

複素グラフィックスの世界 (0)

SHIMURA Masato

JCD02773@nifty.com

URL:http://homepage3.nifty.com/asagaya_avenue

2013 年 9 月 17 日

目次

1	ピタゴレアンを極座標で	2
2	フェルマー素数と正多角形	6
3	指数関数 $f(x) = e^{ix}$ を 3D で描く	16
4	ジュリア集合とマンデルブロート集合	19
付録 A	J と複素数	23

はじめに

高次方程式の解に複素数が現れることをタルタリアやカルダノ達のルネッサンス後期の数学者がを見つけ、これを嬉々として使ったのはオイラーである。社会科学分野では、複素数に触れることは稀で、多くは解のうち実数を選ぶというルネッサンスレベルで足りた。今般、「インドらの真珠ークラインの夢見た世界」(D. マンフォード、C. シリーズ、D. ライト著) が上梓された。群論と複素グラフィックスを多用しており、読み解いて、複素数を J で扱い、複素幾何グラフィックスの深奥を理解するため、本棚に仕舞ってあった数冊の本を取り出した。

金谷「クォータニオン入門」を読んでいたらクォータニオンと複素数が次のようにさらりと表現されていた

クォータニオンを用いる利点は3次元のベクトルが「数」としての性質を持つことである。ちょうど2次元のベクトルを複素数であらわせたように

複素数はデュオ、双子の数であり、この視点で数学や色々なコンピューター言語を見てみると

数学 数学は区分して慎重に対処している

- パスポートが必要 $A \in \mathbb{C}$
- 複素数内で実数と虚数が住み分け $a + bi$
- 計算は各自手計算で行う $(a + bi)(c + di)$

C言語 C言語は手続きが厳格でとても面倒なので各種のライブラリを活用する

- 呼び出してから舞台登場 `\include <complex>`
- ユニフォームは所属チーム毎に異なる `complex alpha,beta(1,2),gamma(3,4)`

J言語 数学者 *K.E.Iverson* が作った *APL* や *J* 言語は数学よりも柔軟で、複素数を普通の数として取り扱うことができる。演算過程で $a \in \mathbb{C}$ と断る必要は全く無く、複素数の関数も整備されており、*J* の演算記号でそのまま高級電卓として演算することもできる

1 ピタゴレアンを極座標で

1.1 円関数から求める

ピタゴラス派のシンボルのような定理であるが、エジプトでは3,4,5のエジプト縄としてピラミッドの計測に用いられ、メソポタミアでは粘土板に刻まれてたものが発掘されている。

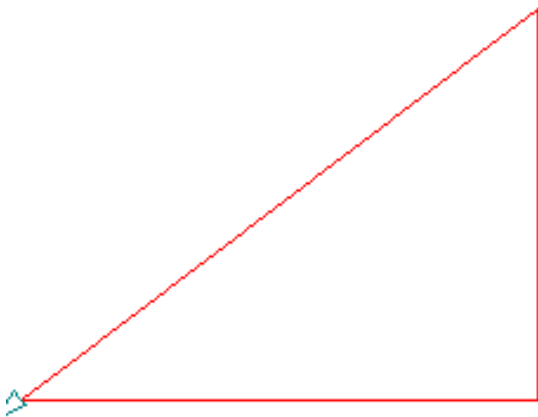
ピタゴレアンの直角を挟む2辺を3 4 とすると斜辺は5で、これは極座標の絶対値になる。しかし、偏角 (θ) を求めることは稀である。

1.2 円関数から求める

円関数から求める	経過と解説
<pre> _3 o. 3r4 4r3 0.643501 0.927295 arctan 4r3 3r4 0.927295 0.643501 dfr arctan 4r3 3r4 53.1301 36.8699 53.1301°と 36.8699° </pre>	<ul style="list-style-type: none"> • 3r4 は $\frac{3}{4}$。tan であり、x,y • 偏角 (θ) は \tan^{-1} で求める。 • J の円関数で arctan は _3 o. である。 arctan=: _3&o. はラジアン表示 • require 'trig' で arctan と dfr を用いることができる • dfr (degree from radiun) ラジアンをデグリーに変換する dfr=: *&(180%pi) rfd=: *&(pi%180)

1p1 は π である

タートルグラフィックスは極座標で描画する



```
require 'turtle'
```

```
show rt 90 fd 4 lt 90 fd 3 lt (90+ 36.8699) fd 5
clip ''
```

1.3 極座標から求める

極座標から求める	経過と解説
<pre>*. 3j4 4j3 5 0.927295 5 0.643501</pre>	<ul style="list-style-type: none"> • 3j4 2 辺 (x,y) の複素数表示。x,y を一つの数として取り扱う • *. (length/angle) 極座標の絶対値と偏角 (ラジアン) を求める

- 複素数の絶対値 (|) で極座標の辺長 (絶対値) が求められる

```
| 3j4 4j3
5 5
```

- $\sqrt{x^2 + y^2}$

```
(%:@+/@:^&2)(L:0) 3 4;4 3 NB. sqrt sum ^2
+---+
|5|5|
+---+
```

