

精度保証付き数値計算の力学系への応用について

On Applications of Rigorous Computing to Dynamical Systems

京都大学大学院理学研究科 荒井迅 (Zin ARAI)

Department of Mathematics, Kyoto University

email: arai@math.kyoto-u.ac.jp

1 はじめに

本稿では、近年盛んになって来た精度保証付き数値計算の力学系の研究への応用について解説する。応用が目的であるので精度保証付き数値計算そのものについてはここでは立ち入らないが、[12, 13]などを参照されたい。

一口に応用といってもその手法や対象は多岐にわたり、とてもここで全てを解説することはできない。本稿ではそれらの結果を列挙するのではなく、典型的な結果の例を Lorenz 方程式を例にとりていくつか紹介し、それらに共通する考え方を明らかにしたい。

以下ではまず § 2 で Lorenz 方程式の導入といくつかの結果を紹介を行ない、数値積分に関する話題に触れる。議論は最終的にポアンカレ写像の解析になるため、§ 3 では対象を常微分方程式から一般の写像の与える離散力学系に移してコンレイ指数を用いた解析について解説を行なう。最後に § 4 において、これらの研究のために開発されたいくつかのソフトウェアパッケージを紹介する。

2 Lorenz 方程式

2.1 Lorenz 方程式とは

気象学者の E. Lorenz は、熱対流現象を記述するある偏微分方程式を大胆に単純化して次のような \mathbb{R}^3 の常微分方程式を導いた。

$$\begin{aligned}\dot{x} &= -\sigma x + \sigma y \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

これが Lorenz 方程式と呼ばれる方程式である。ここで σ, ρ, β は実数値を取るパラメータであり、Lorenz は $(\sigma, \rho, \beta) = (10, 28, 8/3)$ とおいた数値実験を 1963 年に行ない、いわゆるストレンジアトラクターを発見した(図 1)。

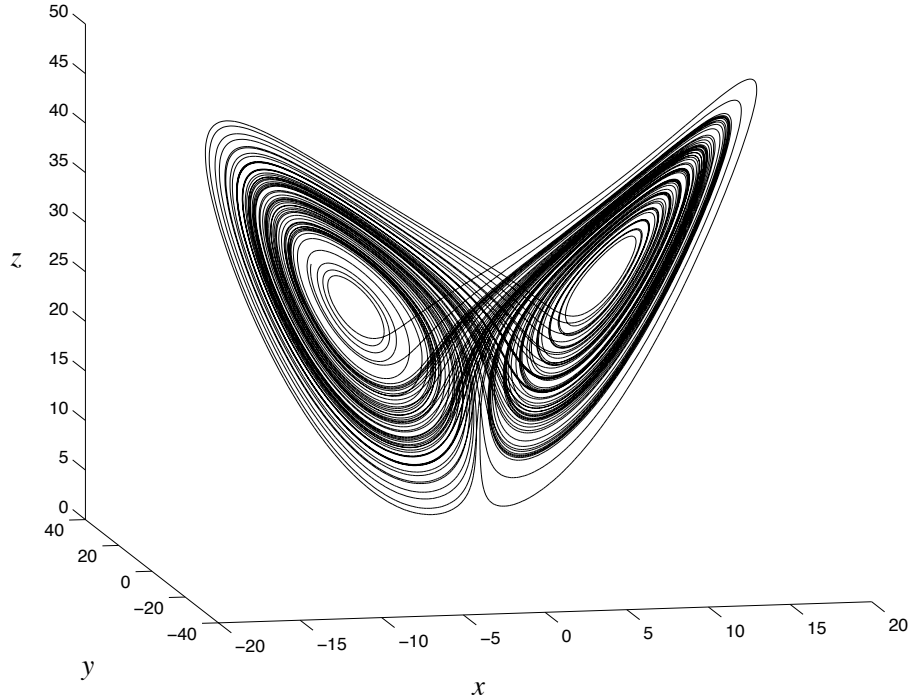


図1 Lorenz 方程式のアトラクター

気象学者である彼はこの結果から天気の長期予測が原理的に不可能であると結論したが、彼の議論は数値的に描かれた軌道に基く観察であり、数学的には Lorenz 方程式が与える力学系は本当にカオス的なのであろうかという疑問が生じる。この問題の重要性は Hilbert の 23 問題に習い S. Smale が 21 世紀の数学者のために作成した 18 の未解決問題集の第 14 番目が「Lorenz 方程式は本当にストレンジアトラクターを持つのだろうか」という問いであったことにも示されている。

