

## 1 緒言

グラフの描画とは、関数の解となる点を描画することである。数学的には点は大きさを持たないが、表示される点は大きさを持つ領域である。その領域内に解が存在するのであれば、数学的に正しいと考えても問題はない。初期の描画方法は出発点から曲線上の次の点を求めて解曲線を追跡していく方法である。しかし、この方法は正しいグラフを表示するとは限らない。特に陰関数を正確に描画することが難しい。そこで、先行研究<sup>[1]</sup>で陰関数の正確な描画に関する方法が研究なされ、Risa/Asir に実装されている。本研究では Risa/Asir での計算結果の正確さを保ったまま画像に出力する関数を開発する。

## 2 研究のアプローチ

グラフを正しく描画する方法として、グラフ描画領域を Cell (四角形) に分割し、描画しない Cell を消去することでグラフを描画する Cell 型アルゴリズムがある。消去の判定基準の精度によって種類があり、本研究では Signature Character と Precise Character を用いる。

Signature Character とは、Cell の端点の関数値の符号が等しい場合、不適切な Cell と判定する方法である。利点は、符号判定だけなので計算が軽い、関数が代数関数でない場合も利用可能な点である。しかし、Cell の一辺に偶数回、解曲線が交わる場合や、Cell 内部に完全に含まれてしまう解曲線・孤立点は描画できない。

Precise Character とは、Signature Character の問題点である Cell の一辺に偶数回解曲線が交わる場合を解決したアルゴリズムである。スツルムの定理を用いて Cell の境界線上に解が存在するかを判定し、4 辺とも存在しない場合は不適切な Cell と判定する方法である。

スツルムの定理とは方程式  $F(x, y) = 0$ ,  $y = 0$ , のとき  $F(x, 0)$  と  $F'(x, 0)$  により、ユークリッドの互除法を用いてスツルム列  $f_n(x)$  を求める。

$$\begin{cases} f_0(x) = F(x, 0) \\ f_1(x) = F'(x, 0) \\ f_n(x) = -\{f_{n-2}(x) - f_{n-1}(x)q_{n-1}(x)\} & (n = 2, 3, \dots, m) \\ f_{m+1}(x) = 0 & q_{n-1}(x) \text{ は } \frac{f_{n-2}(x)}{f_{n-1}(x)} \text{ の商} \end{cases}$$

上式で求められたスツルム列  $f_n(x)$  に値を代入したときの符号変化の回数を  $V(x)$  とする。区間  $(a, b]$  内にある  $y = 0$  上の解の個数は  $V(a) - V(b)$  で与えられる。

Precise Character の利点は、解が存在するとした Cell には確実に解が存在することが保証されている点である。問題点はスツルムの定理を利用しているため、描画適用関数が有利係数代数方程式に限られてしまう点と、Cell 内部に完全に含まれてしまう解曲線・孤立点は描画できない点である。

これらのアルゴリズムを Risa/Asir を用いて用いて計算した結果を PostScript に出力する関数を開発する。

## 3 結果

作成した関数での描画例として以下の関数、表示領域は  $-1.5 \leq x, y \leq 1.5$ , Cell サイズ  $2^{-5}$  とする。

図 1, 2 は Signature Character での描画、図 3, 4 は Precise Character での描画である。

$$\text{Lemniscate} = x^2 - y^2 - (x^2 + y^2)^2 = 0$$

$$\begin{aligned} \text{Heart} = & \frac{93392896}{15625}x^6 + \left(\frac{94359552}{625}y^2 + \frac{91521024}{625}y - \frac{249088}{125}\right)x^4 \\ & + \left(\frac{1032192}{25}y^4 - 36864y^3 - \frac{7732224}{25}y^2 - 207360y + \frac{770048}{25}\right)x^2 + 65536y^6 \\ & + 49152y^5 - 135168y^4 - 72704y^3 + 101376y^2 + 27648y - 27648 = 0 \end{aligned}$$

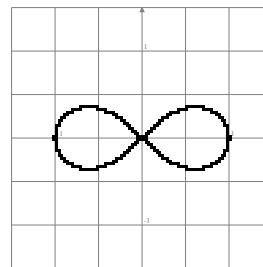


図 1 Lemniscate

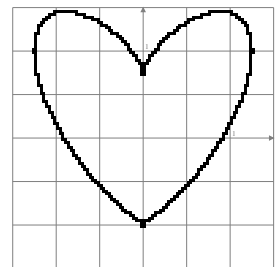


図 2 Heart

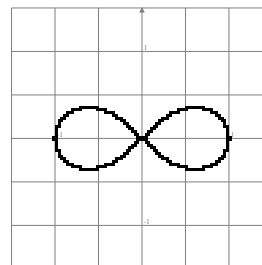


図 3 Lemniscate

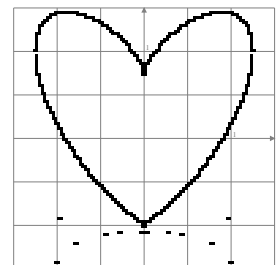


図 4 Heart

Signature Character では描画される Cell が、Precise Character で描画することができない部分がある。原因として、スツルムの定理の結果が負数となる箇所があり、計算時に少数へと近似を行っていたため誤差が生じたと考えられる。また、 $y = \frac{1}{x}$ ,  $x = 0$  等の計算は通常求めることができないため、処理が止まってしまうことがある。

## 4 今後の発展

今後の発展として、Precise Character で描画されない Cell の対策をする。描画できない関数の場合に描画計算処理を行わないようにする。また、より良い描画方法の導入を検討している。

## 参考文献

- [1] 近藤祐史, 齋藤友克. “数式処理における関数零点の描画”. 数式処理, Vol.12, No.1, pp.33-46, 2005.
- [2] Adobe Systems. 『PostScript® リファレンスマニュアル第3版』(桑沢清志 訳). アスキー. 2001.