# ヤコビの楕円関数の数値計算プログラム

2018.10.5　　鈴木　実

## 1　はじめに

ヤコビ (Jacobi) の楕円関数の数値計算プログラムを示す．本プログラムは「電子計算機のための数値計算法 III」[1] に拠っている．そこで述べられているように，楕円テータ関数から計算する．楕円テータ関数の数値計算プログラムはすでに本 HP に掲載したので [2]，ここでは，それを関数プログラムとして利用する．計算精度は楕円テータ関数の計算と同じく 10 桁である [2]．本プログラムは別のエントリーでも使われている [3].

ヤコビの楕円関数 $\mathrm{sn}\,u$， $\mathrm{cn}\,u$， $\mathrm{dn}\,u$ の形を実軸上で描くと，図 1 のようになる．

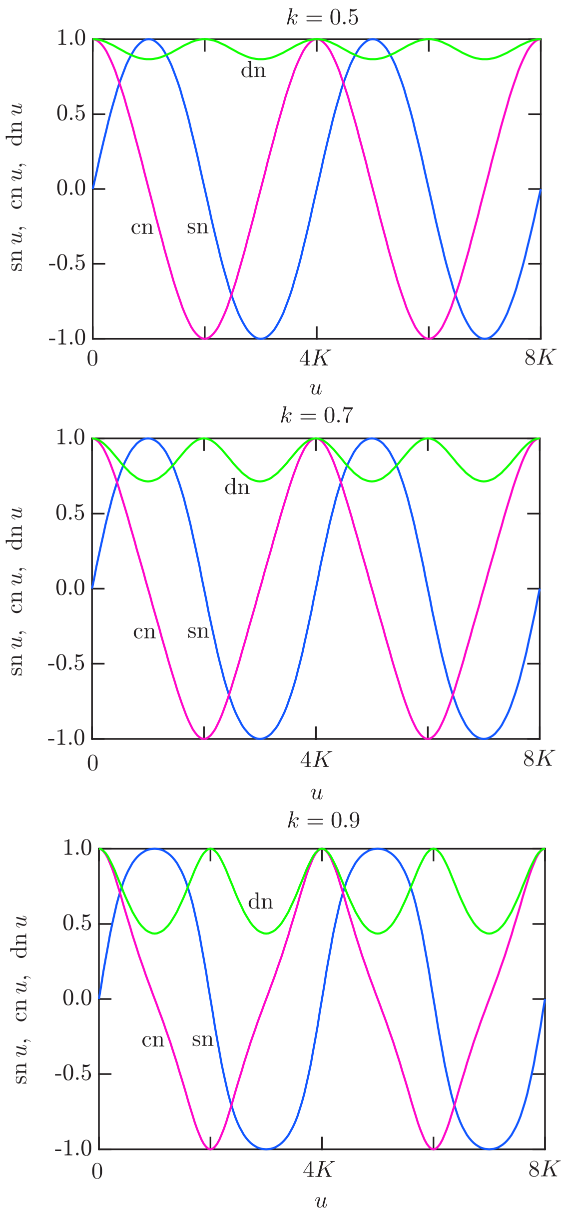


図 1: 実軸上のヤコビの楕円関数 $\mathrm{sn}\,u$， $\mathrm{cn}\,u$， $\mathrm{dn}\,u$. $k = 0.5$，0.7，および 0.9 の場合.

## 2 計算式

### 2.1 ヤコビの楕円関数 $\mathrm{sn}\,u$，$\mathrm{cn}\,u$，$\mathrm{dn}\,u$

第 1 種不完全楕円積分を $u = F(\varphi, k)$ とすると，

$$u = F(\varphi, k) = \int_0^{\varphi} \frac{\mathrm{d}\theta}{\sqrt{1 - k^2 \sin^2 \theta}} \tag{1}$$

