テクニカルノート

多数桁の円周率を計算するための公式の改良：ガウス–ルジャンドルの公式とボールウェインの4次の収束の公式

高橋大介[†] 金田康正[†]

本論文では，多数桁の円周率を計算する際に使われることが多いガウス–ルジャンドルの公式，およびボールウェインの4次の収束の公式を，多倍長数どうしの乗算回数の削減により改良する手法について述べる．

Improvement of the Algorithms for π Calculation: The Gauss-Legendre Algorithm and the Borwein's Quartically Convergent Algorithm

Daisuke Takahashi[†] and Yasumasa Kanada[†]

This paper describes for improvement of the Gauss-Legendre algorithm and the Borwein's quartically convergent algorithm for π calculation. Improvements are focused to the reduction of multiple-precision multiplication.

1. はじめに

本論文では，多数桁の円周率を計算する際に使われることが多いガウス–ルジャンドルの公式，およびボールウェインの4次の収束の公式の改良について述べる．これらの改良は多倍長数どうしの乗算回数を削減することにより実現するものであり，実際の計算においても十分に効果を発揮する．以下，2章でガウス–ルジャンドルの公式の改良について，3章でボールウェインの4次の収束の公式の改良について述べる．最後の4章はまとめである．

2. ガウス–ルジャンドルの公式

1976年にアメリカのサラミンとオーストラリアのブレントの2人によって，まったく独立に，arctan の級数展開を用いた公式よりも格段に優れた方法が発表された[1),2)]．

この方法は楕円積分に関する Gauss の計算法と，Legendre の関係式を用いたもので，ある初期値の相加・相乗平均の反復で π を求める方法である．

$a_0^2 = b_0^2 + c_0^2$ を満たす正である a_0, b_0, c_0 を用いて，数列 $a_n, b_n, c_n\ (> 0)$ を次のように定める．

$$a_n = \frac{1}{2}(a_{n-1} + b_{n-1}), \quad b_n = \sqrt{a_{n-1}b_{n-1}}$$
$$c_n^2 = a_n^2 - b_n^2 \tag{1}$$

すると，a_n および b_n は2次の収束を示し，

$$\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = \mathrm{agm}(a_0, b_0) \tag{2}$$

となる．このように定義される $\mathrm{agm}(a_0, b_0)$ を用いて π は，

$$\pi = \frac{4\mathrm{agm}(1, k)\mathrm{agm}(1, k')}{1 - \sum_{j=1}^{\infty} 2^j (c_j^2 + c_j'^2)} \tag{3}$$

と書ける．ただし，$k^2 + k'^2 = 1$ なる正の k, k' に対して $a_0 = a_0' = 1$, $b_0 = k$, $b_0' = k'$ として $\mathrm{agm}(a_0, b_0)$, $\mathrm{agm}(a_0', b_0')$ を計算し，その計算の途中で計算される $c_j^2 (\equiv a_j^2 - b_j^2)$, $c_j'^2 (\equiv a_j'^2 - b_j'^2)$ を使用する．

$k = k' = 1/\sqrt{2}$ の場合に，ガウス–ルジャンドルの公式による π の計算方法は次のようになる．

† 東京大学大型計算機センター
Computer Centre, University of Tokyo

$A = 1,\ B = 1/\sqrt{2},\ T = 1/4,\ X = 1$ として，A と B の差が必要とする精度より大きな間，次に示すカッコ { とカッコ } で囲まれる部分を繰返し実行する．

$$\{Y := A;\ A := \frac{A+B}{2};\ B := \sqrt{B \times Y};$$
$$T := T - X \times (Y-A)^2;\ X := 2 \times X\}$$

