

# 計算数学 I 最終レポート

05-190502 安達 充慶

## 1 取り組んだ問題

今回取り組んだ問題は次のようなものである。

Euclid 空間内の曲面のパラメータ表示が与えられたとき、その曲面の Gauss 曲率を求めよ。また、その曲面が定曲率であるかを判定せよ。

今回はこの問題を Python を用いて解く。SymPy を用いることにより、簡単な微分の計算は全て厳密に行うことができる。

その前にこのような問題を設定した動機について説明する。曲面の Gauss 曲率は曲面を計量込みで考えるときの基本的な量であり、比較的具体例の計算もしやすいものである。しかし、後述のように実際に計算しようとするとパラメータ表示の二回微分を計算する必要がある、うまいパラメータ表示を見つけないと手計算するのは面倒である。例えば球面のパラメータ表示として

$$\begin{aligned}(x, y) &\mapsto (x, y, \sqrt{1 - x^2 - y^2}), \\ (\theta, \varphi) &\mapsto (\cos \varphi \sin \theta, \sin \varphi \sin \theta, \cos \theta)\end{aligned}$$

の二種類をとることができる。前者は見つけやすいパラメータ表示ではあるが手計算には向かず、後者は手計算はしやすいが見つめるのはなかなか難しい。これは別の例を見ることで分かるだろう。負の定曲率であることで有名な擬球

$$z = -\sqrt{1 - x^2 - y^2} + \operatorname{sech}^{-1} \sqrt{x^2 + y^2}$$

の計算しやすいパラメータ表示を求めるのは簡単ではない。一方で安直なパラメータ表示ならすでに見つっている。このような場合の曲率の計算の困難さは微分の計算が大変であることに起因するのだから、その部分をコンピュータに任せてしまえばこのようなパラメータ表示でも曲率の計算が簡単にできることになる。

## 2 プログラム

まずは任意にパラメータ表示が与えられたときに Gauss 曲率を返すようなプログラムを書く。Gauss 曲率は第一基本形式の行列式と第二基本形式の行列式で書ける。詳しいことは [1] を参照していただきたい。先に実際に書いたプログラムを提示しておく (実際に書いたコードとコメントアウトの位置が違ってしまっているが、修正の仕方が分からなかったので正しいバージョンはソースコードを参照していただきたい)。

```
from sympy import *
import numpy as np
import random
```

```

x = Symbol('x')
y = Symbol('y')

def curv(f1,f2,f3,x0,y0): 曲面のパラメータ付けと曲率を求める点の座標#
    f = np.array([f1, f2,f3])
    fx = diff(f,x)
    fy = diff(f,y) 接ベクトル#
    fxx = diff(f,x,2)
    fxy = diff(f,x,y)
    fyy = diff(f,y,2) ヘシアン#
    v = np.cross(fx,fy)
    a = v[0]*v[0]+v[1]*v[1]+v[2]*v[2]
    n = v/a 単位法ベクトル#第一基本形式

    #
    E = np.dot(fx,fx)
    F = np.dot(fx,fy)
    G = np.dot(fy,fy)第二基本形式

    #
    L = np.dot(v,fxx)
    M = np.dot(v,fxy)
    N = np.dot(v,fyy)

    K = (L*N-M**2)/((E*G-F**2)*a)

    K0 = K.subs(x,x0)
    K1 = K0.subs(y,y0)
    return(K1)

```

まずは曲面の枠場  $fx, fy$  を求め、第二基本形式の計算のため曲面の単位法ベクトル場  $n$  を求めた。以下第一基本形式と第二基本形式を計算するのだが、定義通りに第二基本形式を計算しようとする  $v$  をその長さ  $a$  で割る必要があり、平方根が出てきてしまう。しかし実際には第二基本形式の行列式を使うので、 $L, M, N$  を求めるときには  $\sqrt{a}$  で割らずに Gauss 曲率  $K$  を求めるところで除算をした。

これを使って具体的に計算しよう。まずは球面から実行してみる。パラメータ表示は

$$(x, y) \mapsto (x, y, \sqrt{1 - x^2 - y^2})$$

を利用する。先ほどのコードに引き続いて

```

s = curv(x,y,sqrt(1-x**2-y**2),0,0)
print(s)

```

と入力すると、1 という値が返ってくることが分かる。擬球の場合も考える。

$$\operatorname{sech}^{-1} t = \cosh^{-1} \frac{1}{t} = \log \left( \frac{1}{t} + \sqrt{\frac{1}{t^2} - 1} \right)$$

であることに注意して

$$(x, y, -\sqrt{1 - x^2 - y^2} + \log \left( \frac{1}{\sqrt{x^2 + y^2}} + \sqrt{\frac{1}{x^2 + y^2} - 1} \right), (x, y) = (0, 0.5)$$

を代入する。

```
t = curv(x,y,-sqrt(1-x**2-y**2)+log(1/sqrt(x**2+y**2)+sqrt(1/(x**2+y**2)-1)),0.5,0.5)
print(t)
```

と入力すれば、-1.0000000000000000 と出力される。

次に、与えられた曲面が定曲率であるかを判定するプログラムを作る。先ほどの curv というコードを使って書いた。

```
def const_curv(f1,f2,f3): 定曲率かを判定#
    i = 0
    j = true
    while j == true:
        if i == 0:
            K = curv(f1,f2,f3,random.uniform(-0.3,0.3),random.uniform(-0.3,0.3))
            i = i+1
        elif 0 < i < 1000:
            K1 = curv(f1,f2,f3,random.uniform(-0.3,0.3),random.uniform(-0.3,0.3))
            if -0.0000000001 < K1 - K < 0.0000000001: 誤差の設定はなんとなくで根拠はない#
                K = K1
                i = i+1
            else:
                j = 'nonconst'
        else:
            if K > 0.0000000001:
                j = 'positive const'
            elif K < -0.0000000001:
                j = 'negative const'
            else:
                j = 'zero const'
    print(j)
```

本来は curv で求めた  $K$  が一定の値であることを示すべきだが、コンピュータ上でそれを行うのは難しいのでパラメータに具体的な値をたくさん代入してみて確かめることにした。したがって、このプログラムを使うだけでは本当に定曲率かは分からない。また、代入の際に近似値を求めることになるのを考慮して Gauss 曲率の値は多少ぶれても構わないことにした。

これに先ほど使った球面のパラメータ表示を代入する。

```
const_curv(x,y,sqrt(1-x**2-y**2))
```

とすれば positive const と返ってくる。また、擬球のパラメータ表示を代入すれば、

```
const_curv(x,y,-sqrt(1-x**2-y**2)+log(1/sqrt(x**2+y**2)+sqrt(1/(x**2+y**2)-1))))
```

に対して negative const と返ってくる。なお、

$$(x, y) \mapsto (x, y, 0)$$

を代入して zero const が返ってくることも分かる。

### 3 余談

本論とは全く関係がない話だが、擬球のパラメータ表示として

$$(u, v) \mapsto \left( \frac{\cos u}{\cosh v}, \frac{\sin u}{\cosh v}, v - \tanh v \right)$$

がとれる．これは数理科学研究科棟の大講義室の近くの模型に書いてあった．

## 参考文献

[1] 今野宏, 『微分幾何学』, 東京大学出版会, 2013.