

28.

ifplot アルゴリズム

齋藤友克 (上智大学)

三好善彦 (埼玉女子短大)

近藤祐史 (詫間電波高専)

28.1 始めに

代数関数のグラフを描く問題は、様々なアルゴリズムが提示されている。しかし、与えられた精度で関数のグラフを確実に描くアルゴリズムは現在まで知られていない。この問題を我々は ifplot 問題と呼び解決に取り組んできた。これまで中間値アルゴリズム、strum アルゴリズム、区間演算アルゴリズム等有用なアルゴリズムを開発して来た。しかし、孤立特異点、微小な解等判定できない場合が存在した。今回この ifplot 問題に対し 2 変数代数関数の場合に限って一般的なアルゴリズムを発見し、問題に対する解決を見た。

28.2 ifplot の定義

ifplot において与えられた関数のグラフを描くとは任意の与えられた領域 S と関数 $f = 0$ の零点集合がなんらかの意味で共通部分を持つかどうか判定出来れば良い。そのために必要な定義は、領域に関する定義として基本単位系、どのような共通部分を持つかに対する定義として描画関数を次のように定める。

28.2.1 描画基本単位系

D 上定義された χ の基本単位系 S とは

$$\left\{ \begin{array}{l} S_i \text{ は } D \text{ の compact subset,} \\ S \text{ は } S_i \text{ の有限集合,} \\ D = \bigcup_{i=1}^n S_i, \\ S_i^i \cap S_j^i = \phi \quad i \neq j \end{array} \right.$$

ここで S^i は S の内点の集合である。

28.2.2 描画関数

描画関数 $\chi(D, S)(f)$ とは

$$\left\{ \begin{array}{l} 1. \quad \chi : S \rightarrow \{0, 1\} \\ 2. \quad \chi(S_i) = 0 \quad S_i \text{ の任意の元 } x \text{ に対し } f(x) \neq 0 \end{array} \right.$$

この $\chi(D, S)$ を D における S による character とよぶ。

28.2.3 様々な描画関数

■ signature character

D 上 S による signature character $\chi(D, S)$ とは

$$\left\{ \begin{array}{l} 1. \quad \chi \text{ は } S \text{ による character} \\ 2. \quad \chi(S_i) = 1 \quad \exists x, y \in \partial S_i \quad \text{s.t.} \quad f(x)f(y) \leq 0 \end{array} \right.$$

ここで ∂S_i は集合 S_i の境界からなる集合である。

■ boundary character

D 上 S による boundary character $\chi(D, S)$ とは

$$\left\{ \begin{array}{l} 1. \chi \text{ は } S \text{ による } \textit{character} \\ 2. \chi(S_i) = 1 \quad \exists x \in S_i \quad \text{s.t. } f(x) = 0 \end{array} \right.$$

■ faithful character

D における S による faithful character χ とは

$$\left\{ \begin{array}{l} 1. \chi \text{ は } S \text{ による } \textit{character} \\ 2. \chi(S_i) = 1 \quad \exists x \in S_i \quad \text{s.t. } f(x) = 0 \end{array} \right.$$

28.3 過去の経緯

ifplot 問題の解決法として現在まで確立されたアルゴリズムは中間値アルゴリズム、sturm アルゴリズム、区間演算アルゴリズムが提案されている。

28.3.1 中間値アルゴリズム

このアルゴリズムは、signature character を求めるため T.Saito により 1990 年に提案されたもので描画する最小単位の外周の関数値の符号の変化により描画をするものである。中間値の定理を素朴に利用するため関数の変化係数にたいし丈夫でありアルゴリズムとしては強固である。一般的な利用に対しては十分実用に耐え得るアルゴリズムである。

しかし、判定を行う区間に関数の解が偶数個存在した場合および孤立特異点が内部に存在した場合検出出来ない。

このアルゴリズムは Risa/Asir の ifplot として提供されている。

28.3.2 sturm アルゴリズム

中間値アルゴリズムを発展させたアルゴリズムで M.Noro, T.Takeshima により 1992 年に提案されたアルゴリズムである。このアルゴリズムを適用することにより boundary character を求めることが出来る。

判定に用いる理論として中間値の定理ではなく sturm の定理を利用し全ての計算を有理数演算としておこなう。外周部に解が存在すれば確実に判定できる。

しかし、このアルゴリズムを利用しても内部にある孤立特異点と内部にある閉じた 1 次元の解は検出できない。

このアルゴリズムは Risa/Asir の ifplot の precise モードにより提供されている。

28.3.3 区間演算アルゴリズム

1993 年 Y.Kondoh, T.Saito により提案されたアルゴリズムで図を正確に求めるよりも孤立特異点の位置を推定するために開発されたものである。関数演算を区間演算を利用して計算することにより解が存在する可能性の高い範囲を求めることが出来る。

