

Dibujando mediante iteraciones

M. Benito y J. J. Guadalupe

Existen muchos libros y artículos relacionados con el estudio de iteraciones de funciones holomorfas o meromorfas en el plano complejo. Su estudio teórico conduce a los sistemas dinámicos y a la aparición de conjuntos fractales. El conjunto de Mandelbrot es el más conocido de los que aparecen en la geometría fractal. Sus admiradores se complacen en asegurar que es el objeto más complicado de las matemáticas y, sin embargo, tiene una definición muy simple.

Para cada número complejo $c \in \mathbb{C}$ designamos por Q_c a la aplicación de \mathbb{C} en \mathbb{C} definida por

$$Q_c(z) = z^2 + c.$$

Dado un $z_0 \in \mathbb{C}$ llamamos órbita de z_0 por Q_c a la sucesión $\{Q_c^n(z_0)\}_{n \in \mathbb{N}}$.

Se denomina conjunto de Mandelbrot al subconjunto de \mathbb{C} formado por los c para los que la órbita de 0 por Q_c está acotada. Lo denotaremos por M .

Para cada número complejo c , se denomina conjunto de Julia J_c al subconjunto de \mathbb{C} formado por todos los $z_0 \in \mathbb{C}$ tales que su órbita por Q_c está acotada.

G. Julia [6] y P. Fatou [5] encontraron que, si la órbita de 0 por Q_c está acotada, entonces J_c es un conjunto conexo; en cambio, si no está acotada, J_c es un conjunto completamente desconexo.

La siguiente proposición ayuda a decidir cuándo la órbita de un punto no está acotada.

Si $|z| > \max\{2, |c|\}$, la órbita de z por Q_c tiende a infinito.

Su demostración es sencilla:

Sea $c = \rho e^{i\alpha}$, $z = r e^{i\beta}$. Por la condición $|z| > |c|$ existe un número real $\gamma_0 > 0$ tal que $r = \rho(1 + \gamma_0)$. Por tanto $|Q_c(z)| = |\rho(1 + \gamma_0)^2 e^{2\beta i} + \rho e^{i\alpha}| =$

$$\rho |r(1 + \gamma_0) e^{i2\beta} + e^{i\alpha}| \geq \rho |r(1 + \gamma_0) - 1| = \rho |r - 1 + r\gamma_0| \geq \rho |1 + 2\gamma_0|$$

Sean. $Q_c^j(z) = r_j e^{i\beta_j}$, $r_j = \rho(1 + \gamma_j)$, $j > 0$. Por lo anterior, $\gamma_j \geq 2\gamma_{j-1}$ y así $\gamma_j \geq 2^j \gamma_0$. Si $\rho \neq 0$, como ρ y γ_0 son fijos, la órbita de z por Q_c tiende a infinito. Si $\rho = 0$, entonces $|Q_0^n(z)| = |z|^{2^n} > 2^{2^n}$ y por consiguiente la órbita de z tiende a infinito.

Si $c = -2$, tenemos que $Q_{-2}(0) = -2$, $Q_{-2}(-2) = 2$ y $Q_{-2}(2) = 2$, luego la órbita de 0 por Q_{-2} está acotada. Así, $-2 \in M$.

Dado que $Q_c(0) = c$, $Q_c^2(0) = c^2 + c$, y que para $|c| > 2$, $|c^2 + c| = |c||c+1| > |c|$, la anterior proposición prueba que el conjunto de Mandelbrot está contenido en el círculo de centro 0 y radio 2.

Hay un aspecto en el estudio computacional de las iteraciones que escapa del análisis y que trata de mostrar la belleza de los dibujos que se pueden generar con ayuda de los ordenadores. Existen varias estrategias para generar dibujos mediante iteraciones y aquí describiremos alguna de ellas poniendo de manifiesto, en algunos casos, su estabilidad.

En la literatura son frecuentes programas como el programa 1, que intentan dibujar el conjunto de Mandelbrot; aquí lo hemos realizado en Ubasic [7], que nos permite intentar efectuar los cálculos en forma exacta, esto es utilizando números racionales y no aproximaciones decimales.

