

## 区間演算による充填ジュリア集合の内点の保証

大墨 礼子\*

近藤 祐史†

藤村 雅代‡

関東学院大学

香川高等専門学校

防衛大学校

(Received October 31, 2024    Revised January 31, 2025    Accepted February 3, 2025)

### 概 要

The aim of this study is to find a method to draw filled-in Julia sets with less computational effort and fewer errors. There are a number of algorithms for drawing filled-in Julia sets, as well as methods for controlling errors and reducing computational costs. In this study, we primarily use the level set method as the drawing algorithm and interval arithmetic as the computational technique. The level set method is a simple algorithm that plots the interior points of the filled-in Julia set. However, near the boundaries of the set, interior points may not be plotted correctly, or exterior points may be erroneously included due to computational errors.

In this paper, we analyze the computational behavior of experiments involving the drawing of filled-in Julia sets using interval arithmetic, which is implemented in the Risa/Asir computer algebra system. Moreover, we present a computational method that guarantees the inclusion of interior points in the Julia set by combining the level set method with interval arithmetic.

### 1 はじめに

反復計算は数理学のシミュレーションの際の数値計算において一般的に使用される手法である。数値計算では計算には数値が使用されるが、浮動小数点数を用いた反復計算では、誤差の累積が起こることはよく知られている。そのため数値計算の分野では、誤差の抑制のための研究が多くあるが、ここでは数式処理を用いての反復計算について考える。数式処理を用いての計算では、得られる結果は浮動小数点数を扱わない限り数学的に正確である。しかし、数値計算と比較すると有理数計算など数学的な正確さを保証する計算は計算速度が遅く、メモリも多く必要となる。

筆者らは、数式処理システムを用いて計算し、充填ジュリア集合を数学的に正確に描画することを目指している。多項式  $p$  の充填ジュリア集合  $K(p)$  は、

$$K(p) = \{z \in \mathbb{C}; p^n(z) \not\rightarrow \infty \text{ as } n \rightarrow \infty\},$$

\*osumi@kanto-gakuin.ac.jp

†kondoh@di.kagawa-nct.ac.jp

‡masayo@nda.ac.jp

と定義される.  $z$  は初期値となる点であり,  $p^n$  は  $p$  の  $n$  回目の反復を表している.

充填ジュリア集合の定義は非常に単純であるが, 正確な描画は難しい. 充填ジュリア集合の描画アルゴリズムは数多く研究されており, レベルセット法 (LSM), 逆反復法 (IIM), 距離推定法 (DEM) 等があるが, それぞれに長所, 短所がある.

ここで, 非常に単純なアルゴリズムである LSM を用いて充填ジュリア集合をできるだけ誤差を排除して描画することを考える. LSM は, 境界付近の描画に難点があることは知られており, 充填ジュリア集合の内点となる点が描画されない, また逆に内点ではない点が内点として描画される場合もある. さらに, 反復回数が多いほど描画精度が高いとされているが, 反復回数をどの程度とするかを推定する方法は知られていない.

充填ジュリア集合の描画においては,

1. 描画された点は充填ジュリア集合の内点であるのか?
2. 反復回数をどの程度とするか?

が本質的な問題である. 本論文では, 1つ目の問題である「描画された点が充填ジュリア集合の内点であるのか」について議論する.

## 2 準備

### 2.1 複素力学系からの準備

この章では, 本論文で必要になる複素力学系の用語やテクニックについて簡単に解説する. より詳しい内容については, 複素力学系の標準的な教科書である [1] や [3] を参照してほしい.

以降,  $p$  は  $d(\geq 2)$  次多項式を表すものとする.  $\widehat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$  とする.  $p$  の第  $n$  回反復を  $p^n$  で表す. すなわち,  $p^n = \underbrace{p \circ \cdots \circ p}_n$  とする.  $p$  の **ファトウ集合**  $F(p)$  は次で定まる集合である.

$$F(p) = \{z \in \widehat{\mathbb{C}}; z \text{ のある近傍で } \{p^n\} \text{ は正規族}\}.$$

一方,  $F(p)$  の補集合を **ジュリア集合** と呼び  $J(p)$  と記す.

