

完全楕円積分の高速計算法

寒川 光

日本アイ・ビー・エム株式会社 東京基礎研究所

完全楕円積分は、グリーン関数法や境界要素法で軸対称問題を解く場合、数値解析プログラムの主要なカーネルとして現れる。代表的な計算法として、対数関数を経由する Hastings 公式を用いるものと、算術幾何平均を用いるものがある。本稿では高速計算に焦点をあてる。はじめに上記の方法の精度と速度について述べ、次に区間分割してテーブル駆動型を用いる高速計算法を提案し、最後に完全楕円積分の第1種と第2種が特殊が形で結合したグリーン関数の応用例について述べる。この例では完全楕円積分を独立に計算しても、第1種と第2種が打消しあうため、特別な配慮が必要になる。

High Performance Computation of Complete Elliptic Integral

Hikaru Samukawa

IBM Research, Tokyo Research Laboratory

Complete elliptic integral appears as a major kernel of numerical analysis programs, in which Green function method or boundary element method is applied on axis-symmetric problems. Two typical methods are used, minimax approximation of Hastings form using logarithmic function and arithmetic-geometric mean method. In this paper we focus on high-performance computation of the integral. At first accuracy and performance of the above two methods are described, then a fast method by table driven algorithm is proposed, and finally an application of Green function is described. In this application, since complete elliptic integrals of the first kind and the second kind cancel their precisions each other, special treatment for accuracy is required.

1 はじめに

第1種完全楕円積分の標準形は次式で定義される。

$$K(k) = \int_0^1 \frac{dx}{\sqrt{1-x^2}\sqrt{1-k^2x^2}} \quad (1)$$

$x = \sin \theta$ とおくと次のようにも定義される。

$$K(k) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1-k^2 \sin^2 \theta}} \quad (2)$$

第2種完全楕円積分も同様に標準形と三角関数を用いた定義ができる。

$$E(k) = \int_0^1 \frac{\sqrt{1-k^2x^2}}{\sqrt{1-x^2}} dx \quad (3)$$

$$= \int_0^{\pi/2} \sqrt{1-k^2 \sin^2 \theta} d\theta \quad (4)$$

母数 k は $0 \leq k \leq 1$ の範囲を考える。

$K(k)$ は次のようにべき級数展開される。

$$K(k) = \frac{\pi}{2} \left(1 + \left(\frac{1}{2}\right)^2 k^2 + \left(\frac{1 \cdot 3}{2 \cdot 4}\right)^2 k^4 + \dots \right) \quad (5)$$

第 $r+1$ 項を示す¹。

$$\left(\frac{(2r-1)!!}{(2r)!!} \right)^2 k^{2r} \quad (6)$$

$E(k)$ は次のようにべき級数展開される。

$$E(k) = \frac{\pi}{2} \left(1 - \left(\frac{1}{2}\right)^2 k^2 - \left(\frac{1 \cdot 3}{2 \cdot 4}\right)^2 \frac{k^4}{3} - \dots \right) \quad (7)$$

第 $r+1$ 項を示す [1]。

$$- \left(\frac{(2r-1)!!}{(2r)!!} \right)^2 \frac{k^{2r}}{2r-1} \quad (8)$$

$K(k)$ は $k=0$ で $K(0) = \pi/2$ をとり、 k の増加とともに単調に増加し、 $k=1$ の近傍で急激に大きくなり、 $K(1) = \infty$ となる。 $E(k)$ は $k=0$ でやはり $E(0) = \pi/2$ をとり、 k の増加とともに単調に減少し、 $E(1) = 1$ となる。

本稿では、はじめに代表的な既存の2つの計算法の精度と速度について、次にテーブル駆動型の高速

¹ただし $(2r-1)!! = (2r-1)(2r-3)\dots 3 \cdot 1$ また、 $(2r)!! = (2r)(2r-2)\dots 4 \cdot 2$ とする。