多くの数学者がこの疑問に答えようと努力を続けて来たが、1990 年代に入りいくつかの部分的な結果が出はじめ、やがてほぼ完全な解決が W. Tucker によって得られた。

それらの結果を述べるために、まず記号力学系の定義を思い出そう。自然数 k に対し $\Sigma_k := \prod_{i=0}^{\infty} \{1, 2, \dots, k\}$ を k 個の記号列からなる full-shift の空間とし、その上のシフト写像 $s : \Sigma_k \rightarrow \Sigma_k$ を $s(x_0, x_1, \dots) = (x_1, x_2, \dots)$ で定義する。また $k \times k$ 整数行列 $A = (a_{ij})$ に対し、 $\Sigma_A := \{(s_n) \in \Sigma_k \mid a_{s_n s_{n+1}} \neq 0\}$ とおく。 $s(\Sigma_A) = \Sigma_A$ に注意。

定理 1 (Mishaikow-Mrozek[8, 9, 10]). Lorenz 方程式のパラメータ (σ, ρ, β) が十分 $(10, 28, 8/3)$ に近ければポアンカレ切断 $I \subset \{z = 27\}$ とその上のポアンカレ写像 P が well-defined となる。さらに $\pi : \text{Inv}(I, P) \rightarrow \Sigma_6$ が定義され連続となり、 $\pi \circ P = s \circ \pi$

および $\Sigma_A \subset \pi(\text{Inv}(I, P))$ が成り立つ．ただし A は

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

で与えられる行列である．

定理 2 (Galias-Zgliczyński[6]). *Lorenz* 方程式のパラメータ (σ, ρ, β) が十分 $(10, 28, 8/3)$ に近ければポアンカレ切断 $I \subset \{z = 27\}$ とその上のポアンカレ写像 P が *well-defined* となり，さらに連続写像 $\pi : \text{Inv}(I, P^2) \rightarrow \Sigma_2$ が定義され $\pi \circ P^2 = s \circ \pi$ が成立し， π は全射となる．

おおまかに言うと，上の二つの定理は *Lorenz* 方程式のアトラクターの中に記号力学系で力学系が記述できる不変集合が取れるということを主張している．

定理 3 (Tucker[14]). 古典的パラメータ $(\sigma, \rho, \beta) = (10, 28, 8/3)$ において *Lorenz* 方程式は “*robust strange attractor*” を持つ．

ここでは *robust strange attractor* の意味は解説しないが ([4] など参照のこと)，Tucker の結果は Mischaikow-Mrozek や Galias-Zgliczyński の結果よりも大域的であり，アトラクター全体の構造についてのより深い結果であることだけ注意しておく．この違いはどこから来るかと言うと，特異点の取り扱いである．ベクトル場の特異点の近くを通る軌道を精度保証付きで数値積分すると，速度が 0 に近づくために特異点の近傍を抜けるまでに必要な数値積分のステップ数が莫大になり，よい精度が得られない．そこで Mischaikow-Mrozek や Galias-Zgliczyński は原点にある特異点の周りにはあきらめて，軌道が原点の近くを通らないような部分集合にのみに注目した．一方で Tucker は原点の近傍の外ではやはり精度保証付きの数値計算をするのだが，原点の近傍では Normal Form を用いて解析的に取り扱い，それらを上手く組み合わせることでアトラクター全体の解析に成功した．

本稿ではそれぞれの証明の具体的なところには触れられないが，非常におおまかに言うと，どの結果においても証明は

1. 軌道を厳密に数値積分する (Tucker の場合は Normal Form も用いる) ことによつて Poincaré 写像を求める
2. 得られた Poincaré 写像を解析することでカオスの存在を示す

という 2 つのステップに分かれる．すなわち，常微分方程式の研究を数値積分などにより写像の研究に帰着させるという方針である．ポアンカレ写像の解析に関してはより一般的な状況で § 3 で扱うことにし，この節の残りでは数値積分に関わることについて触れよう．

2.2 数値積分の誤差をどう抑えるか

常微分方程式を数値的に積分してポアンカレ写像を求める時に生じると考えられる問題を大きくまとめると次のようになる．

- ベクトル場を積分する際に用いる近似法が含む誤差
- ポアンカレ断面との交わりを求めるときの誤差
- 数値計算によって生じる丸め誤差

それぞれについて簡単に触れてみよう．

まず数値積分に用いる近似法であるが，Tucker は Euler 法，Galias-Zgliczyński は 4 階の Taylor 展開，Mischaikow-Mrozek は 4 階の Runge-Kutta 法を用いている．普通の数値積分ならば 4 階の Runge-Kutta 法が最適な解に思われるが，精度保証をする場合には一概にそうとは言えない．通常の数値積分と異なり，精度保証をする場合には誤差項の厳密な評価も必要になるからである．実際 Galias-Zgliczyński は研究の初期段階において 4 階の Runge-Kutta を用いて積分を行っていたが，誤差項の項数が増えすぎて計算時間が増大してしまったために，誤差項の評価が簡単な 4 階の Taylor 展開に乗り替えている．

