)(-1,-1)`などとすればよい。
２つ目の括弧が、原点の位置を指定する。

2.4　点を打つ

`\FQuartic*(1,2,3,4,5)`は４次関数のグラフを描く関数であるが、最初に * を付けると、プロットした点に●を打つ。この点を折れ線で結んでグラフを描いている。
他の関数も同じである。

2.5　範囲を制限する

`\FQuartic[-1,3][-2,2](0,0,1,2,3)`は４次関数を使って２次関数のグラフ $y = 0x^4 + 0x^3 + 1x^2 + 2x + 3$ を描く。最初の [ ] で、x 座標を制限する。つまり定義域を指定する。
次の [ ] で、y 座標を制限する。x 座標の制限だけを指定することはできるが、y 座標だけを制限することはできない。
省略された場合、表示されている枠全体にグラフを描く。
他の関数も同じである。

2.6　細かく点を打つ

標準で x 軸方向に 0.1 だけずらして点をプロットしている。曲線が縦に動くとき、きれいな曲線にならないときがある。この幅は`\Fdx`に保存されているので、この値を変更すれば、もっと細かく点を取ることができる。`\def\Fdx{0.01}`と書き込めば、10 倍の細かさで点を打つことになる。この変更は以降も有効になるので、必要な場面が終われば元に戻すべきである。その場だけで有効にしたい場合は、`{`, `}`で囲むと良い。

2.7 方程式を解く

単純な関数 $f(x)$, $g(x)$ について、方程式 $f(x)=g(x)$ を解くことができる。関数を `\Fn#1#2`, `\Gn#1#2` とするとき、`\FSolve(\Fn,\Gn)[1,2]{\va}` とすると、定義域 [1,2] の間にある交点の x 座標が`\va` に入る。x 軸との交点の場合は、`\FSolve(\Fn,0)[1,2]{\va}`とする。`Knode` を使えば、交点に点を打ち処理することができる。【注意】

- 関数はその区間で連続であること。
- 関数値は、端点で符号が逆であること
- 関数値が 0 になる点が複数ある場合は、どれが出るか、あるいはでないか不確定である。

2.8 ２点を通る直線を描く

schlfigure.sty schlgraph2.sty を用いて、定義された２点を通る直線を枠一杯に引くことができる。`\EKLine{PQ}` で、２点 P,Q を通る直線を描く。定義は、schlfigure.sty にある。引数には、複数の点のセットを与えることができる。その場合は「,」で区切ること。

2.9 自作関数の使い回し

ユーザー定義関数の定義は、`\begin{documant}`と`\end{document}`の間に入れなくてもかまいません。プリアンブルすなわち、`\begin{document}`の前にいれても、他のスタイルファイルに入れて、`\usepackage{}` で読み込んでもかまいません。常に使うのなら、プリアンブルに入れて、そのファイルをコピーして使えば、いつでも自作の関数定義を使うことができます。また、使うときが限定されていれば、自分で sty ファイルを作って、その中に自作の定義を入れておき、必要になったら、`\usepackage{}`で読み込むようにすれば良いでしょう。

2.10 領域に斜線を入れる 2007.11.26

区間 $[a,b]$ において、2 つの関数 $f(x)$, $g(x)$ $\ (f(x) \leqq g(x))$ で挟まれた部分の領域に斜線を入れたいときは、`\Fhatch[d,r](a,b)(f,g)` とする。ここで、`[d,r]` は省略可能である。d は斜線と斜線の間隔を表す。省略されると、0.5 となる。r は斜線の傾きを表す。マイナスも指定できる。省略されると 1 となる。
現在の問題点は、斜線が曲線とくっついていないことである。
これに対応するために、`\Fdx` を 0.01 として計算している。そのため、計算に時間がかかる。きちんと境界を判断するようにすれば、`\Fdx` を 0.1 に戻すこともでき、境界まで線を書くことも、一定の距離だけ離すことも可能になのだが、やってみると、x 座標の増分が小さい分だけ難しいようだ。
f と g の大小を確認しているので、実は斜線を引く x の範囲を厳密には気にしなくても、ちょうど良い範囲の斜線を引いてくれる。

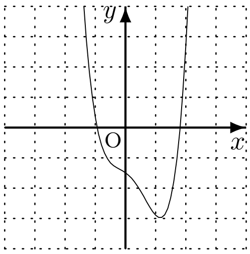
2.11 今後の予定

- 点の指定
- 媒介変数による描画
- 領域に網掛けをする
- 曲線を点線で描く

3 組み込み関数、機能

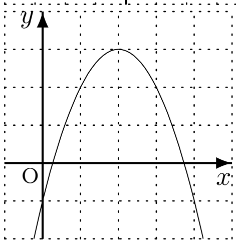
3.1 ４次関数

```
\unitlength=5mm%
\begin{AxesDotC}(8,8)%
\FQuartic(1,-0.9,-0.8,-0.7,-1.5)
\end{AxesDotC}%
```



$y = x^4 - 0.9x^3 - 0.8x^2 - 0.7x - 1.5$ のグラフである。
１次関数から３次関数までは、適当な係数値を 0 にすれば良い。

```
\unitlength=5mm%
\begin{AxesDot}(6,6)(-1,-2)%
\FQuartic(0,0,-1,4,-1)
\end{AxesDot}%
```



$y = -x^2 + 4x - 1$ のグラフである。

3.2 定数

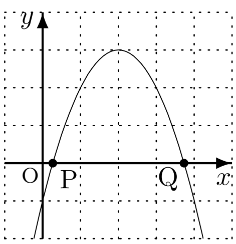
\FCnstPI= 3.14159265
\FCnstE= 2.71828183
を定義している。