La parte real de c recorre el intervalo $[-2, 2]$ de centésima en centésima, y para cada uno de estos valores su parte imaginaria varía de centésima en centésima entre -2 y 2 . Para cada valor de c el programa calcula el número de iteraciones necesarias para que la órbita de 0 por Q_c alcance el exterior del círculo de centro en 0 y radio 2. Entonces asigna a la celdilla (*pixel*) correspondiente a ese valor de c un color en función de dicho número, dejando sin colorear (en negro) los puntos que en un determinado número k de pasos, fijado previamente, no ha alcanzado el exterior del círculo.

```

5      ' programa 1
6      ' dibuja el conjunto de Mandelbrot con calculos exactos
10     screen 23
20     for X=0 to 400
30     for Y=0 to 400
40     X1=(X-200)//100:Y1=(Y-200)//100
50     R=X1*X1+Y1*Y1
60     if R>4 then pset (X,Y),1:goto 180
70     X3=X1 : Y3=Y1
80     N=1
90     repeat
100    N=N+1
110    if N>10 then 180
120    X2= (X3*X3-Y3*Y3+X1)
130    Y2= (2*X3*Y3+Y1)
140    R2=(X2*1.)\^2+(Y2*1.)\^2
150    X3=X2 : Y3=Y2
160    until R2>4
170    pset (X,Y) , (N@16)+1
180    next Y
190    next X

```

¿Podemos estar seguros de que el dibujo representa con precisión el conjunto de Mandelbrot? ¿Cuántas iteraciones debemos calcular para estar seguros de que el punto c no está en M ?

Debemos hacer notar que el color asignado a una determinada celdilla puede depender del representante escogido para dicha celdilla, así la celdilla correspondiente a $c = -2$ quedará en negro, mientras que, si escogemos como representante de dicha celdilla $-\frac{4001}{2000}$, el color que le corresponde es el azul.

En nuestro ordenador, para un valor de k superior a 10 se produce un error de desbordamiento. Si se pretenden realizar ampliaciones de algunas zonas, el número de iteraciones abordables es aún menor.

Cambiando, en la instrucción 40, los símbolos // por /, los cálculos se realizan en forma aproximada. Así, se puede aumentar el valor de k ; un valor frecuente en la literatura es $k = 100$.

Esto deja patente que el color asignado a cada celdilla depende en este caso de la forma como el programa realice los redondeos.

Si el valor de un número real x aproximado hasta las milésimas es $2'437$, el número x verifica

$$2'4365 < x < 2'4375.$$

Del número $x + x$ sólo podemos afirmar que

$$4'8730 < x + x < 4'8750.$$

Por tanto, al efectuar cálculos aproximados en procesos iterativos, aunque la precisión de los números de partida sea de un número elevado de cifras, en general, en pocas iteraciones, los números obtenidos no tendrán ninguna precisión.

A pesar de los anteriores inconvenientes, debemos reconocer la belleza de los dibujos relacionados con el conjunto de Mandelbrot, teniendo siempre presente que ellos pueden depender de los procesos de redondeo realizados por el programa y de la tarjeta gráfica empleada.

Existen numerosas referencias sobre fractales y el conjunto de Mandelbrot, entre las cuales podemos citar [4], [8], [9], [3].

El método de las iteraciones de funciones ha sido utilizado por M. F. Barnsley para la generación de conjuntos fractales (véanse [1], [2]). Nosotros estamos interesados en una idea distinta para obtener dibujos mediante iteraciones. Para llevarla a cabo se usa el programa 2, que está realizado en Qbasic; la opción de pantalla elegida permite utilizar 16 colores, al negro le corresponde el número 0 y al blanco el número 15.

```

REM programa 2
INPUT «menor valor de x»; xp
INPUT «menor valor de y»; yp
INPUT «mayor valor de x»; xg
INPUT «mayor valor de y»; yg
INPUT «incremento de x»; xc
INPUT «incremento de y»; yc
INPUT «color del fondo 0=negro, 15=blanco»; f
INPUT «primera iterada a pintar»; k0
INPUT «cuantas iteradas quieres pintar»; k1
INPUT «desplazamiento del color»; k2
K0 = k0 - 1: k1 = k0 + k1
SCREEN 12
PAINT (0, 0), (f)
WINDOW (-3, 3) - (3, -3)
FOR x = xp TO xg STEP xc
FOR y = yp TO yg STEP yc
x1 = 0: y1 = 0: k = 0
70   xa = x1 * x1 - y1 * y1 + x
80   y1 = 2 * x1 * y1 + y: x1 = xa
k = k + 1: IF INKEY$ = CHR$(27) THEN STOP
r = x1 * x1 + y1 * y1
IF r > 18 THEN 180
IF k > k0 THEN PSET (x1, y1), k - k2
IF k < k1 THEN 70
IF ABS(x1) < 3 AND ABS(y1) < 3 THEN PSET (x1, y1), 0
180  NEXT y
NEXT x
210  IF INKEY$ <> CHR$(27) THEN 210