今求めたいのはポアンカレ断面からそれ自身への写像であるが，数値積分法による離散近似を続けていくと，正確にポアンカレ断面に戻って来ず，一般には飛び越えてしまう．よってベクトル場の Taylor 展開などを用いて軌道とポアンカレ断面との交わりを求めることになるが，やはりここでも誤差を正確に評価しなくてはならない．

最後の丸め誤差については，精度保証付きの厳密な区間演算を用いることで評価できる．ただしここで区間演算によって別の問題が生じる．それが “wrapping effect” である．

2.3 Wrapping Effect

区間演算は基本的に一次元の区間に対する演算なので，高次元の常微分方程式を積分をする場合には各座標に対して区間演算を用いることになる．従って一点もしくは区間の直積を積分した像として得られる集合は，やはり区間の直積として表わされることになる．ここで図 2 における左側の正方形 X を考え，常微分方程式によってこの正方形を積分した像が右の X' のように細長く斜めの図形になったとする．すると実際には面積が

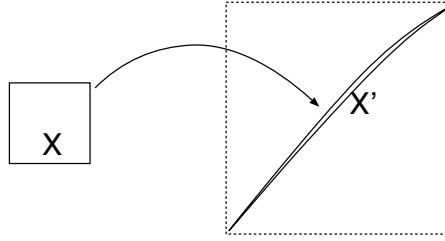


図2 wrapping effect

強く縮小しているにもかかわらず X' を覆う区間の直積は図の点線で描かれた正方形のように大きなものになってしまい、計算結果は精度の悪いものになってしまう。これが wrapping effect と呼ばれる現象であり、とくに Lorenz 方程式のように強い拡大方向と縮小方向を持っている場合には深刻な問題となる。

この wrapping effect を抑えるために Tucker が採った方法は、定義域となる領域 X を小さく分割し直すことで歪みを抑えることである。分割の要素となる各領域はやはり wrapping effect の影響を受けるが、それぞれの像の和集合を取ると全体としては像の拡大は抑えられる。

また Galias-Zgliczyński は積分を区間の直積に対してではなく一点に対して行なうことで wrapping effect を抑えている。すなわち点 x を中心とする半径 ϵ の球体 $B(x, \epsilon)$ を時間 h だけ積分した像を求めるために、まず x の軌道を時間 h だけ積分してその像 P を求める。ここで P は区間の直積となる。その後、logarithmic norm を用いた微分の評価を用いて領域 P を少し膨らませた領域を P' とすると、求めたい $B(x, \epsilon)$ の像は P' に含まれているということが示される。

Mischaikow-Mrozek が wrapping effect を抑えるために用いたのはポアンカレ断面を一つではなく沢山取るという方法である。すなわち、 $\{z = 27\}$ という断面を出た軌道がアトラクターの中を一周して帰ってくる、その道筋に数多くの断面を取って（パラメータにもよるが、何十個も取っている）おき、それぞれの断面の間のポアンカレ写像を求める。最終的に欲しい $\{z = 27\}$ の上のポアンカレ写像は、そうして求めたポアンカレ写像たちの合成となる。こうすると各断面の間の数値積分は短い時間に抑えられ、各断面を上手く取れば wrapping effect の効果を抑えられる。

各ステップで座標を取り直すことで wrapping effect を抑えるという方法も考えられる。例えば図2の場合であれば、 (x, y) という座標から $(x + y, x - y)$ という座標に変換すれば X' を小さな面積で包み込めることがわかる。このような座標の取り直しを効果的に行なう Lohner アルゴリズム ([16] など参照) と呼ばれるアルゴリズムが知られている。

3 Results for Discrete Dynamical Systems

この節では精度保証付き数値計算を用いて一般の離散力学系を研究する手法について解説する．常微分方程式から導かれたポアンカレ写像の場合でも考え方は同じである．

たとえ精度保証付きであっても，数値計算によって得られた情報は有限時間の軌道の振舞いについての情報しか含まない．そこから軌道の漸近的な挙動を知るためには不動点定理のような何らかの数学的な道具が必要となる．本節ではそのような道具のなかでも特に広い適用範囲を持つと思われるコンレイ指数 [11] を用いる．