1.4 複素数の極座標

- 複素数 z の絶対値 ($|z|$) と偏角 $\theta = \arg z$
- z の極座標表示

$$z = |z|(\cos\theta + i\sin\theta)$$

- 指数関数 $e^{i\theta}$

オイラーの恒等式。これが複素グラフィックスのエンジンとなる。

$$e^{i\theta} = (\cos\theta + i\sin\theta)$$

- これは r で求められる。更に $+$ で実数/虚数分離を行いグラフィックデータを得る。

絶対値は拡大を、偏角は回転を表す。

$$z = re^{i\theta} = r(\cos\theta + i\sin\theta)$$

- 数値例 $3e^{i\frac{\pi}{4}} = 3(\cos\frac{\pi}{4} + i\sin\frac{\pi}{4})$

1. $r \times \cos\theta + i\sin\theta$

$$3 * (\cos 1r4p1) + j. \sin 1r4p1$$

$$2.12132j2.12132$$

2. $r \times e^{i\theta}$

$$3 * \hat{j}.1r4p1$$

$$2.12132j2.12132$$

3. r . (angle/polar)

$$3 * r. 1r4p1$$

$$2.12132j2.12132$$

- J のグラフィックスは \cos から x 座標、 \sin から y 座標を求め、最終的には複素座標で描画している。

2 フェルマー素数と正多角形

2.1 フェルマー素数

19 歳のガウスが正 17 角形の作図が可能であることを発見し、数学をやろうと決心したとの逸話のある正 17 角形を描いてみよう。

ユークリッドの許した直線定規とコンパスのみを用いてという制約でいくつの正多角形を描けることができるか。

ユークリッドは正 3,4,5,6 角形と正 15 角形を作図していた。

ガウスは正 n 角形 $\{n\}$ が定規とコンパスだけで作図できるのは n の奇素因数が異なった「フェルマー素数」

$$F_k = 2^{2^k} + 1$$

である場合に限られることを発見した。

これを計算してみると

```
fermer_prime=: 3 : '(i.y),. x: >: 2^2^ i. y'
```

*1

		fermer_prime	8
0	$F_0 = 2^1 + 1 = 3$	0	3
1	$F_1 = 2^2 + 1 = 5$	1	5
2	$F_2 = 2^4 + 1 = 17$	2	17
3	$F_3 = 2^8 + 1 = 257$	3	257
4	$F_4 = 2^{16} + 1 = 65537$	4	65537
5	$F_5 = 2^{32} + 1 = 4294967297$	5	4294967297
		6	18446744073709524992
		7	340282366920937443432214307588402053120

F_5 以上の数の素数テスト 1 p: を用いる。1 は素数で 0 は合成数。p:は素数関連の関数

```
fermer_primetest=: 3 : '(i.y),.1 p: x: >: 2^2^ i. y'
```

*1 J に限らず最近の数値計算言語は明示、黙示で長大倍数や有理数をサポートしている。J では $12x$ のように末尾に x を付ける。

```

fermer_primetest 10      p: {: fermer_prime 5
0 1
1 1                      11 821651
2 1
3 1
4 1                      F5 以上の数は素数でなく合成数であることを示
5 0                      している。フェルマーは F5 が素数であると思っ
6 0                      込んでいた。オイラーは F5 が合成数であると証
7 0                      明している
8 0                      F9 より多くの Fk が合成数であると証明されてい
9 0                      るようだ。F9 までは計算できたが手元のノート
                      PC ではメモリが苦しい。

```

1898 にはリシュローとシュウエンディンハイムが正 257 角形を、その 10 年後に J. ヘルメスが正 65537 角形の作図をした。

2.2 ガウスの正 17 角形の作図

正 n 角形の作図は円周分割方程式の解を得ることである。正 17 角形の円周分割方程式は次の多項式であらわされる。

$$z^{17} - 1 = 0$$

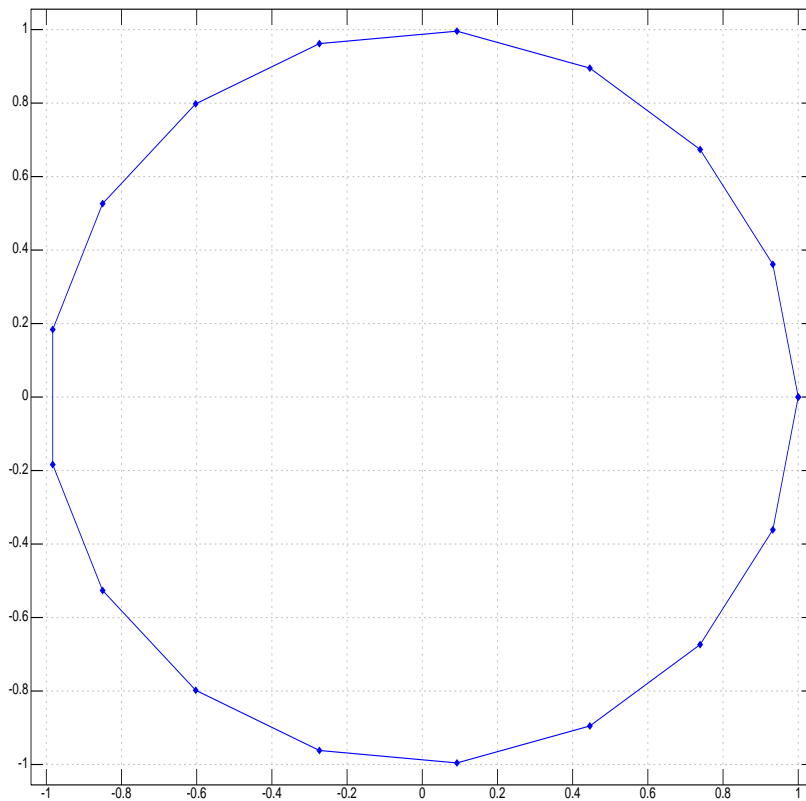
2.2.1 指数関数 $e^{i\theta}$ で

コンパスと定規に限定しなければ正 17 角形は $e^{i\theta}$ から簡潔に座標を求めることができ、1 行で計算と作図ができる。

```

require 'plot numeric trig' NB. trig is circle function
'line marker' plot {: (+.r. 2p1* (i.17)%17),1 0

