複素力学系入門

京都大学大学院理学研究科 数学教室

京都大学オープンキャンパス 2006年8月11日

はじめに: 漸化式

高校の数学で, 漸化式というものを習ったと思います.

$$x_{n+1} = f(x_n) \qquad (n = 0, 1, 2, \ldots)$$

という形のものや, より一般に,

$$x_{n+1} = f(x_n, x_{n-1}, \ldots) \qquad (n = 0, 1, 2, \ldots)$$

といった形で数列を定義するものです. 例えば,

$$\begin{aligned} x_{n+1} &= x_n + c &\Rightarrow\quad & x_n = x_0 + nc, \\ x_{n+1} &= a x_n &\Rightarrow\quad & x_n = a^n x_0, \\ x_{n+1} &= a x_n + b x_{n-1} &\Rightarrow\quad & x_n = \alpha\lambda^n + \beta\mu^n \\ & & & (\lambda, \mu \text{ は } x^2 = ax + b \text{ の解.}) \end{aligned}$$

のように, 簡単な場合は一般項を求めることができます. では,

$$x_{n+1} = x_n^2 + c \tag{1}$$

の場合はどうでしょう? 非常に特殊な場合 ($c = 0, -2$) を除いては, n についての簡単な式では表せないことが知られています.

漸化式と力学系

漸化式 (1) で, $c \leq \frac{1}{4}$ のときは, 適当な d, e をとって $y_n = dx_n + e$ とおくと,

$$y_{n+1} = a y_n (1 - y_n) \tag{2}$$

という形になります. これはロジスティック写像と呼ばれるもので, ある容器の中に飼っている虫の数の増減のモデルとして導入されたものです. n番目の世代での虫の数 (最大生息可能数に対する比率) を y_n として, 時間発展を記述しているわけです. このような時間発展を記述する数学的モデルを力学系と呼びます. ロジスティック写像のように時間が整数 (離散時間) の場合は, ある写像 (または関数) f によって,

$$\begin{aligned} x_{n+1} &= f(x_n) & (n = 0, 1, 2, \ldots) \\ x_n &= f^n(x_0) & \text{ただし, } f^n = \underbrace{f \circ f \circ \cdots \circ f}_{n} \end{aligned}$$

のように記述されます. 各 x_n が時刻 n での状態を表す実数やベクトルなどです. また時間が実数 (連続時間) の場合は, $x = x(t)$ は時刻 t の関数として,

$$\frac{dx}{dt} = f(x)$$

という微分方程式で表されます. 時刻 0 のときに点 x_0 から始めて時間を進めていった時に通る点, つまり, 離散時間の場合の $x_0, x_1, x_2, \ldots$ という点列や, 連続時間の場合の $x(t)$ という関数のことを, (初期値 x_0 の) 軌道 と呼びます. 時間を逆に戻せる (負の時間についても解ける) 場合は, それも含めて軌道と呼ぶこともあります.

こういったモデルにおいて, いろいろな点 x_0 から始めて時刻$\to\infty$ を考えた時に何が起きるか? また $f(x)$ に含まれるパラメータを変化させた時に何が起きるか? ということを考えることが重要になります.

複素力学系

ロジスティック写像のような, 非常に簡単な力学系においても, 初期値やパラメータの小さな違いが大きな影響を及ぼすことがあり, 実際の挙動は非常に複雑 (カオス的挙動) になります. このような現象は, 複素力学系の世界においては非常に美しいフラクタル集合として捉えることができます. 複素力学系とは, 漸化式で与えられるような離散時間の力学系を, 複素数の範囲で考えたものです. ここでは漸化式として, (1) を考えます. つまり, c を複素数のパラメータとして,

$$p_c(z) = z^2 + c$$

を考え, 各複素数 z に対して, その軌道 $\{p_c^n(z)\}_{n=0}^{\infty}$ のふるまいを調べます. 実は軌道は有界にとどまる (ある $R > 0$ があって, すべての n について $|p_c^n(z)| < R$ となる) か, 無限大に発散するかのどちらかになることが知られています. そのため以下のような p_c の(充填)ジュリア集合と呼ばれる, 複素平面 $\mathbb{C}$ 内の集合を考えます.

$$K_c = \{z \in \mathbb{C} | \ \{p_c^n(z)\} \text{ は有界}\}\text{: 充填ジュリア集合}$$

ジュリア集合はこの境界です.

