数理解析研究所講究録
第2067巻 2018年 132-141

KETCindy の C 呼び出し機能と曲線・曲面論の教材の作成

東邦大学・理学部　野田　健夫 (Takeo Noda)
高遠　節夫 (Setsuo Takato)
Faculty of Sciences,
Toho University

1 はじめに

高専・大学における数学教育では，平面図形・空間図形による直観的理解は非常に重要であり，正確で見やすい挿図が常に求められている．著者の一人（高遠）をはじめとする開発チームは，TEX 文書の挿図作成用のマクロパッケージである KETpic と動的幾何ソフトウェア Cinderella を連携させ，インタラクティブな作図環境 KETCindy を構築した [1, 2].

KETCindy によって作成される曲面は輪郭線からなる線画であり，シンプルで必要な情報が絞り込まれているので概念的に把握しやすい．しかし，従来のシステムでは陰線処理の複雑な計算を Scilab や R などのインタプリタ型言語で処理しているので，出力を得るのにある程度の時間を要する．

他方において，現在 KETCindy は動画入り PDF スライド教材を作ることが可能になっており [3]，動画のために一度にたくさんの図形データを作成する需要が生じ，処理速度の向上が期待されていた．

今回の研究では，KETCindy からの C コンパイラの呼び出しを実装し，それにより曲面描画にかかる時間を大幅に短縮することに成功した．本稿ではまず C 呼び出しの実装と使用について説明し，描画の高速化を活かして作成した曲面の動画を含む PDF スライド教材の例を紹介する．

2 C 呼び出し

KETpic は，TEX 描画用の Tpic special コードを生成する Scilab または R のマクロパッケージである．なお，海外では pdfLATEX が広く用いられていることに鑑み，pict2e コードの出力にも対応している．KETCindy は，Cinderella のプログラム言語である CindyScript のマクロパッケージで，そのコマンドを記述して

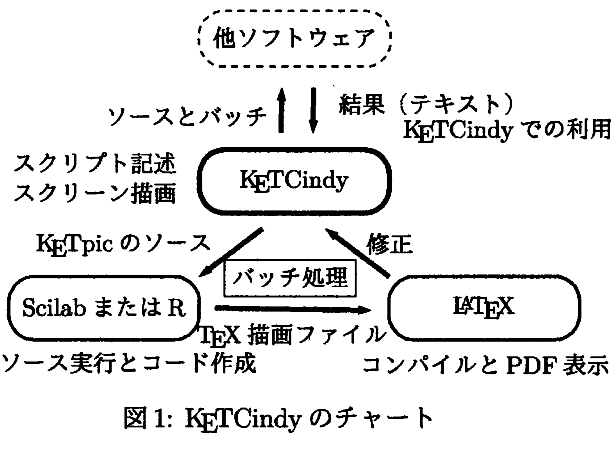

図 1: KETCindy のチャート

いくことでスクリーンに描画することができる．さらに，バッチ処理のボタンを押すことによりTEXの描画ファイルを作成する．

最近は，同様なバッチファイルを作成・実行することでMaxima, Scilab, RなどのソフトウェアをKETCindyから利用するコマンドを順次追加している．例えば，

```
Mxfun("1","integrate",["sin(x)^2","x"],[""]);
```