すると，π の値は $(A+B)^2/(4T)$ となる．ただし，$A,\ B,\ T,\ Y$ の各値は，求めようとする精度以上で計算しておく必要がある．

この公式は 2 次の収束を示すので，求めようとする桁数を N とすると，$\log_2 N$ 回程度の反復で N 桁の π の値が求まる．平方根の計算や逆数計算はよく知られているニュートン法により，加減算と乗算に置き換えることができる．N 桁の多倍長数の加減算の計算量は明らかに $O(N)$ であるので，N 桁の多倍長数の乗算に必要な計算量を $M(N)$ とすれば，π は $O(M(N)\log N)$ の計算量で求まることになる．

ガウス–ルジャンドルの公式においては，1 回の反復につき，1 回の多倍長数の乗算（$a_{n-1}b_{n-1} := B \times Y$）と 1 回の多倍長数の自乗計算（$c_n^2 := (Y-A)^2$），そして 1 回の多倍長数の平方根の計算（$\sqrt{a_{n-1}b_{n-1}} := \sqrt{B \times Y}$）が必要である．$N$ 桁の多倍長数の加減算および，多倍長数と単精度定数の乗算は，$O(N)$ のオーダーで求まるので，ここでは考慮に入れないものとする．

N 桁の多倍長数 A と B の積を求めるのに FFT を用いると，$O(N(\log N)(\log\log N))$ の計算量で済むことが知られている[3)]．数千桁以上の乗算においては，FFT を用いる方法が最も高速に計算できる[4)]ことから，本論文では乗算に FFT を使うものとして議論する．

A と B の積を求める際に A と B それぞれの順 FFT を求めることになるので，順 FFT が 2 回，畳み込みの後の逆 FFT が 1 回の合計 3 回の FFT が必要となるが，多倍長数 A の自乗を計算するには，A の順 FFT が 1 回，逆 FFT が 1 回の合計 2 回で済む．

2.1 ガウス–ルジャンドルの公式における改良

本節では，ガウス–ルジャンドルの公式において，1 回の多倍長乗算が 1 回の多倍長の自乗演算に置き換えることができることを示す．

多倍長数の乗算の計算時間のほぼ 85%以上が FFT ルーチンに費やされる[4)]ので，FFT の回数が減ることは，実行時間の大幅な短縮になる．

まず c_n^2 の定義より，$b_n^2 = a_n^2 - c_n^2$ となる．また b_n の定義より $b_n^2 = a_{n-1}b_{n-1}$ となる．したがって，$a_{n-1}b_{n-1}$ の乗算が，a_n^2 と c_n^2 の 2 つの自乗計算に置き換わっていることが分かる．ところが，c_n^2 はいずれにしろ計算しなければならない値なので，実質的に $a_{n-1}b_{n-1}$ の乗算が a_n^2 の 1 回の計算で求まることになる．

表 1 ガウス–ルジャンドルの公式での各反復における計算回数の比較

Table 1 Comparison with the number of operations in each iteration of the Gauss-Legendre algorithm.

	従来の方法	提案する方法
乗算	1	0
自乗計算	1	2
平方根計算	1	1

さらに最初の反復において，a_1, b_1, c_1 はそれぞれ，

$$a_1 = \frac{a_0+b_0}{2} = \frac{2+\sqrt{2}}{4}, \quad b_1 = \sqrt{a_0 b_0} = \frac{1}{2^{1/4}}$$
$$c_1^2 = a_1^2 - b_1^2 = \frac{3-2\sqrt{2}}{8} \qquad (4)$$

となるので，1 回目の反復を省略することができる．ガウス–ルジャンドルの公式においては，N 桁の π を求めるのに必要な反復回数は $O(\log N)$ であるため，1 回でも反復回数が減ることは計算時間の大幅な短縮につながる．これらにより改良されたガウス–ルジャンドルの公式による π の計算方法は，次のようになる．

$A = (2+\sqrt{2})/4,\ B = 1/2^{1/4},\ T = (\sqrt{2}-1)/8,\ X = 2$ として，A と B の差が必要とする精度より大きな間，次に示すカッコ { とカッコ } で囲まれる部分を繰返し実行する．

$$\{A := \frac{A+B}{2};\ B := (A-B)^2;$$
$$T := T - X \times B;\ B := \sqrt{A^2 - B};$$
$$X := 2 \times X\}$$