3.1 離散力学系のコンレイ指数

離散力学系とはここでは局所コンパクト距離空間 X 上の連続写像 $f : X \rightarrow X$ のこととする．力学系の解析では X 全体における f の挙動を一度に考えるのではなく，不変集合（すなわち $f(S) = S$ となる $S \subset X$ ）上での f のふるまいと，不変集合たちの間の軌道の繋がり方に問題を分割するのが常套手段である．ところが不変集合そのものを対象として解析を行なうのは一般に難しい．不変集合は無限に微細な構造を持っていたり，摂動によりその構造が変化したり消えてしまったりする．計算誤差により不変集合の構造が変化してしまう可能性を考えると，数学的に厳密な結果を数値計算から得るのは難しい．

そこで我々は次のような「よい」不変集合のクラスを考える．以下 $\text{int } N$ で N の内点集合を表すことにする．

定義 4. 不変集合 S はあるコンパクト近傍 N が存在して S が N の最大不変集合となるとき，孤立不変集合であるという．すなわち，

$$S = \text{Inv}(N, f) := \{x \in N \mid \exists \{x_i\}_{i \in \mathbb{Z}} \subset N \text{ s.t. } x_0 = x, f(x_i) = x_{i+1} \text{ for all } i \in \mathbb{Z}\}$$

かつ $S = \text{Inv}(N, f) \subset \text{int } N$ となることである．このとき N を S の孤立化近傍という．

空集合も孤立不変集合の定義を満たすことに注意する．ここで重要なのは孤立化近傍であるという性質が微細な摂動に対し安定である，つまり N がある f に対し孤立化近傍であれば， f と十分 (C^0 位相で) 近い g に対しても N は孤立化近傍であり続けるということである． $\text{Inv}(N, f)$ と $\text{Inv}(N, g)$ の構造は一般に異なるが， N が孤立化近傍であるという性質は安定であり，この事実がコンレイ指数が計算機で求められる根拠になっている．

よって以下では主に孤立化近傍に注目する．知りたいのはあくまで不変集合についての情報であるが，それは扱いが難しいのでまず大雑把にその近傍を観察し，そこで得られた

情報から不変集合について何らかの結論を得ようという方針である．

では具体的な力学系が与えられたときに孤立化近傍をどのようにして構成すればよいのか，また孤立化近傍からどのような情報を得ればそこに含まれる孤立不変集合上の力学系について理解できるのか．以下これらについて解説する．

定義 5. 孤立不変集合 S の index pair とは，コンパクト集合対 $P = (P_1, P_0)$ であって，
 (イ) $P_1 \setminus P_0$ の閉包が S の孤立化近傍であり (ロ) $f(P_0) \cap P_1 \subset P_0$ と (ハ) $f(P_1 \setminus P_0) \subset P_1$
 の 3 つの条件が成立するもののことである．このとき P_1/P_0 を P_1 の中で P_0 を一点に潰した空間とし (P_0 を潰して得られた点を $[P_0]$ と書く)， $f_P : P_1/P_0 \rightarrow P_1/P_0$ を

$$f_P([x]) := \begin{cases} [f(x)] & f(x) \in P_1 \text{ のとき} \\ [P_0] & \text{その他} \end{cases}$$

と定義すると f_P は連続写像となるが，これを index map と言う．

この index map は S の近傍での f のふるまいを記述していると考えられ，ここから何らかの情報を引き出したい．位相幾何学的に言えば適当な関手により不変量を取り出そうという事であるが，具体例に応用するためには実際に計算できる関手でないといけない．計算ホモロジー理論 [7] の発展により，ホモロジー群ならば計算機で求めることができるのでこれを適用することにしよう．すると空間 P_1/P_0 から加群 $H_*(P_1/P_0, [P_0])$ が，写像 f_P から自己準同形 $f_{P*} : H_*(P_1/P_0, [P_0]) \rightarrow H_*(P_1/P_0, [P_0])$ が得られる．ここで $H_*(P_1/P_0, [P_0])$ は位相空間対 $(P_1/P_0, [P_0])$ の双対ホモロジーを表わす．また $H_*(P_1/P_0, [P_0])$ は $H_k(P_1/P_0, [P_0])$ を直和した次数付き加群であり f_{P*} はその上の次数 0 の準同形である．とくにある次数 k を指定して表示したい場合は $f_{P*k} : H_k(P_1/P_0, [P_0]) \rightarrow H_k(P_1/P_0, [P_0])$ などと書くことにする．これらは S の近傍での力学系についての何らかの情報を持っているが，index pair の選び方は一般に無数にあり $H_*(P_1/P_0, [P_0])$ も f_{P*} もその選び方に依存してしまう．そこで次のような同値関係を考える．