である．$k$ は母数である．このとき，ヤコビの楕円関数 $\mathrm{sn}\,u = \mathrm{sn}\,(u, k)$ は,

$$\mathrm{sn}\,u = \mathrm{sn}\,(u, k) = \sin \varphi \tag{2}$$

と定義される．$\varphi = \mathrm{am}\,u = \mathrm{am}\,(u, k)$ は振幅，あるいは振幅関数と呼ばれる．

同じく，ヤコビの楕円関数 $\mathrm{cn}\,u$， $\mathrm{dn}\,u$ は

$$\mathrm{cn}\,(u, k) = \cos \varphi \tag{3}$$

$$\mathrm{dn}\,(u, k) = \sqrt{1 - k^2 \sin^2 \theta} \tag{4}$$

という関係式を用いて計算できる．

ランデン変換を用いて $k$ を十分 0 に近い値に変換すれば，$\mathrm{sn}\,u$ は $\sin u$ に十分近くなるので，$\sin u$ と係数となる分数の計算をすることに帰着する．しかし，この方法は収束が遅い場合もあるので，ノームによる収束の速い楕円テータ関数の数値計算プログラムを用いて計算する [1].

楕円テータ関数とヤコビの楕円関数の関係は

$$\mathrm{sn}\,u = \mathrm{sn}\,(u, k) = \frac{1}{\sqrt{k'}} \frac{\vartheta_1(v)}{\vartheta_4(v)} \tag{5}$$

$$\mathrm{cn}\,u = \mathrm{cn}\,(u, k) = \frac{\sqrt{k'}}{\sqrt{k}} \frac{\vartheta_2(v)}{\vartheta_4(v)} \tag{6}$$

$$\mathrm{sn}\,u = \mathrm{sn}\,(u, k) = \sqrt{k'} \frac{\vartheta_3(v)}{\vartheta_4(v)} \tag{7}$$

である．ただし，

$$v = \frac{u}{2K} \tag{8}$$

で，$K$ は第 1 種完全楕円積分である．したがって，ヤコビの楕円関数の計算は，すでに述べた楕円テータ関数の計算でほとんど尽きている．

## 参考文献

[1] 山内二郎，宇野利雄，一松信,「電子計算機のための数値計算法 III」（培風館）, 1971 年, p.258.

[2] 「楕円テータ関数の数値計算プログラム」（2017/8/7 のエントリー）
http://totoha.web.fc2.com/Elliptic_theta_functions.pdf

[3] 「Onsager の 2 次元イジングモデル厳密解の論文を読む　その 12（完）」（2018/7/1 のエントリー）
http://totoha.web.fc2.com/Onsager_paper12.pdf

[4] 「第 1 種および第 2 種完全楕円積分の数値計算プログラム」（2017/8/3 のエントリー）
http://totoha.web.fc2.com/Complete_Elliptical_Integral.pdf

## プログラムソース

第 1 種完全楕円積分の計算は関数 QKK(double *p) で行う．この関数は，QKKEE(double *p)[4] から第 2 種完全楕円積分の計算を除いたものである．計算は,

```
v=u/(2*K);
TH4=THETA(4, v, q);
SN=THETA(1, v, q)/(sk*TH4);
CN=THETA(2, v, q)*skbis/(sk*TH4);
DN=THETA(3, v, q)*skbis/TH4;}
```

として，SN，CN，DN がヤコビの楕円関数 $\mathrm{sn}\,u$， $\mathrm{cn}\,u$， $\mathrm{dn}\,u$ を与えることになる．

以下のプログラムは，実質，楕円テータ関数の関数サブプログラム THETA(I, v, q) である．THETA(I, v, q) の使い方は，[2] のようにすれば良い．main() では，$u = 0$ から $u = 4K$ までの計算がされる．計算結果は，周期が $2\pi$ または $\pi$ の場合である．