3.3 交点を計算する

$y = \sin x$ と $y = \cos x$ のグラフの交点を計算し、その交点に点を打つ。例２で交点の x 座標を求めている部分は、\FSolve(\FESin,\FECos)[0,1]{\va}である。
２つの関数 \FESin, \FECos の交点の x 座標を、区間 $[0,1]$ の範囲で求め、\va に入れる。交点の y 座標は、関数を使って計算すればよい。
x 軸との交点は、２つ目の関数として 0 を与え計算する。例１の \FSolve(\UsrFn,0)[0,1]{\va} が点 P の x 座標を求めるものである。

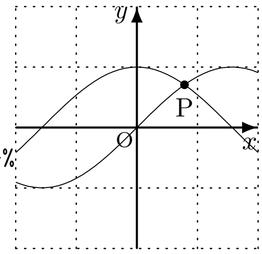
3.3.1 例 1 $y = -x^2 + 4x - 1$ と x 軸の交点

```
% 関数の定義
\def\UsrFn#1#2{% y=x(4-x)-1
\Sub{4}{#1}{#2}%
\Mul{#1}{#2}{#2}%
\Sub{#2}{1}{#2}}%
% 描画と計算
\unitlength=5mm%
\begin{AxesDot}(6,6)(-1,-2)%
\FDraw{\UsrFn}%
\FSolve(\UsrFn,0)[0,1]{\va}%
\Knode*(\va,0){P}[\KSame][rb]%
\FSolve(\UsrFn,0)[3,4]{\va}%
\Knode*(\va,0){Q}[\KSame][lb]%
\end{AxesDot}%
```



3.3.2 例 2 $y = \sin x$ と $y = \cos x$ の交点

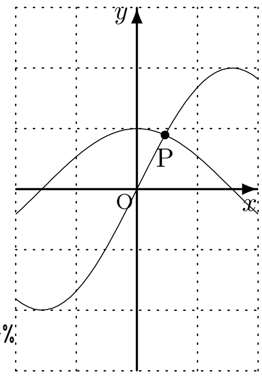
```
% 関数の描画と交点の表示
\unitlength=8mm
\begin{AxesDot}(4,4)(-2,-2)
\FDraw{\FESin}%
\FDraw{\FECos}%
\FSolve(\FESin,\FECos)[0,1]{\va}%
\FESin{\va}{\vb}%
\Knode*(\va,\vb){P}[\KSame][b]%
\end{AxesDot}%
```



3.4 例 3 $y = 2\sin x$ と $y = \cos x$ の交点

少し複雑な関数との交点を考えている。組み込み関数でないので、ユーザー定義関数を使っている。

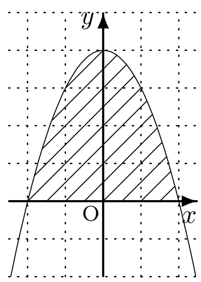
```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\FESin{#1}{#2}%
\Mul{#2}{2}{#2}}%
% 関数の描画と交点の表示
\unitlength=8mm
\begin{AxesDot}(4,6)(-2,-3)
\FDraw{\UsrFn}%
\FDraw{\FECos}%
\FSolve(\UsrFn,\FECos)[0,1]{\va}%
\UsrFn{\va}{\vb}%
\Knode*(\va,\vb){P}[\KSame][b]%
\end{AxesDot}%
```



3.5 斜線を引く

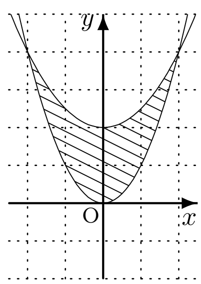
3.5.1 例 1 $y = -x^2$ と x 軸で囲まれた図形

```
\unitlength=5mm
\begin{AxesDot}(5,7)(-2.5,-2)
\def\UsrFnA#1#2{%
\Sub{0}{0}{#2}%
}% y=0
\def\UsrFnB#1#2{%
\Mul{#1}{#1}{#1}%
\Sub{4}{#1}{#2}%
}% y=-x^2+4
\FDraw{\UsrFnB}%
\Fhatch(-2,2)(\UsrFnA,\UsrFnB)%
\end{AxesDot}%
```



3.5.2 例 2 $y = x^2$ と $y = 0.5x^2 + 2$ で囲まれた図形

```
\unitlength=5mm
\begin{AxesDot}(5,7)(-2.5,-2)
\def\UsrFnA#1#2{%
\Mul{#1}{#1}{#2}%
}% y=x^2
\def\UsrFnB#1#2{%
\Mul{#1}{#1}{#2}%
\Mul{#2}{0.5}{#2}%
\Add{#2}{2}{#2}%
}% y=1/2x^2+2
\FDraw{\UsrFnA}%
\FDraw{\UsrFnB}%
\Fhatch[0.3,-0.5](-2,2)(\UsrFnA,\UsrFnB)%
\end{AxesDot}%
```



3.6　要素関数

無理関数、分数関数、三角関数、指数関数、対数関数、ユーザー定義関数の例をあげる。

3.6.1　無理関数

\FESqrt#1#2 が、無理関数である。\Sqroot#1#2 としても良い。
\FESqrt{2}{\va} とすると、\va に $\sqrt{2} = 1.41421356$ が入る。

3.6.2　分数関数

\FEFrac#1#2 が、分数関数である。
\FEFrac{3}{\va}とすると、\va に $\dfrac{1}{3} = 0.33333333$ が入る。

3.6.3　三角関数

\FESin#1#2 が正弦関数、\FECos#1#2 が余弦関数、\FETan#1#2 が正接関数である。
正弦関数は \Sin#1#2 、余弦関数は \Cos#1#2 として良い。
\FESin{3}{\va}とすると、\va に $\sin 3 = 0.14112001$ が入る。他の関数についても同様である。

