橢圓繪圖的演算法

翁義聰

一、前　言

若橢圓交 x 軸於 $(b, 0)$、$(-b, 0)$ 且交 y 軸於 $(0, a)$、$(0, -a)$，則其方程式通常以無參數式或參數式表示如下：

$$a^2x^2+b^2y^2=a^2b^2 \cdots\cdots\cdots(1)$$

$$\begin{cases} x=b\cos\phi \\ y=a\sin\phi \end{cases},\ 0\leqslant\phi\leqslant 2\pi \cdots\cdots\cdots(2)$$

在電腦繪圖時常利用旋轉及對稱的技巧[1]，給一個快速的演算法；本文先給圓的演算法，並證明誤差小於½，並依此推廣為橢圓的演算法，結束前我們試著把所有運算簡化為最基本的運算。

二、繞直角坐標原點旋轉的演算法

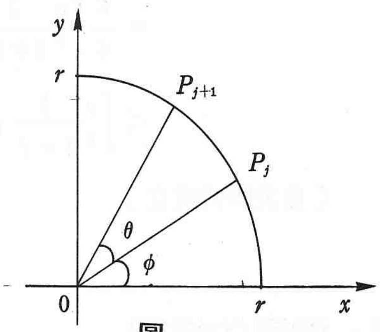
我們認為圓 $x^2+y^2=r^2$ 上一點 $P_j(r\cos\phi, r\sin\phi)$ 以原點為軸逆時針旋轉弳度 θ 後到 P_{j+1}，如圖一所示，而 P_{j+1} 的坐標可由 P_j 點求得：

$$[r\cos(\phi+\theta), r\sin(\phi+\theta)]$$

$$=[r\cos\phi, r\sin\phi]\cdot\begin{bmatrix}\cos\theta & \sin\theta \\ -\sin\theta & \cos\theta\end{bmatrix}$$

$$\cdots\cdots\cdots(3)$$

當(3)式中的 θ 趨近 0，則 $\cos\theta=1$、$\sin\theta=\theta$；我們可簡化(3)式的運算及運算次數，式中的三角函數亦省略了，若 P_j 點坐標為 (x_j, y_j)

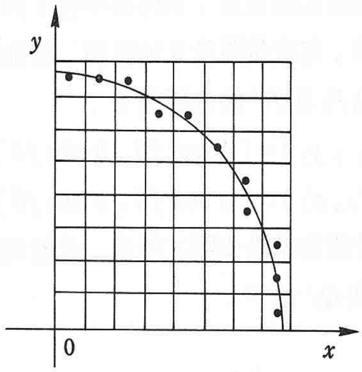


圖一

、P_{j+1}點坐標爲（x_{j+1}，y_{j+1}），則

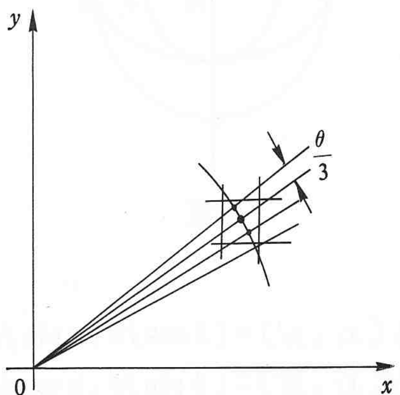
$$\begin{cases} x_{j+1} = x_j - y_j * \theta \\ y_{j+1} = y_j + x_j * \theta \end{cases} \quad \cdots\cdots\cdots\cdots\cdots\cdots(4)$$

現在剩下的問題是：經過多次的遞迴計算，誤差會不會太大？因爲我們的目的祇在螢幕上繪圖，而螢幕上的圖形是由沿曲線上（或旁）一個一個亮點表示，我們只要顯示有限個點即可表示（如圖二），故不需把旋轉逕度訂得



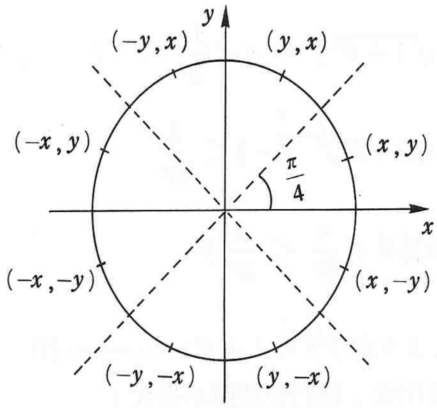
圖二

太小，訂得太小好幾次計算所得的結果顯示在螢幕上，還是同一個亮點（如圖三）。



圖三

設旋轉次數爲N，則經N次旋轉與描點可產生 1/8 個圓，再由對稱性質產生整個圓，如圖四所示，令



圖四

$$N = 2r = \frac{\pi}{4} \div \theta$$

$$\theta = \frac{\pi}{8r} \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots(5)$$