多項式  $p$  に対して, 無限遠点  $\infty$  は, 常に超吸引不動点である. この無限遠点の吸引鉢を  $A(\infty)$  と記す. すなわち,

$$A(\infty) = \{z \in \widehat{\mathbb{C}}; p^n(z) \rightarrow \infty \text{ as } n \rightarrow \infty\}.$$

$A(\infty)$  は連結集合であり, 定義から開集合であることがわかる. したがって,  $A(\infty)$  の補集合は閉集合であり, これを  $K(p)$  と書き  $p$  の **充填ジュリア集合** と呼ぶ. さらに, 充填ジュリア集合  $K(p)$  の境界は, ジュリア集合  $J(p)$  に一致することが知られている (例えば, [3, Lemma 9.4] を参照).

### 2.2 充填ジュリア集合の描画アルゴリズム

充填ジュリア集合の描画アルゴリズムとして様々なものが知られているが, ここでは, 今後の議論で必要になるレベルセット法 (LSM) と距離推定法 (DEM) を紹介する.

### 2.2.1 レベルセット法 (LSM)

レベルセット法 (LSM) は, 多項式  $p$  の充填ジュリア集合  $K(p)$  の描画アルゴリズムとしては最も単純なものであり,  $p$  の無限遠点の吸引鉢  $A(\infty)$  の補集合が充填ジュリア集合  $K(p)$  であることを利用したアルゴリズムである.

$R$  を十分大きな正の数 ( $R > 4$ ) とし,  $N$  を自然数とする. 各点  $z \in \mathbb{C}$  に対して,

$$z_n = p^n(z) \quad (n = 0, 1, 2, \dots) \quad \text{および} \quad n(z, R) = \min\{n; |z_n| \geq R\}$$

を計算する. このとき, 点  $z$  は  $n(z, R) = \infty$  のときに限り,  $K(p)$  に属する.

次のようにおく,

$$\begin{aligned} K(p)_{\text{LSM}}(N, R) &= \{z \in \mathbb{C}; n(z, R) > N\} \\ &= K(p) \cup \{z \in A_0; n(z, R) > N\}, \end{aligned}$$

ただし,  $A_0 = A(\infty) \setminus \{\infty\}$  とする.

このとき,  $N \rightarrow \infty$  とすれば  $K(p)_{\text{LSM}} \downarrow K(p)$  となる. これがレベルセット法のアルゴリズムである. しかし, 残念ながら  $K(p)_{\text{LSM}}$  が常に  $K(p)$  の良い近似を与えるとは限らない.

### 2.2.2 距離推定法 (DEM)

距離推定法 (DEM) は, 充填ジュリア集合  $K(p)$  の外側から  $K(p)$  の境界までのユークリッド距離を評価することで  $K(p)$  を描くアルゴリズムである (DEM 法について詳しい解説は, [5] や [4] を, この方法の精度については [2] を参照のこと). DEM は  $K(p)$  が連結集合のとき, すなわち  $A(\infty)$  が単連結集合のときに, 良い近似を与えることが知られている.

単位円板を  $\mathbb{D}$  と表すことにする. 次で定義される ベトヒャ写像  $\varphi$

$$\varphi(z) = \lim_{n \rightarrow \infty} \varphi_n(z) \quad \text{ただし} \quad \varphi_n(z) = (p^n(z))^{1/d^n}$$

は  $A(\infty)$  から  $\widehat{\mathbb{C}} \setminus \overline{\mathbb{D}}$  上への正則写像であり,

$$\varphi(p(z)) = (\varphi(z))^d,$$

をみたま (例えば [3, §9, 18] を参照のこと). このとき, ケーベの 4 分の 1 円定理を適用して,

$$\frac{1}{4} \cdot \frac{|dz|}{d(z, \partial K(p))} \leq \frac{|dw|}{d(w, \partial \mathbb{D})} \leq 4 \cdot \frac{|dz|}{d(z, \partial K(p))}, \quad z \in A_0, \quad w = \varphi(z),$$