を実行すると，以下の手順でMaximaを実行して，結果がCinderellaに戻される．

(1) `integrate(sin(x)^2,x)` を主コードとするMaximaのソースファイルを作成

(2) Maximaを呼び出して，結果をテキストファイルに書き出すバッチ処理を実行

(3) Cinderellaに戻って，ファイルから必要な文字列を抽出して変数 `mx1` に代入

```
mx1="(x-sin(2*x)/2)/2"
```

空間図形の描画については，空間曲線や多面体などの描画の場合，KETCindyで直接データを生成するが，曲面描画の場合は，陰線処理の計算が複雑であるため，従来はScilabのKETpicパッケージをそのまま用いて，Maximaと同様な呼び出しを行っていた．しかし，Scilabはインタプリタ型の言語であるが故に，多くの計算時間を要する．例えば，Scilab5.5.2によって図2のグラフを作成するには，著者（高遠）のMacbook Air 2015年モデルでは100秒程度の時間がかかっていた．実は，2007年にはKETpicの開発を始め，程なくして曲面描画のコマンドを実装したのであるが，当時から描画の高速化が課題であった．KETpicでは関数データを次のような文字列で与える．

```
"z=3*(1-(2*u^2+v^2)*exp(-u^2-v^2))"
```

それをScilabの文字列評価関数execstrで関数値を求めるが，Cなどのコンパイル型言語にとっては難しい処理であった．

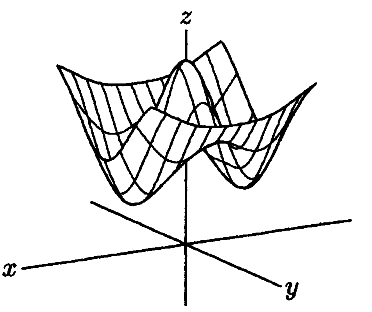

図2: $z = 3(1-(2x^2+y^2)e^{-x^2-y^2})$

最近になって，gccなどのCコンパイラが考えていた以上に速いことを知り，ヘッダファイルとメインファイルを書き出して，コンパイラに渡せばよいことに考え至った．すなわち，Maximaの呼び出しなどと同様にバッチ処理を利用すればよい．具体的には，

(1) Cindy Scriptで関数データ，Cでの実行コマンド，表示オブジェクトのリストを定義する．

```
FdC=[["double u2=pow(u,2.0),v2=pow(v,2.0)","x=u","y=v",
"z=3*(1-(2*u2+v2)*exp(-u2-v2))"],"u=[-1.5,1.5]","v=[-1.5,1.5]",""]
```

(2) ヘッダとメインファイルの書き出し，バッチ処理，画面表示のコマンドを記述して実行する．

```
Cheader(); Cmain(); kcC(); DisplayC();
```

作成されるヘッダファイルとメインファイルは次のようになる．

- ヘッダファイル

```
const double Urng[2]={-1.5,1.5}, Vrng[2]={-1.5,1.5};
const double XMIN=-4.500000,XMAX=3.593115;
const double THETA=1.311622,PHI=1.047198;
  //THETA:75.15, PHI:60.00;
const int DrawW=1, DrawE=1, DrawS=1, DrawN=1;
const int Mdv=50, Ndv=50;
#define DsizeLL 5000
#define DsizeL 3000
#define DsizeM 500
#define DsizeS 200
const double Eps=0.00001, Eps1=0.02, Eps2=0.1, Eps3= 1;
void surffun(double u, double v, double pt[3]){
  double u2=pow(u,2.0),v2=pow(v,2.0);
  pt[0]=u;
  pt[1]=v;
  pt[2]=3*(1-(2*u2+v2)*exp(-u2-v2));
}
```

- メインファイル

```
#include <stdio.h>
#include <math.h>
#include "mantheader.h"
#include "ketcommon.h"
#include "surflibhead.h"
#include "surflib.h"
double cutfun1(double u, double v){
  double p[3],val;
  surffun(u,v,p);
  val=1;
  return val;
}
double cutfun(int ch, double u, double v){
  double val;
  if(ch==1){val=cutfun1(u,v);}
  return val;
}
int main(void){
  double out[DsizeL][3], data[DsizeLL][3];
  int i, j, nall, din[DsizeS][2];
  double sfbd[DsizeL][3];
  sfbdparadata(sfbd);
```

```
    output3h("sfbd3d","sfbdh3d","mantsfbd.txt",sfbd);
    dataindexd3(3,sfbd,din);
    Surf[0][0]=0;
    appendd3(2,din[1][0],din[1][1],sfbd,Surf);
     (中略)
    return 0;
}
```