3.6.4　指数関数

\FEExp#1#2 が指数関数である。
\FEExp{1}{\va}とすると、\va に $e = 2.71828183$ が入る。

3.6.5　対数関数

\FELog#1#2 が対数関数である。
\FELog{2}{\va}とすると、\va に $\log_e 2 = 0.69314718$ が入る。

3.6.6　ベキ関数

$x^r = e^{r \log_e x}$ であるから、対数関数と指数関数を組み合わせて定義することもできるが、便宜のために提供する。
\FEPowerIndex として指数 r を与える。この関数を使う直前に、再定義すること。
関数は \FEPower#1#2 である。
\FEPower{3}{\va}とすると、\va に 3^r が入る。ただし、r は直前で再定義した\FEPowerIndex の値である。

3.6.7　絶対値

\Abs#1#2 とすると、#1 の絶対値が#2 に入る。

3.6.8　ユーザー定義関数

要素関数を組み合わせて、ユーザー定義関数を作ることができる。その際、次の四則演算を使って組み合わせる。
\Add#1#2#3 は#3=#1+#2
\Sub#1#2#3 は#3=#1-#2
\Mul#1#2#3 は#3=#1×#2
\Div#1#2#3 は#3=#1÷#2 を表す。
例えば、$y = 2x + \sin x$ という関数を作ろう。

```
\def\UsrFn#1#2{%
    \FESin{#1}{#2}%   #2 に sin(#1) の値が入る
    \Mul{2}{#1}{\va}% \va に 2 ×#1 の値が入る
    \Add{#2}{\va}{#2}%#2 に 上の和が入る
    % 最後に #2 に入った値が返る値である。
}%
```

このように、要素関数と四則演算で作ることのできる関数を定義し使うことができる。関数名は任意であるが、現在定義されているものと同じ名前を使うとおかしな動きになったりエラーになったりするので、\UsrFn, \UsrFnA, \UsrFnB, ··· を勧めます。
もちろんユーザー定義関数を使って新しいユーザー定義関数を作ることも可能です。

3.7　要素関数の汎用化と描画の例

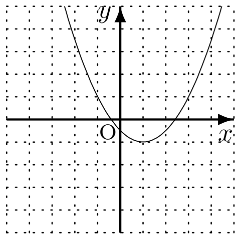
単純な関数\Fn#1#2 を使って、

```
\FDraw{\Fn}[*][x1,x2][y1,y2](a,b,c,d)%
```

とかくことで、オプションを選んで、グラフを描くことができる。
最初の*は省略可能である。グラフは折れ線で描いているが、*を書き込んだ場合、その折れ線の線分と線分の交わった点に●を打つ。滑らかさなどを確認したいときに点を打つと便利である。
次の [x1,x2] は、省略可能である。これを書き込んだ場合、グラフの定義域をこの範囲に制限する。グラフの一部を描いたり、一部を太くしたり、√のように、本来定義域が制限されている場合に使う。省略された場合は、グラフ全体でグラフを描く。
次の [y1,y2] は、省略可能である。これを書き込んだ場合、値域を制限する。[x1,x2] を省略した場合 [y1,y2] だけを記述することはできない。省略した場合は、グラフ全体でグラフを描く。
(a,b,c,d) は、省略可能である。書き込まれた場合、$y = f(x)$ でなく、$y = af(bx + c) + d$ のグラフを描く。書き込まれない場合は、単に $y = f(x)$ のグラフを描く。
【注意】省略を認識するために、*, [, (があるかどうか調べているので、最後に%を付けること。また、これは、描画したとき位置がずれないようにするおまじないでもある。

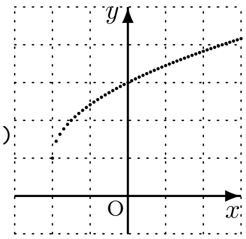
3.7.1 ２次関数 $y=\frac{1}{2}(x-1)^2-1$

```
% ユーザー関数の定義
\def\SqrFn#1#2{%
\Mul{#1}{#1}{#2}}% y ← x^2
% ユーザー関数の描画
\unitlength=3mm
\begin{AxesDotC}(10,10)
\FDraw{\SqrFn}(0.5,1,-1,-1)%
\end{AxesDotC}%
```



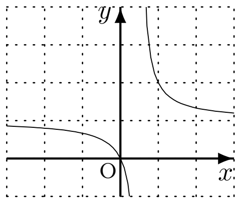
3.7.2 無理関数 $y=2\sqrt{\frac{x}{2}+1}+1$

```
\unitlength=5mm%
\begin{AxesDot}(6,6)(-3,-1)%
\FDraw{\FESqrt}*[-2,3](2,0.5,1,1)
\end{AxesDot}%
```



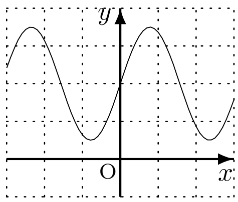
3.7.3 分数関数 $y=\frac{\frac{1}{2}}{x-\frac{1}{2}}+1$

```
\unitlength=5mm%
\begin{AxesDot}(6,5)(-3,-1)%
\FDraw{\FEFrac}(0.5,1,-0.5,1)
\end{AxesDot}%
```



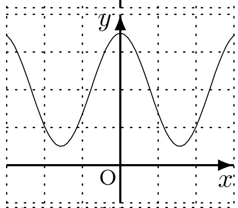
3.7.4 三角関数

```
\unitlength=5mm%
\begin{AxesDot}(6,5)(-3,-1)%
\FDraw{\FESin}(1.5,2,0,2)
\end{AxesDot}%
```



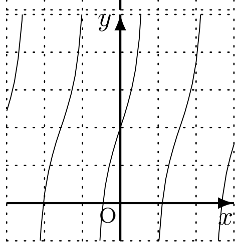
$y=\frac{3}{2}\sin(2x)+2$ のグラフ

```
\unitlength=5mm%
\begin{AxesDot}(6,5)(-3,-1)%
\FDraw{\FECos}(1.5,2,0,2)
\end{AxesDot}%
```