すると，π の値は $(A+B)^2/(4T)$ となる．ただし，$A,\ B,\ T$ の各値は，求めようとする精度以上で計算しておく必要がある．

この計算方法では，**表 1** に示すように多倍長数同士の乗算を含まず自乗計算だけで構成されていることが分かる．

3. ボールウェインの 4 次の収束の公式

1980 年代以降，カナダの数学者ボールウェイン兄弟によって，それまでに知られている方法と同等の計算量を有する方法がいくつか発見されている[5)]．ボールウェイン兄弟が発見し，これまでに多数桁の円周率計算で使用されている[6),7)]4 次の収束の公式の 1 つを以下に示す．

$a_0 = 6 - 4\sqrt{2}$ および $y_0 = \sqrt{2} - 1$ として

$$y_{n+1} = \frac{1-(1-y_n^4)^{1/4}}{1+(1-y_n^4)^{1/4}} \quad (5)$$

$$a_{n+1} = a_n(1+y_{n+1})^4 - 2^{2n+3} y_{n+1}(1+y_{n+1}+y_{n+1}^2) \quad (6)$$

の級数を求める精度まで繰返し計算すると π の値は $1/a_n$．ただし，a_n, y_n の値は求める精度以上の精度で計算しておく必要がある．

3.1 ボールウェインの 4 次の収束の公式における改良

この公式で式 (5) の y_{n+1} を計算するのに，$y_n^4, (1-y_n^4)^{1/4}$，そして $1/(1+(1-y_n^4)^{1/4})$ の計算とその結果に $(1-(1-y_n^4)^{1/4})$ を掛ける必要がある．

ところが，金田らの行った 1988 年の 2 億桁計算[8]以降の計算ですでに行っている，以下のような工夫を行うことにより，乗算および自乗計算の回数を減らすことができる．

まず式 (5) は次のように変形できる．

$$y_{n+1} = \frac{1-(1-y_n^4)^{1/4}}{1+(1-y_n^4)^{1/4}} = 1 - \frac{2}{1+(1-y_n^4)^{-1/4}} \quad (7)$$

このような変形を行うことによって，分母の逆数をとって 2 倍したものを 1 から引くことで y_{n+1} が求められ，式 (5) では必要であった，分子の多倍長数の乗算が不要になっていることが分かる[9]．

また，分母が 4 乗根の逆数の形になっているので，$a^{1/4}$ を求めるのに必要であったニュートン法による $a^{-1/4}$ の計算[7]の後に，その結果を 3 乗して最後に a を掛ける必要がなくなっている．この 3 乗の計算には，1 回の自乗計算と 1 回の乗算が必要であるので，結局 y_{n+1} の計算において乗算を 3 回，自乗計算を 1 回減らすことができることが分かる．

次に，式 (6) の a_{n+1} の計算において，普通に計算すると，$a_n(1+y_{n+1})^4 = a_n((1+y_{n+1})^2)^2$ で乗算が 1 回と自乗計算が 2 回，$y_{n+1}(1+y_{n+1}+y_{n+1}^2)$ で乗算が 1 回と自乗計算が 1 回となり，合計で乗算が 2 回と自乗計算が 3 回必要となる．

ところが，$y_{n+1}(1+y_{n+1}+y_{n+1}^2)$ の計算は，

$$\begin{aligned} & y_{n+1}(1+y_{n+1}+y_{n+1}^2) \\ &= \frac{(1+y_{n+1})^4 - (1+2y_{n+1}^2+y_{n+1}^4)}{4} \\ &= \frac{(1+2y_{n+1}+y_{n+1}^2)^2 - (1+2y_{n+1}^2+y_{n+1}^4)}{4} \end{aligned} \quad (8)$$