計算法について、最後に完全楕円積分の第1種と第2種が特殊な形で結合したグリーン関数の問題(グリーン関数法で軸対称問題のポアソン方程式を解く問題)について述べる。

2 これまでの計算法

代表的な計算法として、算術幾何平均法と Hastings 公式の最良近似による方法が知られている [2]。

(1) 算術幾何平均法

2数 a, b ($0 < b < a$) に対して, $a_1 = (a+b)/2$, $b_1 = \sqrt{ab}$, 以下順次 $a_n = (a_{n-1} + b_{n-1})/2$, $b_n = \sqrt{a_{n-1}b_{n-1}}$ とおけば, 数列 $\{a_n\}, \{b_n\}$ はともに算術幾何平均 $M(a, b)$ に収束する。

$$\lim_{n \rightarrow \infty} a_n = \lim_{n \rightarrow \infty} b_n = M(a, b)$$

$a = 1, b = \sqrt{1-k^2}$ とすれば, $K(k)$ は次式より得られる。

$$K(k) = \frac{\pi}{2M(1, \sqrt{1-k^2})} \quad (9)$$

つまり, 数列 $\{a_n\}, \{b_n\}$ を必要な桁数まで求めて $\pi/(2a_n)$ で近似する。

さらに $c = k^2$ として $c_n = a_n^2 - b_n^2$ を用いると, $E(k)$ も次式より近似できる。

$$E(k) = \left(1 - \sum_{j=0}^n 2^{j-1} c_j \right) K(k) \quad (10)$$

(2) 最良近似法

Hastings 公式の最良近似(ミニマックス近似)による方法では $t = 1 - k^2$ として, 多項式と, 多項式と対数の積の和として計算する(以下, 対数は自然対数とする)。

$$K(k) = \sum_{j=0}^n a_j t^j + \log \frac{1}{t} \left(\sum_{j=0}^n b_j t^j \right) \quad (11)$$

$$E(k) = \sum_{j=0}^n c_j t^j + \log \frac{1}{t} \left(\sum_{j=0}^n d_j t^j \right) \quad (12)$$

ここで $a_0 = \log 4, b_0 = 0.5, \sum_{j=0}^n a_j = \pi/2, c_0 = 1, d_0 = 0, \sum_{j=0}^n c_j = \pi/2$ である。

2次式の場合の係数(7桁)と4次式の場合の係数(13桁)は最初に Hastings によって得られ [3], また2次から10次式の場合の係数(7桁から20桁)が Cody によって得られた [4]。

以下本節では第1種完全楕円積分を倍精度で計算することを考える。算術幾何平均法には平方根が, 最良近似法には対数関数がいられる。前者は IEEE 浮動小数点数の規格に入っているのが, 後者は入っていないので精度はライブラリ依存となる。

算術幾何平均法は平方根を含む反復計算により M を必要精度まで求め, 最後に除算を行う。文献 [2] に記載されているプログラム(倍精度)は, 算術幾何平均の前回と今回の反復値の差を収束判定値と関数値との積 (1×10^{-15}) によって検定する。

反復回数は k^2 がゼロに近い場合は4回程度, ゼロから離れると10回近くまで増加する。この方法の特長として, 4倍精度を利用することで, 簡単により高い精度を得ることを挙げられる。

最良近似法で上記の2次式と9次式の係数についても倍精度計算してみた。対数関数は XL Fortran ライブラリを用いている。2次式の場合は, 後述する指数区間ごとに検定すると, 区間1 (k^2 が0のから0.5の区間)で精度が最も悪く(最大相対誤差 1.9×10^{-5}), k^2 が大きくなるにつれて向上する。9次式の場合は精度は良好で, 最大相対誤差 2.4×10^{-16} 以下に収まっている。なおこの誤差は, 上記の算術幾何平均法のほぼ半分である。

これ以上の精度を追求する場合は, k を複数の区間に分けて, 区間ごとに異なる近似式を使用する必要がある。この場合, ランデンの変換²などを利用して, 収束の速い式を使用するが, 変換そのものに平方根や除算が含まれるため, 高速化には適さない。