```



- 後尾の, 1 0 はラインの終端と始点を結ぶ

2.2.2 多項式を解く

1. J の多項式の入力と解法

$-3 + x - 2x^2 + 6x^4$ ならば p. _3 1 _2 0 6 と入力

最初の 6 は J の p. 関数の収束までの反復回数

p. _3 2 _1 0 6

```

+--+-----+
|6|_1 0.117487j0.799856 0.117487j_0.799856 0.765026|
+--+-----+

```

最初の 6 のボックスは J の p. での反復回数

2. 多項式の指定ポイントでの値

```
_3 1 _2 0 6&p. 0 1 2 3 4 5
```

```
_3 2 87 468 1505 3702      NB. x=0 1 2 3 4 5 での値
```

3. 正 17 角形の多項式

$$z^{17} - 1 = 0$$

4. J の多項式の入力と解法

```
p. _1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1  f(x) = -1 + z17
```

5. p. で多項式の解を求めた後、+(real/imaginary) で複素解を実数/虚数に分離する

```
(i.17),.(+.tmp) ,.tmp=.;}. p. _1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
```

No.	real/image	分離	複素解
0	1	0	1
1	0.932472	0.361242	0.932472j0.361242
2	0.932472	_0.361242	0.932472j_0.361242
3	0.739009	0.673696	0.739009j0.673696
4	0.739009	_0.673696	0.739009j_0.673696
5	0.445738	0.895163	0.445738j0.895163
6	0.445738	_0.895163	0.445738j_0.895163
7	0.0922684	0.995734	0.0922684j0.995734
8	0.0922684	_0.995734	0.0922684j_0.995734
9	_0.273663	0.961826	_0.273663j0.961826
10	_0.273663	_0.961826	_0.273663j_0.961826
11	_0.602635	0.798017	_0.602635j0.798017
12	_0.602635	_0.798017	_0.602635j_0.798017
13	_0.850217	0.526432	_0.850217j0.526432
14	_0.850217	_0.526432	_0.850217j_0.526432

```

15 _0.982973  0.18375  _0.982973j0.18375
16 _0.982973  _0.18375  _0.982973j_0.18375

```

6. p. の解法は共役順に表示される。正 17 角形を描くため単位円周上に解を並び替える

```

pick17_sub ''
+-----+-----+
|0 2 4 6 8 10 12 14 16|15 13 11 9 7 5 3 1 0|
+-----+-----+

```

```

(i.18),. |:> gauss17 ''
0      1      0
1  0.932472 _0.361242
2  0.739009 _0.673696
3  0.445738 _0.895163
4  0.0922684 _0.995734
5  _0.273663 _0.961826
6  _0.602635 _0.798017
7  _0.850217 _0.526432
8  _0.982973 _0.18375
9  _0.982973  0.18375
10 _0.850217  0.526432
11 _0.602635  0.798017
12 _0.273663  0.961826
13  0.0922684  0.995734
14  0.445738  0.895163
15  0.739009  0.673696
16  0.932472  0.361242
17      1      0

```

17 番目は終点と始点を結ぶ

- Script

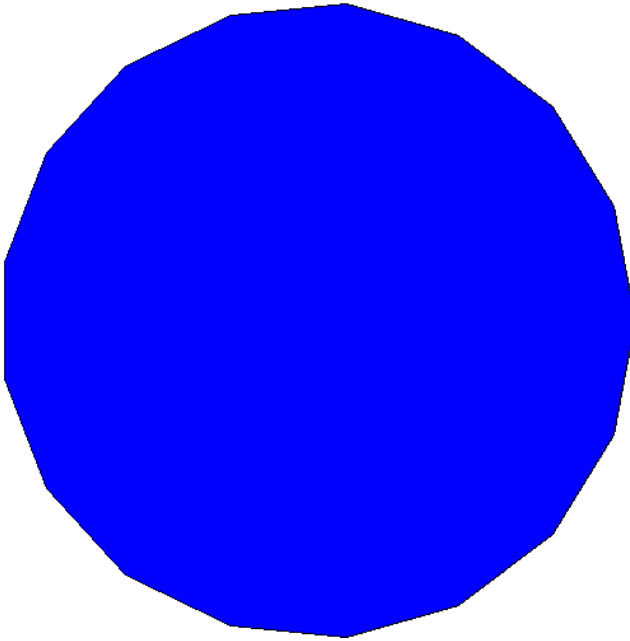
```
gauss17=: 3 : 0
NB. plot gauss17 ''
tmp0=.,. +. ; }. p. _1, (,16#0),1
{|: (;pick17_sub '') {tmp0
)
```

```
pick17_sub=: 3 : 0
NB. pick17_sub ''
tmp0=.{|: 9 2 $ i.17
(({|. tmp0)),(|.@}: L:0 {: tmp0),L:0] 0
)
```

- isigraph 上で polygon で描く
キャンバスは C.Reiter の addon dwin2.ijs を用いた。

```
require 'addon/graphics/fvj3/dwin2.ijs'
```

```
_1 _1 1 1 dwin ''
0 0 255 dpoly gauss17 ''
```



