複素数とグラフィックスー円周等分方程式を巡って

SHIMURA Masato
jcd02773@nifty.ne.jp

2018 年 10 月 30 日

目次

概要

複素数は幾何との親和性がよい。更に J は複素数にも配列環が通っており、数として普通に用いることができ、特別ののファイルを読み込んだり、複素数と断って演算する必要はない。

1 複素数の準備

- 複素数は数学では $a + bi$ と表すが、電気やコンピュータの分野では ajb と一つの数で表現する
- 複素数の生成には j. を用いる
- J では他に pi を 1p1、e を 1x1 として数として取り込んでいる

```
    3.1j_2.2   j._0.9
```

- r. （Angle） $e^{i\theta}$ を計算する

```
    r. 2
_0.416147j0.909297
   ^ j.2
```

```
_0.416147j0.909297
```

- *. Length Angle/極座標変換
  5 は 3j4 のピタゴレアン。Angle は偏角 *arctan* でラジアンで表示

```
      *. 3j4
5 0.927295
```

- +. Real/Imaginary 実部、虚部分離

```
  +. 3j4
3 4
```

2　複素数の定着

1. カルダノ
   3 次方程式の解の公式の過程に複素数は姿を見せるがカルダノは無視した
2. ボンベリ（1526-1572 ボロニア）は干拓土木の技師であったが、手隙の間に数学をやり、カルダノの公式の解の三乗根を開く方法を追求し、代数学の著書は広く読まれた
3. ド・モアブル
   ド・モアブルは 18 歳の時、ルイ 14 世がナントの勅令を廃止し国内の 5% の新教徒を排撃したのでイングランドに逃れ、溢れかえるユグノーの中で市井の数学教師として過ごした。負の数や複素数では足下が覚束ないニュートンに一目置かれ、ライプニッツ攻撃の砲塔も務め、ロイヤルアカデミーの会員に推挙されている。
   ド・モアブルの公式
   ボンベリから 150 年近くたって複素数は華麗な姿を現す

$$(cos\theta + isin\theta)^n = cosn\theta + isinn\theta$$

$\theta = \frac{2}{\pi}$ としてみると

$$(cos\frac{2}{\pi} + isin\frac{2}{\pi})^3 = cos\frac{6}{\pi} + isin\frac{6}{\pi}$$

4. オイラーの美しい公式

ド・モアブルとオイラーの年齢差はちょうど 40 歳

オイラーの公式でド・モアブルの公式と指数関数の橋渡しがなされる

$$e^{i\theta} = cos\theta + isin\theta$$

（この種の計算は慣れが必要、clean は数値計算の微細な塵をとる J の関数）

```
 require 'numeric trig'

    (cos;j.@ sin) 1r6p1  NB. cos 1/6pi  + i 1/6pi
+--------+-----+
|0.866025|0j0.5|

+--------+-----+

     ^  j.1r6p1            NB. e^(i 1/6 pi)
0.866025j0.5
```

このことは指数の計算規則の複素数版である

$$e^a \times e^b = e^{a+b}$$

もう一つの公式も確認しておこう

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

```
   j. 1p1
```

```
   0j3.14159
      clean  ^  j. 1p1
   _1
```

3　グラフィックスは極座標で

3.1　極形式

複素数は極形式という表し方があり、$z = a + bi$ のとき $r = \sqrt{a^2 + b^2}$ となる r がとれ、$z = r(cos\theta + isin\theta)$ と表すことが出来る

*. (Length Angle/極座標変換) を用いると極形式に変換できる

Length はピタゴレアンであり、Angle は arctan(tan^{-1}) でラジアンで表される。

```
require 'trig '
   arctan 1r4p1
0.665774
   _3&o. 1r4p1
0.665774

      *. 3j4
5 0.927295
```

$$z = y(cost + isint)$$

（長さ r, 偏角 t）

3.2　回転行列

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \begin{pmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

