柏新田原数学セミナー
JAPLA 研究会資料 2020/12/5

Jによる「スーパー楕円」のグラフィックス
ー自然界の図形はずっと自由だー

西川　利男

ことの始まりは、相変わらず続けている柏新田原数学セミナーでの話題からである。中学・高校の学校数学でのテーマは、どうして三角形、円から離れられないのだろう。

古典幾何学は古代ギリシャのピタゴラスから王者の学として確立された。

さらに、デカルトの解析幾何学により幾何学の世界はずっと豊かなものになった。

最近では、紙とえんぴつ、グラフ用紙に代わり、コンピュータ画面の上での幾何学となった。そして幾何学は王者の学から、一般庶民のものになったのである。

ところで、学校数学はどうして、こう融通のきかない定型的なのだろうか。柏の新田原数学セミナーでこんな話題が交わされた。自然界にある図形はこのような型にはまったものだけではなく、もっと自由で連続的でもある。

幾何学、コンピュータグラフィックスの対象をもっと広く、さまざまな形へと広げたら良い。このような意味で最近読んだ次の本は印象に残った。[1]

[1] 高木隆司「かたちの不思議」講談社現代新書(1989).

そのような一つの例として、この本の中に「スーパー楕円」という記載がある。この図形をJグラフィックスでプログラミングしてみた。この実験をとおして、もっと開かれた学校数学へと思いをはせた。

１．スーパー楕円とは [1] p.24,26

楕円は円と長方形の途中の図形といっても良いが、普通の楕円に似ているが、もう少しふくらませた楕円もどき形がスーパー楕円である。

ところで、家具のテーブルは本来、長方形だが四隅の角を少し丸くしてある。
一方、日本でも昔から小判型という図形がある。

北欧の建築家・デザイナーのビート・ハイン、ブルーノ・マットソンらによって、このような「スーパー楕円」形はテーブルから、椅子、マット、カーペットなどに人にやさしい形ということで広く用いられるようになった。

さて、スーパー楕円を数学の問題として扱っていこう。
一般に楕円とは次の式で表される図形である。

$$\left(\frac{X}{A}\right)^2 + \left(\frac{Y}{B}\right)^2 = 1$$

これを拡張した式として

$$\left|\frac{X}{A}\right|^M + \left|\frac{Y}{B}\right|^M = 1,\quad M \text{ は任意の実数}$$

で表される図形がスーパー楕円となる。

つまり、Mの値により、もっと丸く太った楕円になったり、すぼまりほっそりした楕円など、いろいろな楕円もどき図形が現れる。

M＝２では通常の楕円

M＝１では菱形

になる。

これらの図形をＪのグラフィックスでプログラミングしてみた。

２．数学図形とＪコンピュータグラフィックス

紙面の上の解析幾何からコンピュータ画面での図形表示には、そのためのいくつかの処理操作が必要になる。

解析幾何学では、紙面の中央を原点(0, 0)として, 左から右へ伸びるＸ軸と下から上に伸びるＹ軸での直交座標系の点の集まり(X, Y)として図形が表示される。

一方、コンピュータグラフィックスの画面では、その画素数を例えば 1000 x 1000 としたとき

(0, 0) ~ (X, Y) ~ (1000, 1000)

の点の集まりで表される。中央が(0, 0)ではない。

J4 の gl2 グラフィックスでは画面の左下(0, 0)、 右上(1000, 1000)である。

またグラフィックスによっては、縦座標が数学座標系と異なり、下から上でなく、上から下のときのものもある。さらには、浮動小数点の実数値ではなく、整数値でなければならないときもある（J9 のグラフィックス）。

