

K_ETpic による作図プログラミング書法について

木更津高専 山下 哲

工学院大学 北原 清志

長野高専 前田 善文

群馬高専 碓氷 久

明治大学 阿原 一志

東邦大学 高遠 節夫

Satoshi Yamashita

Kiyoshi Kitahara

Kisarazu National College of
Technology

Kougakuin University

Yoshihumi Maeda

Hisashi Usui

Nagano National College of
Technology

Gunma National College of
Technology

Kazushi Ahara

Setsuo Takato

Meiji University

Toho University

1 はじめに

K_ETpic は、T_EX で挿図教材を作成するために開発された数式処理システム (Computer Algebra System, 以下 CAS という) のマクロパッケージである。2006 年に有償の CAS である Maple 版から開発が始まり、2D 描画だけでなく 3D 描画機能も装備し、2010 年に無償の CAS である Scilab 版でほとんどの機能を完成させた [1]。2011 年以降は、作表機能、T_EX マクロ作成機能、ページレイアウト機能が開発され、挿図教材作成に関して T_EX 総合支援システムとなっている。K_ETpic が使用できる CAS には、Maple, Scilab 以外にも、有償の Mathematica, Matlab や無償の Maxima, R (統計処理システム) がある。

K_ETpic による挿図教材作成の方法は以前から紹介しており [2]、最近では K_ETpic ユーザーによるオリジナル挿図教材もいろいろ作成されている。ユーザーが作成した挿図作成用 CAS 実行ファイルを手に入れることで、他のユーザーも同じ挿図教材を利用することが可能である。ところが、その実行ファイルに記された K_ETpic プログラム書式がユーザー独自のものであるため、他のユーザーが改変しづらいという現状がある。そこで、K_ETpic によるプログラミング書法を確立するために、2012 年から書法の要件を調査してきた。本稿では、これまでの調査結果を報告するとともに、その要件のいくつかを提案する。本研究は、科学研究費補助金基盤 (C) 「挿図教材のための作図プログラミング書法の確立と教材作成支援ポータルシステムの構築」 (課題番号 25350370) の助成を受けている。

2 楕円の課題について

K_ETpicによる作図プログラミング書法の要件を本格的に調査する前に、2012年12月にプレ調査を行った。プレ調査の課題は「Scilab版K_ETpicを用いて、図1のような楕円をかけ」で、調査対象者はK_ETpicユーザーである数学教員12名、K_ETpic初心者である数学教員8名、K_ETpicの講習会を受講した学生4名であった。課題プログラムの正答例は以下の通りである。

```
cd("c:/work/");
Ketlib=lib("kepicsciL5/");
Ketinit();
Fname='ellipse.tex';
A=2;
B=1;
Setwindow([-A-0.5,A+0.5],[-B-0.5,B+0.5]);
G1=Paramplot("[A*cos(t),B*sin(t)]","t=[0,2*%pi]");
C=sqrt(A^2-B^2);
PF1=[C,0];
PF2=[-C,0];
Windisp(G1,PF1,PF2)

Openfile(Fname,"1cm");
Drwline(G1);
Setpt(4);
Drwpt(PF1,PF2);
Htickmark(-A,"sw","-a",A,"se","a",-C,"s","-c",C,"s","c");
Vtickmark(-B,"sw","-b",B,"nw","b");
Closefile("1");
```

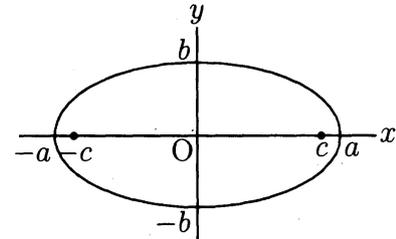


図1. 楕円の課題

このプログラムにおける本調査の観点は以下のようなになる。

(1) Scilabの数値計算機能の利用

ローカル変数を用いて、 a , b の値を的確に設定する。また、数値計算機能を用いて、焦点の座標 c の値を正確に求める。

(2) 数学の知識

媒介変数表示を用いて、頂点 $(\pm a, 0)$ 付近の曲線を滑らかに描画させる。

(3) K_ETpic コマンドの知識

K_ETpic コマンドを的確に利用している。

調査結果は以下の通りである。

A グループ (K_ETpic ユーザーである数学教員12名)

- (1) ローカル変数を使用した人が10名、数値計算機能で焦点の座標 c を求めた人が8名いた。

- (2) 媒介変数表示を用いた人は12名全員であった。
- (3) すべての $\text{K}\epsilon\text{Tpic}$ コマンドを的確に使用した人が10名いた。

Bグループ ($\text{K}\epsilon\text{Tpic}$ 初心者である数学教員8名)