極座標で次の掛け算を実数と複素数とに整理して実数を x 軸に、虚数を y 軸に割り振ると回転行列と同じようになる

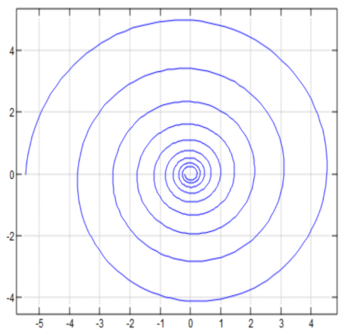
$$(a+bi)(cost+isint)$$
$$=(acost-bsint)+(asint+bcost)i$$

実際の計算は拡大、移動と組み合わせた同時行列を使ったほうが便利である。

3.3 plot の極座標の機能

かって描いたグラフィックスを見ていたらアルキメデスの渦巻きを描いた次のスクリプトがあった。

- th0 は０から１００までを１０００等分して $\frac{1}{12}\pi$ を掛けた直線
- th0;-th0 は plot に渡す XY のフォーム。th0;th0 とすると右回りになる
- polar 1 は plot の機能で、データを極座標に変換する

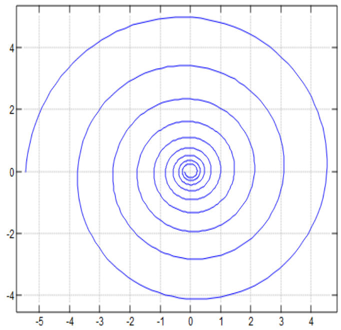



```
require 'plot numeric trig'
th0=: 1r12p1 * steps 0 100 1000
'polar 1' plot th0;-th0

 pd 'save png c:/temp/arch0.png'
```

`plot  th0 r. th0=. 1r12p1 * steps 0 100 1000` でも OK

ベルヌイ螺旋も描いておこう

```
plot (^ 0.06* t) r.  t=: steps _9p1 9p1 1000
```



x r. y は r. の両項型で x はスケーリングファクター

`plot r. steps _1p1 1p1 100` は円になる

4　円周等分方程式

ガウスの表記法

$$X = \frac{x^{n-1}}{x-1}$$

分子 x^{n-1} は $x-1$ で割り切れるので X は次の多項式となる

$$X = x^{n-1} + x^{n-2} + \cdots + x + 1$$

$$x^{n-1}$$

この式は重根を持たないので n 個の解をもち、複素数平面の単位円周上に等間隔で並ぶ。

1 の n 乗根は全部で n 個あるが，それらの和は 0 である。

$n = 4$ の場合

$x^4 - 1 = 0$
$(x^2 - 1)(x^2 + 1) = 0$
$x = \pm 1, x = \pm i$

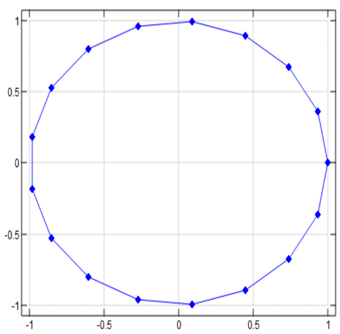
```
   p. _1 0 0 0 1
+-+--------------+
|1|1 0j1 0j_1 _1|
+-+--------------+

 +/  1 0j1 0j_1 _1
0
```

$z^n = 1$ となるような複素数 z を複素平面上にプロットすると半径 1 の円に内接する正 n 角形の頂点全体となる。

ド・モアブルの公式により、複素平面上で $\dfrac{360}{2}$ ずつ順番に回転していった数は全て 1 の n 乗根になることを利用している

円周等分方程式の解を求めてから作図するのは順序通りに整列するのが大変なので次のアルゴリズムを利用し、簡潔に多角形を描くことが出来る

4.1 複素数で多角形を描く



1. 次の方法は正多角形の作図に愛用している J のイディオムである
   これは 2π の円周を n 分割した上で各点の $e^{i\theta}$ の値を求めている。
   (a) 正四角形の辺の位置の計算

```
        clean  +. r. 2p1* (i.4)%4
    1  0
    0  1
```

```
_1  0
 0 _1
```

（b）ガウスの正 17 角形を描く

```
 'line marker' plot {|: (+.r. 2p1* (i.17)%17),1 0
```

（c）複素極座標での計算 J の r. は次の計算をしている

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

（d）円を描く（正３６０角形で代用）

```
plot r. 2p1 *(i.360)%360
```