実際に実行すると，1度目はCのライブラリを読み込むため15秒程度かかるが，図を少し変化させて続けて行う場合は，ほんの数秒である．

3 曲線・曲面論の教材の例

この節では，高速化を活かして作成した，曲面の動画を含むPDFスライド教材の例を紹介する．PDF動画には自動で再生されるムービー型と，手動でページ操作をしてコマ送りを実現するパラパラ型がある[3]．手動操作の方が途中で止めたり逆に戻したり再生の自由度が高いこと，またPDFビューアによらず利用できることから，今回はパラパラ型を用いた．

題材には初等微分幾何の話題からガウス曲率の定義を選んだ．3次元空間内の曲面は視覚化可能な図形であり，適切な図解を提供すればガウス曲率が何を測ろうとしているのか学習者は直観的に理解できると信じるからである．曲線・曲面の微分幾何の詳細については[5]などを参照せよ．

3.1 曲線の曲率

まず，ガウス曲率の定義のための準備段階として，平面曲線$C(t)=(x(t),\ y(t))$の曲率について説明する．曲線C上に任意に1点Pをとり，PにおいてCに2次の接触をする円Γが一意に定まる．これを曲率円とよび，その半径Rを曲率半径と定める．ただし曲率円が退化して直線になるとき，曲率半径は∞であるものとする．

いま，PにおけるCの単位接ベクトルe_1をパラメータtが増加する向きにとり，e_1を正の方向に$\frac{\pi}{2}$回転して得られる単位法ベクトルをe_2とおく．このとき，曲線Cの点Pにおける曲率κは次で定めることができる．

(1) $|\kappa|=\frac{1}{R}$

(2) 曲率円がCに対しe_2側にあるときκの符号は正，逆側にあるとき負とする．

(3) $R=\infty$のとき$\kappa=0$

図3はKETCindyで作図している．曲線Cの成分$x(t), y(t)$を4次多項式におくと，与えられた5点を通る曲線Cは一意に決まる．Cinderella上で作図した5つの幾何点$P_1,\ldots,P_5$を通る曲線としてCを定めれば，マウスで幾何点を動かして曲線の形と曲率

関数 κ のグラフの形を見ながら，説明に都合の良いように変形して選ぶことができる．このように一般性のある曲線を扱いながら数学的に正確な図を作れるのは KETCindy の利点である．

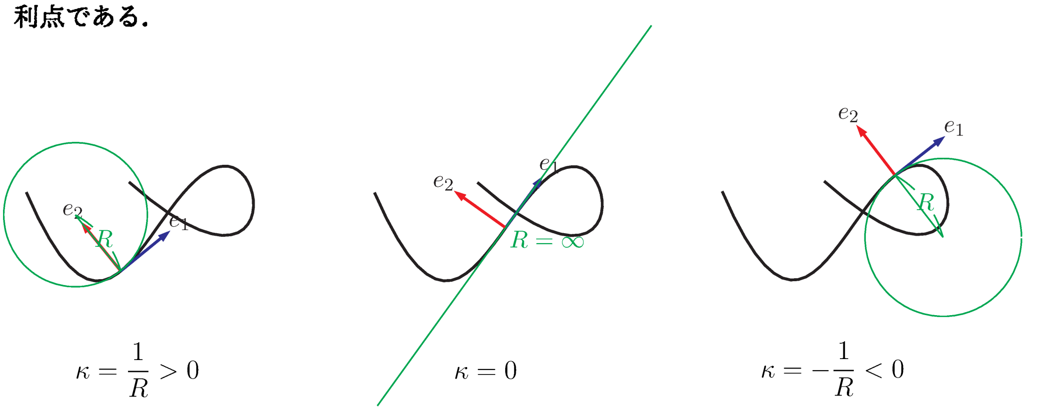

図 3: 曲率の符号

さらに，Cinderella を用いて作図したコンテンツは，CindyJS により描画手続き込みで HTML 形式にフォーマットし，ブラウザ上で操作可能にできる [4]．これにより学習者たちが自ら制御点を調整して曲線の形を自由に変え，曲線の形状が曲率関数にどう反映されるかを体感して学ぶ教材も得られる．

