三角関数を使わないで円を描く方法

2006 年 9 月 2 日

コンピュータの画面上で任意の円を描くことは、既に用意されている描画メソッド (Java であれば、drawOval() など) を用いることで容易にできる。これらの描画メソッドでは、三角関数を用いて円を描く処理をしているかと言えば、そうではない筈である。三角関数 (実際の三角関数は、勿論、これを級数展開したもので近似計算する筈であるが) を用いて、いくつかの点を結んで円 (や楕円) を描くのでは余りに時間がかかり過ぎる。これでは、普通は、瞬時に円を描くことは困難になる。実は、(近似的な) 円を描くには、三角関数を使用しなくてもできるのである。今回は、三角関数を使わないで円を描く手続きについて考えてみることにする。

では、実際に円 $x^2 + y^2 = r^2$ を三角関数を利用しないで描く手続きについて考えてみる。ここで、この円の方程式の両辺を x で微分すると、次の微分方程式が得られる。

$$\frac{\mathrm{d}y}{\mathrm{d}x} = -\frac{x}{y} \tag{1}$$

この微分方程式の解は元の円の方程式になる筈である。

実際に、コンピュータの画面上で円を描画するには、微分方程式 (1) を数値的に解いて、一定数の点の集合 (x_i,y_i) を決定し、これらの点を線で結べば良い。そこで、ここでは微分方程式を数値的に解くのに、最も簡単な Euler 法を用いることにする。

微分方程式 $\dfrac{\mathrm{d}y}{\mathrm{d}x} = f'(x)$ を解き、$y = f(x)$ を求めるのに用いる Euler 法とは、次のような方法である。微分方程式の初期値として、点 (x_0,y_0) が与えられるので、任意の微少量変化量 h を考え、次の漸化式 (2) で、(x_1,y_1)、(x_2,y_2) $\cdots$ を次々の決定していくのである。

$$\begin{aligned} x_{n+1} &= x_n + h \\ y_{n+1} &= y_n + h \cdot f'(x_n, y_n) \end{aligned} \tag{2}$$

従って、円の微分方程式 (1) では、

$$f'(x_n, y_n) = -\frac{x_n}{y_n}$$

であるから、漸化式 (2) より

$$\begin{aligned} x_{n+1} &= x_n + h \\ y_{n+1} &= y_n - h \cdot \frac{x_n}{y_n} \end{aligned} \tag{3}$$

が得られる。ここで、Euler 法では、h は一定値とするが、実際は特に一定にする必要はなく、一般的には

$$x_{n+1} = x_n + h \cdot n$$
$$y_{n+1} = y_n - h \cdot n \cdot \frac{x_n}{y_n}$$

とできる。そこで、問題は $h \cdot n$ をどのように決めれば良いかということになるが、ここでは、任意の定数 A を用意して、次式のように $h \cdot n$ を決めることにする。

$$h \cdot n = A \cdot y_n$$

この方法を採ると、漸化式は

$$\begin{aligned} x_{n+1} &= x_n + A \cdot y_n \\ y_{n+1} &= y_n - A \cdot x_n \end{aligned} \tag{4}$$

と書き換えることができる。これにより、原点を中心にした半径 r の円は、初期条件 $x_0 = r$, $y_0 = 0$ で、漸化式 (4) で与えられる点 (x_1,y_1)、(x_2,y_2) $\cdots$ を結ぶことで描くことができるのである。

最後に、具体的に計算した結果を揚げておことにする。x_0=1.0,y_0=0 として、A を適当に決めた場合、A と共に x^2+y^2 の値がどのように変化するかを確認してみた (表.1)。これが、1 になれば正しい円 (x^2+y^2=1) になる筈である。

表 1: 計算結果

A	$x_{100}^2+y_{100}^2$	$x_{500}^2+y_{500}^2$	$x_{1000}^2+y_{1000}^2$
0.1	2.704813817	144.7727727	20959.15600
0.01	1.010049662	1.051268471	1.105165397
0.001	1.000100005	1.000500125	1.001000501

A を小さくすると (計算回数は増えるが)、それだけ真円に近い円を描くことができることが分かると思う。実際には、どの程度の大きさの円を画面上に描くかで計算回数を変えることができるし、2^n 分の 1 の円弧を描く手続きを作成すれば、円全体を描くことができるから、更に計算回数は少なくてすむ筈である。

最後に、数式処理ソフト Maple を使って、漸化式 (4) を計算 (A=0.025 とした) して、近似円を描かせた結果を載せておく。Maple の手続きは、

```
> x[0]:=1;
> y[0]:=0;
> for i from 1 to 2520 do
>   x[i]:=x[i-1]+0.0025*y[i-1];
>   y[i]:=y[i-1]-0.0025*x[i-1];
> end do:
> datalist:=[seq([x[j],y[j]],j=0..2520)]:
> plot(datalist,color=black,thickness=2);
```

であり、近似円のグラフは、図.1 となる。

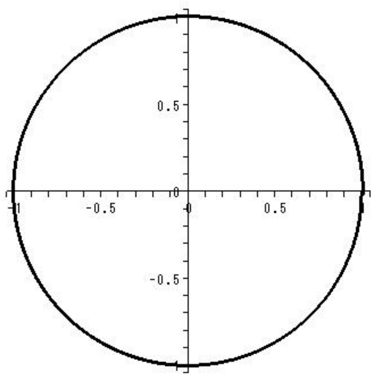

図. 1: Maple による近似円のグラフ