定義 6. (群の) 準同形 $f : X \rightarrow X$ と $g : Y \rightarrow Y$ は，ある自然数 m と準同形 $r : X \rightarrow Y$ ， $s : Y \rightarrow X$ が存在して $r \circ f = g \circ r$ ， $s \circ g = f \circ s$ ， $r \circ s = g^m$ ， $s \circ r = f^m$ となるときシフト同値であると言う．

任意の孤立不変集合 S に対してその index pair は必ず存在し， $P = (P_1, P_0)$ と $Q = (Q_1, Q_0)$ を S の index pair とすると f_{P*} と f_{Q*} はシフト同値であることが証明される．よって次のようにコンレイ指数を定義することができる．

定義 7. 孤立不変集合 S のホモロジーコンレイ指数とは, $P = (P_1, P_0)$ を S の index pair としたときの f_{P^*} のシフト同値類のことである.

コンレイ指数から何がわかるのか. 最も単純な結果は次のようなものである.

定理 8 (Ważewski principle [7, 11]). $P = (P_1, P_0)$ を S の index pair とする. このとき f_{P^*} が $0 : \{0\} \rightarrow \{0\}$ とシフト同値でないならば, S は空集合ではない.

定理 9 (Index pair に対する Lefschetz 不動点定理 [7]). $P = (P_1, P_0)$ を S の index pair とする. $L(f_{P^*}) := \sum_k (-1)^k \text{tr } f_{P^*k} \neq 0$ ならば S は不動点を含む. また $\sum_k (-1)^k \text{tr } f_{P^*k}^n \neq 0$ ならば S 内に f^n の不動点が存在する.

他にもら記号力学系との対応 [7, 15] や connecting orbit の存在 [3] などにもコンレイ指数は使えるが, これらの定理を具体例に適用するためには index pair と index map を構成し, そのホモロジーを計算せねばならない. 次に計算機でこれらを自動的に実行する方法について考える.

3.2 計算機によるコンレイ指数の計算

$X = \mathbb{R}^n$ とする. コンレイ指数の計算を計算機に実行させるためには, \mathbb{R}^n の部分集合を計算機で扱える形で表現しなければならない. 様々な方法が考えられるが, 最も単純に \mathbb{R}^n を n 次元直方体により分割し, それらの有限和として書ける部分集合のみ扱うという方法を採用する. 分割の要素となる n 次元直方体の各辺の長さを d_i ($i = 1 \dots n$) とし,

$$\Omega := \left\{ \prod_{i=1}^n [k_i d_i, (k_i + 1) d_i] : k_i \in \mathbb{Z} \right\}$$

とおくと, \mathbb{R}^n は Ω の要素により被覆される. 直方体の集合 $B \subset \Omega$ に対し, $|B|$ を B に含まれる直方体の和集合として表される \mathbb{R}^n の部分集合とする.

次に f を計算機で扱える形に表現しよう. ここで写像 f に精度保証付き区間演算が適用でき, 各 $\omega \in \Omega$ に対して $f(|\omega|)$ を内点に含む直方体を計算機で求められると仮定する. この直方体を $\tilde{f}(\omega)$ と書く. $\tilde{f}(|\omega|)$ は直方体ではあるが Ω の要素の和ではないので, $\tilde{f}(|\omega|)$ と交わる Ω の要素を全て集めこれを $\mathcal{F}(\omega)$ とおく. すなわち

$$\mathcal{F} : \Omega \rightarrow \{ \Omega \text{ の部分集合} \} : \omega \mapsto \{ \omega' \in \Omega : \tilde{f}(|\omega|) \cap |\omega'| \neq \emptyset \}$$

である. $f(|\omega|) \subset \text{int } \mathcal{F}(\omega)$ が成立することに注意する. $f(|\omega|)$ は正確に求めることができないので, それを外側から近似する $\mathcal{F}(\omega)$ を求めていることになる.