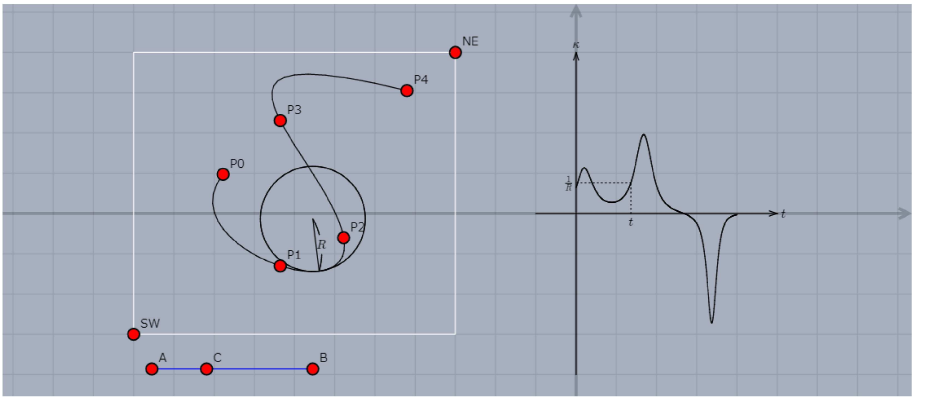

図 4: CindyJS による動的コンテンツ

3.2 曲面のガウス曲率

次に，3 次元空間内の曲面 S を考える．S の各点 P におけるガウス曲率は，次の段階を経て定義される：

(1) 断面曲線の曲率として法曲率を定義

(2) 法曲率の最大値・最小値として主曲率を定義

(3) 主曲率の積としてガウス曲率を定義

以下，順を追って説明しよう．

3.2.1　法曲率

曲面 S 上の点 P を固定し，P における S の単位接ベクトル e_1 と単位法ベクトル e_2 をとる．ここで，e_2 は曲面 S の向きづけを固定しておけば一意に決まる．他方，e_1 は接点 P を始点とし P における接平面上大きさ 1 のベクトルなので円周分の自由度があり，適当な起点をとれば $\theta \in [0, 2\pi]$ でその向きを表すことができる．

すると，各 θ に対して P を通り $e_1(\theta)$, e_2 によって張られる平面 Π_θ が定まる．平面 Π_θ 上曲面 S との共通部分は P を通り e_1, e_2 をそれぞれ単位接ベクトル，単位法ベクトルとする曲線なので，平面曲線としての曲率が定まる．これを $\kappa_\nu(\theta)$ で表し，S の P における θ 方向の法曲率という．

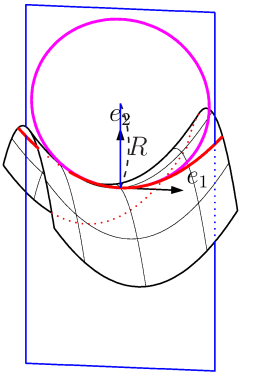

図 5: 法曲率

3.2.2　主曲率

定義より法曲率 $\kappa_\nu(\theta)$ は単位接ベクトル e_1 と同じく円周上の関数になる．PDF スライド教材ではこの 1 周分を 24 コマのパラパラ型動画で表した．