```

Tomamos el rectángulo de lados paralelos a los ejes, de vértice inferior izquierdo $x_p + iy_p$ y vértice superior derecho $x_g + iy_g$. El programa calcula las sucesivas iteradas $Q_c^n(0)$ para una secuencia de puntos que criba dicho rectángulo comenzando en el vértice inferior izquierdo, con pasos verticales y_c y, tras completar una pasada vertical, pasos horizontales x_c . Se empieza a colorear a partir de una determinada iterada numerada como k_0 , fijada inicialmente, de la siguiente manera: los puntos $Q_c^{k_0}(0)$ en color $k_0 - k_2$, $Q_c^{k_0+1}(0)$ en color $k_0 - k_2 + 1, \dots$, $Q_c^{k_0+k_1-2}(0)$ en color $k_0 + k_1 - 1 - k_2$ y $Q_c^{k_0+k_1-1}(0)$ en color negro; cuando una celdilla es alcanzada más de una vez, toma el color correspondiente a la última vez que ha sido alcanzada. El número de iteraciones

pintadas es k_1 , siendo k_2 una constante que provoca un cambio en el color asignado. El lenguaje de programación utilizado no permite notaciones como x_p , por lo que en el programa es sustituida por xp .

Las figuras 1 a 6 corresponden a los parámetros indicados en la tabla

	x_p	y_p	x_g	y_g	x_c	y_c	f	k_0	k_1	k_2
figura 1	-2	-2	-1	2	0'001	01001	15	2	15	0
figura 2	-2	-2	0	2	0'001	01001	15	2	15	0
figura 3	0	-2	0'24	2	0'001	01001	15	2	14	2
figura 4	-2	-2	0'26	2	0'001	0,001	15	4	12	2
figura 5	-2	-2	2	2	0'001	01001	15	2	15	0
figura 6	-2	-2	2	2	0'001	01001	15	2521	15	2520

Si en el programa 2 cambiamos las instrucciones

```
70      xa = x1 * x1 - y1 * y1 + x
80      y1 = 2 * x1 * y1 + y: x1 = xa
```

por

```
70      x1 = x1 * x1 - y1 * y1 + x
80      y1 = 2 * x1 * y1 + y
```

quedan definidas aplicaciones de \mathbb{C} en \mathbb{C} que designamos por $T_c(z)$. La función $T_{a+bi}(z)$ no es holomorfa en ningún abierto no vacío de \mathbb{C} , únicamente cumple las condiciones de Cauchy-Riemann en los puntos

$$z = \frac{1 \pm \sqrt{1-4a}}{2}, \quad \text{si } a < \frac{1}{4},$$

$$z = \frac{1}{2} \pm i \sqrt{\frac{4a-1}{12}}, \quad \text{si } a \geq \frac{1}{4}.$$

Igual que las funciones $Q_c(z)$, también se tiene que $T_c(0) = c$.

Las figuras 7 y 8 se han obtenido con los siguientes parámetros:

	x_c	y_p	x_g	y_g	x_c	y_c	f	k_0	k_1	k_2
figura 7	-2	-2	2	2	0'01	0'01	15	2	15	1
figura 8	-2	-2	2	2	0'01	0'01	15	60	15	59

La figura 7 empieza en $k_0 = 2$. Al aumentar el valor de k_0 las figuras parecen estabilizarse en dos dibujos de forma parecida pero de colores distintos.