$y=\frac{3}{2}\cos(2x)+2$ のグラフ

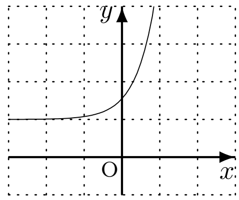
```
\unitlength=5mm%
\begin{AxesDot}(6,6)(-3,-1)%
\FDraw{\FETan}(1.5,2,0,2)
\end{AxesDot}%
```



$y=\frac{3}{2}\tan(2x)+2$ のグラフ

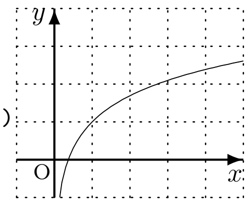
3.7.5 指数関数 $y=\frac{3}{2}e^{2x-1}+1$

```
\unitlength=5mm%
\begin{AxesDot}(6,5)(-3,-1)%
\FDraw{\FEExp}(1.5,2,-1,1)%
\end{AxesDot}%
```



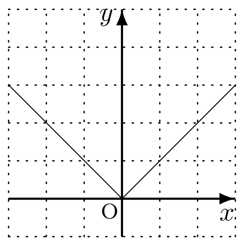
3.7.6 対数関数 $y=\log_e x+1$

```
\unitlength=5mm%
\begin{AxesDot}(6,5)(-1,-1)%
\FDraw{\FELog}[0.0001,5](1,1,0,1)
\end{AxesDot}%
```



3.7.7 絶対値関数 $y=|x|$

```
\unitlength=5mm%
\begin{AxesDot}(6,6)(-3,-1)%
\FDraw{\Abs}
\end{AxesDot}%
```

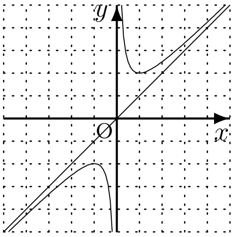


4 ユーザー定義関数の例

組み込み関数を平行移動する程度でなければ、このユーザー定義を行うことになる。
ここでは、数３の教科書にあるグラフを例に取り上げた。

4.1 例１ $y = x + \frac{1}{x}$

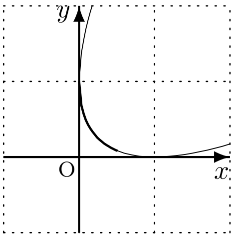
```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Div{1}{#1}{#2}% y ← 1/x
\Add{#2}{#1}{#2}}%y ← y + x
% ユーザー関数の描画
\unitlength=3mm
\begin{AxesDotC}(10,10)
\FDraw{\UsrFn}%
\FQuartic(0,0,0,1,0)%
\end{AxesDotC}%
```



4.2 例２ $\sqrt{x} + \sqrt{y} = 1$

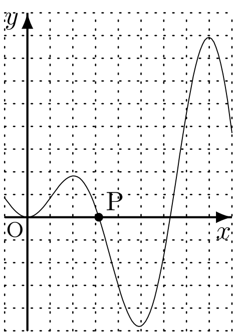
$\sqrt{x} + \sqrt{y} = 1 \iff y = 1 - 2\sqrt{x} + x\ (x \geqq 0)$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\SQRT{#1}{#2}%     y ← √ x
\Mul{#2}{-2}{#2}% y ← -2y
\Add{#2}{#1}{#2}% y ← y + x
\Add{#2}{1}{#2}}%  y ← y + 1
%
\def\UsrFnB#1#2{%
\SQRT{#1}{#2}% y ←  √ x
\Mul{#2}{2}{#2}% y ← 2y
\Add{#2}{#1}{#2}% y ← y + x
\Add{#2}{1}{#2}}% y ← y + 1
% ユーザー関数の描画
\unitlength=1cm
\begin{AxesDot}(3,3)(-1,-1)
{\edef\Fdx{0.03}
\FDraw{\UsrFnB}[0,0.5]%
\thicklines
\FDraw{\UsrFn}[0,0.5]}%
\FDraw{\UsrFn}[0.5,2]%
\end{AxesDot}%
```



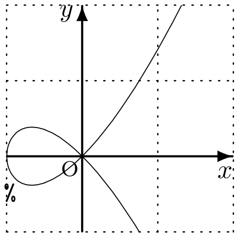
4.3 例３ $y = x \sin x$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Sin{#1}{#2}% y ← Sin(x)
\Mul{#1}{#2}{#2}}% y ← xy
% ユーザー関数の描画
\unitlength=3mm
\begin{AxesDot}(10,14)(-1,-5)
\FDraw{\UsrFn}%
\FSolve(\UsrFn,0)[2,4]{\va}%
\Knode*(\va,0){P}[\KSame][rt]
\end{AxesDot}%
```