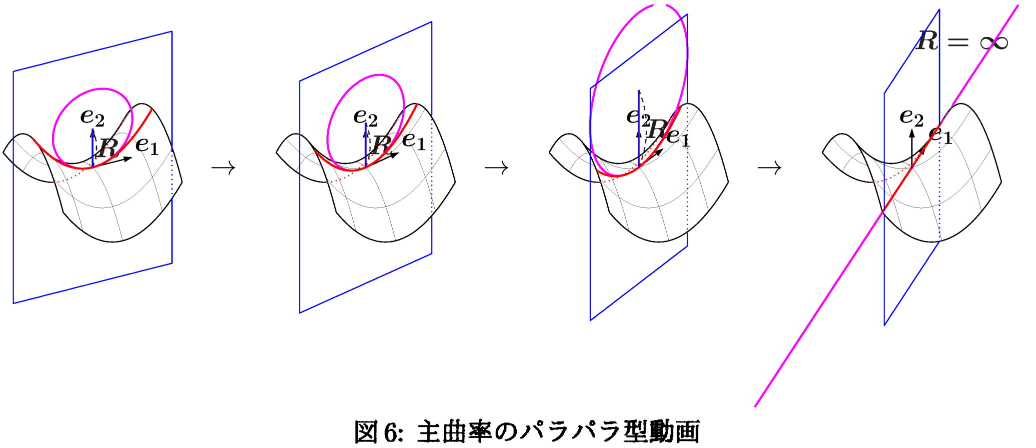

図 6: 主曲率のパラパラ型動画

円周はコンパクトなので，$\kappa_\nu(\theta)$ は必ず最大値と最小値をとる．これを主曲率と定義し，κ_1, κ_2 で表す．次は主曲率を説明するスライドである．

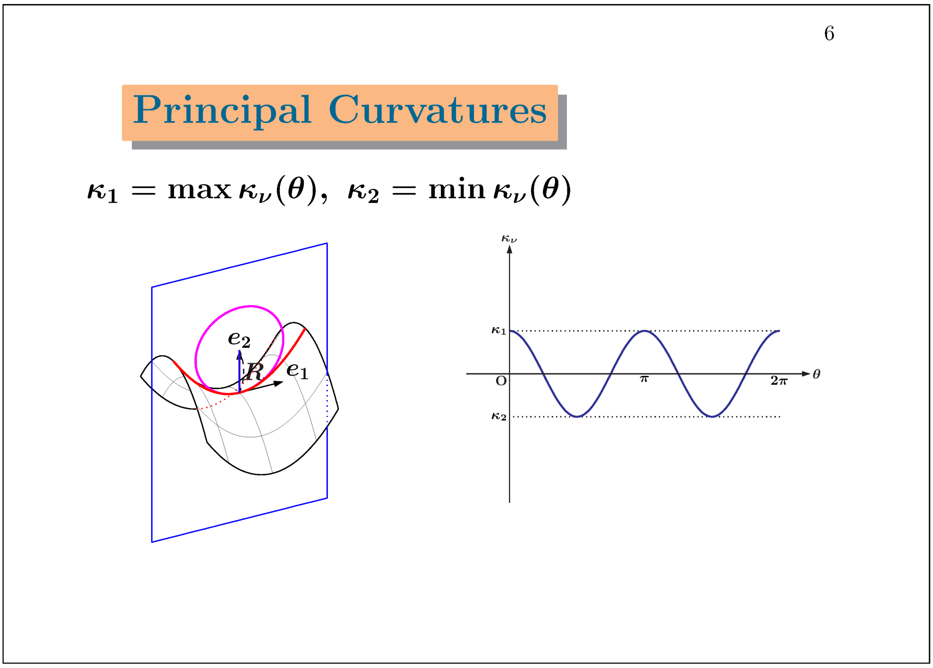

図 7: 主曲率の定義

3.2.3 ガウス曲率

主曲率 κ_1, κ_2 に対し，その積 $K = \kappa_1\kappa_2$ を曲面 S の点 P における**ガウス曲率**という．上に挙げた図の例のように鞍点においては $\kappa_1 > 0$, $\kappa < 0$ なので $K < 0$ が分かる．