コンレイ指数を計算機で求めるためのステップをまとめると次のようになる。

- ステップ 1. 孤立化近傍の候補となる集合 $\mathcal{I} \subset \Omega$ をつくる
- ステップ 2. 孤立化近傍の条件を満たすように \mathcal{I} を修正する
- ステップ 3. \mathcal{I} から index pair を構成する
- ステップ 4. ホモロジーを計算する

以下これらのステップを順に見ていこう。

ステップ 1. コンレイ指数を計算したい不変集合が存在すると予想される領域を被覆する直方体の集合 \mathcal{I} を構成する。これが孤立化近傍の第一近似となる。各直方体が十分小さくないと以後のステップで計算が破綻する、もしくは自明な結論しか導き出せなくなってしまう。そこでまず各直方体を 2 等分し、次に注目する不変集合と関係ない直方体を取り除くという作業を各直方体が十分小さくなるまで繰り返す。

ステップ 2. $|\mathcal{I}|$ が孤立化近傍となる、すなわち $\text{Inv}(|\mathcal{I}|, f) \subset \text{int } |\mathcal{I}|$ が満たされるように \mathcal{I} を再構成する。 $B \subset \Omega$ に対し $o(B) := \{\omega \in \Omega : |\omega| \cap |B| \neq \emptyset\}$, $d(B) := o(B) \setminus B$ とおこう。 $|o(B)|$ は Ω の部分集合により表すことができる $|B|$ の近傍で最小のものである。また $\text{Inv}(B, \mathcal{F})$ を

$$\{\omega \in B \mid \text{ある } \gamma : \mathbb{Z} \rightarrow B \text{ が存在して } \gamma(0) = \omega \text{ かつ } \gamma(k+1) \in \mathcal{F}(\gamma(k)) \text{ を満たす}\}$$

と定義する。 $f(|\omega|) \subset \text{int } \mathcal{F}(\omega)$ が保証されていることから $\text{Inv}(|\mathcal{I}|, f) \subset |\text{Inv}(\mathcal{I}, \mathcal{F})|$ が成り立つので、 $o(\text{Inv}(\mathcal{I}, \mathcal{F})) \subset \mathcal{I}$ が言えれば

$$\text{Inv}(|\mathcal{I}|, f) \subset |\text{Inv}(\mathcal{I}, \mathcal{F})| \subset \text{int } |o(\text{Inv}(\mathcal{I}, \mathcal{F}))| \subset \text{int } |\mathcal{I}|$$

となり目標が達成される。勝手に構成した \mathcal{I} は $o(\text{Inv}(\mathcal{I}, \mathcal{F})) \subset \mathcal{I}$ を満たさないの、この包含関係が成立するまで \mathcal{I} に直方体を付け加えてゆくアルゴリズム [7] を用いて \mathcal{I} を再構成する (どんどん \mathcal{I} を大きくしてもいつまでも成立しない場合もある)。

ステップ 3. $|\mathcal{I}|$ が f の孤立化近傍のとき、 $B = \text{Inv}(\mathcal{I}, \mathcal{F})$,

$$(\mathcal{P}_1, \mathcal{P}_0) = ((d(B) \cap \mathcal{F}(B)) \cup B, d(B) \cap \mathcal{F}(B))$$

とおくと $P = (|\mathcal{P}_1|, |\mathcal{P}_0|)$ が $\text{Inv}(|\mathcal{I}|, f)$ の index pair となる [7] 。

ステップ 4. 計算ホモロジー理論 [7] を用いて $H_*(|\mathcal{P}_1|/|\mathcal{P}_0|, [|\mathcal{P}_0|])$ と f_{P^*} を計算機上で f の近似表現 \mathcal{F} から計算する。 § 4 で紹介する CHomP を利用するとこれらがほぼ自動的に実行できる。単体近似定理の場合と同様に、各直方体が十分小さく、 \mathcal{F} による近似が