が得られる. ただし,  $d$  はユークリッド距離を表すものとし,

$$d(z, \partial K(p)) \approx \frac{|\varphi(z)| - 1}{|\varphi'(z)|} \approx \frac{|\varphi_n(z)| - 1}{|\varphi_n'(z)|}, \quad z \in A_0.$$

である. さらに, もし  $z \in A_0$  が  $K(p)$  の境界に近ければ, 次が成り立つ.

$$|\varphi_n(z)| - 1 \approx \log |\varphi_n(z)| = \frac{1}{d^n} \log |p^n(z)|,$$

$$|\varphi'_n(z)| = \frac{1}{d^n} |(p^n)'(z)| |p^n(z)|^{\frac{1}{d^n} - 1} = \frac{|\varphi_n(z)| |(p^n)'(z)|}{d^n |p^n(z)|} \approx \frac{|(p^n)'(z)|}{d^n |p^n(z)|}.$$

これにより,  $z \in A_0$  が  $K(p)$  の境界に近くて,  $n$  が十分に大きければ, 次を得る.

$$d(z, \partial K(p)) \approx d_n(z, \partial K(p)), \quad (1)$$

となる. ただし,

$$d_n(z, \partial K(p)) = \frac{|z_n| \log |z_n|}{|z'_n|} \quad (z_n = p^n(z), z'_n = (p^n)'(z)).$$

また,  $z_n$  と  $z'_n$  は

$$z_0 = z, z'_0 = 1, \quad z_{n+1} = p(z_n), z'_{n+1} = z'_n p'(z_n),$$

を満たすので, (1) は  $K(p)$  の  $\varepsilon$  近傍の (ような) 集合を描くアルゴリズムを与える. これが DEM による  $K(p)$  の描画アルゴリズムである.

### 2.3 数式処理システム Risa/Asir の区間数および区間演算

区間演算は数値計算において精度保証に用いられる演算手法である. 浮動小数点数の計算では, 丸め誤差等の誤差の影響を考える必要があるが, 数値を区間で表現し, その区間の中に真値が必ず含まれることを保証する.

数式処理システム Risa/Asir では区間演算の実装がなされている [6]. Risa/Asir 上での区間演算及び区間数とは,

$$A = \{x; \underline{a} \leq x \leq \bar{a}\} \quad x, \underline{a}, \bar{a} \in \mathbb{R}$$

なる  $A$  を区間数と呼び,  $A = [\underline{a}, \bar{a}]$  と表す. ただし,  $\underline{a} \leq \bar{a}$  とする.  $\underline{a}, \bar{a}$  それぞれを区間数の下限, 上限と呼ぶ.

また, 2つの区間数  $A = [\underline{a}, \bar{a}], B = [\underline{b}, \bar{b}]$  の間の演算を次のように定義する. ここで, 英大文字は区間数, 英小文字は実数を表す.

$$A + B = [\underline{a}, \bar{a}] + [\underline{b}, \bar{b}] = [\underline{a} + \underline{b}, \bar{a} + \bar{b}],$$

$$A - B = [\underline{a}, \bar{a}] - [\underline{b}, \bar{b}] = [\underline{a} - \bar{b}, \bar{a} - \underline{b}],$$

$$A \cdot B = [\underline{a}, \bar{a}] \cdot [\underline{b}, \bar{b}] = [\min(\underline{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\underline{b}, \bar{a}\bar{b}), \max(\underline{a}\underline{b}, \underline{a}\bar{b}, \bar{a}\underline{b}, \bar{a}\bar{b})],$$

ただし  $0 \notin B$ ,

$$A / B = [\underline{a}, \bar{a}] / [\underline{b}, \bar{b}] = [\underline{a}, \bar{a}] \cdot [1/\bar{b}, 1/\underline{b}]$$

$$= [\min(\underline{a}/\bar{b}, \underline{a}/\underline{b}, \bar{a}/\bar{b}, \bar{a}/\underline{b}), \max(\underline{a}/\bar{b}, \underline{a}/\underline{b}, \bar{a}/\bar{b}, \bar{a}/\underline{b})].$$

Risa/Asir では特に設定しない場合は, 区間の範囲  $a, b$  は倍精度浮動小数点数となる.