- (1) ローカル変数を使用した人が1名、数値計算機能で焦点の座標 c を求めた人が1名で、同じ人だった。
- (2) 媒介変数表示を用いた人は8名全員であった。
- (3) すべての $\text{K}\epsilon\text{Tpic}$ コマンドを的確に使用した人が7名いた。

Cグループ ($\text{K}\epsilon\text{Tpic}$ 講習会を受けた学生4名)

- (1) ローカル変数を使用した人も、数値計算機能で焦点の座標 c を求めた人もいなかった。
- (2) 媒介変数表示を用いた人はいなかった。
- (3) すべての $\text{K}\epsilon\text{Tpic}$ コマンドを的確に使用した人は4名全員だった。

以上の結果から、 $\text{K}\epsilon\text{Tpic}$ による作図プログラミングについて以下のことが明らかになった。

- (1) ローカル変数の使用や数式処理による計算などプログラミングの知識が必要である。
- (2) グラフを均等に描くためには媒介変数表示を使った方がよいという数学的知識が必要である。
- (3) $\text{K}\epsilon\text{Tpic}$ コマンドを的確に利用することは難しくない。

また、エラーメッセージの表示を避けるために Fail-safe コマンドを利用していたが、 $\text{K}\epsilon\text{Tpic}$ を初期化するコマンド Ketinit 以外は使用しづらいことも明らかになった。現在では、Fail-safe コマンドは、エラーメッセージを表示させる原因となっているコマンドの内部で実行できるようにしている。

3 作図プログラミング書法と Symbolic Thinking

第1節で述べたように、挿図教材を共有するために作図プログラミング書法の統一は欠かせない。そこで、 $\text{K}\epsilon\text{Tpic}$ による作図プログラムを調査することにより、書法の要件を探究した。今回調査したプログラムは、 $\text{K}\epsilon\text{Tpic}$ 初心者で数学研究者が作成した52個のプログラムである。この結果、作図プログラムの読みやすさにとって、これまで提唱してきた Symbolic Thinking が重要であることが明らかになった [2]。Symbolic Thinking とは、数学的な作図手順に従ってコマンドを実行することで、図の全体像を明確に認識しながら質的な改良に集中できることである。本節では、そのうちのプログラムを1個取り上げて、プログラムの読みやすさと Symbolic Thinking の関連性を具体的に説明する。

次ページの図2は三角比の定義に利用する図である。図2を描画するために、 $\text{K}\epsilon\text{Tpic}$ により、次のような作図プログラムが作成された。

```

1 f1=Listplot([[0,0],[4,0]]);
2 f2=Listplot([[0,0],[4,2]]);
3 f3=Listplot([[2.6,0],[2.6,1.3]]);
4 A1=Anglemark([4,0],[0,0],[4,2]);
5 P1=Paramark([0,0],[2.6,0],[2.6,1.3],0.3);
6 B1=Bowdata([0,0],[2.6,0],1,1);
7 B2=Bowdata([2.6,1.3],[0,0],1,1);
8 B3=Bowdata([2.6,0],[2.6,1.3],2,0.8);
9 Windisp(f1,f2,f3,A1,P1,B1,B2,B3)
10 Setwindow([0,5],[-0.5,2]);
11
12 Openfile("zu5_1_i.tex","1cm");
13 Drwline(f1,f2,f3,A1,P1,B1,B2,B3);
14 Expr([1,0.3],"sw","\alpha");
15 Letter([4,0],"e","X");
16 Letter([2.6,1.3],"n","P");
17 Letter([4,2],"ne","Y");
18 Letter([0,0],"w","O");
19 Htickmark(2.6,"s","\mathrm{Q}");
20 Bowname(B1,"\mathrm{底辺}");
21 Bowname(B2,"\mathrm{斜辺}");
22 Bowname(B3,"\ \ \mathrm{高さ}");
23 Closefile("0");