従ってこれらの仕様に合わせる変換が必要になる。

今回の J4 の gl2 グラフィックスでは、その変換として

```
adj =: 3 : ‘500 + 100*y.’
```

により、数学値よりピクセル値にに変換した。つまり数学値を 100 倍して、コンピュータ画面の中央(500, 500)に移動する。

実際にスーパー楕円を描くには、先の式を変形して、値xから値yをあらわに求める。

$$y = B \times \left(\sqrt[M]{1 - \left| \frac{x}{A} \right| \wedge M} \right)$$

その部分のJのコーディングは次のようになる。

```
X =: (i. N) % ((N-1)%A)
Y =: B * (| 1 - (X%A)^ M)^(1%M)
```

なお、一般のプログラミング言語では図形を描くのにどうしても繰り返しループが必要である。ところが、Jでは各、点の値(X, Y)を配列として与えてやって、それをつなげば図形が出来あがる。つまり、Jではループを用いることはないのである。

図形の表示はgllinesにより行った。なお、簡便のため、第１象限(x >0, y>0)で値を計算して、第２象限、第３１象限、第４象限では値の符号を変えて、図形を貼り付けた。

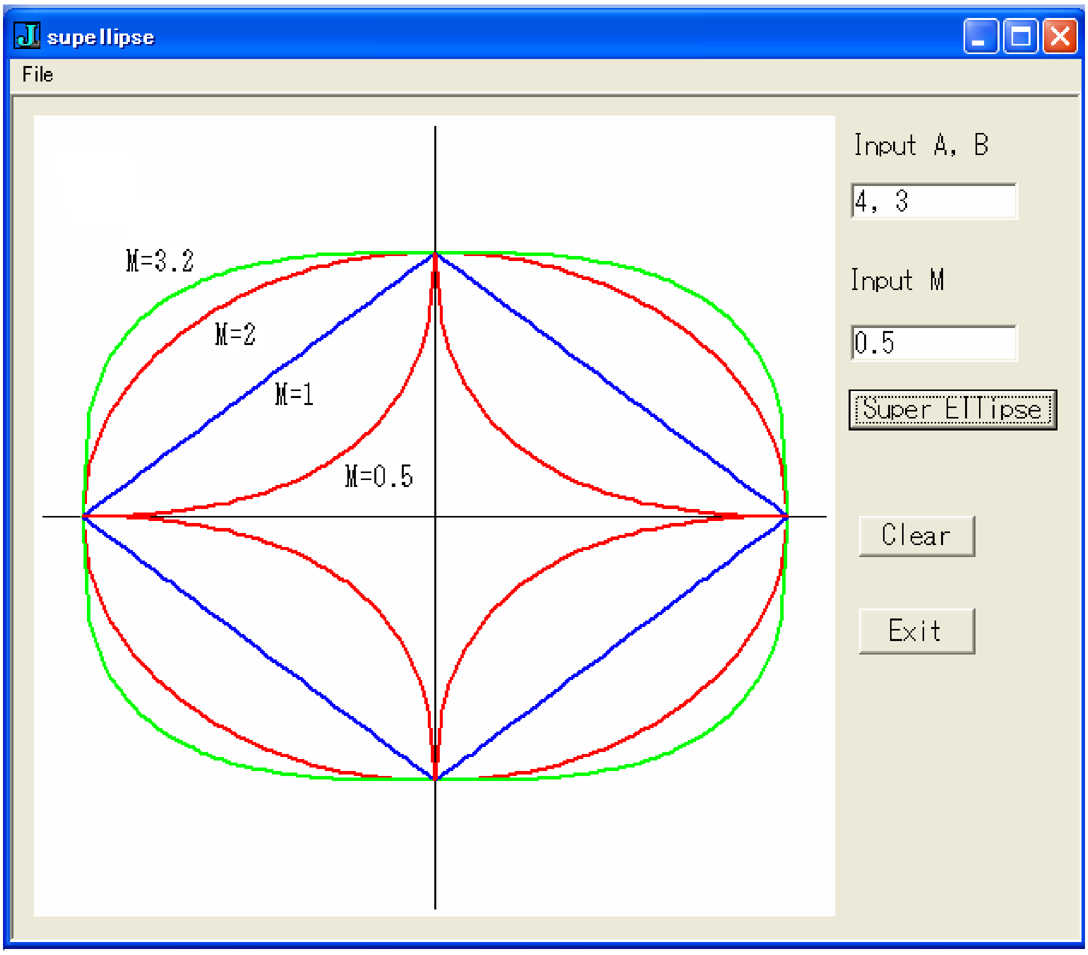
```
gllines adj , X ,. Y
gllines adj , X ,. (-Y)
gllines adj , (-X) ,. Y
gllines adj , (-X) ,. (-Y)
```