Bibliografía

- [1]. Barnsley, M. F. (1988). *Fractals everywhere*. Academic Press.
- [2] Barnsley, M. F.; S. Demko. (1985). *Iterated function systems and the global construction of fractals*, Proc. R. Soc. London, A 399. pp. 243-275.
- [3] Devaney, R. L. (1999). *The Mandelbrot set, the Farey tree, and the Fibonacci sequence*, Amer. Math. Monthly, 106. pp. 289-302.
- [4] Guzmán, M. de; M. A. Martín; M. Morán; M. Reyes. (1993). *Estructuras fractales*. Labor, Barcelona.
- [5] Fatou, P. (1926). *Sur l'iteration des fonctions transcendentes entières*, Acta Math., 47, pp. 337-370.
- [6] Julia, G. (1918). *Memoire sur l'iteration des fonctions rationnelles*, J. Math., 8. pp. 47-245.
- [7] Kida, Y.: Ubasic, *Programa disponible en ftp : //rkmath.rikkyo.ac.jp/ubibm*
- [8] Mandelbrot, B. B. (1977). *The fractals geometry of nature*. Freeman and Co.
- [9] Peitgen, H. O.; P. H. Richter. (1986). *The beauty of fractals*. Springer Verlag.

Manuel Benito, Departamento de Matemáticas. I. Práxedes Mateo Sagasta. Logroño. mbenit8@palmera.pntic.mec.es

José J. Guadalupe (1945-2000). Departamento de Matemáticas y Computación. Universidad de La Rioja, Logroño