実際に計算機で絵を書く時には, 順番に $p_c^n(z)$ を計算していって, 絶対値がある程度 (例えば 10) より初めて大きくなった n によって点 z を色分けしていきます. n が一定の値 ($n = 100$ など) になるまで調べても上の条件を満たさない場合は, 点 z は充填ジュリア集合の点である (だろう) と判断します.

ジュリア集合の例

ここで説明した方法で, いろいろな p_c に対応したジュリア集合を描いたものは, もう1枚のポスターで見ることができます. (同じ例がコンピュータでも見られるようになっています.) ここでは最も簡単な場合のジュリア集合について説明したいと思います.

$c = 0$, つまり $p_0(z) = z^2$ とします. この時は $p_0^n(z) = z^{2^n}$ となるので,

$$\lim_{n=0} |p_0^n(z)| = \begin{cases} 0 & |z| < 1 \text{ のとき,} \\ 1 & |z| = 1 \text{ のとき,} \\ \infty & |z| > 1 \text{ のとき,} \end{cases}$$

となります. つまり, 充填ジュリア集合は単位円板 $\{z; \ |z| \leq 1\}$ で, ジュリア集合は単位円 $\{z; |z| = 1\}$ です.

次に $c = -2$, $p_c(z) = z^2 - 2$ を考えます. この場合, 区間 (線分) $[-2, 2] = \{z; \ z \text{は実数で}, -2 \leq z \leq 2\}$ は充填ジュリア集合に含まれることがわかります. グラフを描いてもわかりますが, 少し違う方法で説明します. $z \in [-2, 2]$ は $z = 2\cos\theta$ と表わせます. すると三角関数の倍角の公式 $\cos(2\theta) = 2\cos^2\theta - 1$ より,

$$p_{-2}(z) = (2\cos\theta)^2 - 2 = 2\cos(2\theta),$$

つまり p_{-2} は角度 θ を2倍するという作用に対応しています. 従って $p_{-2}^n(z) = 2\cos(2^n\theta)$ となり, このことから $p_{-2}^n(z) \in [-2, 2]$ がわかります. 実は, p_{-2} の (充填) ジュリア集合は $[-2, 2]$ に一致することが知られています.

ジュリア集合の連結性とマンデルブロート集合

上の例で見られるように, パラメータ c の値によって, ジュリア集合は連結な場合と連結でない場合があります. (「連結」という言葉の数学的に正確な定義は省きますが, おおざっぱに言えば1つにつながっている, ということです. ちなみにこの場合は, 連結でなければカントール集合と呼ばれる (非可算) 無限個ある全ての点がばらばらにわかれたような集合になることが知られています. 従って, 例えばちょうど2つにわかれるような場合などは存在しません.)

実はこれは, $z = 0$ の軌道だけで完全に決まります:

$$K_c \text{ が連結} \Leftrightarrow z = 0 \text{ の軌道が有界.}$$

(なぜ $z = 0$ が重要なのかについては詳しくは述べませんが, $p_c(z)$ の微分が 0 になる唯一の点である, ということが関連しています.) そこで, マンデルブロート集合 M を以下で定義します.

$$\begin{aligned} M &= \{c \in \mathbb{C} | \ K_c \text{ が連結}\} \\ &= \{c \in \mathbb{C} | \ \{p_c^n(0)\}_{n=0}^{\infty} \text{ が有界}\}. \end{aligned}$$

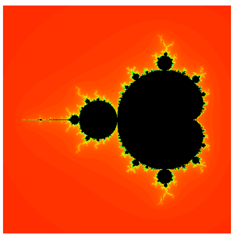

Figure 1: マンデルブロート集合

マンデルブロート集合は連結であることが知られています.