2.3 正多角形を描く

コセクターにプトレマイオスとリッチモンドの正5角形の内接方法が紹介されている。ユークリッド流に定規とコンパスは本当に大変だ。

複素数を極座標に変換すると

$$z = x + yi \rightarrow r(\cos\theta + i\sin\theta)$$

のかたちになる。 $r = 1$ とすると拡大がなくなり、回転のみとなる。

この $\cos + i\sin$ の形が xy 座標となる

ド・モアブルの公式

$$(\cos \alpha + i \sin \alpha)^n = \cos n \alpha + i \sin n \alpha$$

は回転を n 回繰り返すことになる。特に

$$\omega = \cos \frac{2\pi}{n} + i \sin \frac{2\pi}{n}$$

をかけることは $\frac{2\pi}{n}$ の回転になり、ここから単位円上の回転は内接する正 n 角形 $\{n\}$ の頂点が得られる

この n 個の頂点は円周等分方程式 $z^n = 1$ の根を表す

$$1, \omega, \omega^2, \omega^3, \dots, \omega^{n-1}$$

ここから次の方程式が成り立つ

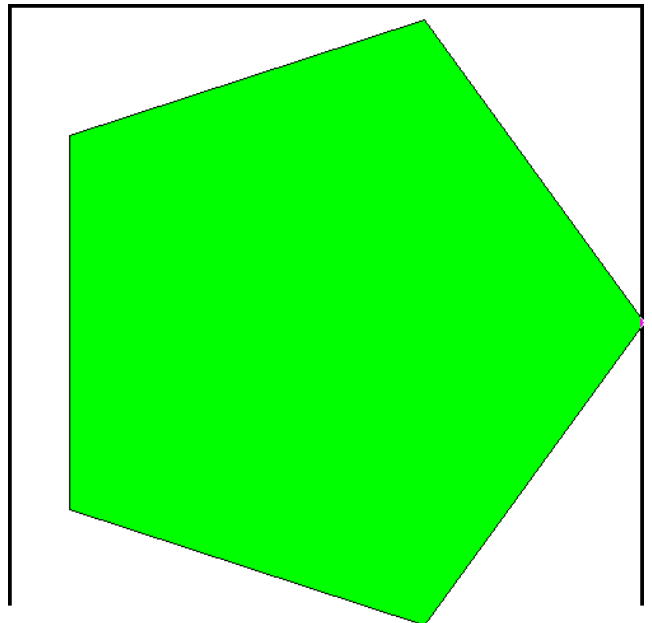
$$z^n - 1 = (z - 1)(z - \omega)(z - \omega^2)(z - \omega^3) \cdots (z - \omega^{n-1})$$

正多角形の作図を定規とコンパスのユークリッドの制約なしで PC で行おう。

プログラムを組むまでもないが複素数を多用する。 $e^{i\theta} = \cos\theta + i\sin\theta$ なので $e^{i\theta}$ を用いる。 $\theta = \frac{2\pi}{n}$ をもちいる。

キャンパスは *C.Reiter* の `addon/graphics/fvj3/dwin.ijs` を用いた。

```
+. r. 2p1* (i.5)%5
      1      0
0.309017  0.951057
_0.809017  0.587785
_0.809017 _0.587785
0.309017 _0.951057
```



```
_1 _1 1 1 dwin ''
0 0 255 dpoly +. r. 2p1* (i.5)%5
```

2.3.1 経過と解説

1. $\frac{i.5}{5}$ $\frac{i.n}{n}$ は n 等分するスパンを指定

```
(i.5)%5
```

```
0 0.2 0.4 0.6 0.8
```

2. $2p1 \times$ 単位円 (2π) を n 分する

```
2p1* (i.5)%5
0 1.25664 2.51327 3.76991 5.02655
```

3. 複素数に変換し、 e^{ix} を求める。 $\cos\frac{2\pi}{n} + isin\frac{2\pi}{n}$ と同じ

```
(tmp=. 2p1 * (i.5)%5),. (r. tmp) ,. 1x1 ^ j. tmp
```

```
tmp=(2p1*(i.5)%5) r. tmp      1x1 ^ j. tmp
```

```
-----
          0              1              1
1.25664   0.309017j0.951057   0.309017j0.951057
2.51327  _0.809017j0.587785  _0.809017j0.587785
3.76991  _0.809017j_0.587785 _0.809017j_0.587785
5.02655   0.309017j_0.951057   0.309017j_0.951057
```

4. r は絶対値 1 の $e^{i\theta}$ の単位複素数表示

```
r. 1
0.540302j0.841471
1x1 ^ j.1
0.540302j0.841471
| r. 1
1
```

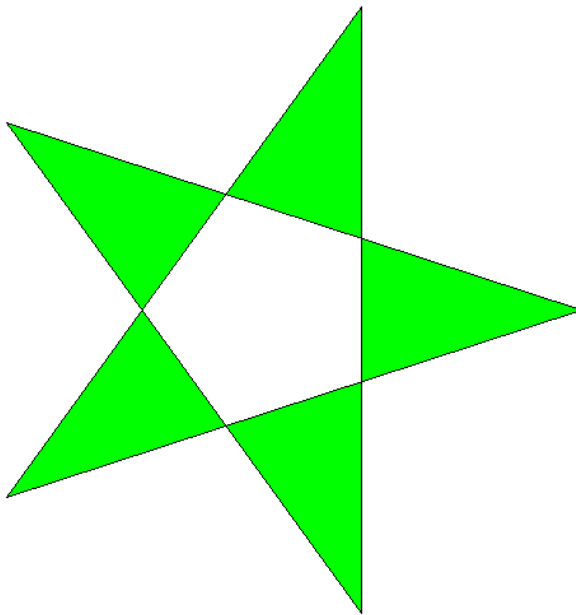
5. 実数と虚数の分離 順に xy 座標になっている。 $plot$ なら複素数のままでも描くことができる

```
+. r. 2p1* (i.5)%5
          1          0
0.309017  0.951057
_0.809017  0.587785
_0.809017 _0.587785
```