## ヤコビの楕円関数の計算プログラム

```
/*  QKK.c  */
// provides values for the Jacobi elliptic functions from u.
// using the algorhythm in the textbook 電子計算機のための数値計算法 III p.271
/*  M. Suzuki 2018.6.24 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>

int QKK(double *p)
// QKK(p),  p[0]=q, p[1]=qbis, p[2]=kk, p[3]=kkbis, p[4]=kork2, p[5]=M;
{
    double k, k2, kork2, q, qbis, kk, kkbis;
    double eps, eps4, kbis, sqkbis, kkovpi, lnq;
    double pi=3.14159265359;
    int LARGE, M;

    kork2=p[4];
    M=p[5]+0.0001;
    if(M==1) k2=kork2*kork2;
    else k2=kork2;
    LARGE=k2>0.5;
    if(LARGE) k2=1.0-k2;
    if(k2==0)
    {
        q=0;
        qbis=1.0;
        kk=pi/2.0;
        kkbis=1e124;
    }
    else
    {
        kbis=sqrt(1.0-k2);
        sqkbis=sqrt(kbis);
        eps=0.5*k2/((1.0+kbis+2.0*sqkbis)*(1.0+kbis));
```

```
        eps4=pow(eps, 4);
        q=((15.0*eps4+2.0)*eps4+1.0)*eps;
        kkovpi=(1.0+2.0*pow(q, 4))/(1.0+sqkbis);
        kkovpi*=2*kkovpi;
        kk=pi*kkovpi;
        lnq=log(1.0/q);
        kkbis=kkovpi*lnq;
        qbis=exp(-9.86960440109/lnq);
    }
    if(LARGE)
    {
        p[0]=qbis;
        p[1]=q;
        p[2]=kkbis;
        p[3]=kk;
    }
    else
    {
        p[0]=q;
        p[1]=qbis;
        p[2]=kk;
        p[3]=kkbis;
    }
    return 0;
}

double THETA(int I, double v, double q)
{
    double w, q2, q3, q6, lnq, qbis, plq;
    double theta;
    double pi=3.14159265359;
    double doub_pi=6.28318530718;
    int S;


    if(I==2) v+=0.5;
    if(I==3) v-=0.5;
    if(I==1 || I==2)
    {
        S=(v>0)-(v<0);
    }
    else
    {
        S=1;
    }
    v=fabs(v);
    while(v>1)
    {
        v-=1.0;
        if(I==1 || I==2) S*=-1;
    }
    if(I==1 || I==2)
    {
        if(q==0 || v==0)
        {
            theta=0;
            return theta;
        }
    }
    else
    {
        if(q==0)
```

```
        {
            theta=1;
            return theta;
        }
    }
    if(q>0.043)
    {
        lnq=log(q);
        plq=9.8690440109/lnq;
        qbis=exp(plq);
        q2=qbis*qbis;
        q6=q2*q2*q2;
        if(v>0.5) v=1.0-v;
        if(I==1 || I==2) w=-exp(2.0*v*plq); else w=exp(2.0*v*plq);
        theta=S*(1.7724538509/sqrt(-lnq))*exp((v*v-v+0.25)*plq);
        theta*=(((((q6*w+q2)*w+1.0)*w+1.0)*w+q2)*w+q6)/w/w;
        return theta;
    }
    else
    {
        if(I==1 || I==2)
        {
            w=pi*v;
            q2=q*q;
            theta=2*S*sqrt(sqrt(q))*((sin(5.0*w)*q2*q2-sin(3.0*w))*q2+sin(w));
            return theta;
        }
        else
        {
            w=doub_pi*v;
            q2=q*q;
            q3=q2*q;
            theta=2.0*((-cos(3.0*w)*q2*q3+cos(2.0*w))*q3-cos(w))*q+1;
            return theta;
        }
    }
}


int main()
{
    //  QKK(p),  p[0]=q, p[1]=qbis, p[2]=kk, p[3]=kkbis, p[4]=kork2, p[5]=M;
    FILE *fp;
    double k, k2, kk, kk2, kbis, sk, skbis, p[6], q, v, dv;
    double SN, CN, DN, TH4;
    int i, n;
    char *filenameout="Jacobi_0.9.txt";

    k=0.9;
    k2=k*k;
    p[5]=1;
    p[4]=k;
    QKK(p);
    q=p[0];
    kk=p[2];
    kk2=2*kk;
    sk=sqrt(k);
    skbis=sqrt(sqrt(1.0-k2));
    n=200;
    dv=4.0/n;
    fp=fopen(filenameout, "w");
    for(i=0;i<n+1;i++)
```

```
    {
        v=i*dv;
        TH4=THETA(4, v, q);
        SN=THETA(1, v, q)/(sk*TH4);
        CN=THETA(2, v, q)*skbis/(sk*TH4);
        DN=THETA(3, v, q)*skbis/TH4;
        printf("%lf\t%lf\t%lf\t%lf\n", v, SN, CN, DN);
        fprintf(fp, "%lf\t%lf\t%lf\t%lf\n", v, SN, CN, DN);
    }
    fclose(fp);
}
```