全体のプログラムは最後にあげた。

３．J によるスーパー楕円のグラフィックスの実際

プログラムの実行は、run ‘ ‘ とすると画面が現れる。
Input A, B により、長径、短径を入力し、続いて Input Mによりスーパー楕円のべき乗数 M の値を入力する。

ボタン Super Ellipse を押すとべき乗数 M に対応したスーパー楕円が現れる。
違う M の値を入れると次々と色を代えて、スーパー楕円が表示される。

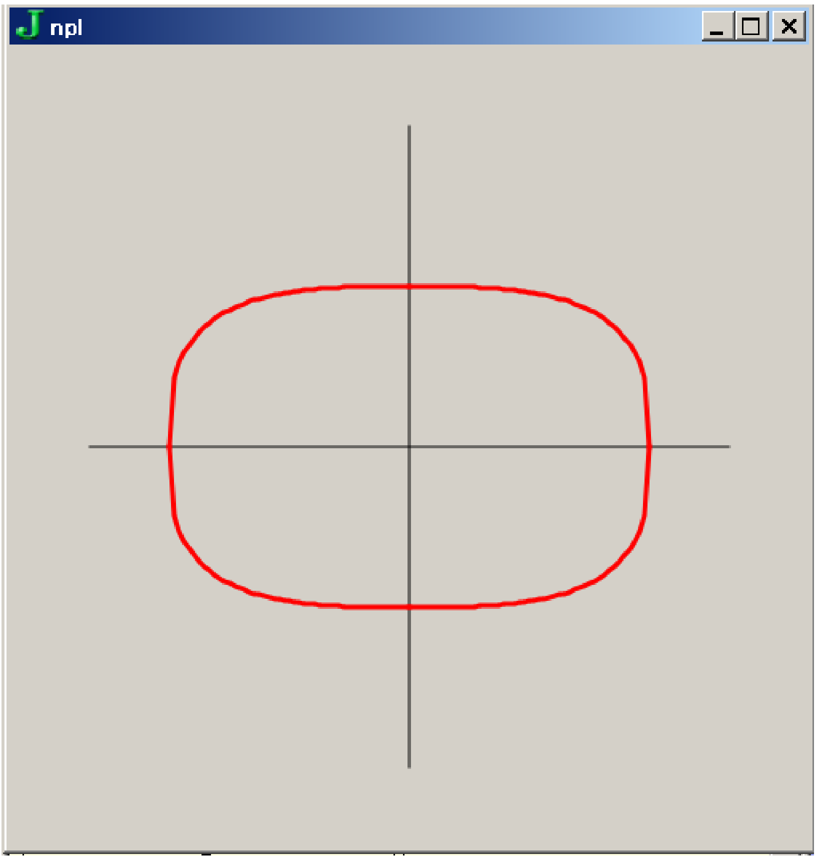


４．Ｊ９によるスーパー楕円のグラフィックスの実際

J9 でも動くようにプログラムを作った。
J9 では、Form_editor がサポートされず、またボタン操作も出来ない。従って、run の引数として、スーパー楕円の次数 M の値を入れて、次のようにして実行する。