また, Risa/Asir の区間演算では, ゼロ書き換えモードが実装されている. ゼロ書き換えとは, 区間演算の結果がゼロが含まれる区間数となった場合, その区間をゼロに置き換える演算のことである. ゼロ書き換えの機能は本論文では使用しない. 充填 Julia 集合の描画において, 反復計

算の途中の値をゼロに書き換えることは与えられた多項式  $p(z)$  の点  $z=0$  の軌道  $\{p^n(0)\}$  を計算することに相当する. 詳しくは第3節で述べるが, 本論文での計算設定で使用するのは  $c$  をマンデルブロ集合の点とした多項式  $f_c(z) = z^2 + c$  である. そのため, 常に  $f_c^n(0) \rightarrow \infty$  が成り立ち, 本来は判定不能な初期値であってもゼロ書き換えにより, 初期値は充填 Julia 集合の点であると判定されてしまうためである.

### 3 区間演算を用いた充填ジュリア集合の描画実験と計算結果の分析

本章では, 区間演算を用いて充填ジュリア集合の計算実験を行い, その結果について分析する. よく知られている通り, 浮動小数点数を用いた計算では, 計算結果に誤差が含まれる可能性がある. 充填ジュリア集合の計算でも, 浮動小数点数を用いれば, 高速に多くの反復回数でも計算可能である. しかし, その結果には誤差が含まれる可能性があるため, 得られた点が必ずジュリア集合であるかどうかの判断はできない. もちろん, 多倍長計算を用いれば, より長い桁を用いて計算精度を向上させることは可能であるが, 誤差を含むことには変わりはない. 図1は, 多倍長計算を用いて計算した結果である.

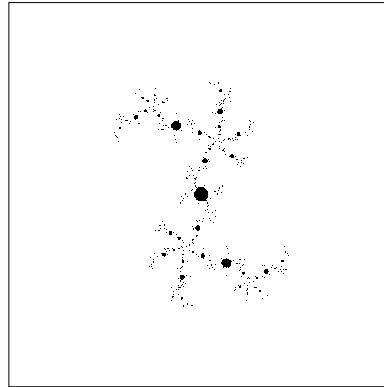


図1: 多倍長浮動小数点数での計算結果 (計算精度 100 桁, 反復 500 回,  $c = 0.4433256+0.3729624i$ ).

数学的に正確な計算をするためには, 有理数を用いて計算する必要がある. 数式処理システムを用いれば, 有理数の計算も可能である. しかし, 数式処理システムでの有理数の計算では, 特に充填ジュリア集合の計算は数多くの反復計算を行うため, 分子, 分母ともに巨大な数となり計算を終えることができない.

そこで, 我々は数式処理システム Risa/Asir 上で, 区間演算を用いて充填ジュリア集合の計算を行うことを考える.

#### 3.1 計算設定

ここで,  $c$  はマンデルブロ集合の点とし,  $f_c(z) = z^2 + c$  とする. この時, 充填ジュリア集合  $K(f_c)$  は連結であることが知られている. 計算アルゴリズムに LSM を用い,  $R$  を 5 とし,  $D = \{n; |z| < 5\}$  であるとする. また  $N$  を反復計算の上限とする. 本論文での計算設定では, 充

填 Julia 集合は半径 2 の閉円版に含まれることが分かっている．ある反復回数と十分大きな  $R$  に対して，初期値が  $|f_c^{m_0}(z)| > R > 2$  を満たせば， $|f_c^{m_0+1}(z)| \approx R^2$  が成り立つので，領域  $D$  の半径が例えば 10 から 100 に変わったとしても反復回数の増加は 1 回程度である．したがって， $R$  の大きさは反復回数にほとんど影響しないため，上記の値とした．