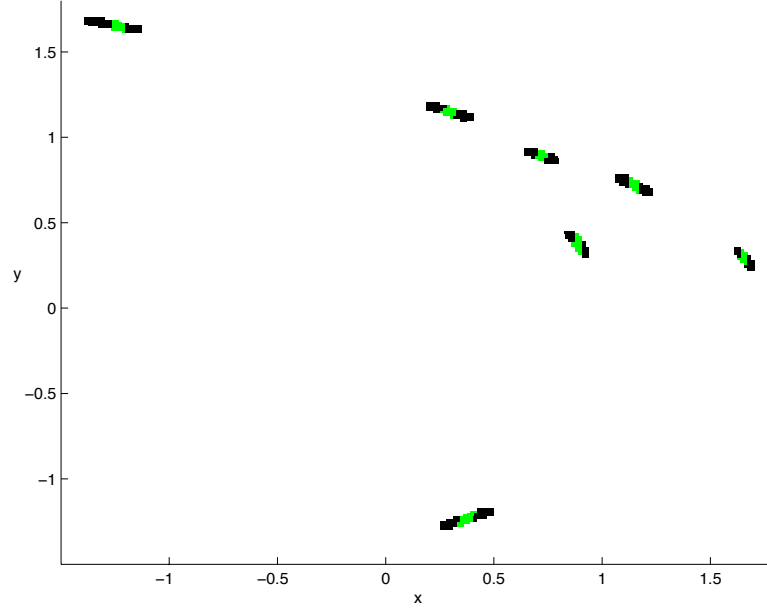


図3 エノン写像のある7周期点に対する index pair

十分よいことが計算可能性の必要条件となるが，一方で直方体の数が多くなると計算に必要なメモリの量及び実行時間が大きな障害となる．

例としてエノン写像 $H_{a,b} : \mathbb{R}^2 \rightarrow \mathbb{R}^2 : (x, y) \mapsto (a - x^2 + by, x)$ に対して定理9を適用して周期点の存在を証明しよう．Hénon 写像は天文学者の Hénon が Lorenz 方程式のポアンカレ写像を解析するためにより簡単なモデルとして提案した写像である．パラメータは $a = 1.4, b = 0.3$ とする．図3は上記のステップにより構成した index pair であり，灰色の領域が $P_1 \setminus P_0$ ，黒の領域が P_0 を表わす．CHomP による計算を実行すると

$$f_{P_*1} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} : \mathbb{Z}^7 \rightarrow \mathbb{Z}^7$$

となり，1以外の次数では f_{P_*} は0であることがわかる． $\text{tr}((f_{P_*1})^7) = 7$ であるから定理9より $\text{Inv}(P_1 \setminus P_0)$ は f^7 の不動点を含む． P_1 が f の不動点を含まないことは簡単に示せるので，結局 $\text{Inv}(P_1 \setminus P_0)$ は7周期点を含むことが証明された．

残念ながらここでは最も単純な場合である周期点の存在しか例として扱えなかったが，Hénon 写像の解析についてはホモクリニック接触の発生を証明した [3]，一様双曲性なパラメータ領域の存在を証明した [2] などの結果がある．また本節の内容については [1] により詳しい解説がある．

4 Software Packages

本節では本稿の内容に関連するソフトウェアについて解説する．残念ながら現状ではどれもドキュメントの整備が進んでいるとは言い難い状況であり，各自の目的に合わせて利用するためにはソースコードを読む必要が生じる場合がある．ソースを読むのは面倒なことではあるが，ソースコードを読んだことがないプログラムを証明に用いるのは証明を見たことがない定理を自分の結果に使うのと同じことなので，計算機援用証明をする目的ならば重要な部分だけでもソースコードに目を通したほうが良いと思われる．

4.1 GAIO (Global Analysis of Invariant Objects)

<http://math-www.uni-paderborn.de/~agdellnitz/gaio/>

M. Dellnitz and O. Junge らによって開発された力学系の研究のための汎用パッケージである．Python 版と MATLAB 版の二種類のインターフェースがあるが，現在メンテナンスされているのは MATLAB 版のみのものである．もちろん無料で利用できるが利用するためには作者らにメールで連絡を取る必要がある．

GAIO には § 3 で述べたような空間の分割を扱うためのデータ構造が二分木を用いて実装されている．GAIO そのものは標準では精度保証付き数値計算を用いない設計になっているが，以下に述べる PROFIL などを書いた C/C++ の関数を MATLAB から呼び出して組み合わせることができる．

4.2 CHomP (Computational HOMology Project)

<http://www.math.gatech.edu/~chomp/>

位相空間のホモロジー群や，その上の連続写像がホモロジー群に導く準同形を求めるためのソフトウェア．そもそも Conley 指数を具体的な力学系で計算したいという動機のもとに開発が始まったソフトウェアであるが，力学系の研究だけでなく偏微分方程式が生成するパターンの解析やイメージプロセッシングなどにも広く応用されている．多くの数学者が開発にかかわっているが，コーディングの中心となっているのは P. Pilarczyk である．

4.3 BIAS (Basic Interval Arithmetic Subroutines)

<http://www.ti3.tu-harburg.de/knueppel/profil/>

O. Knüppel により開発された C 言語用の区間演算ライブラリである (下に述べる PROFIL と共に配布されている)。区間は BIASINTERVAL という構造体により表現される。C では演算子のオーバーロードができないため、複雑な計算を書くためには以下で触れる PROFIL や b4m といったラッパーを用いるほうが便利であろう。

注意しなくてはならないのは、BIAS による厳密な区間演算ができるのは四則演算に限られることである。BiasF.c での関数の定義を見ればわかる通り、BIAS が返してくる \sin , \cos , \exp などの初等関数の値は数学的に保証されていない。例えば BiasExp でやっていることは、入力された区間の両端で libm の \exp を呼び出して、その値を少し膨らませた区間を返すだけである。現実問題として BIAS が返す区間から真の値が外れることはないかも知れないが、これでは証明をしたことにはならないので注意されたい。

