数理解析研究所講究録
第 1909 巻 2014 年 217-222

pict2e を用いた Maxima 版 KETpic の再実装について

呉工業高等専門学校・自然科学系分野　深澤　謙次
Kenji Fukazawa
Department of Natural Sciences
Kure National College of Technology

東邦大学・理学部　高遠　節夫
Setsuo Takato
Faculty of Science
Toho University

1　はじめに

数学や物理学の研究者や教育者の中には，論文の作成に LaTeX を用いる者が多くいるが，教材の作成となると LaTeX ではなく， Microsoft Word などのワープロを使用する者も少なくない．その理由の 1 つは， LaTeX が図を扱うのが得意ではないことが考えられる．

教材には，きれいで正確な図が不可欠である．言葉や数式で説明してもなかなかわからないことが，図を 1 つ見せるだけで理解できることもある．したがって， LaTeX 文書にきれいで正確な図を簡単に入れられるようにならない限り，教材作成に LaTeX を使うようにはならないと思われる．

TeX 文書にきれいで正確な図を挿入するためのツールとして開発されたものの 1 つに KETpic がある [1]． KETpic は数学の配布用印刷教材の作成のために開発が始められ，数式処理システム（以下，CAS）上で動作するパッケージとして提供されている．現在の開発は主に Scilab 上で行われている．

KETpic では TeX 文書用の挿図を作成するために， Tpic を利用する．Tpic とは TeX 用に開発された図形プリプロセッサ及びそれが出力する special コマンドセットの名称である． Tpic を用いて TeX 文書に図を挿入するには，図を描くための一連の Tpic のコマンドの並びをファイルに書き込み，そのファイルを \input 文を用いて TeX のマスターソースファイルに読み込めばよい． KETpic はこの Tpic のソースファイルを作成するための CAS 上で動作するプログラム群として実装されている． KETpic を用いることで，ユーザーは Tpic のコマンドを知らなくても Tpic を利用した図が作成でき，この結果として， KETpic には以下のような特徴が得られている．

- TeX との親和性が良い（図の中に本文と同じ書体で数式が書ける）．
- 形と大きさに関して正確な図が描ける．

- 図の中に様々な装飾が付けられる．
- 豊かな表現力を持ったモノクロ線画が描ける．
- 修正が容易である．

KETpicを用いて挿図を作成する手順を模式的に図示すると，図1のようになる．ユーザーは CAS 上で KETpicのコマンドを使って図を描くための一連のコマンドの並びを書き， Tpic ファイルを作成する．このファイルを LATEX ソースファイルに読み込みコンパイルすると，挿図入りの dvi ファイルが得られる．図を修正したい場合は， CAS 上に戻り KETpicのコマンドを修正後，同じことを繰り返せばよい．コマンドリファレンスなどは以下のサイトから自由にダウンロードできる．

`https://ketpic.com`

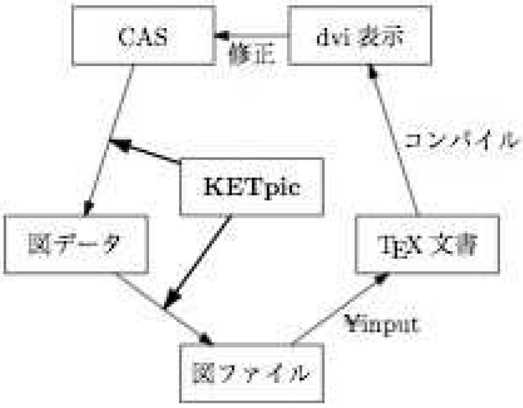

図 1: KETpicによる作図手順

KETpicでは図の中に様々な装飾を付けるために曲線の交点などの補助データを内部で計算している．例えば，曲線で囲まれた領域にハッチングを付けるためにはそれらの曲線と直線との交点の座標を求めなければならない．そのために KETpicではこれらの曲線の基本データ（プロットデータ）を作成し，基本データを基にして交点を見つけている．以上をまとめると以下のようになる．

(a) 基本データ（プロットデータ）を作成する

(b) 基本データを基に交点などの補助データを計算する

(c) 補助データを基にハッチングなどの2次データを作成する

(d) 以上のデータを基に，図の tpic (or pict2e) ファイルを作成する

(e) アクセサリなどを追加する

KETpicの Maxima への移植は以前にもなされているが，空間曲面の描画機能の実装で行き詰まってしまい，移植は終了していない．その理由の1つは，補助データを直接計算することと関係している．一般に，基本データを基にして交点などの補助データを直接計算するには以下のような問題がある．

- 実用的な時間内で計算を終了するためには，基本データの量を制限しなければならない．
- 他の CAS への移植に手間が掛かる

一方， Maxima などの CAS には標準で様々な便利な関数が用意されており，利用することができる．例えば， Maxima で用意されている関数には以下のものがある．