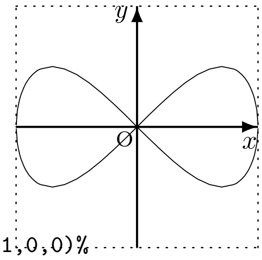
4.4 例４ $y^2 = x^2(x+1)$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Add{#1}{1}{#2}% y ← x+1
\SQRT{#2}{#2}%    y ← √ y
\Mul{#2}{#1}{#2}}%y ← xy
% ユーザー関数の描画
\unitlength=1cm
\begin{AxesDot}(3,3)(-1,-1)
{\edef\Fdx{0.01}
\FDraw{\UsrFn}[-1,-0.5](-1,1,0,0)%
\FDraw{\UsrFn}[-1,-0.5]}%
\FDraw{\UsrFn}[-0.5,2](-1,1,0,0)%
\FDraw{\UsrFn}[-0.5,2]%
\end{AxesDot}%
```



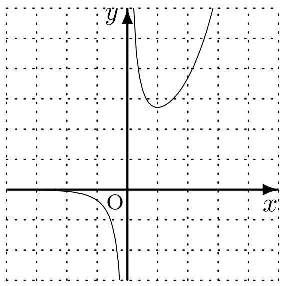
4.5 例５ $y^2 = x^2(1-x^2)$ レムニスケート

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Mul{#1}{#1}{#2}% y ← x^2
\Sub{1}{#2}{#2} % y ← 1-y
\SQRT{#2}{#2}%     y ← √ y
\Mul{#2}{#1}{#2}}% y ← xy
% ユーザー関数の描画
\unitlength=1.6cm
\begin{AxesDot}(2,2)(-1,-1)
{\edef\Fdx{0.01}
\FDraw{\UsrFn}[-0.9999,-0.8]%
\FDraw{\UsrFn}[ 0.8,0.9999]%
\FDraw{\UsrFn}[-0.9999,-0.8](-1,1,0,0)%
\FDraw{\UsrFn}[ 0.8,0.9999](-1,1,0,0)}%
\FDraw{\UsrFn}[-0.8,0.8]%
\FDraw{\UsrFn}[-0.8,0.8](-1,1,0,0)%
\end{AxesDot}%
```



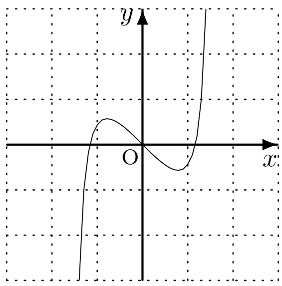
4.6 例６ $y = \dfrac{e^x}{x}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\FEExp{#1}{#2}% y ← e^x
\Div{#2}{#1}{#2}}% y ← y/x
% ユーザー関数の描画
\unitlength=4mm
\begin{AxesDot}(9,9)(-4,-3)
\FDraw{\UsrFn}%
\end{AxesDot}%
```



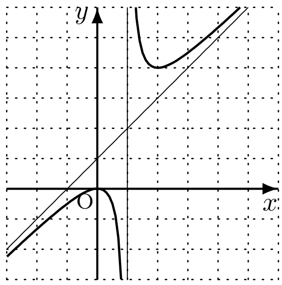
4.7 例７ $y = \tan x - 2x$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\FETan{#1}{#2}%
\Add{#1}{#1}{\vaUsrFn}%
\Sub{#2}{\vaUsrFn}{#2}}%
% ユーザー関数の描画
\unitlength=6mm
\begin{AxesDot}(6,6)(-3,-3)
\FDraw{\UsrFn}%
\end{AxesDot}%
```



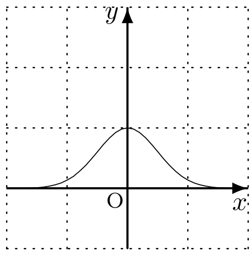
4.8 例 8 $y = \frac{x^2}{x-1} = x + 1 + \frac{1}{x-1}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Sub{#1}{1}{#2}%
\FEFrac{#2}{#2}%
\Add{#2}{#1}{#2}%
\Add{#2}{1}{#2}}%
% ユーザー関数の描画
\unitlength=4mm
\begin{AxesDot}(9,9)(-3,-3)
\thicklines
\FDraw{\UsrFn}%
\thinlines
\FQuartic(0,0,0,1,1)%
\path(1,-3)(1,6)%
\end{AxesDot}%
```



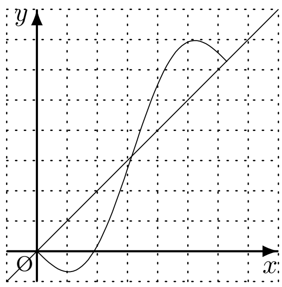
4.9 例 9 $y = e^{-2x^2}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Mul{#1}{#1}{#2}%
\Mul{#2}{-2}{#2}%
\FEExp{#2}{#2}}%
% ユーザー関数の描画
\unitlength=8mm
\begin{AxesDot}(4,4)(-2,-1)
\FDraw{\UsrFn}%
\end{AxesDot}%
```



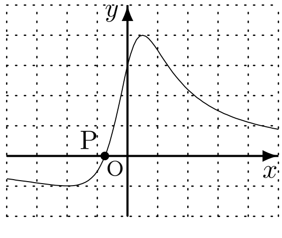
4.10 例 10 $y = x - 2\sin x$, $y = x$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Sin{#1}{#2}%
\Mul{#2}{-2}{#2}%
\Add{#2}{#1}{#2}}%
% ユーザー関数の描画
\unitlength=4mm
\begin{AxesDot}(9,9)(-1,-1)
\FDraw{\UsrFn}[0,6.2830]%
\FQuartic(0,0,0,1,0)%
\end{AxesDot}%
```



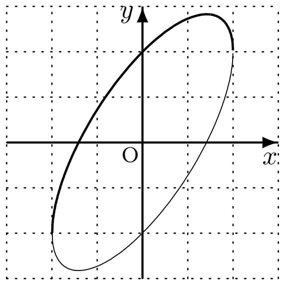
4.11 例 11 $y = \dfrac{4x+3}{x^2+1}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Mul{#1}{#1}{#2}%
\Add{#2}{1}{#2}%
\Mul{#1}{4}{\vtUsrFn}%
\Add{\vtUsrFn}{3}{\vtUsrFn}%
\Div{\vtUsrFn}{#2}{#2}}%
% ユーザー関数の描画
\unitlength=4mm
\begin{AxesDot}(9,7)(-4,-2)
\FDraw{\UsrFn}%
\Knode*(\va,0){P}[\KSame][lt]%
\end{AxesDot}%
```