0.309017 _0.951057

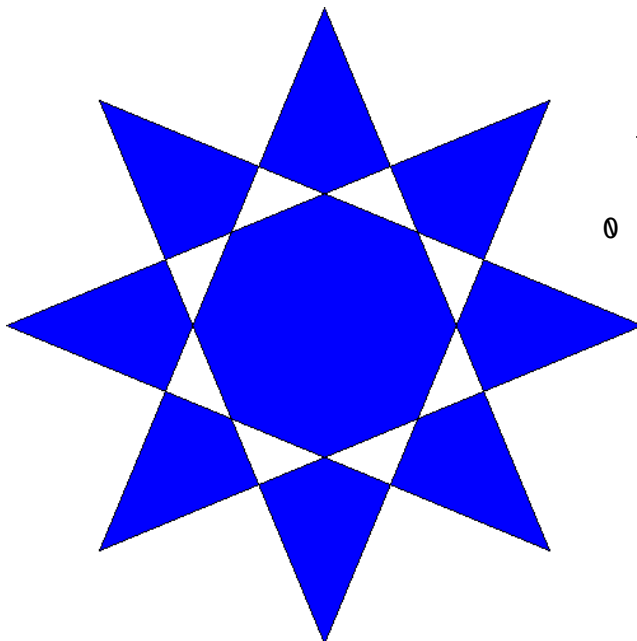
2.4 星型多角形

コセクターは星型多角形を群から多角形を導き出すとしているが、各頂点が分かっているの逆に取り出して描くことができる。頂点(0)から鉛筆で行く先を追っていくと回転群ができる



```
_1 _1 1 1 dwin ''  
0 255 0 dpoly  
0 2 4 1 3 0 { +. r. 2p1* (i.5)%5
```

$$\begin{pmatrix} 0 & 2 & 4 \\ 1 & 3 & 0 \end{pmatrix}$$



```
_1 _1 1 1 dwin ''  
0 0 255 dpoly  
0 3 6 1 4 7 2 5 0 { +. r. 2p1* (i.8)%8
```

$$\begin{pmatrix} 0 & 3 & 6 \\ 1 & 4 & 7 \\ 2 & 5 & 0 \end{pmatrix}$$

3 指数関数 $f(x) = e^{ix}$ を 3D で描く

$$e^{i\theta} = \cos(\theta) + i\sin(\theta)$$

である。(オイラーの公式)

オイラーは大胆に次のように e^{ix} を級数展開した

$$e^{ix} = 1 + ix + \frac{(ix)^2}{2!} + \frac{(ix)^3}{3!} + \dots$$

i の整数乗は周期 4 で繰り返す性質を持っている。 $i = \sqrt{-1}$, $i^2 = -1$, $i^3 = -i$, $i^4 = 1$

$$e^{ix} = 1 + ix - \frac{x^2}{2!} - \frac{ix^3}{3!} + \frac{x^4}{4!} + \dots$$

大胆に式の順序を変えた

$$e^{ix} = \left(1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots\right) + i\left(x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots\right)$$

既に $\cos x$, $\sin x$ の級数は知られていた。そして次の式のたどり着いた。

$$e^{ix} = \cos x + i\sin x$$

3.1 複素数のグラフィックス座標

複素数の幾何学的解釈は次の 3 人がほぼ同時に考えたがベッセル座標と言われることが少ないのはデンマーク語で書かれていたからだった。

*2

- 1799 ノルウエーの測量技師 カスパー・ベッセル
- 1806 アルガン (フランス)
- 1810 ガウス

*2 フランス、ロシア系はアルガン座標、ドイツ系はガウス座標の名で呼ぶ。最小二乗法などガウスにはこの対立が多く付きまとう

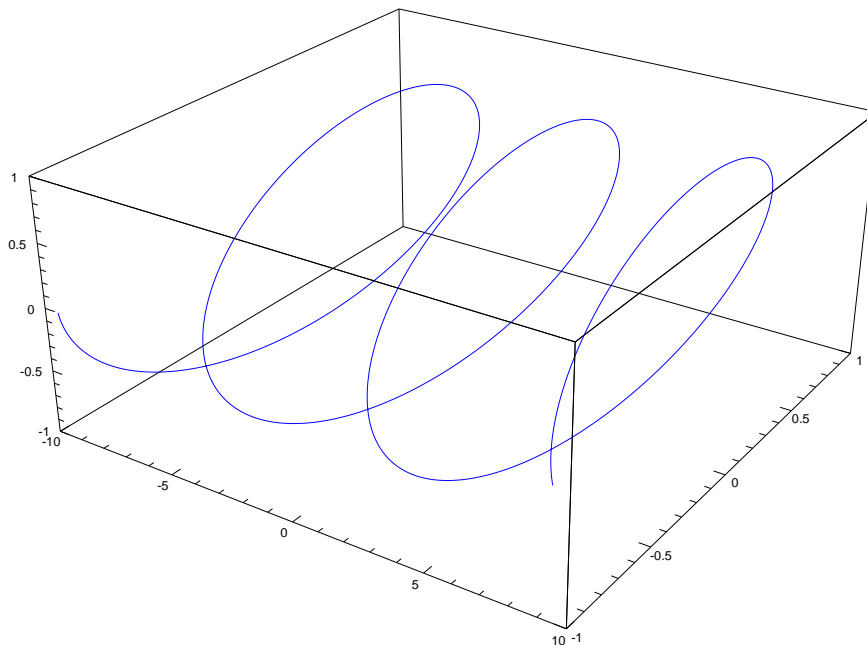
複素数の実態は数に忍び入った双子 (2次元のベクトル) であった。これはガウスの仕事には粗雑ではないかと思ったが代数幾何では $2D$ で十分であり、複素グラフィックスはやはりコンピューターの領域のようだ。

これら双方を $3D$ で描いてみよう。

3.2 指数関数 $f(x) = e^{ix}$

-3π から 3π の間で $f(x) = e^{ix}$ のグラフを描く

```
plot {@|: a,. +. r. a=. steps _3p1 3p1 1000
```