また，図 8 のように凸な点においては $K > 0$, 柱面の点においては $K = 0$ になることも図を見ればすぐわかるであろう．

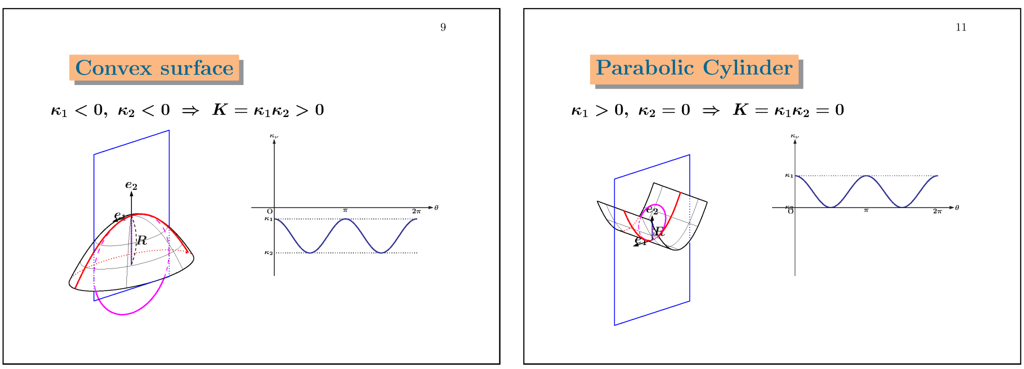

図 8: ガウス曲率

3.3 パラパラ型PDF動画の作成

最後に，パラパラ型動画の作成方法について簡単に触れておこう．今回の教材を作るために KETCindy の CindyScript として以下のリストを書いた．ただし，簡単のため曲面と法曲率をとる平面以外の図形は省略してある：

```
FheadCorg="nc";

// 曲面の定義
FdC=[
  ["x=u","y=v","z=0.5*pow(u,2.0)-0.5*pow(v,2.0)+1.0"],
  "u=[-1.5,1.5]","v=[-1.5,1.5]","nwes",
  [100,100],[5000,3000,500],[0.01,0.1]
];

Ctr=1;
mf(s):=(
  FheadC=FheadCorg+text(Ctr);
  Changestyle3d(Datalist3d(),["nodisp"]);

  // 曲面の接ベクトル e1 と法ベクトル e2 で張られる平面
  Defvar("e1=[cos(s),sin(s),0]");  Defvar("e2=[0,0,1]");
  p1=[0,0,1]+2*e1+3*e2;  p2=[0,0,1]-2*e1+3*e2;
  p3=[0,0,1]+2*e1-3*e2;  p4=[0,0,1]-2*e1-3*e2;
  Spaceline("1",[p1,p2,p4,p3,p1],["nodisp"]);
  pln="("+text(sin(s))+")*x+("+text(-cos(s))+")*y";

  MainC=[
    "writesfbd",["sfbd"],
    "writewire",["wire",4,[[-1.0,0,1.0],[-1.0,0,1.0]]],
    "writecut",["sfcut",pln],  "writesc",["sl3d1"]  ];
  DispC=[
    "sfbd",["dr",black],  "sfbdh",["do",red],
    "wire",["dr,0.3",black],  "sfcut",["dr,2",red],
    "sfcuth",["do,2",red],  "sl3d1",["dr,1",blue],
    "sl3d1h",["do",blue]  ];

  Cheader();
  Cmain();
  kcC();
  DisplayC();

  Ctr=Ctr+1;
);
Setpara("normal_curvature","mf(s)","s=[0,2*pi]",["m","Div=24"]);
```

mf(s) として定義した部分がC呼び出しによる曲面描画に相当し．最後の Setpara がパラパラ型動画作成を行う．

Cinderella の画面上の ParaF ボタンを押すことでパラパラ用ファイルの作成が実行さ

れ，ketwork/fig フォルダ下の normal_curvature フォルダを作成し，区間 $[0, 2\pi]$ を 24 等分して $s = 0, \frac{2\pi}{24}, \ldots, 2\pi$ に対応する 25 個の tex ファイルを出力する．