```

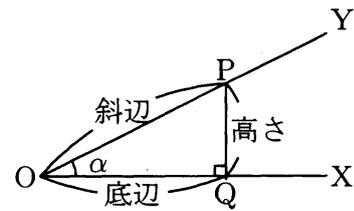


図 2. 三角比の定義

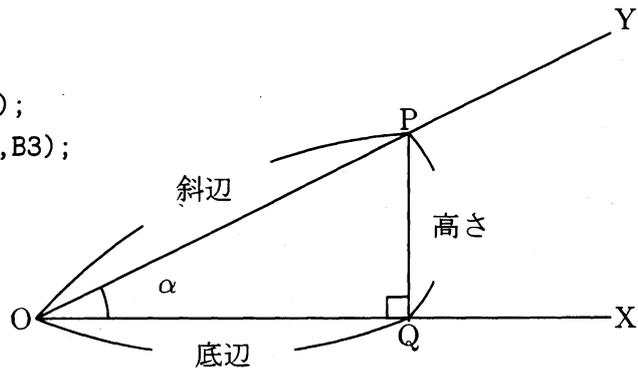


図 3. 三角比の定義の拡大図

このプログラムでsymbolicでない部分を抽出したところ、以下の6点が挙げられ、いずれもプログラムを読みづらくしている原因であることが明らかになった。

(1) コマンドの配置

10行目のSetwindowがプログラムの先頭にないため、描画範囲がわかりづらい。

(2) 適切な名前

(a) 図形の頂点がいろいろな場所で使用されているが、頂点に名前を付けていないため、どの頂点を指定しているかがわかりづらい。

(b) 1行目から3行目にかけて線分のプロットデータに名前を付けているが、名前の付け方が線分の端点を指定していないため、どの線分のデータを作成しているかわかりづらい。

(3) CASの計算機能

3, 5, 8行目で半直線OY上の点[2.6,1.3]を指定するのに、OYの方向ベクトル $\mathbf{v}_y = [1, 1/2]$ の定数倍を使用していないため、座標が煩雑になり、OY上の点を指定しているかわかりづらい。

(4) 基準点による位置の指定

14行目で角XOYの大きさ α を表示するとき、4行目のプロットデータA1における

中点の座標を使用せずに, [1,0.3] と目分量で指定したため, 図3のように拡大するとA1から離れていく.

(5) KETpic コマンドの適切な使用

19行目で点Qの名前を書き出すとき, Letterではなく, 座標軸に目盛を入れる Htickmark を使用したため, 点Qの位置において線分OXの下側に座標軸の目盛マークが飛び出している(図3参照).

以上の6点を修正すると, 以下のように読みやすいプログラムになる.

```

1 cd("c:/work/");
2 Ketlib=lib("ketpicsciL5");
3 Ketinit();
4 Fname="zu5_1_iy.tex";
5
6 Setwindow([0,4],[-0.5,2]);
7 Po=[0,0];
8 Tmax=Xmax();
9 Px=[1,0]*Tmax;
10 Vy=[1,1/2];
11 Py=Vy*Tmax;
12 Lxoy=Listplot(Px,Po,Py);
13 T0=2.6;
14 Pq=[1,0]*T0;
15 Pp=Vy*T0;
16 Lpq=Listplot(Pp,Pq);
17 Am=Anglemark(Px,Po,Py);
18 Pm=Paramark(Po,Pq,Pp,0.3);
19 Boq=Bowdata(Po,Pq,1,0.8);
20 Bpo=Bowdata(Pp,Po,1,0.8);
21 Bqp=Bowdata(Pq,Pp,3,0.5);
22 Windisp(Lxoy,Lpq,Am,Pm,Boq,Bpo,Bqp)
23
24 Openfile(Fname,"1.5cm");
25 Drwline(Lxoy,Lpq,Am,Pm);
26 Dottedline(Boq,Bpo,Bqp)
27 Pm=Bowmiddle(Am);
28 Expr(Pm,"n-0.5e2","\alpha");
29 Letter(Po,"w","0",Px,"e","X",Py,"ne","Y",Pp,"n","P",Pq,"s","Q");
30 Bowname(B1,'\mathrm{底辺}');
31 Bowname(B2,'\mathrm{斜辺}');
32 Bowname(B3,'\ \ \mathrm{高さ}');

```

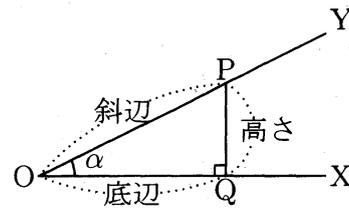


図4. 三角比の定義 (修正版)

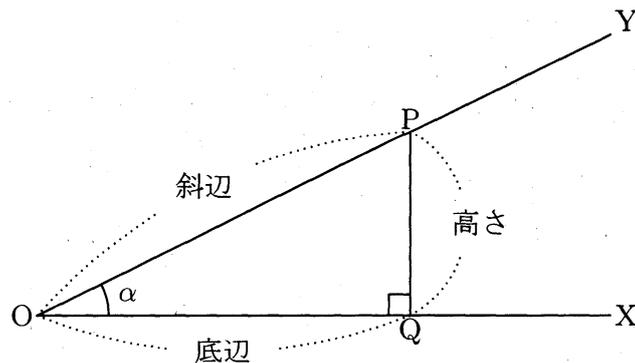


図5. 三角比の定義 (修正版) の拡大図

```
33 Closefile("0");
```

1行目から4行目までは、どの挿図作成用実行ファイルでも必ず実行すべき部分である。1行目で作業ディレクトリを指定し、2行目でK_ETpicパッケージをScilabに読み込み、3行目でK_ETpicを初期化し、4行目でこれから作成する挿図ファイル名を指定する。