ジュリア集合やマンデルブロート集合 (の境界) は, フラクタル集合と呼ばれる, どこを拡大しても, もとの集合と同じ複雑さを持っている奇妙な集合になっています. 例えばマンデルブロート集合の境界の至る所に(稠密に), マンデルブロート集合の小さいコピーが埋めこまれていることが知られています. このような性質は自己相似性と呼ばれ, フラクタル集合を特徴づける重要な性質の1つです.

ソフトウェアの使い方

皆さんにコンピュータを使っていろいろな絵を実際に見てもらうために, **iDynamics** というソフトウェアがインストールされています. iDynamics は, 京都大学大学院理学研究科の宍倉光広教授が作ったソフトウェアで, ここで紹介したマンデルブロート集合やジュリア集合の絵を, 領域やパラメータを変えて表示することができます.

印刷もできるようにしてありますので, ぜひ自分の気にいった一枚を印刷して記念に持って帰ってください.

以下に簡単な使い方を説明します. (iDynamics のアプリケーションと一緒に Read Me が置いてあります. そちらにより詳しい説明が載っていますのでそちらも参照してください.)

起動と終了 iDynamics のアイコンをダブルクリックして起動します. 起動時や「File」メニューの「New」を選んだときには, 「Parameter for new picture」というダイアログが表示されます. ここでウィンドウの名前, マンデルブロート集合 (Parameter space) とジュリア集合 (Julia set) のどちらを描くか, ウィンドウのサイズ (pixels), 描く領域の中心 (center) の座標と一辺の長さの半分 (size), 最大計算回数 (iteration) の値などを入力することができます. 「OK」ボタンを押すと新しくウィンドウが開いて図が描かれます.

終了するときは, 「File」メニューの「Quit」を選びます.

ズーム 図の描かれたウィンドウの中で, マウスをドラッグすると点線で長方形が描かれます. この内部をクリックすると, 囲まれた領域を新しいウィンドウに拡大して描画します. Shift キーを押しながらドラッグすると正方形の領域が選択できます.

選択範囲の外側をクリックすると選択が解除されます.

領域の移動 Option キーを押しながらドラッグすることで, 領域を移動することができます. キャンセルしたい場合は画面上のメニューバーまでドラッグして離してください.

最大計算回数を変更する 拡大しすぎた場合など, 最大計算回数が足りなくなった場合は「Special」メニューの「More iteration」を選ぶと計算回数を変更できます. 最大計算回数を減らすと描画が速くなりますし, 増やすとより正確な図が描けます.

ジュリア集合 マンデルブロート集合のウィンドウでダブルクリックするとその点のパラメータに対応するジュリア集合を新しいウィンドウに描画します.

リアルタイムにジュリア集合を表示する 「Special」メニューの「Real time Julia set」を選ぶと, 「[Julia set]」というウィンドウが表示されます. このとき, マンデルブロート集合が描かれたウィンドウでクリックやドラッグすると, 対応するジュリア集合をリアルタイムに描画します.

軌道の表示 ジュリア集合のウィンドウで, コマンドキー (⌘) を押しながらクリックするとその点の軌道を表示します.

情報の表示 「Window」メニューの「Show Info」を選ぶと, 現在前面にあるウィンドウに表示されている図に関する, パラメータや領域などの情報が表示されます. また, 同じく「Window」メニューの「Show Point」を選ぶと, 前面にあるウィンドウについて, 矢印の指し示す点の値が表示されます.

表示色の変更 「Color」メニューには, 表示色の変更に関する機能がまとめられています. アニメーションしたり, 別のパレットに変更したりできます. いろいろ試してみてください.

印刷 気に入った図が描けたら, ぜひ印刷して持って帰ってください. 「File」メニューの「Print」を選んで, 表示されたダイアログで「プリント」ボタンを押すと印刷されます. パラメータや領域の情報も一緒に印刷されます. 残念ながら紙のサイズにあわせて自動的に拡大／縮小はしないので, 図のサイズ (Pixels) は 400x400 か, 500x500 にすることをおすすめします.