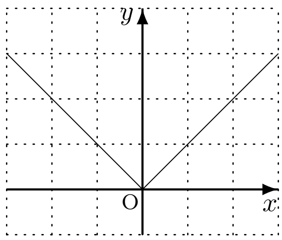
4.12 例 12 $y = x + \sqrt{4 - x^2}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Mul{#1}{#1}{#2}%
\Sub{4}{#2}{#2}%
\FESqrt{#2}{#2}}%
\def\UsrFnB#1#2{%
\UsrFn{#1}{#2}%
\Add{#1}{#2}{#2}}%
\def\UsrFnC#1#2{%
\UsrFn{#1}{#2}%
\Sub{#1}{#2}{#2}}%
% ユーザー関数の描画
\unitlength=6mm
\begin{AxesDot}(6,6)(-3,-3)
\thicklines
{\edef\Fdx{0.01}
\FDraw{\UsrFnB}[-2,-1.5]%
\FDraw{\UsrFnB}[1.5,1.99]}%
\FDraw{\UsrFnB}[-1.5,1.5]%
\thinlines
{\edef\Fdx{0.01}
\FDraw{\UsrFnC}[-2,-1.5]%
\FDraw{\UsrFnC}[1.5,1.99]}%
\FDraw{\UsrFnC}[-1.5,1.5]%
\end{AxesDot}%
```



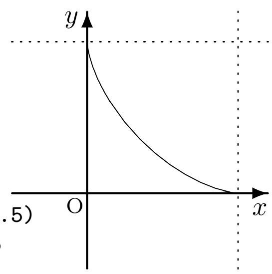
4.13 例 13 $y = |x|$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\AbsSign{#1}{#2}{\dummy}}%
% ユーザー関数の描画
\unitlength=6mm
\begin{AxesDot}(6,5)(-3,-1)
\FDraw{\UsrFn}%
\end{AxesDot}%
```



4.14 例 14 $x^{\frac{2}{3}} + y^{\frac{2}{3}} = 1$

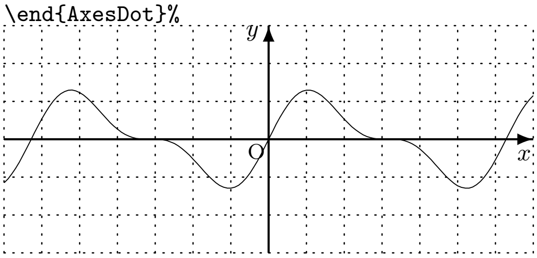
```
% ユーザー関数の定義
$x^\frac23+y^\frac23=1\iff y=(1-x^\frac23)^\frac32$
\def\UsrFn#1#2{%
\def\FEPowerIndex{0.66666667}% 2/3
\FEPower{#1}{#2}%
\Sub{1}{#2}{#2}%
\def\FEPowerIndex{1.5}% 3/2
\FEPower{#2}{#2}}%
% ユーザー関数の描画
% だいぶ時間がかかります！
\unitlength=2cm
\begin{AxesDot}(1.7,1.7)(-0.5,-0.5)
{\def\Fdx{0.008}% 0.01 に す る
と Arithmetic overflow になる
\FDraw{\UsrFn}[0.001,0.15]%
\FDraw{\UsrFn}[0.9,0.99]}%
\FDraw{\UsrFn}[0.15,0.9]%
\end{AxesDot}%
```



4.15 例 15 $y = \sin x(1 + \cos x)$

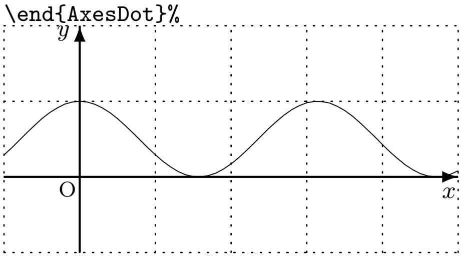
```
% ユーザー関数の定義
\def\UsrFn#1#2{%% y = \sin x(1+cos x)
\Cos{#1}{#2}%
```

```
\Add{#2}{1}{#2}%
\Sin{#1}{\va}%
\Mul{#2}{\va}{#2}}%
\unitlength=5mm
\begin{AxesDot}(14,6)(-7,-3)
\FDraw{\UsrFn}%
\end{AxesDot}%
```



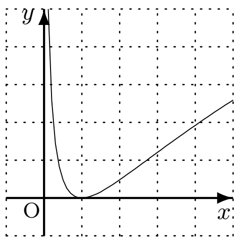
4.16 例 16 $y = \sin^2 x$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%% y = sin^2x
\Cos{#1}{#2}%
\Mul{#2}{#2}{#2}}%
\unitlength=5mm
\begin{AxesDot}(5,5)(-1,-3)
\FDraw{\UsrFn}[0.01,5]%
\end{AxesDot}%
```



4.17 例 17 $y = (\log x)^2$

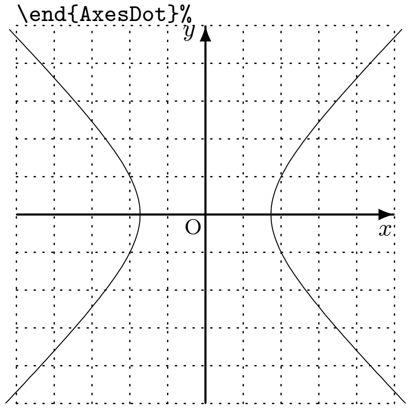
```
% ユーザー関数の定義
\def\UsrFn#1#2{%% y = (log x)^2
\Cos{#1}{#2}%
\Mul{#2}{#2}{#2}}%
\unitlength=5mm
\begin{AxesDot}(5,5)(-1,-3)
\FDraw{\UsrFn}[0.01,5]%
\end{AxesDot}%
```



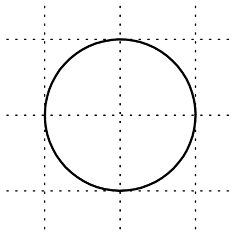
5 パラメータによる図形の描画

5.1 パラメータを使って、x と y を交換する