各格子点は各軸とも  $-2$  から  $2$  の範囲で  $\frac{1}{150}$  の幅で刻み，数式処理システム Risa/Asir を用いて計算および描画を行う．初期値  $Z$  は，格子点  $z$  の区間幅を計算機イプシロンとして実部，虚部ともに区間数としたものである．同様に， $c$  の値も，区間幅が計算機イプシロンの区間数とする．本論文の計算実験では，Risa/Asir の bigfloat を用い，有効桁数は 100 桁に設定した．

計算後の各点の判定方法は，区間数  $f_c^N(Z)$  の 4 つの端点（頂点）がすべて  $D$  内の点であれば，格子点  $z$  は  $K(f_c)$  の点と判定する．

図 2 と図 3 はこの設定により描画した結果である．図 1 と図 2 はいずれも  $c = 0.4433256 + 0.3729624i$  の  $K(f_c)$  の場合の，LSM+ 浮動小数点数，LSM+ 区間数を用いて計算した結果を示している．

2 つの結果を比較して，図 1 と図 2 は  $K(f_c)$  の点として信頼できる結果であるか? という問題を考える．この 2 つの図に描画された点は，計算結果の信頼性の面で異なっている．図 1 は浮動小数点数での計算である．多倍長計算を用いて 100 桁の桁数を用いているため，倍精度浮動小数点数を用いた計算と比較すれば計算精度は向上しており，含まれている誤差もごく小さくなっている．しかし，100 桁の桁数を用いたとしても，計算結果には微細な値であるとはいえ丸め誤差を含んでいるため，その結果が数学的に正しいものであるかの保証はできない．一方の図 2 は区間演算を用いた計算結果である．区間演算は，2.3 に記した通り，計算結果の区間内に必ず真値が含まれていることが保証されている．つまり，図 2 は，得られた結果は真値であることが保証可能である．

次節では，計算結果の判定プロセスについてより詳しく説明する．

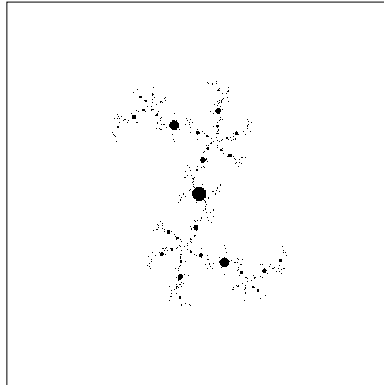


図 2: 区間演算での計算結果 (計算精度 100 桁, 反復 500 回,  $c = 0.4433256 + 0.3729624i$ ).

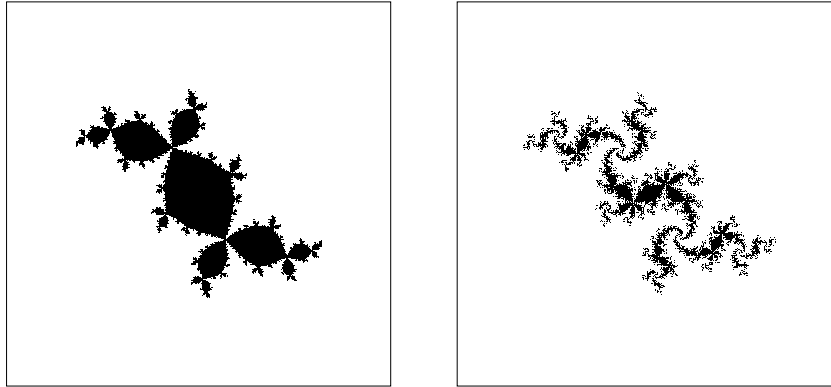


図 3: レベルセット法と区間演算による描画 (左:  $c = -0.1226 + 0.7449i$ , 右:  $c = -0.222 + 0.71838i$ )

### 3.2 レベルセット法と区間演算を用いて計算した充填ジュリア集合の検証

ここで、まず区間数の特徴を再確認する。前述したとおり、区間数とは、幅を持ち、その間に真の値があることを保証して計算が可能である。区間数の両端は浮動小数点数であるので、設定された桁数の上限を超えた場合は丸めによる誤差が必ず発生し、計算を繰り返すことによって、誤差の累積により区間は計算前の幅よりも広がる。