3.2.1 経過と説明

1. `steps` は `numeric.ijs` に入っているツール。 `require 'plot numeric trig'`
2. `steps` での区間の設定 -3π から 3π までを 1000 で刻む
`a=.steps _3p1 3p1 1000`
3. e^{ix} を $r. a$ (`length/angle`) で求める
4. 複素数を実数と虚数に分離 `+. r. a`
5. 縦 3 列のマトリクスを `x; y; x` の列にした後、{@|: で J の `plot` のフォーム (横のボックス) にあわせる

次は円になる

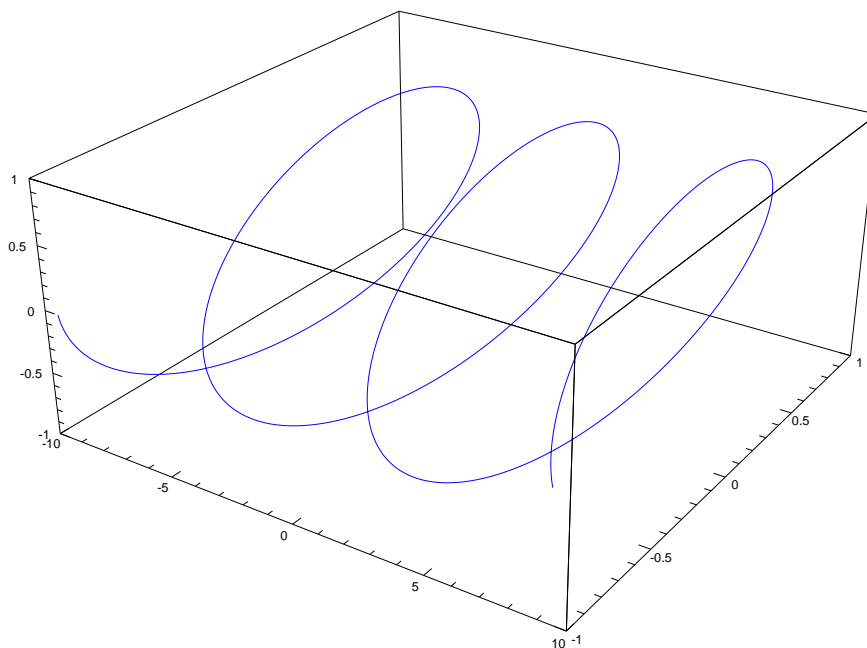
```
plot { |: +. r. steps _2p1 2p1 1000
```

```
plot { |: +. (cos a) +j. sin a=. steps _2p1 2p1 1000 NB. j. sin の順に
```

3.3 $\cos\theta + i\sin\theta$

$e^{i\theta} = \cos(\theta) + i\sin(\theta)$ である。オイラーの公式は 恒等式である

```
plot {@|: a,. +. (cos a) + j.sin a=. _3p1 3p1 steps 1000
```



4 ジュリア集合とマンデルブロート集合

マンデルブロート集合 .

マンデルブロート集合は次のざん化式からなる。 z_n が無限大に発散しない c の値を図示したもの

$$\begin{cases} z_{n+1} = z_n^2 + c & (N = 0, 1, 2) \\ z_0 = 0 \end{cases}$$

1. $c = 2$ の場合 → 発散する

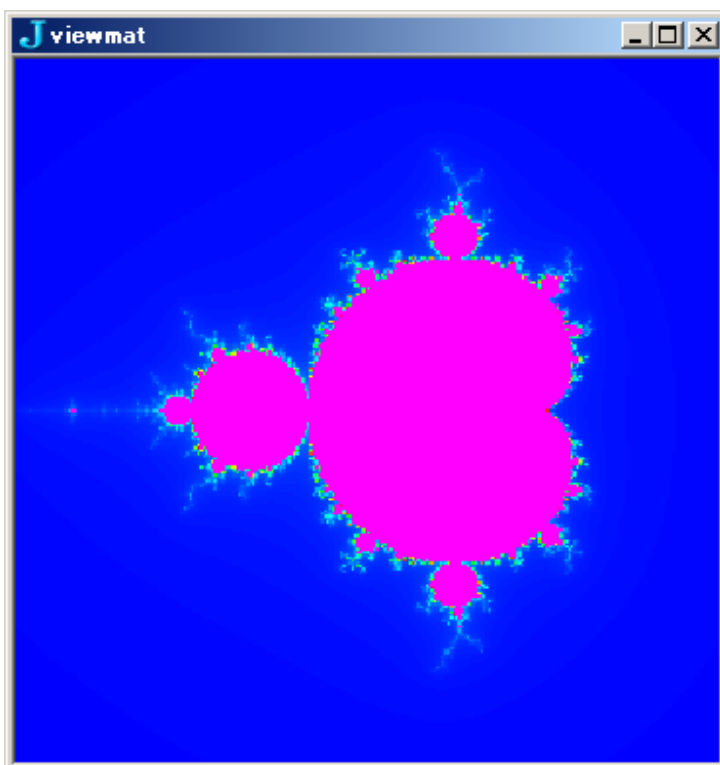
$$\begin{aligned}
z_1 &= 0^2 + 2 &= 2 \\
z_2 &= 2^2 + 2 &= 6 \\
z_3 &= 6^2 + 2 &= 38 \\
z_4 &= 38^2 + 2 &= 1446 \\
&\vdots
\end{aligned}$$