```
% ユーザー関数の定義
\def\UsrFn#1#2{\Add{#1}{0}{#2}}% y=x
\def\UsrFnA#1#2{%% x^2-y^2=3 x=\pm\sqrt{y^2+3}
\Mul{#1}{#1}{#2}%
\Add{#2}{3}{#2}%
\Sqroot{#2}{#2}}%
\def\UsrFnB#1#2{%% x^2-y^2=3 y=\pm\sqrt{x^2-3}
\UsrFnA{#1}{#2}%
\Sub{0}{#2}{#2}}%
\unitlength=5mm%
\begin{AxesDot}(10,10)(-5,-5)%
\PDraw(\UsrFnA,\UsrFn)[-5,5]%
\PDraw(\UsrFnB,\UsrFn)[-5,5]%
\end{AxesDot}%
```

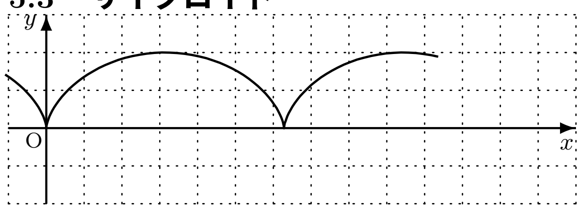


5.2 円



```
\unitlength=2cm
\begin{PicDotC}(3,3)
\thicklines
\PDraw(\Cos,\Sin)[0,6.4]%
\end{PicDotC}
```

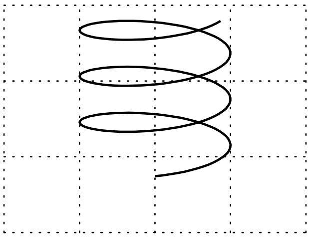
5.3 サイクロイド




```
\unitlength=5mm
\begin{AxesDot}(15,5)(-1,-2)%
\def\UsrFnA#1#2{%
\Sin{#1}{#2}%
\Sub{#1}{#2}{#2}%
}%
\def\UsrFnB#1#2{%
```

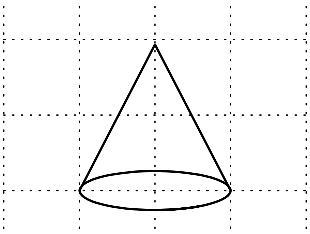
```
\Cos{#1}{#2}%
\Sub{1}{#2}{#2}%
}%
\thicklines%
\PDraw(\UsrFnA,\UsrFnB)[-2,10]%
\end{AxesDot}%
```

5.4 コイル



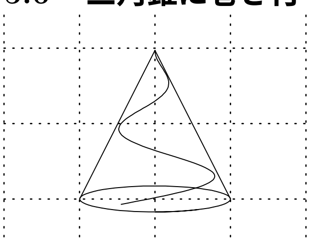
```
\EKView(0,-15)% 図形を下に 15 度回転させて表示する
\unitlength=1cm
\begin{PicDot}(4,3)(-2,-1)
\def\UsrFn#1#2{\Mul{#1}{0.1}{#2}}% y=0.1x
\thicklines
\PSDraw(\Cos,\Sin,\UsrFn)[0,20]%
\end{PicDot}}%
```

5.5 三角錐



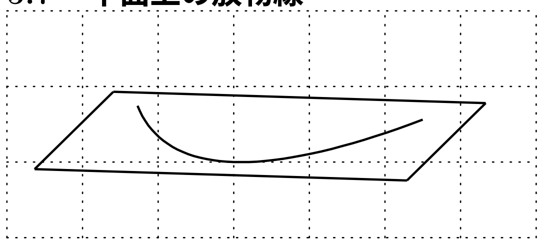
```
\EKView(0,-15)% 図形を下に 15 度回転させて表示する
\unitlength=1cm
\begin{PicDot}(4,4)(-2,-1)
\def\UsrFn#1#2{\Sub{0}{0}{#2}}% y=0
\thicklines
\PSDraw(\Cos,\Sin,\UsrFn)[0,7]%
\Snode(0,0,2){P}
\Snode(0,-1,0){A}
\Snode(0,1,0){B}
\KPath{APB}
\end{PicDot}%
```

5.6 三角錐に巻き付く空間曲線



```
\EKView(0,-10)% 図形を手前下に 15 度回転させて表示する
\unitlength=1cm
\begin{PicDot}(4,3)(-2,-0.5)
\def\UsrFn#1#2{\Sub{0}{0}{#2}}% y=0
\def\UsrFnA#1#2{%
\Mul{#1}{0.1}{#2}% <-
\Cos{#2}{#2}%
\Mul{#2}{#1}{#2}%
\Mul{#2}{0.1}{#2}% <-
}%
\def\UsrFnB#1#2{%
\Mul{#1}{0.1}{#2}% <-
\Sin{#1}{#2}%
\Mul{#2}{#1}{#2}%
\Mul{#2}{0.1}{#2}% <-
}%
\def\UsrFnC#1#2{%
\Mul{#1}{0.1}{#2}% <-
\Sub{1}{#2}{#2}%
\Mul{#2}{2}{#2}%
}%
\PSDraw(\Cos,\Sin,\UsrFn)[0,7]%
\Snode(0,0,2){P}
\Snode(0,-1,0){A}
\Snode(0,1,0){B}
\KPath{APB}
\PSDraw(\UsrFnA,\UsrFnB,\UsrFnC)[0,10]
\end{PicDot}%
```

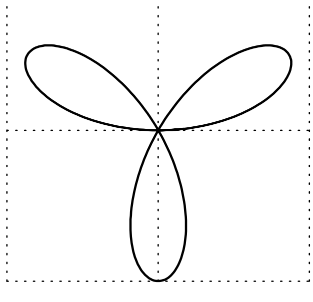
5.7 平面上の放物線