4.4 b4m (BIAS for MATLAB)

<http://www.ti3.tu-harburg.de/zemke/b4m>

J. Zemke により書かれた MATLAB から BIAS を呼び出すためのパッケージである。例えば `x = interval(1,2)` とおくだけで `x` は閉区間 $[1, 2]$ となり、あとは通常の実数と同様の感覚で計算を進められる。インタープリタ言語である MATLAB の簡明さとあいまって非常に手軽で便利であるが、内部で BIAS を呼び出していることから上で書いた BIAS に対する注意がそのままあてはまることに注意。

4.5 PROFIL (Programmer's Runtime Optimized Fast Interval Library)

<http://www.ti3.tu-harburg.de/knueppel/profil/>

BIAS 同様 O. Knüppel により開発された C++ 言語用の区間演算ライブラリである。やはり内部で BIAS を呼び出すため、BIAS に対する注意がそのままあてはまる。

4.6 CAPD (Computer Assisted Proofs in Dynamics)

<http://capd.wsb-nlu.edu.pl/>

§ 2 で紹介した [6] の Z. Galias や P. Zgliczyński, また CHomP の P. Pilarczyk など一連のポーランドの研究者が開発した精度保証付き数値計算を用いた計算機援用証明のためのパッケージである。§ 2 の最後で触れた Lohner アルゴリズムなどを含むベクトル場の厳密な積分のためのルーチンが数多く含まれている。

5 おわりに

計算機を援用した研究には常に「数学の証明は紙と鉛筆のみで理解できるものであるべきではないか」という議論がつきまとう。確かに自分の手で計算を追いかけないと本当に理解したという気分にはなれない，それは当然の感覚であろう。しかし一方で精度保証計算を続けていると，通常の数値計算をしている時には感じなかった計算機への信頼とでも呼ぶべき感覚が育って来るのも事実である。数学と計算機をめぐる議論について深く考えるためにも，ぜひ皆様に精度保証計算を実際に試してみていただきたい。

参考文献

- [1] 荒井 迅, 計算機支援による離散力学系の解析, *応用数理*, **15** (2005), 20–31.
- [2] Z. Arai, On Hyperbolic Plateaus of the Hénon Maps, preprint.
- [3] Z. Arai and K. Mischaikow, Rigorous Computations of Homoclinic Tangencies, preprint.
- [4] C. Bonatti, L. Díaz and M. Viana, *Dynamics Beyond Uniform Hyperbolicity*, Encyclopaedia of Mathematical Sciences, **102**, Springer-Verlag, 2005.
- [5] M. Dellnitz and O. Junge, The algorithms behind GAIO - set oriented numerical methods for dynamical systems, *Ergodic theory, analysis, and efficient simulation of dynamical systems*, Springer, Berlin, 2001, 145–174, 805–807.
- [6] Z. Galias and P. Zgliczyński, Computer assisted proof of chaos in the Lorenz equations, *Physica D*, **115** (1998), 165–188.
- [7] T. Kaczynski, K. Mischaikow and M. Mrozek, *Computational Homology*, Applied Mathematical Sciences, **157**, Springer-Verlag, 2004.
- [8] K. Mischaikow and M. Mrozek, Chaos in the Lorenz equations: a computer-assisted proof, *Bull. Amer. Math. Soc. (N.S.)*, **3** (1995), 66–72.
- [9] K. Mischaikow and M. Mrozek, Chaos in the Lorenz equations: a computer-assisted proof. II. Details, *Mathematics of Computation*, **67** (1998), 1023–1046.
- [10] K. Mischaikow and M. Mrozek, Chaos in the Lorenz equations: a computer-assisted proof. III. Classical parameter values, *J. Differential Equations*, **169** (2001), 17–56.
- [11] K. Mischaikow and M. Mrozek, *The Conley index theory*, Handbook of Dynamical Systems II, North-Holland, 2002, 393–460.
- [12] 中尾充宏・山本野人, 精度保証付き数値計算, 日本評論社, 1998.
- [13] 大石進一, 精度保証付き数値計算, コロナ社, 2000.
- [14] W. Tucker, A rigorous ODE solver and Smale’s 14th problem, *Found. Comput. Math.*, **2** (2002), 53–117.
- [15] A. Szymczak, The Conley index and symbolic dynamics, *Topology*, **35** (1996), 287–299.
- [16] P. Zgliczyński, C^1 Lohner algorithm, *Found. Comput. Math.*, **2** (2002), 429–465.