Fig. 1

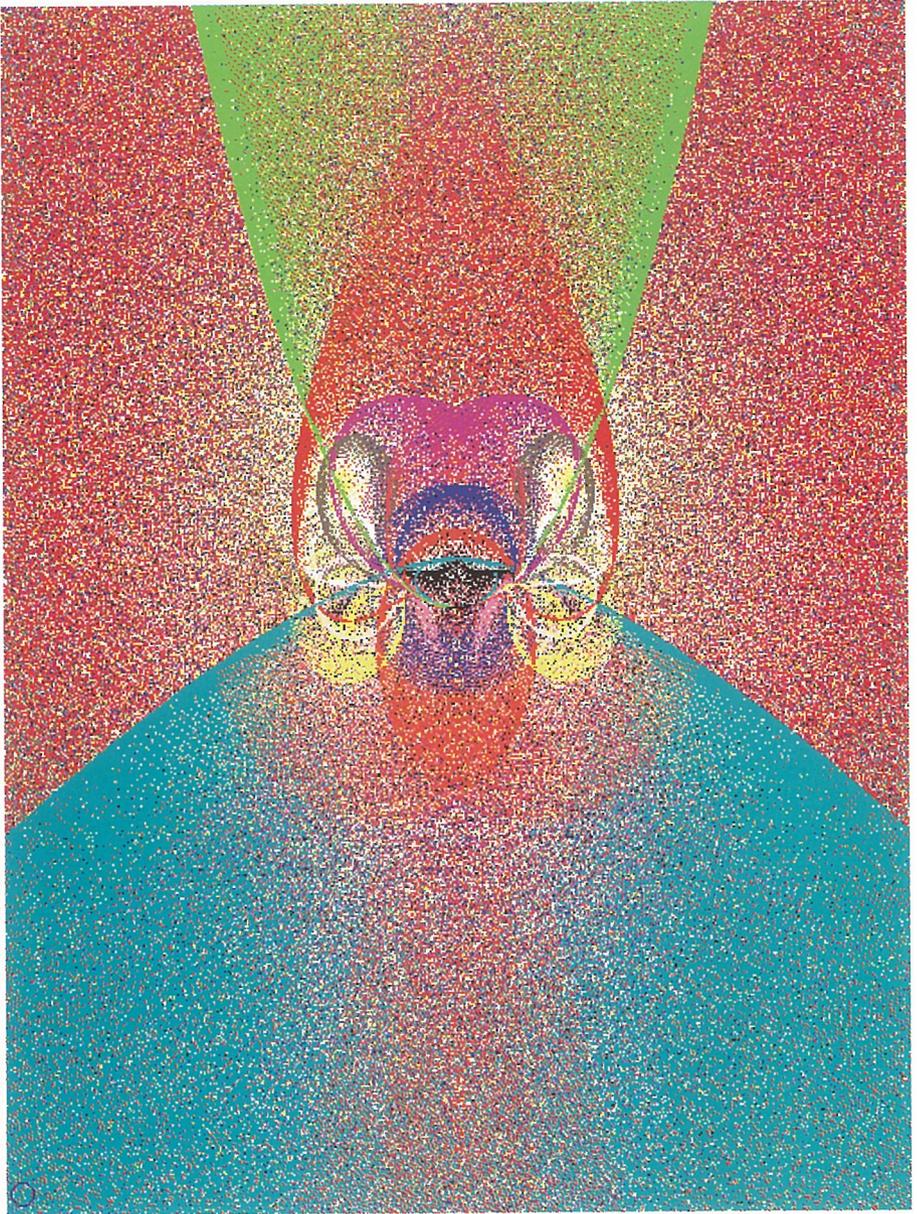


Fig. 2



Fig. 3



Fig. 4

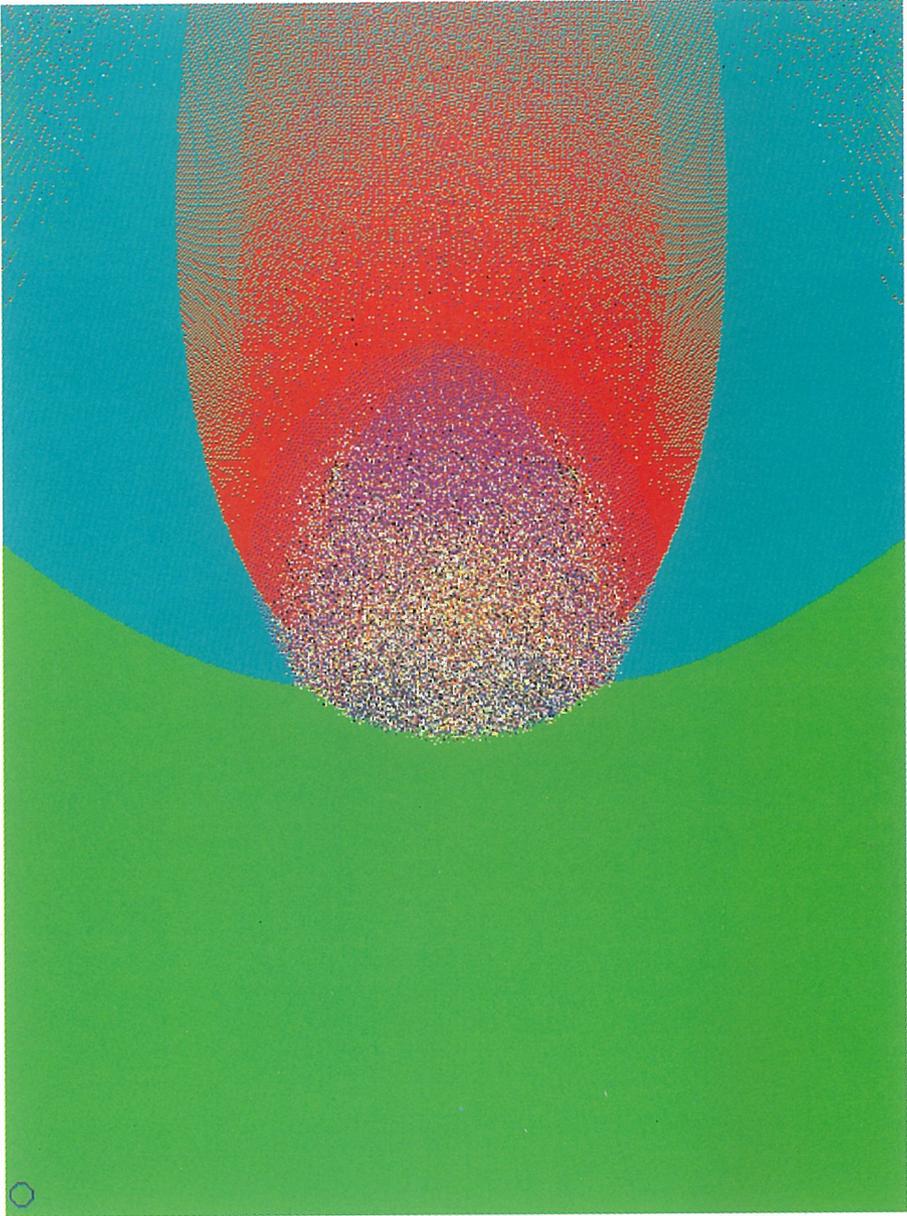