またこのアルゴリズムを繰り返し適用することにより計算量は別として確実に必要な精度で関数の解の特異点を含め求めることが出来る。

28.4 2変数多項式の faithful character

28.4.1 前提

faithful character を求める理論を展開するには、基本単位系と描画領域が同一である場合を考察すればよい。よって以下のことを仮定する。

$$\left\{ \begin{array}{l} S = \{S\} \text{ とする。} \\ \text{基本単位系 } S \text{ の成分 } S \text{ は長方形である。} \end{array} \right.$$

■事前判定

解が領域の境界と共通部分を持つならば事前に boundary character を求めることにより解の存在を計算できる。よって faithful character を求めるためには以下の仮定を付けて良い。

$$\partial S \text{ 上には } f = 0 \text{ となる点は存在しない。}$$

■注意

以上の仮定により解は 0 次元であるか 1 次元の円周と同相な図形である。

28.4.2 0次元の解の存在判定

0次元の解は孤立特異点であるから $f(x, y) = 0$ と $\partial f / \partial x = 0, \partial f / \partial y = 0$ の共通零点である。よって

$$\begin{cases} f(x, y) = 0, \\ \partial f / \partial x = 0, \\ \partial f / \partial y = 0 \end{cases}$$

の解を Gröbner Base を用いて求めることにより判定できる。

28.4.3 1次元の解の存在判定

前段階の操作により 0 次元の解の存在を判定することで領域 S 内には 0 次元の解は存在しないと仮定できる。また解はもし存在するならば仮定により S に含まれる閉路である。よって、閉路に囲まれた境界を含む領域は compact であるから、その上の連続関数は極大もしくは極小点を持つ。その点を Ω とする。

$$\Omega = \{(x_1, y_1), \dots, (x_m, y_m)\}$$

Ω の各点は $\partial f / \partial x = 0$ と $\partial f / \partial y = 0$ のすべての共通零点であるから連立方程式

$$\begin{cases} \partial f / \partial x = 0, \\ \partial f / \partial y = 0 \end{cases}$$

の解を Gröbner Base を用いて計算すれば、 Ω は求めることが出来る。

そこで各々の閉路の中には必ず極小もしくは極大点が存在するかつその点は、閉路により S の境界点と解によって区切られている。故に Ω の各点と S の任意の境界点とを結ぶ線分と解は必ず交わる。そこでその線分上の代数関数は 1 変数代数関数に変換出来ることより Sturm の定理を適用すれば解の存在が判定できる。以上をまとめると次の定理が成り立つ。

Theorem faithful character $\chi(S, S)(f)$ は ∂S 上の点 (a, b) に対して

$$\chi(S, S)(f) = \begin{cases} 1 & (a, b) \text{ と } (x_k, y_k) \text{ を結ぶ線分上に } f \text{ の解が存在する。} \\ 0 & (a, b) \text{ と全ての } (x_k, y_k) \text{ } k = 1, \dots, m \text{ を結ぶ線分上に } f \text{ の解が存在しない。} \end{cases}$$

で与えられる。

28.5 結論

これまでの結果を総合すると

- 2 変数多項式の描画は全ての場合にアルゴリズムは確立された。
- アルゴリズムの計算量は、多項式オーダーである。

●3 変数多項式の描画は boundary character のアルゴリズムが確立した。

今後 2 変数多項式の描画は計算量を如何に減らすかが問題である。

■他のアルゴリズムとの比較

現在数式処理システムには、いろいろな描画機能が盛り込まれている。例えば mathematica には、領域を線分により分割しその線分上の解を数値計算により求め線分間の解を逐次結んで描く方法がある。このアルゴリズムの場合線分方向にのびた解が存在した場合描画出来ない。また孤立特異点は理論上発見することが不可能である。

■計算量

計算量に関しては、このアルゴリズムは Gröbner Base の計算は描画 1 回につき高々 2 回計算すれば良くその解が 0 次元イデアルであることが一般論より明らかであるので Risa/Asir にインプリメントされている ifplot の precie モードと比較してもほとんど時間的に差はない。

参考文献

- [1] 近藤、三好、齋藤, 陰関数描画に関する一つの試み, 数理解析研究所講究録, vol.920, pp 168-174, Aug 1995.