初期値である格子点は、ごく小さな矩形または点である。これを  $Z$  とおく。区間数を適用した充填ジュリア集合の計算を考える。計算開始前は、各格子点の値は区間または点でスタートし、反復を繰り返すことで、図 4 のように複素平面上を移動する。移動の際には、区間数であるため、計算を繰り返すたびに、区間は少しずつ広がっていく。充填ジュリア集合の計算では、実軸、虚軸方面どちらも区間数であるため、少しずつ大きな矩形に変化しながら移動する。

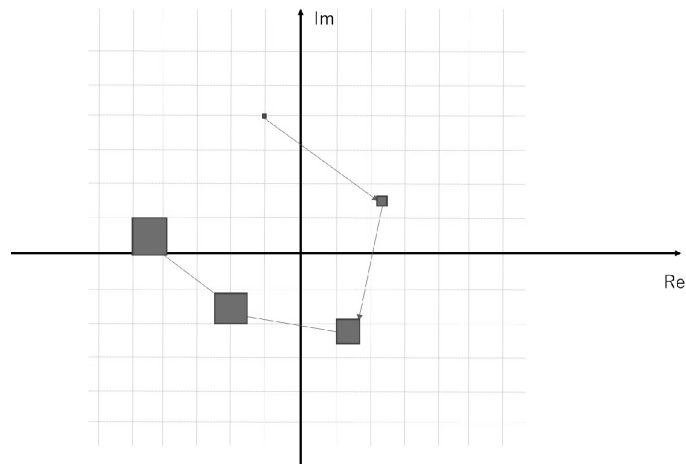


図 4: 反復計算時の区間数の移動イメージ

ここで、計算結果の判定方法について考える。反復計算を 1 回行うごとに、格子点の区間が  $D$  内にあるかどうかを判定する。もし、格子点の区間が  $D$  内にあり、かつ反復回数が  $N$  未満の場合は、反復計算を継続する。そうでない場合はその格子点の計算を終了する。

次に計算結果の示す意味について考える。計算結果は、 $Z$  が

- (1)  $K(f_c)$  の内点である, (2)  $K(f_c)$  の外の点となる, (3) 判別不可能,

のいずれかのケースとなる。それぞれのケースを詳しく見ていこう。

(1) の場合:  $f_c$  で  $Z$  を  $N$  回反復したあと、 $f_c^N(Z)$  のすべての端点が  $D$  内に残っていれば、その初期格子点は  $K(f_c)$  の点であると判定する (図 5)。一般に区間数を用いて反復計算を繰り返すと、図 4 に示す通り区間の幅は大きくなる。しかしこの場合、 $f_c^N(Z) \subset D$  であることと、真の値が  $f_c^N(Z)$  にあることから初期格子点は充填ジュリア集合の内点であることが保証できる。

したがって、区間演算を用いて充填ジュリア集合の内点であると判定された点は実際に充填ジュリア集合の内点であることが保証可能である。

しかし、すべてのジュリア集合の内点が検出できるわけではない。反復計算によって区間が膨らみ、本来であれば内点であるものがそうではないと判定されている可能性もある。

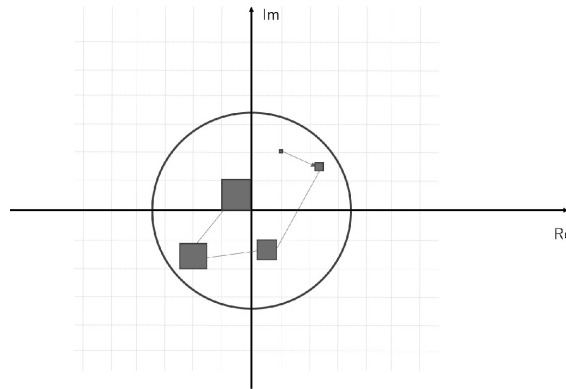


図 5: (1) の場合の計算例 (計算結果がすべて  $D$  の内側)