圓的演算法敍述如下：

1. 輸入圓的半徑 r
2. 計算旋轉逕度 $\theta := \pi / 8r$
3. 計算旋轉次數 $N := \pi / 4\theta$
4. 令 $X := r$，$Y := 0$
5. 當 $N > 0$ 時執行
6. 　$X := X - Y * \theta$
7. 　$Y := Y + X * \theta$
8. 　$N := N - 1$
9. 　顯示 8 個對稱點
10. 結束

三、證明圓的演算法誤差小於$\frac{1}{2}$

設旋轉前的點爲（r，0），旋轉 j 次後的點爲P_j，與原點距離爲r_j，則數列（r_j）爲一公比$\sqrt{1+\theta^2}$ 的遞增等比數列：

因爲 $r_{j+1}^2 = x_{j+1}^2 + y_{j+1}^2$

$$= (x_j - y_j * \theta)^2 + (y_j + x_j * \theta)^2$$

$$= (x_j^2 + y_j^2) \cdot (1 + \theta^2)$$

所以 $r_{j+1} = r_j * \sqrt{1+\theta^2}$

對圓的繪圖而言，旋轉$N=\pi/4\theta$次才能完成 1/8 個圓。我們想要證明：$r_N - r < 1/2$，亦即證明

$$r(\sqrt{1+\theta^2})^N - r < \frac{1}{2}$$

$$\Leftrightarrow (1+\theta^2)^{\frac{1}{2}\cdot\frac{\pi}{4\theta}} - 1 < \frac{1}{2r}$$

（因爲 $\theta = \dfrac{\pi}{8r} < \dfrac{1}{2r}$）

$$\Leftarrow (1+\theta^2)^{\frac{\pi}{8\theta}} < 1+\theta \cdots\cdots\cdots\cdots\cdots (7)$$

爲了證明(7)式，我們先證明不等式：

$$\frac{\pi}{4}\int_0^\theta \frac{1}{1+t^2}\,dt < \int_0^\theta \frac{1}{1+t}\,dt \cdots\cdots (8)$$

證明：因爲 $\dfrac{\pi}{4}\displaystyle\int_0^\theta \frac{1}{1+t^2}\,dt - \int_0^\theta \frac{1}{1+t}\,dt < 0$

$$\Leftrightarrow \int_0^\theta \left(\frac{\pi}{4}\cdot\frac{1}{1+t^2} - \frac{1}{1+t}\right)dt < 0$$

$$\Leftrightarrow \int_0^\theta \frac{\pi(1+t)-4(1+t^2)}{4(1+t^2)(1+t)}\,dt < 0$$

$$\Leftarrow \pi(1+t)-4(1+t^2) < 0$$

$$\Leftrightarrow -4t^2+\pi t+(\pi-4) < 0$$

$$因 -4\left(t^2-\frac{\pi}{4}t\right)+(\pi-4)$$

$$= -4\left(t-\frac{\pi}{8}\right)^2+\frac{\pi^2}{16}+\pi-4$$

$$\leqslant \frac{\pi^2}{16}+\pi-4$$

$$< 0$$

故得證。

現在證明(7)式如下：

$$(1+\theta^2)^{\frac{\pi}{8\theta}} < 1+\theta$$

$$\Leftrightarrow \frac{\pi}{8\theta}\ln(1+\theta^2) < \ln(1+\theta)$$

$$\Leftrightarrow \frac{\pi}{8\theta}\int_0^\theta \frac{2t}{1+t^2}\,dt < \int_0^\theta \frac{1}{1+t}\,dt$$

$$因 \frac{\pi}{8\theta}\int_0^\theta \frac{2t}{1+t^2}\,dt \leqslant \frac{\pi}{8\theta}\cdot 2\theta\cdot\int_0^\theta \frac{1}{1+t^2}\,dt$$

$$= \frac{\pi}{4}\int_0^\theta \frac{1}{1+t^2}\,dt$$

$$< \int_0^\theta \frac{1}{1+t}\,dt$$

（由於(8)式成立）

四、橢圓的繪圖

若橢圓的方程式如(2)式所示，則依參數ϕ從 0 至2π可繪出一橢圓，但經過對稱性質與並行計算的使用，祇要計算$0 \leqslant \phi \leqslant \pi/4$的範圍，其計算次數如下：