2. 複素数 $c = 0.2i$ の場合 $z_{n+1} = z_n^2 + c$

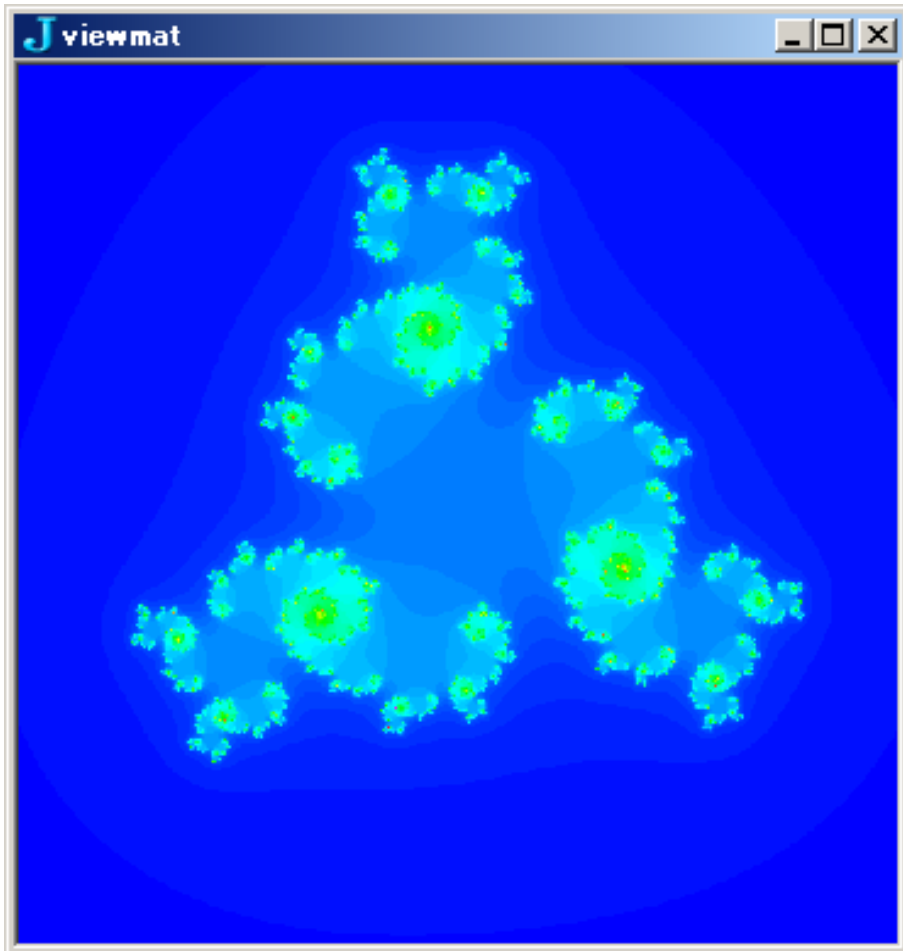
$$\begin{aligned}
z_1 &= 0^2 + 0.2i &= 0.2i \\
z_2 &= (0.2i)^2 + 0.2i &= -0.04 + 0.2i \\
z_3 &= (-0.04 + 0.2i)^2 + 0.2i &= -0.0384 + 0.184i \\
&\vdots
\end{aligned}$$

この点は収束するが、複素数でも収束と発散するポイントが近接している。マンデルブロート集合の図は収束と発散の境界を示す

```
viewmat mandelt fmx _2j_1.5 256
```



```
ジュリア集合 . f7=: +&_0.2j0.8@(^&3) NB. triple -> add _0.2j0.8
viewmat f7 julia ''
```



Script *C.Reiter* による

```

NB. -Julia-----
require 'viewmat'

f5=: +&_0.2j0.8@*: NB. square -> add _0.2j0.8
f6=: +&_0.2j0.6@*: NB. square -> add _0.2j0.6
f7=: +&_0.2j0.8@(^&3) NB. (fine!) triple -> add _0.2j0.8
f8=: +&_0.4j0.6@*: NB. fine square -> add _0.2j0.8
NB. direct formula
NB. viewmat ((+&_0.678j0.312@*:) julia0 4 100) fjx _1.5 512
fj0=: 3 : '+&y@*:'

```

```

fjx=: 3 : '|.|: j./ ~ ({. y)+3*(i.%<:) {: y'
NB. jullia fx(make canvas)
esc0=: 1 : 0 NB. for test
NB. f4 esc 0.3 1 // OK
(,u@{:})^:(<&3@# *. <&10@:(+/@:|@:{:})^:_ ,: y
)

escapet=: 2 : 0
NB. (f4 esc 10 3) 0.3 1 // OK
NB. Usage: #@ (f4 esc 10 3) 0.3 1
NB. exchange complex number to color code
'a0 b0'=: n
TMP=.(,u@{:})^:(<&b0@# *. <&a0@:(+/@:|@:{:})^:_ ,: y
# TMP
)

julia0=: 2 : '>(u escapet n) L:0 {@> y'
julia=: 2 : '>(u escapet 4 256) L:0 {@> fjx _1.5 512'
NB. Usage: viewmat (+&_0.678j0.312@*:) julia ''
NB. Usage: (f5 julia0 4 100) fjx _1.5 512
NB. fjx n1 n2 is position

NB. ---mandelbrot-----
mandelt0=: 3 : 'y &+@*: escapet(10 255) 0'
mandelt=: 3 : '> mandelt0 L:0 {@> y'
NB. mandelbrot fx
fmx=: 3 : '({. y) + 3*|.|: j./~ (i.%<:) {: y'
NB. viewmat mandelt fmx _2j_1.5 256
NB. fx (x=)_2j_1.5 (n=)5/15/256
NB. (m;n) position parameter
NB. m is left<-->right n is up/down
NB. _2j_1.5 is position <-- center
NB. 3* is scale 3 is just!!

```

付録 A J と複素数

A.1 J の複素数関数

複素数を生成 .