run 3.2

また、プログラムもいろいろと手直しが必要となったが、そのプログラムは最後にあげた。



J のプログラム(J4)

```
NB.  super_ellipse.ijs
NB.   2020/11/2 T.Nishikawa
NB. 高木隆司「かたちの不思議」p.24, 26, 講談社現代新書 (1989).

require 'gl2'

SUPELLIPSE=: 0 : 0
pc supellipse;
menupop "File";
menu new "&New" "" "" "";
menu open "&Open" "" "" "";
menusep ;
menu exit "&Exit" "" "" "";
menupopz;
xywh 247 133 34 12;cc cancel button;cn "Exit";
xywh 6 5 234 208;cc sellipse isigraph;
xywh 244 76 61 11;cc supel button;cn "Super Ellipse";
xywh 244 59 50 11;cc inputM edit ws_border es_autohscroll;
xywh 244 44 50 10;cc inpm static;cn "Input M";
xywh 245 9 50 10;cc inab static;cn "Input A, B";
xywh 244 22 50 11;cc inpab edit ws_border es_autohscroll;
xywh 247 109 34 11;cc Clear button;
pas 6 6;pcenter;
rem form end;
)

run =: supellipse_run
supellipse_run=: 3 : 0
wd SUPELLIPSE
NB. initialize form here
wd 'pshow;'
gllines 10, 500, 990, 500
gllines 500, 10, 500, 990
glshow ''
)
```

```
supellipse_close=: 3 : 0
wd'pclose'
)

supellipse_cancel_button=: 3 : 0
supellipse_close''
)

supellipse_Clear_button=: 3 : 0
glclear ''
gllines 10, 500, 990, 500
gllines 500, 10, 500, 990
glshow ''
)

adj =: 3 : 0
500 + 110 *  y.
)

supellipse_supel0_button=: 3 : 0
glrgb 255 0 0
glpen 4 0
NB. gllines 200 700 800 700 800 300 200 300 200 700
gllines adj 3, 2, _3, 2, _3, _2, 3, _2, 3, 2
glshow ''
)

A =: 3
B =: 2

N =: 49

Col =: (255, 0, 0); (0, 255, 0); (0, 0, 255)
ColN =: 0
```

```
supellipse_supel_button=: 3 : 0
X =: (i. N) % ((N-1)%A)
Y =: B * (| 1 - (X%A)^inM)^(1%inM)

glrgb 255 0 0
glrgb (> (3|ColN) { Col)
glpen 4 0
gllines adj , X ,. Y
gllines adj , X ,. (-Y)
gllines adj , (-X) ,. Y
gllines adj , (-X) ,. (-Y)
ColN =: ColN + 1
glshow ''
)

supellipse_inputM_button=: 3 : 0
inM =: ". inputM
)

supellipse_inpab_button=: 3 : 0
'A B' =: ". inpab
)
```

J9 のプログラム

```
NB. super_ellipse.ijs
NB. j9 - version ====================================
NB.    2020/12/2 T.Nishikawa
NB.
NB. Enter run 3.2 , run 1.8, etc => super ellipse
NB. run 2 => ordinary ellipse

load 'graph'
load 'numeric trig'
coinsert 'jgl2'

NPL=: 0 : 0
pc npl closeok;
minwh 500 500;cc g isigraph flush;
pas 0 0;
)

npl_run =: 3 : 0
M =: y
wd NPL
wd 'pshow'
)

N =: 49
A =: 3
B =: 2

NB. M =:  2.4

NB. X =: (i. N) % ((N-1)%A)
NB. Y =: B * (| 1 - (X%A)^M)^(1%M)

adj =: 3 : 0
250 + 50 *  y
)
```

```
npl_g_paint=: 3 : 0
wh=: glqwh ''
gllines 50, 250, 450, 250  NB. x-axis
gllines 250, 50, 250, 450  NB. y-axis
glrgb 255 0 0
glpen 3


X =: (i. N) % ((N-1)%A)
Y =: B * (| 1 - (X%A)^M)^(1%M)

gllines roundint adj , X ,. Y
gllines roundint adj , X ,. (-Y)
gllines roundint adj , (-X) ,. Y
gllines roundint adj , (-X) ,. (-Y)
)

run =: npl_run
```