Fig.5



Fig. 6

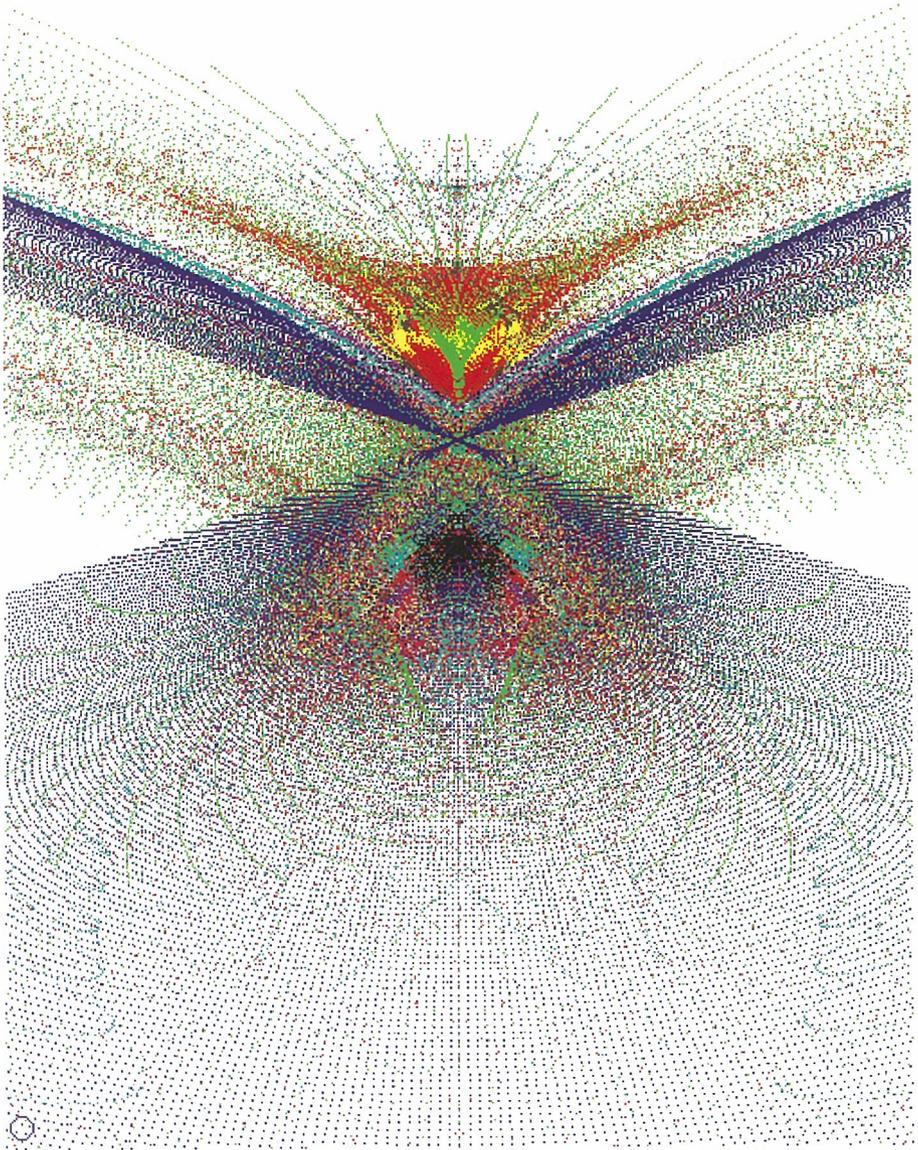


Fig. 7

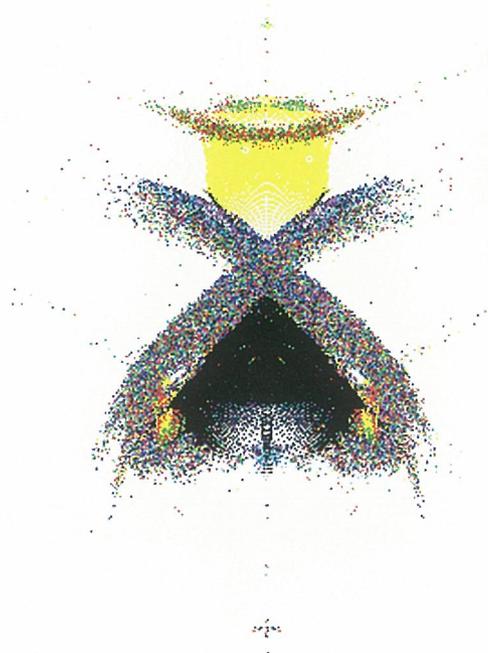


Fig. 8