また計算時間については, 上記の9次式的最良近似法を算術幾何平均法と比較すると, 最良近似法がおおよそ半分の計算時間で済む(後述)。さらに最良近似法は対数関数を高速化すれば数10%の高速化は可能であろう。

同じ精度をより少ない数の係数で実現できれば, 関数近似の効率が上がったと評価するなら, 被近似関数の解析的な振舞いをより多く含む近似式のほうが一般的には効率がよいと考えられる。Hastings 公式の特長である $\log 1/t$ を用いる形は, 完全楕円積分を超幾何級数を用いて変換した修正ルジャンドル公式に類似していることから, 被近似関数の解析的な振舞いをよく近似して, 非常に高い精度を得ることができる [4]。

² $k_1 = (1-d)/(1+d)$ ただし $d = \sqrt{1-k^2}$ により $K(k) = (1+k_1)K(k_1)$ を求める。

3 完全楕円積分の高速計算法

Hastings 公式の最良近似法を高速化するには、対数関数を高速化する必要がある。対数関数は最近の言語ライブラリでは多くの縮小区間に分割して、区間ごとにテーブル値を参照する方法で実装されている（テーブルも係数の数に含めると係数の数は多くなる）。このプログラムの高速化にはかなり手間のかかるので、完全楕円積分を直接テーブル駆動型で近似する方法が自然に発想される。

本稿では $K(k)$ を多数の異なる k について連続的に計算することを考える。したがってインターフェースは、複数の引数をまとめて計算するベクトル型が可能である。RISC 計算機でベクトル型インターフェースの関数を高速化するには、アンローリングによる。このため全区間で同一の形の式（次数の同じ多項式で、係数だけが区間ごとに異なる式）を用いる。

3.1 インターフェース

引数は k^2 にする。これは完全楕円積分の代表的な例題である力学の単振り子の周期の問題にも見られるように ($k = 0.5\omega_0\sqrt{l/g}$)、 k^2 が自然に得られる場合が多く、 k が自然に求められる場合に k^2 を求めるのは 1 回の乗算でできるが、逆は開平を必要とするからである。以下 $x = k^2$ として K 、 E を x を変数と見た関数 $K(x)$ 、 $E(x)$ として記述する。

引数 $x = k^2$ の範囲 $[x_{min}, x_{max}]$ として、 $x_{min} = 0$ から (IEEE 形式の倍精度浮動小数点数の場合、1 のゼロ側の ulp は 2^{-53} なので) $x_{max} = 1 - 2^{-53}$ について考察する³。この場合、 $K(x_{max}) = 19.75$ である。また $K(1) = \infty$ とする。もし $1-x$ を引数とすると、 2.225×10^{-308} までが引数の範囲に入り、この値に対する K は 355.58 となる。不正規化数も許すとさらに大きな K を考慮する必要がある。しかし通常倍精度演算でこの引数を生成することは自然でないこと⁴、また、この範囲まで含めて高速計算することが困難なため k^2 を引数とする。

3.2 縮小区間

関数計算ルーチンの内部では $t = 1-x$ を実質的な変数として近似計算を行う。この変換は数学的に

³ulp は Unit in the Last Place の略で、浮動小数点数の仮数の最下位のビットの値として定義される。

⁴4 倍精度で k^2 と $1-k^2$ を求めてから倍精度に丸める必要がある。

は何でもないが、浮動小数点演算では精度を失う。図 1 に変数変換の様子を、仮数部が 4 ビットの 2 進浮動小数点数として示した。 $1/2 < x$ では t は x よりも細かい浮動小数点のスクリーンを与えられているが、 $x < 1/2$ では反対に粗くなるので、 $t = 1-x$ 変換で丸められて精度を失う（図に、 $1/4 < x < 1/2$ の浮動小数点数が、IEEE 浮動小数点数の偶数丸め規約によって丸められる様子を示した）。