(2) の場合: 計算後の区間  $f_c^N(Z)$  の端点がすべて  $D$  の外側であり、かつ  $f_c^N(Z) \cap D = \emptyset$  を満たす場合を考える (図 6)。この場合 (1) と同様に初期格子点  $Z$  が区間を膨張させながら移動し、最終的に区間すべての端点が、 $D$  の外側に  $D$  と交わることなく存在するため、膨張後であっても充填ジュリア集合の外の点であると判定される。

(3) の場合: この場合は以下の二つが考えられる。

- ・計算後の区間の端点  $f_c^N(Z)$  がすべて  $D$  の外側であるが、区間が大きく膨張して図 7 のように区間自体が  $D$  を覆ってしまう場合。この場合は、 $f_c^N(Z)$  は  $K_c(f_c)$  の内点を含む可能性があるため、 $Z$  は  $K_c(f_c)$  の内点であるとも外の点であるとも判定不可能である。

- ・計算結果が図 8 のようになる場合。すなわち  $f_c^N(Z)$  の端点のうちいくつかは  $D$  の外であるが、残りは内部に残るケース。この場合も、 $Z$  は  $K_c(f_c)$  の内点とも外の点であるとも判定不能である。



次節では、(3)の后者の区間の端点の一部が $D$ の中となるため判定不能となる場合に $Z$ が $K_c(f_c)$ の内点であるか外部の点であるかの評価が可能かを議論する。

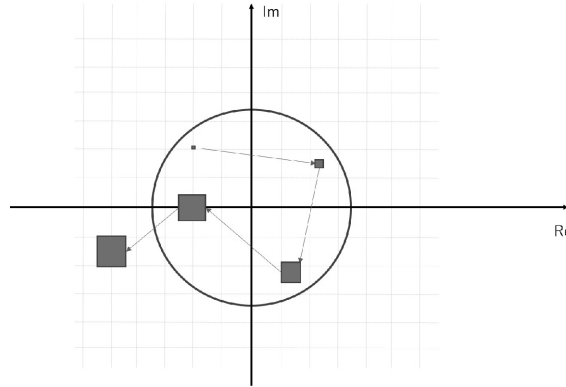


図 6: (2) の場合の計算例 (計算結果がすべて  $D$  の外側)

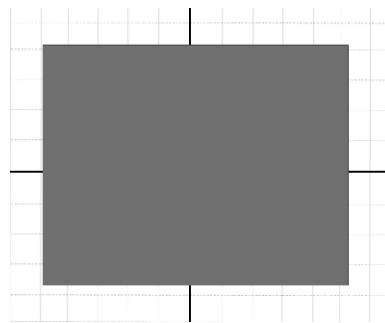


図 7: (3) の場合の計算例 (計算結果の端点  $Z$  が  $D$  の外側かつ  $D$  を覆っている)

### 3.3 DEM と区間演算で得られるジュリア集合の外側の領域の確定

ここでは、前節にて内点とも外の点とも確定不可能であった点  $Z$  を、DEM を用いて充填ジュリア集合の外側の領域であるかを判定することを考える。DEM は充填ジュリア集合の  $\varepsilon$  近傍を描画する方法であり、充填ジュリア集合の外の点からのユークリッド距離を用いて評価する方法である。DEM を区間演算を用いて計算し、LSM と同様に、区間の端点  $Z$  がすべて外部である条件を満たす場合を、充填ジュリア集合の外部の点であると評価する。LSM の結果と DEM の結果を組み合わせることにより、確実に充填ジュリア集合の外側の点であることが保証できる。図 9 は、区間演算と DEM を用いて充填ジュリア集合外の点および  $\varepsilon$  近傍を描画したものである。グレーのエリアが外部、黒で示した部分が  $\varepsilon$  近傍となる。