PDF スライド作成機能も KETCindy には用意されている．Beamer 風の簡易的なコードをテキストファイルで書いておくと，Cinderella の Slide ボタンを押すことにより対応する tex ファイルを自動生成しコンパイルする．たとえば，上述のパラパラ型動画に含まれる 25 個のファイルから PDF スライドを作るには以下のようなファイルを用意すればよい：

```
new::Normal Curvature//
%repeat=,para=normal_curvature:{0}:s{25}{10}:input:0.8//
```

スライド作成に関するより詳しい説明は，ketsample\samples\s07slides フォルダ内の howtouseslideJ に書かれている．

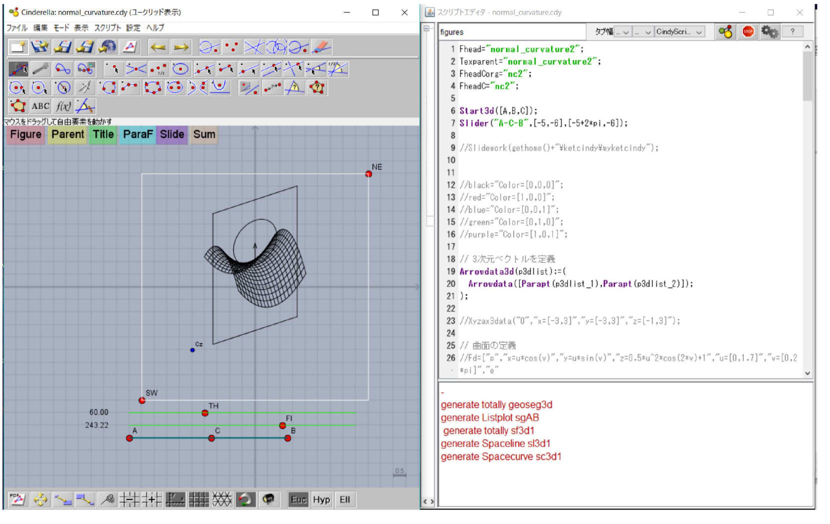

図 9: Cinderella の画面

4 まとめと今後の課題

今回の曲面描画の高速化により，PDF 動画スライドの表現の幅が確実に広がった．本稿ではガウス曲率の教材の例を挙げたが，同様に多変数関数の微積分や，ベクトル解析においても曲面と動画表現を組み合わせて分かりやすい教材ができるようになるだろう．

また，出力が PDF であることからファイルサイズが小さく，ビューアの他に特殊なアプリケーションを必要としないことから，学習者はスマートホンなど携帯端末で容易にスライドを自ら操作することができるのは利点であろう．

今後は描画の精度を上げていくとともに，実際の教育で印刷教材，スライド教材，動的幾何ソフトの操作などを組み合わせることにより，高い教育効果の実現を模索していきたい．

謝辞

本研究は，京都大学数理解析研究所共同事業「数学ソフトウェアとその効果的教育利用に関する研究」による成果である．

参考文献

[1] 高遠節夫，KeTCindy 開発チーム，KeTCindy の開発について，数理解析研究所講究録 1978, pp. 173–182, 2015

[2] Takato S.,What is and How to Use KeTCindy—Linkage Between Dynamic Geometry Software and LaTeX Graphics Capabilities— , Lecture Notes in Computer Science 9725, Springer, pp.371–279, 2016

[3] 山下哲，K$_E$TCindy による図入り PDF 教材の作成，数理解析研究所講究録 2022, pp. 59–64, 2016

[4] 金子真隆，CindyJS によるアクティブラーニングの可能性，数理解析研究所講究録 2022, pp. 48–58, 2016

[5] 小林昭七，曲線と曲面の微分幾何，裳華房，1995

[6] `https://sites.google.com/site/ketcindy`

[7] `http://www65.atwiki.jp/ketcindy`

[8] `http://www.cinderella.de`