テーブルの参照には t の指数部を用いる。IEEE の倍精度演算では 53 区間になる。ここでは第 1 区間は 2^{-1} から 2^0 を、第 m 区間は 2^{-m} から $2^{-(m-1)}$ を扱うことにする。第 2 区間以上では、 t は元の変数より高い解像度を持っている（第 m 区間は $t = 1-x$ 変換で下位 m ビットがゼロになる）。以下、この区間を指数区間と呼ぶ。

しかし第 1 区間では、 t は元の変数より解像度が低くなって（変数が丸められて）いるので、一般的には正確な計算が困難である。しかし完全楕円積分の場合は、べき級数展開式（式 (5)、(7)）から分るとおり、 $x = 0$ の近傍で傾きが $\pm\pi/8$ （約 ± 0.4 ）で、またここで関数値が $\pi/2$ の近傍にある。 $\pi/2$ の ulp は 2^{-52} なので、 $h = 2^{-53}$ とすると、 $K(x)$ 、 $E(x)$ の関数値が変化するには、 x が $5h$ 移動する必要がある。言い換えると、変数が丸められることが関数値の精度に殆ど影響せず、結果的に救われるのである。

3.3 区間の細分割

高速計算を目的とする場合には、近似式の次数を上げるよりも、区間を増やすほうが有利な場合が多い。そこでより多くの区間を用いる場合には、指数区間を細分化する。細分化で、仮数部の上位ビットを直接使用すると、区間を等分割することになるが、これは精度的に不利である。第 1 種完全楕円積分で指数区間を 1:1 に等分割場合と、1: $\sqrt{2}$ に等比分割する場合を比較すると、最大誤差で 4 倍の違いがある。そこで精度を均等にするために、指数区間を等比分割した。この場合、近似計算プログラムでは、2 分割の場合は t^2 の指数部、4 分割の場合は t^4 の指数部からテーブルを引けばよいので、計算時間への影響は小さい。16 分割すると 2^{-52} が 2^{-832} となるので、これが倍精度 IEEE 浮動小数点数での上限となる。細分割は 1 に対する ulp と、0 に対する ulp の差を区間細分化に利用しているのである。な

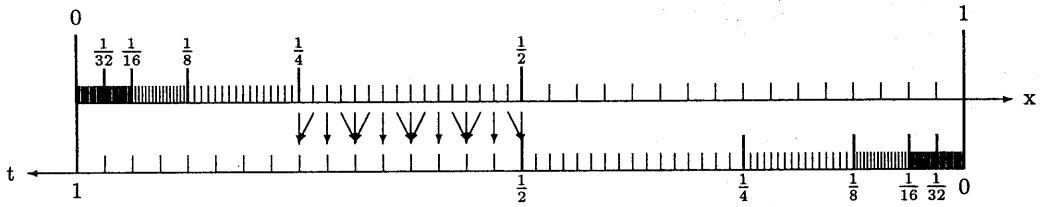


図 1: $t = 1 - x$ 変換

お 1 の近傍の、区間内部に数点しか存在しないような区間は、近似式を使用せずに（直接表を参照することで）関数値を求めることもできる。

区間を縮小しすぎると、単調性に関して 2 つの危険が生ずる。近似関数に（被近似関数が単調増加でも）減少する部分がでてくると、（精度の低い近似式を使用する場合）区間と区間の境界で近似関数が不連続になることである（後述）。

3.4 最良近似式の作成

区間ごとの近似式はチェビシエフ多項式で近似された係数を反復改良（最良近似）して得る [5]。これは被近似関数 $f(x)$ を、区間 $a \leq x \leq b$ で近似式 $g(x) = g(x, c_0, c_1, \dots, c_n)$ によって近似する。 $R(x) = g(x) - f(x)$ として、

$$\min \left(\max_{a \leq x \leq b} |R(x)| \right) \quad (13)$$

となるように $g(x, c_0, c_1, \dots, c_n)$ の c_i を定める。

$g(x)$ が最良近似であるときは、 $n+2$ 個の偏差点があり⁵、 $R(x_j) = (-1)^j \epsilon$ および $R'(x_j) = 0$ となっている。この条件を満たす c_0, c_1, \dots, c_n は通常次の式を連立させて解くことで求める。