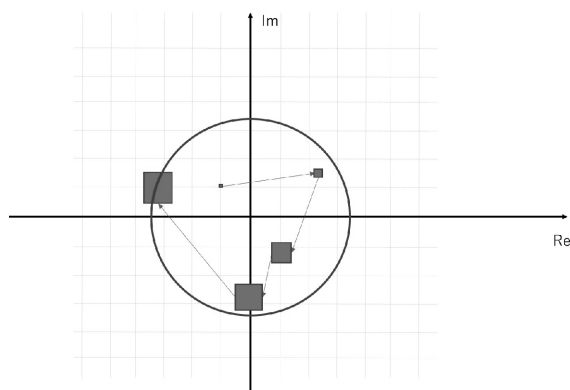


図 8: (3) の場合の計算例 (計算結果の端点が  $D$  の内側と外側が混在)

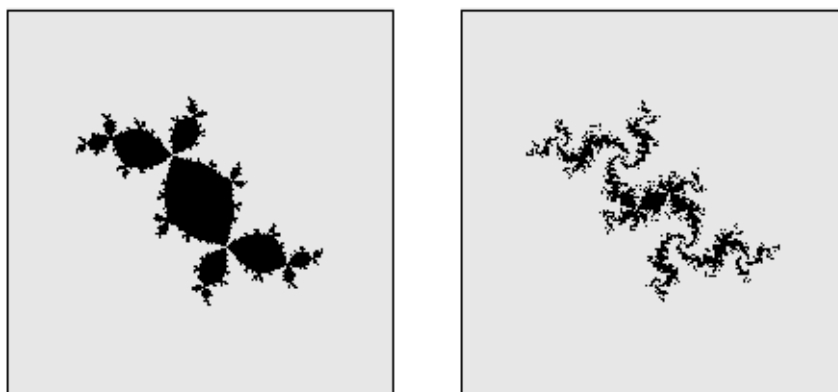


図 9: DEM と区間演算での  $K(f_c)$  の描画 (左:  $c = -0.1226 + 0.7449i$ , 右:  $c = -0.222 + 0.71838i$ ).

#### 4 まとめと今後の課題

LSM と DEM を区間演算を用いて計算することにより，充填ジュリア集合の内点および外点が判定できることを示した．しかし，2つのアルゴリズムを併用しても判定できない点に関しては，区間の膨張を抑える方法の検討など今後のさらなる実験及び検証が必要である．

また，計算時間に関しては，区間演算では区間の端点すべてに対する計算を行う必要があるため，浮動小数点数での計算の場合に比べれば遅くなる．しかし，有理数での計算と比較すれば，十分現実的な計算時間で結果を得られるため，充填ジュリア集合の計算においては，区間演算は有効な手段であると言える．

#### 謝辞

我々の原稿を注意深く読み，有益なコメントをいただいた査読者に感謝の意を表する．

### 参考文献

- [1] Beardon, A.F.: Iteration of rational functions, Graduate Texts in Mathematics, Vol. 132. Springer-Verlag, New York (1991). <https://doi.org/10.1007/978-1-4612-4422-6>.
- [2] Fujimura, M., Gotoh, Y., Yoshida, S.: Explicit estimates on distance estimator method for Julia sets of polynomials. Kodai Math. J. Vol.36, No.3, pp. 491-507 (2013). <https://doi.org/10.2996/kmj/1383660695>.
- [3] Milnor, J.: Dynamics in one complex variable, Annals of Mathematics Studies, vol. 160. Princeton University Press, Princeton, NJ, third edn. (2006).
- [4] Milnor, J.: Self-similarity and hairiness in the Mandelbrot set. In: Computers in geometry and topology (Chicago, IL, 1986), Lecture Notes in Pure and Appl. Math., Vol. 114, pp. 211-257. Dekker, New York (1989).
- [5] Peitgen, H.O., Richter, P.H.: The beauty of fractals. Springer-Verlag, Berlin (1986). <https://doi.org/10.1007/978-3-642-61717-1>.
- [6] 近藤祐史: 区間演算と数式処理の歴史, 数式処理 Vol.12, No.1, pp.23-31 (2005).
- [7] 大墨礼子, 近藤祐史, 藤村雅代: 区間演算を用いた反復計算について, 京都大学数理解析研究所講究録 2255, pp.26-31 (2023).