$$r = \max\{a, b\}$$

$$\theta = \pi/8r$$

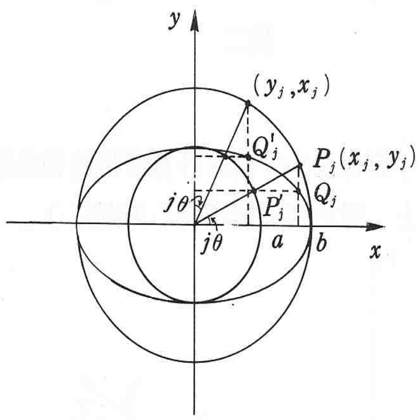
$$N = \pi/4\theta$$

利用第二段圓的演算法，同時將半徑b與a的兩圓上的點，每次做弳度θ的旋轉，當旋轉j次後所得點P_j與P_j'的坐標爲：

$$P_j(x_j, y_j) = [b\cos j\theta, b\sin j\theta]$$

$$P_j'(x_j', y_j') = [a\cos j\theta, a\sin j\theta]$$

將此並行計算的結果依圖五所示，分配到橢圓上得點Q_j與Q_j'如下：



圖五

$$Q_j(x_j, y_j') = [b\cos j\theta, a\sin j\theta]$$

$$Q_j'(y_j, x_j') = [b\sin j\theta, a\cos j\theta]$$

顯然地，他們都滿足方程式(2)。再進一步，仿前面所述讓θ趨於0，則第$j+1$次的點爲：

$$Q_{j+1}(x_{j+1}, y'_{j+1}) = [x_j - y_j * \theta, y'_j + x'_j * \theta]$$

$$Q'_{j+1}(y_{j+1}, x'_{j+1}) = [y_j + x_j * \theta, x'_j + y'_j * \theta]$$

這樣經過N次的遞迴計算，描點的誤差仍小於1/2。

橢圓的演算法如下：

1. 輸入橢圓的X、Y軸長b、a
2. 判斷長軸 $r := \max\{b, a\}$
3. 計算旋轉弳度 $\theta := \pi/8r$
4. 計算旋轉次數$N := \pi/4\theta$
5. 令$X := b$，$Y := 0$，$X' := a$，$Y' := 0$
6. 當$N > 0$時執行
7. $X := X - Y * \theta$
8. $Y := Y + X * \theta$
9. $X' := X' - Y' * \theta$
10. $Y' := Y' + X' * \theta$
11. $N := N - 1$
12. 顯示8個對稱點(X, Y')，(Y, X')，$(-X, Y')$，$(-Y, X')$，$(-X, -Y')$，$(-Y, -X')$，$(X, -Y')$，$(Y, -X')$
13. 結束

五、一個構想

曾有人把橢圓的演算法中的乘除法簡化爲加減法〔2〕，因此本文亦嘗試以不同的觀點進一步的簡化(4)式。我們將式中相當小的θ值以2^{-I}取代，得到

$$\begin{cases} x_{j+1} := x_j - y_j / 2^I \\ y_{j+1} := y_j + x_j / 2^I \end{cases} \quad \cdots\cdots\cdots\cdots\cdots (9)$$

其中的I爲滿足$2^I \geqslant r$的最小整數。

設某數以二進位狀態存在暫存器中，再除以2^I，即將此二進位狀態的位元數往右移(shift)I個位元(bit)，左邊空缺補0。

例如：$13.375_{10} \div 2^3 = 1101.011_2 \div 2^3$
$= 0001.101011_2$

因此，此類型的除法皆以右移位元運算(shift-right)代替，記爲shr。故建議將(9)式改爲：

$$\begin{cases} x_{j+1} := x_j - y_j \text{ shr } I \\ y_{j+1} := y_j + x_j \text{ shr } I \end{cases} \quad \cdots\cdots\cdots\cdots (10)$$

至此圓與橢圓的演算法皆簡化爲$0 \leqslant \theta \leqslant \pi/4$與基本運算了。

最後我們還是有些煩惱的問題：如某些電腦語言雖提供位元數右移或左移的運算，但仍限制此位元數爲整數型，以此執行(10)式誤差會擴大。如描點時使用四捨五入法會出現誤差。如$r_N - r < 1/2$式中這個值1/2，在r較小時就嫌大些。也因爲這些煩惱，以上僅稱它爲一個構想。

六、誌謝

本人在此感謝杜詩統教授在第三段證明時給予的指導，以及董世平師在演算法的指引。

七、參考資料

1. D. F. Rogers and J. A. Adams *Mathematical elements for computer graphics*, McGraw-Hill Book company, 1976, pp.92－107.
2. Jerry R. Van Aken *An Efficient Ellipse-Drawing Algorithm*, *IEEE CG & A*, Vol.4 No.9 Sept. 1984, pp.24－35.

(對本文相關程式有興趣的讀者歡迎來信與本人連繫)

——本文作者任教於崑山工專——