$$R'(x_j) = 0 \quad (j = 1, 2, \dots, n) \quad (14)$$

$$R(x_j) + R(x_{j+1}) = 0 \quad (j = 0, 1, \dots, n) \quad (15)$$

具体的には $f(x)$ と $g(x)$ の導関数から偏差点 x_j を求め、次の連立 1 次方程式を解くことで c_0, c_1, \dots, c_n を得る。

$$\begin{pmatrix} a_{01} & \dots & a_{0n} \\ \vdots & \ddots & \vdots \\ a_{n0} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} \Delta c_0 \\ \vdots \\ \Delta c_n \end{pmatrix}$$

⁵ $x_0 = b > x_1 > x_2 > \dots > x_n > x_{n+1} = a$ とする。

$$= \begin{pmatrix} -R(x_0) - R(x_1) \\ \vdots \\ -R(x_n) - R(x_{n+1}) \end{pmatrix} \quad (16)$$

係数行列は近似関数を $g(x) = c_0 + c_1x + c_2x^2 + \dots$ とすると、 $R(x_j) + R(x_{j+1}) = 0$ より次の形になり、

$$a_{ij} = \frac{\partial R(x_{i-1})}{\partial c_{j-1}} + \frac{\partial R(x_i)}{\partial c_{j-1}} = x_i^{j-1} + x_{i+1}^{j-1} \quad (17)$$

チェビシエフ多項式 $g(x) = c_0T_0(x) + c_1T_1(x) + c_2T_2(x) + \dots$ のまま計算する場合は次の形になる。

$$a_{ij} = T_{j-1}(x_{i-1}) + T_{j-1}(x_i) \quad (18)$$

このステップは、 c_i に関しては線形だが x_j に関しては非線形なので、反復する必要がある。なおこの近似式の生成プログラムは 4 倍精度で行い、関数値は算術幾何平均法により求めた。

この方法で得られた近似式を用いて高精度の計算をしてみたところ、ほぼ正しく丸められた結果を得るためには、14 次で各指数区間を 4 細分割、または 21 次で各指数区間は細分割なし、が必要であることが分った。なおこの場合チェビシエフ多項式のままで近似計算するほうが精度的にもやや有利である。また、係数 c_0 には補正項を用いて、丸めを向上させた（チェビシエフ近似の場合、 c_0 がもっとも大きな係数で、かつ最後に計算されるため、丸めの制御を入れやすい）。

3.5 精度の緩和

この方法は精度を緩めることで高速化できるが、関数の連続性についての注意が必要である。

(1) 単調性の保存

被近似関数 $f(x)$ の傾きが小さく単調な場合、誤差 ϵ を大きくすると、近似関数 $g(x) = f(x) + R(x)$ が単調でなくなる危険性がある。しかし次数 5 で、各指数区間は分割しない場合について調べてみたこと

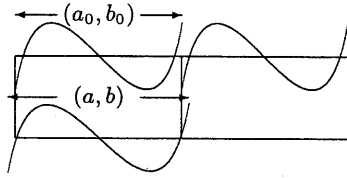


図 2: 隣接区間での誤差の変化と区間の拡大

ろ、区間両端での $R(x)$ の差から得られる平均的な傾きは、 ϵ よりも数桁大きく、単調性が逆転することはない。

(2) 区間境界

最良近似を行うと、区間の端点が偏差点となるため、精度を緩めると、区間境界で関数値が不連続になる。図 2 で、2 次式近似の場合の誤差を示した。本来の引数の指数部から得られる区間を (a_0, b_0) としている。この場合、 m 番目の区間の右端の点と、 $m+1$ 番目の区間の左端の点とは、図のように不連続になる。