```
\EKView(-80,-10)% 図形を手前下に 15 度回転させて表示する
\unitlength=1cm
\begin{PicDot}(7,3)(-3,-1)
\def\UsrFn#1#2{\Sub{0}{0}{#2}}% y=0
\def\UsrFnA#1#2{\Mul{#1}{#1}{#2}}% y=x^2
\def\UsrFnB#1#2{\Add{#1}{0}{#2}}% y=x
\thicklines
\PSDraw(\UsrFnB,\UsrFnA,\UsrFn)[-2,2]%
\Snode(-2.5,-1,0){A}
\Snode(-2.5,5,0){B}
\Snode(2.5,5,0){C}
\Snode(2.5,-1,0){D}
\KPath{ABCDA}
\end{PicDot}%
```

6 極座標による曲線の描画

6.1 正葉曲線



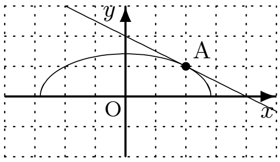
```
\unitlength=2cm
```

```
\begin{PicDotC}(2,2)%
\def\Fdx{0.01}%
\def\UsrFn#1#2{%
\Mul{#1}{3}{#2}%
\Sin{#2}{#2}%
}%
\thicklines%
\FDrawP{\UsrFn}[0,3.2]%
\end{PicDotC}%
```

7 平面図形との合わせ技

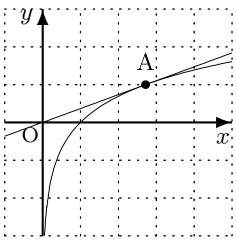
7.1 例 1 $\dfrac{x^2}{8}+\dfrac{y^2}{2}=1$ と接線 $x+2y-4=0$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Mul{#1}{#1}{#2}%
\Div{#2}{4}{#2}%
\Sub{2}{#2}{#2}%
\Sqroot{#2}{#2}}%
% ユーザー関数の描画
\unitlength=4mm
\begin{AxesDot}(9,5)(-4,-2)
{\edef\Fdx{0.001}%
\FDraw{\UsrFn}[-2.828,-2.7]%
\FDraw{\UsrFn}[2.7,2.827]}%
\FDraw{\UsrFn}[-2.7,2.8]%
\FQuartic(0,0,0,-0.5,2)%
\Knode*(2,1){A}[\KSame][rt]
\end{AxesDot}%
```



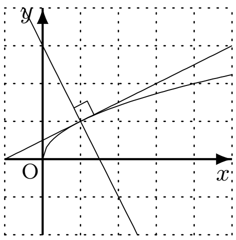
7.2 例 2 $y=\log x$ と接線 $y=\dfrac{1}{e}x$

```
% 関数の描画
\unitlength=5mm
\begin{AxesDot}(6,6)(-1,-3)
{\edef\Fdx{0.01}%
\FDraw{\FELog}[0.01,0.1]}%
\FDraw{\FELog}[0.1,5]%
\Knode*(\FCnstE,1){A}[\KSame][t]
\FEFrac{\FCnstE}{\vaE}%
\FQuartic(0,0,0,\vaE,0)%
\end{AxesDot}%
```



7.3 例 3 $y=\sqrt{x}$ と法線 $y=-2x+3$

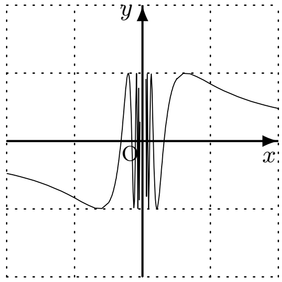
```
% 関数の描画
\unitlength=5mm
\begin{AxesDot}(6,6)(-1,-3)
\FDraw{\FESqrt}[0,5]%
\FQuartic(0,0,0,0.5,0.5)%
\FQuartic(0,0,0,-2,3)%
\Knode(0,3){A}%
\Knode(1,1){B}%
\Knode(3,2){C}%
\KNinty{ABC}%
\end{AxesDot}%
```



8 病的な例

8.1 例 1 $y=\sin\dfrac{1}{x}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Div{1}{#1}{#2}%
\Sin{#2}{#2}}%
% ユーザー関数の描画
\unitlength=9mm
\begin{AxesDot}(4,4)(-2,-2)
{\edef\Fdx{0.01}%
\FDraw{\UsrFn}[-0.5,-0.05]%
\FDraw{\UsrFn}[0.05,0.5]}%
\FDraw{\UsrFn}[-2,-0.5]%
\FDraw{\UsrFn}[0.5,2]%
\end{AxesDot}%
```



8.2 例 2 $y=x\sin\dfrac{1}{x}$

```
% ユーザー関数の定義
\def\UsrFn#1#2{%
\Div{1}{#1}{#2}%
\Sin{#2}{#2}}%
% ユーザー関数の描画
\unitlength=17mm
\begin{AxesDot}(2,2)(-1,-1)
{\edef\Fdx{0.001}%
\FDraw{\UsrFn}[-0.05,-0.02]%
\FDraw{\UsrFn}[0.02,0.05]}%
{\edef\Fdx{0.01}%
\FDraw{\UsrFn}[-0.5,-0.05]%
\FDraw{\UsrFn}[0.05,0.5]}%
\FDraw{\UsrFn}[-1,-0.5]%
\FDraw{\UsrFn}[0.5,1]%
\FQuartic(0,0,0,1,0)%
\FQuartic(0,0,0,-1,0)%
\end{AxesDot}%
```