6行目から22行目までは、挿図のプロットデータを作成する部分である。6行目で描画範囲を指定し、7行目は原点を指定、8行目から11行目までは方向ベクトルを用いて点X, Yを指定、12行目で折れ線XOYのプロットデータを作成する。13行目から15行目で方向ベクトルを用いて点P, Qを指定し、16行目で線分PQのプロットデータを作成する。17行目は∠XOYのマークのプロットデータ、18行目は∠OPQの直角マークのプロットデータを作成する。19行目から21行目は3辺OQ, PO, QPの弓形のプロットデータを作成し、22行目で各プロットデータの作成状況をScilabのグラフィック・ウィンドウ上で確認する。いずれの行も作図手順に従ってプロットデータを作成しているため、Symbolic Thinkingしやすいプログラムになっている。

24行目から34行目までは、挿図ファイルへの書き出し部分である。24行目で挿図ファイルを開き、25行目で実践部分、26行目で点線部分を書き出す。27行目で∠XOYのマークのプロットデータから中点の座標P_mを抽出し、28行目で文字αを中点P_mの北東方向に置く。中点の座標P_mもsymbolicに抽出しているため、プログラムが読みやすい。29行目で他の点の名前を置く。30行目から32行目で弓形に底辺、斜辺、高さの文字を置き、33行目で挿図ファイルを閉じる。

4 まとめと今後の課題

第2節でプレ調査から得られた3つの要件を紹介したが、第3節で行った作業を52個のプログラムを調べたところ、さらに詳細な要件が明らかになった。以下の9つの要件は52個のプログラム調査から得られたものであり、()内は要件が満たされていなかったプログラム数とその割合を表している。

- | | |
|-----------------------------------|----------------|
| (1) コマンドの適切な配置 | (41/52, 78.8%) |
| (2) ローカル変数やプロットデータなどの適切な名前 | (44/52, 84.6%) |
| (3) CASの計算機能の利用 | (36/52, 69.2%) |
| (4) K _E Tpicコマンドの適切な使用 | (40/52, 76.9%) |
| (5) プログラムのブロック化 | (19/52, 36.5%) |
| (6) 基準点の指定 | (24/52, 46.2%) |
| (7) リスト構造の利用 | (21/52, 40.4%) |
| (8) for文の使用 | (1/52, 1.9%) |
| (9) ローカル変数の定義 | (27/52, 51.9%) |

(1)から(4)までの要件がほとんど満たされていなかった。作成者がプログラミング初心者であるため、第2節の(1)の要件が満たされていなかったと考えられる。(5)の要件については、プログラムの雛形通りに作成したため、プログラミングの知識がなくても自然と満たされていた。(6)から(9)までの要件が満たされていない割合が少ないようであるが、これらの要件に関わるプログラム数が少ないことに起因している。例えば、(8)の要件については、for文を利用すべきプログラムは1個しかなかったため、本来ならば(1/1, 100%)とすべきであろう。残念ながら、今回の調査では要件に関するプログラム数の確認までできなかった。

最後に、今後の課題は以下の通りである。

- (1) プログラミング上級者の数学教員による作図プログラムを調査し、上記の9つの要件で十分であるか確認する。
- (2) (1)で調査した結果を基にして、作図プログラムの基本型を構築する。
- (3) オリジナル図入り教材を作成している数学教員に対して、作図に関するインタビューを行う。微分積分や線形代数を教えている大学教員に対して、図入り教材に関するアンケートを実施する。
- (4) エラーメッセージを調査し、頻出するエラーや原因がわかりづらいエラーを確認する。また、この調査を基にして、 $K_{\text{E}}\text{T}_{\text{P}}\text{i}c$ の機能を改良する。
- (5) 図入り教材作成支援ポータルサイトを立ち上げる。

参考文献

- [1] CASTeX 応用研究会, 「 $K_{\text{E}}\text{T}_{\text{P}}\text{i}c$ で楽々 $\text{T}_{\text{E}}\text{X}$ グラフ」, イーテキスト研究所, 2011年.
- [2] 山下哲, 高遠節夫, 「 $K_{\text{E}}\text{T}_{\text{P}}\text{i}c$ による教材作成と Symbolic Thinking」, 京都大学数理解析研究所講究録 1780 「数学ソフトウェアと教育」, pp.72-82, 2012年.