単項		両項	
<i>imaginary</i>	$j. y \rightarrow 0j1 * y$	<i>complex</i>	$x j. y \rightarrow x + j. y$
$j. y$	$j. _3$	$x j. y$	$3 j. 4$
	$0j_3$		$3j4$

極座標 極座標の絶対値と偏角 (ラジアン) を求める

単項		両項	
*. <i>Length/Aangle</i>	*. 3j4 5 0.927295	<i>LCM/And</i>	
<i>r.</i> <i>Angle</i>	$r. y \rightarrow \hat{j}. y$ 絶対値 l とした値の複素数 表示 r. 3 _0.989992j0.14112 r. 3 1 $e^{i\theta}$ である 1x1 ^ j. 3r4 0.731689j0.681639 r. 3r4 0.731689j0.681639	<i>r.</i> <i>Polar</i>	$x r. y \rightarrow x * r. y$ r. 3 _0.989992j0.14112 4 r. 3 _3.95997j0.56448

実数と虚数の分離 グラフィックスでは、 xy 座標 (デカルト、アルガン/ガウス座標) なら x,y の値になる。

単項		両項	
+. <i>Real/Imaginary</i>	+. 3j4 4j3 3 4 4 3	<i>GCD/or</i>	

+ <i>conjugate</i> 共役	$3j_4 \ 4j_3$ $3j_4 \ 4j_3$ + $3j_4 \ 4j_3$ $3j_{-4} \ 4j_{-3}$	<i>plus</i>	
-----------------------------	--	-------------	--

A.2 複素数の四則演算

	$\alpha_1 = a_1 + ib_1 \quad \alpha_2 = a_2 + ib_2$	$a = 3j_4 \ 5j_6$ +. a 3 4 NB. a1 b1 5 6 NB. a2 b2
和	$\alpha_1 + \alpha_2 = (a_1 + a_2) + i(b_1 + b_2)$	 + / a 8j10 + / +. a 8 10
差	$\alpha_1 - \alpha_2 = (a_1 - a_2) - i(b_1 + b_2)$	 - / a _2j_2 - / +. a _2 _2

積	$\alpha_1\alpha_2 = (a_1a_2 - b_1b_2) + i(a_1a_2 + b_1b_2)$	<pre> */ a NB. alpha1 * alpha2 _9j38 -/ */ +. a _9 +/. * +. a 38 NB. +/. * パワー 行列式の要素をかける </pre>
---	---	--

1. 複素数のノルム

$\alpha_x + i\alpha_y$ とすると

$$\|\alpha\| = \sqrt{\alpha_x^2 + \alpha_y^2}$$

```

| 3j4 5j6
5 7.81025

```

```

%:@+/@: ^&2 @ +. 3j4 5j6
5 7.81025

```

$$\|\alpha\|^2 = \alpha^* \alpha$$

```

(+ 3j4 5j6) * 3j4 5j6
25 61

```

```

^&2 | 3j4 5j6
25 61

```

A.2.1 各種の複素数演算

1. $i^2 = -1$

```

clean a, . ^& (a=.1 2 3 4 5 6) j. _1
x^n i
1 0j_1
2 _1
3 0j1
4 1
5 0j_1
6 _1

```

2. 加法定理 $\cos(\theta_1 + \theta_2) = \cos\theta_1\cos\theta_2 - \sin\theta_1\sin\theta_2$

```

] theta=: {"1 *. 3j4 4j3
0.927295 0.643501
clean cos +/- theta
0
*/ L:0 (cos ;sin)theta
+-----+-----+
|0.48|0.48|
+-----+-----+
- / ; */ L:0 (cos ;sin)theta
0

```

3. ド・モアブルの公式

$$\cos n\theta + i\sin\theta = (\cos\theta + i\sin\theta)^n$$

```

(cos 3 *theta) + j. sin 3* theta
_0.936j0.352 _0.352j0.936

```

```

^&3 (cos theta) + j. sin theta
_0.936j0.352 _0.352j0.936

```

4. オイラーの公式

$$e^{i\theta} = \cos\theta + i\sin\theta, \quad -\pi \rightarrow \pi$$

```
clean (i.11),.a,.(^ j.a) ,.(cos a) + j. sin a=. steps _1p1 1p1 10
```

No.	a	e ^{i theta}	cos theta + i sin theta
0	_3.14159	_1	_1
1	_2.51327	_0.809017j_0.587785	_0.809017j_0.587785
2	_1.88496	_0.309017j_0.951057	_0.309017j_0.951057
3	_1.25664	0.309017j_0.951057	0.309017j_0.951057
4	_0.628319	0.809017j_0.587785	0.809017j_0.587785
5	0	1	1
6	0.628319	0.809017j0.587785	0.809017j0.587785
7	1.25664	0.309017j0.951057	0.309017j0.951057
8	1.88496	_0.309017j0.951057	_0.309017j0.951057
9	2.51327	_0.809017j0.587785	_0.809017j0.587785
10	3.14159	_1	_1

5. $e^{\pi i} = -1$

```
clean ^ j. 1p1
_1
```

A.3 複素数と plot

J の *plot* は 3D や複素数を描く機能を持っている。

複素数は $3j4$ などのままでアルガン/ガウス座標に描くことができるがデカルト座標との混在はできない

References

H.S.M. コセクター (銀林 浩訳) 「幾何学入門」(上下) 明治図書出版 (1982)/ちくま学芸文庫 (2009)

E. マオール (伊理由美訳) 不思議な数 e の物語 岩波書店 1999

D. マンフォード、C. シリーズ、D. ライト (小森洋平訳) 「インドらの真珠ークラインの夢見た世界」日本評論社 2013

金谷一朗 「3D-CG プログラマのためのクォータニオン入門」工学社 2004

土基善文 「 x の x 乗のはなし」日本評論社 (はじめよう数学シリーズ 5) 2002