`solve`	$solve([< eqn_1 >, \ldots, < eqn_n >], [< x_1 >, \ldots, < x_n >])$ 方程式の解を求める
`mnewton`	$mnewton(< FuncList >, < VarList >, < GuessList >)$ Newton method で解を求める
`cspline`	$cspline(< points >, < option1 >, < option2 >, \ldots)$ 3次スプライン法による多項式補間を計算する

そこで，本論文では Maxima で用意されている関数を利用して補助データを計算する方法について議論する．ここでは平面図形を扱うこととする．なお， pdftex は tpic 拡張機能はサポートされていないため，この再実装では tpic の代わりに pict2e を用いることとする．

2 曲線の交点(接点) の求め方

2.1 陽関数で表される平面上の曲線

2つの関数 $f(x), g(x)$ の交点(接点) の求め方の概要は以下の通りである．

(a) 2つの関数 $f(x), g(x)$ の差の関数 $h(x) = f(x) - g(x)$ を定義する

(b) $h(x)$ が単調な関数になるような小区間に分割する

(c) $h(x)$ のゼロ点を計算する

(b) の単調な関数になるような小区間に分割する方法は，以下のようになる．

(1) $h'(x)$ を計算する (Maxima で用意されている **diff** 関数を利用)

(2) 区間を小区間に等分割する

(3) 関数 $h'(x)$ の符号が変わる小区間を選び出す

(4) 小区間の幅がある値 (default 0.1) 以下になるまで (2) - (3) を繰り返す

(5) 選んだ小区間での $h'(x)$ のゼロ点を線形近似で決める

(6) (5) で求めた点での関数 $h(x)$ の値の絶対値が

- 小さければ，接点の候補
- それ以外は $h(x)$ の区間の分割点

とする

(7) (6) で求めた分割点で分割して小区間を決める

(c) の $h(x)$ のゼロ点の計算は，(b) で決めた各小区間で $h(x)$ のゼロ点の近似 (候補) を求め，ゼロ点の近似値と接点の候補を初期値として newton 法（ Maxima で用意されている **mnewton** 関数を利用） で正確な値を求める．

以上のアルゴリズムを用いて，交点(接点) を求める関数 $intersect\_contact\_points$ を定義し，具体的な例として関数

$$f(x) = sin(x), \qquad g(x) = \frac{1}{2}\left(x + \sqrt{3} - \frac{\pi}{3}\right)$$

を定義し，[-5, 5] の範囲でこれらの関数に適用すると結果は

$$x_1 = -2.246005589297974, \qquad x_2 = 1.047197544757387$$

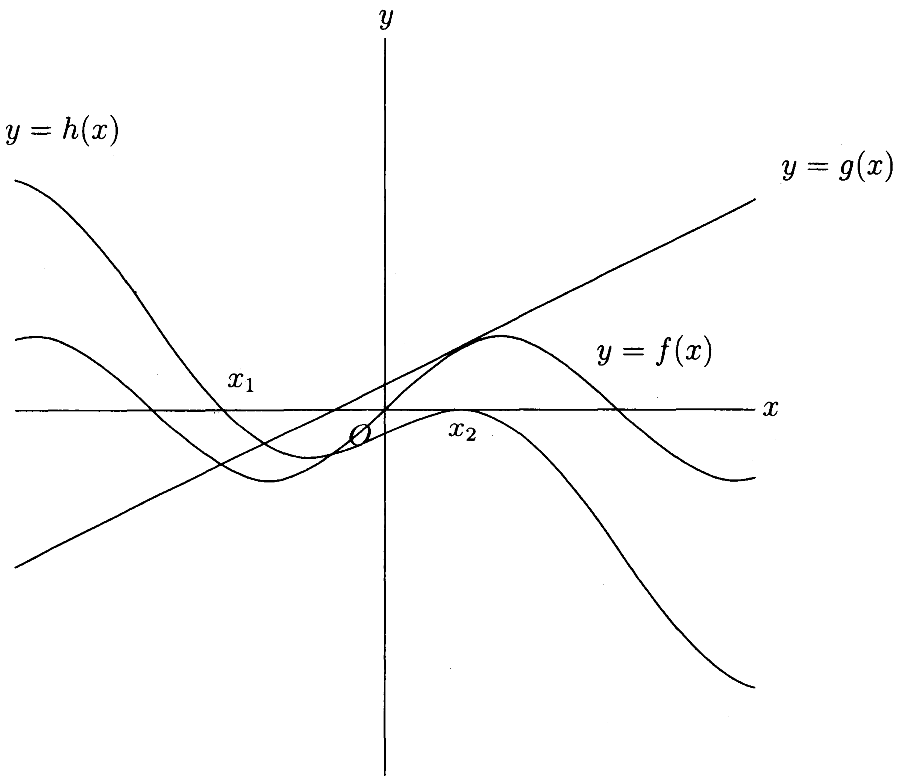

図 2: pict2e によるグラフの描画

となり，$h(x_1), h(x_2)$ の値が

$$h(x_1) = -4.4408920985006262 \times 10^{-16}, \qquad h(x_2) = 0.0$$

となることより x_1, x_2 は $f(x), g(x)$ の交点または接点であることがわかる（ 実際には，x_1 は交点であり， x_2 は接点である）．

Maxima 版のこの実装による精度として，定数 Δ の値を変えて関数 $h(x) - \Delta$ の接点を上で定義した関数 *intersect_contact_points* を用いて接点が求まるかどうか調べると，結果は以下のようになる．