そこで区間を仮想的に拡張して、 a_0 と b_0 とで正確になるような区間 (a, b) を用いる。これには誤差 $R(t)$ を区間の関数 $R(t, a, b)$ と考えて、次の式を反復して解くことで $R(a_0, a, b) = 0$ および $R(b_0, a, b) = 0$ となる a と b を求めた。

$$\begin{pmatrix} \partial R(a_0, a, b)/\partial a & \partial R(a_0, a, b)/\partial b \\ \partial R(b_0, a, b)/\partial a & \partial R(b_0, a, b)/\partial b \end{pmatrix} \begin{pmatrix} \Delta a \\ \Delta b \end{pmatrix} = \begin{pmatrix} -R(a_0, a, b) \\ -R(b_0, a, b) \end{pmatrix} \quad (19)$$

なお最良近似によって c_i を最適化する反復は、この反復の内側に入る。

この方法で、区間境界での連続性を満たしたまま、近似精度を落としてみたところ、次数 5 で、各指数区間は分割しない場合に、6 桁（単精度）の精度があった。このとき ϵ が 10^{-7} まで緩めても、区間境界では精度を 10^{-16} （倍精度）に維持することができる。また図 2 からも分るように、次数は偶数のほうが誤差のつながりが自然になる。

3.6 計算時間の比較

IBM の RISC System/6000 の 990 型 (71.5MHz) で 100 万点のデータについて測定した (表 1)。算術幾何平均法は平方根を含む反復を数回行うために遅い。Hastings 公式は対数関数を使用するが、これに約 60 クロックを使用している。対数関数をベクトルインターフェース (MASS) に変更しても、半分

表 1: 計算時間の比較

方法	時間 (秒)	クロック数
算術幾何平均法	3.95	282
Hastings 公式 (9 次)	2.06	147
縮小区間 (14 次 4 細分割)	0.40	28
縮小区間 (6 次)	0.30	21

以下の時間で計算することは困難と予想される [6]。

これらの方法に比較してここで述べた方法は格段に速く、単独で行われる除算 1 回強の時間で関数値 1 つを計算できる。なお完全楕円積分では、対数関数などの初等関数と異なり、区間ごとに異なる多項式係数を用いるので、係数が 2 次元配列にある。初等関数では係数をレジスタにおいてループを回せるが、ここでは配列要素を指すための整数命令とロード命令のために、Mflop/s 性能値で見ると初等関数よりも遅くなる。

4 グリーン関数の計算

有限な領域 V の境界面 S 上で境界条件が与えられた場合、 V 内部のポアソンまたはラプラス方程式の解は、グリーン関数を用いて求めることができる。電磁気学の問題で、半径 a の円形導線に電流 I が流れている場合の、点 P における磁束密度を考える (図 3)。磁束密度に対するベクトルポテンシャル $A(r, \theta)$ は次式で表される [7]。

$$A(r, \theta) = \frac{4I\sqrt{a}}{c\sqrt{r}\sin\theta} \left[\frac{(2-k^2)K(k) - 2E(k)}{2k} \right] \quad (20)$$

$$k^2 = \frac{4ar\sin\theta}{a^2 + r^2 + 2ar\sin\theta} \quad (21)$$

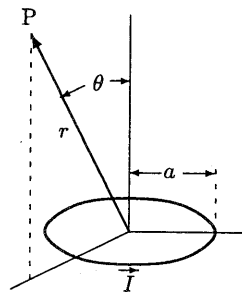


図 3: 円電流による磁束密度

$x = k^2$ として、被近似関数として次の形を考える。

$$f(x) = \frac{(1-x/2)K(x) - E(x)}{\sqrt{x}} \quad (22)$$

以下 $f(x)$ をグリーン関数と呼ぶ。グリーン関数の分子に着目してこれを $G(x)$ とする。 $G(x)$ は式(5), (7)より次のようにべき級数展開されるが、 $(1-x/2)$ のために x の1次の項も消える。

$$G(x) = \frac{\pi}{2} \left(2 \cdot 2 \left(\frac{1}{2 \cdot 4} \right)^2 x^2 + \dots \right) \quad (23)$$

x の r 乗の項は次の形である。

$$\frac{\pi}{2} \left(2r(r-1) \left(\frac{(2r-3)!!}{(2r)!!} \right)^2 x^r \right) \quad (24)$$