Maxima 版 (再実装)		結果
$\Delta = 10^{-16}$	接点が求まらない	OK
$\Delta = 10^{-17}$	接点が求まる	NG

この結果から， Maxima 版 KETpic の再実装の精度は 10^{-16} 程度と考えられる．ただし，この精度は Maxima の設定によって変えられることは認識しておいた方がよい．

2.2 関数のグラフの描画

pict2e による関数のグラフの描画では，ベジェ曲線を利用する．本論文では，3 次のベジェ曲線を利用する場合について説明する．

3 点 A, B, C を制御点とするベジェ曲線は

$$\vec{r}(t) = \overrightarrow{OA}(1-t)^2 + 2\overrightarrow{OC}\, t(1-t) + \overrightarrow{OB}\, t^2$$

で表される．ここで C は A, B での接線の交点であるから， 2 点 A, B での接線ベクトルがわかれば，点 C が求められる．したがって，ベジェ曲線を利用して関数のグラフを描画するには，適当な小区間に分割し，各小区間の端点を A, B として小区間の部分曲線をベジェ曲線として描けばよい．

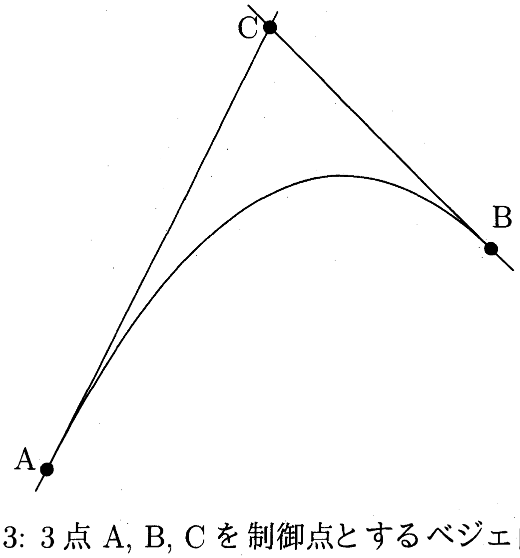

図 3: 3 点 A, B, C を制御点とするベジェ曲線

2.3 陰関数で表される平面上の曲線

2つの陰関数 $\vec{r_1}(t), \vec{r_2}(s)$ の交点(接点) の求め方の概要は以下の通りである.

(1) 2つの陰関数 $\vec{r_1}(t), \vec{r_2}(s)$ の差の関数 $\vec{r}(t,s) = \vec{r_1}(t) - \vec{r_2}(s)$ を定義する

(2) $\vec{r}(t,s)$ が単調な関数になるように t, s のそれぞれについて小区間に分割する

(3) すべての2組の小区間の組み合わせについて, 部分曲線が交わる可能性があるものを以下のルールに従って選ぶ

小区間の端点をベジェ曲線の制御点 A, B として3つ目の制御点 C を計算し, △ABC を作る. 2組の小区間に対応する2つの三角形が交わるか一方が他方を含む場合に, 部分曲線が交わる可能性があるとする.

(4) (3) で選ばれた各小区間の組み合わせについて, 小区間の中間の値を初期値として newton 法(Maxima で用意されている **mnewton** 関数を利用) で正確な値を求める.

(3) の部分曲線が交わる可能性がある小区間の組み合わせを選ぶ理由は, 全く交わらない2つの部分曲線について Maxima で用意されている **mnewton** 関数を使うとエラーになりプログラムが途中で終了してしまうからである. この手順に従って計算することによって, 2つの陰関数 $\vec{r_1}(t), \vec{r_2}(s)$ の交点(接点) を正しく求めることができる.

3 まとめと今後の課題

本論文では Maxima で用意されている関数を利用して補助データを計算する方法について議論し, 数式で表すことができる平面上の曲線については交点(接点) は精度良く求められることがわかった. この方法の利点は, 計算が比較的早いことと他の CAS への移植が比較的容易であることであるが, 現状では数式で表せない場合については(対応可能と予想してはいるが) 対応できていない.

今後の最も大きな課題は, 空間図形(空間曲線・ 曲面) への対応である. 空間曲線については, 小区間に分割した後, 各部分曲線を 3平面($x-y$ 平面, $y-z$ 平面, $z-x$ 平面) へ射影して 3平面上で交わる可能性のある部分曲線に対応する小区間の組み合わせについて, 交点(接点) を Maxima で用意されている **mnewton** 関数を使って求めればよいと思われる. 空間曲面については, spline 関数を利用することで, 例えば稜線が計算できる可能性があるが, 今後の検討課題である.

参考文献

[1] *CAS*TEX 応用研究会(編),「 KETpic で楽々 TEX グラフ」, イーテキスト研究所, 2011.