すなわちグリーン関数は、 $K(x)$ や $E(x)$ を独立に求めても打消しがあること、 $\lim_{h \rightarrow 0} f(h) = 0$ であることのために、数値計算としては $K(x)$ や $E(x)$ を単独に計算するよりも難しい問題になっている。

$K(x)$ と $E(x)$ を経由してグリーン関数を求めるには、4倍精度の算術幾何平均法を用いる必要がある⁶。前節で述べた縮小区間を用いる方法で直接グリーン関数を近似する場合も、 $x=0$ の近傍を特別扱いする必要がある。

まず変数が丸められることを避けるために、 $x < 1/2$ の区間では t ではなく x のままで近似計算する。これにより近似計算プログラムには if ブロックが入り遅くなるが、全区間で同じ次数の近似式を用いれば、主要部のアンローリングは可能である。

次に $x=0$ の近傍でグリーン関数の高精度な近似多項式を作りたいが、これは困難である。そこで代わりに $G(x)$ を近似することにする。これは $G(x)$ のべき級数の展開式からテレスコピングによって得ることができる。

3つめは、 $x=0$ の近傍で精度が不足するところ (x_1 とする) を調べ (14 次の多項式で指数区間を4細分割した場合は、 $x < 0.02$ では相対誤差が1桁増える)、 x_1 のゼロ側を別処理する。

完全楕円積分の近似計算プログラムでは、引数が特殊な値のチェックを、ゼロ以下、1以上としているが、この部分を、 x_1 以下、1以上として、 x_1 以下の場合はさらにゼロ以下か以上かで分類する。 $[0, x_1]$ はテレスコピングによる近似式より $G(x)$ を求め、

⁶Hastings の近似式は $t = 1-x$ 変換で変数が丸められるので使えない。

これを \sqrt{x} で割るのでアンローリングは適用できない。したがって x_1 以上の区間に4ウェイのアンローリングを適用した場合、連続する4つのデータに1つでも x_1 以下の値があると性能の低下を招く。

5 おわりに

完全楕円積分の代表的な計算法には、算術幾何平均法と Hastings 公式の最良近似法があるが、後者のほうが高速である。そこでこれを RISC 計算機向けに高速化しようとする、対数関数を改訂する必要に迫られる。言語ライブラリでは初等関数はテーブル駆動型で実装されているので、完全楕円積分を直接被近似関数として最良近似する方法を思い付く。このようにすると“精度を緩めて速度を優先させる”というような実装も実現しやすい。

完全楕円積分の場合は、変数の範囲が $[0, 1]$ であり、また変数を k^2 に選ぶことで区間分割が適用しやすい。この方法で最良近似による高精度な方法と同程度の精度を保ったまま、5倍程度の性能を達成した。また区間と区間の繋ぎ目をスムーズにしたまま精度を緩めた高速版も実現した。最後に、完全楕円積分を単独に計算したのでは精度を失う個所を持つグリーン関数の問題に対しても、この方法に改良を加えれば対応できることを示した。

参考文献

- [1] 安藤四郎: 楕円積分・楕円関数入門, 日新出版 (1970).
- [2] 大野 豊, 磯田和男監修: 新版 数値計算ハンドブック, オーム社, (1990).
- [3] Abramowitz, M. and Stegun, I. (editor): Handbook of Mathematical Functions, Dover Publications, New York, (1968)
- [4] Cody, W. J.: Chebyshev approximations for the elliptic integrals K and E, Mathematics of Computation, 19, pp.105-112, (1965)
- [5] 浜田穂積: 近似式のプログラミング, 培風館, (1995).
- [6] Andersson, S., Bell, R., Hague, J., Holger, H., Mayes, P., Nakano, J., Shieh, D., and Tuccillo, J.: RS/6000 Scientific and Technical Computing: POWER3 Introduction and Tuning Guide, IBM Corp. SG24-5155 (1998).
- [7] J. D. Jackson: Classical Electrodynamics (second edition), John Wiley & Sons (1975), 西田 稔訳: 電磁気学, 吉岡書店 (1994).