と変形できる．したがって，a_{n+1} を計算するためには，y_{n+1}^2，$y_{n+1}^4 = (y_{n+1}^2)^2$，$(1+y_{n+1})^4 = (1+2y_{n+1}+y_{n+1}^2)^2$，$a_n(1+y_{n+1})^4$ を計算すればよいので，結局乗算が 1 回と自乗計算が 3 回で済むことが分かる．つまり，乗算が 1 回減ることになる．

表 2　ボールウェインの 4 次の収束の公式での各反復における計算回数の比較

Table 2　Comparison with the number of operations in each iteration of the Borwein's quartically convergent algorithm.

	従来の方法	提案する方法
乗算	5	1
自乗計算	6	3
逆 4 乗根計算	1	1
逆数計算	1	1

また，式 (6) において y_{n+1}^4 の値を保存しておけば，式 (5) で $n \to n+1$ とした場合の反復において，y_{n+1}^4 を計算しなくて済む．したがって，この方法でも 4 乗計算が 1 回，つまり自乗計算が 2 回減ることが分かる．

これらの工夫を行うことにより，そのままの形でボールウェインの 4 次の収束の公式を計算する方法（従来の方法）に比べ，1 回の反復につき，乗算を 4 回，自乗計算を 3 回減らすことができる（**表 2** 参照）．

表 1 のガウス–ルジャンドルの公式における各反復における計算回数と比較すると，ボールウェインの 4 次の収束の公式では各反復の計算回数が多くなっているが，4 次の収束であることから π を計算するのに必要な反復回数は 2 次の収束のガウス–ルジャンドルの公式の約 1/2 で済むことに注意しておく．実際の計算時間では同じ程度の計算時間を必要とする[7]．

また，初期値を計算する際に $y_0 = \sqrt{2} - 1$ であるが，$y_0^4 = 17 - 12\sqrt{2}$ であるので，最初に y_0^4 を計算しておけば，1 回目の反復において自乗計算を 2 回減らすことができる．

以上の工夫を行ったボールウェインの 4 次の収束の公式の計算方法は次のようになる．

$A = 6 - 4\sqrt{2}$, $Y = 17 - 12\sqrt{2}$, $X = 2$ として，Y が必要とする精度より大きな間，次に示すカッコ { とカッコ } で囲まれる部分を繰返し実行する．

$$\begin{aligned} \{ & Y := 1 - \frac{2}{1+(1-Y)^{-1/4}}; \\ & B := Y^2;\ W := (1 + 2 \cdot Y + B)^2; \\ & A := A \cdot W - X \times (W - (1+B)^2); \\ & X := 4 \times X \} \end{aligned}$$

すると，π の値は $1/A$ となる．ただし，A, B, W, Y の各値は，求めようとする精度以上で計算しておく必要がある．

さらに，式 (5) と式 (6) において，10 進で N 桁を

求める際には y_n が4次の収束を示すことから，最後の反復では $y_n^4 < 10^{-N/4}$ となるため，以下に示すように逆4乗根および逆数計算を省略することができる．

まず式 (7) の計算において，$(1-y_n^4)^{-1/4}$ をテイラー展開すると，

$$(1-y_n^4)^{-1/4} = 1+\frac{1}{4}y_n^4+\frac{5}{32}y_n^8+\frac{15}{128}y_n^{12}+\cdots \quad (9)$$

となる．これより，

$$y_{n+1}=\frac{1}{8}y_n^4+\frac{1}{16}y_n^8+\frac{21}{512}y_n^{12}+\cdots \quad (10)$$

となるので，式 (6) において，

$$(1+y_{n+1})^4 = 1+\frac{1}{2}y_n^4+\frac{11}{32}y_n^8+\frac{17}{64}y_n^{12}+\cdots \quad (11)$$

となり，

$$y_{n+1}(1+y_{n+1}+y_{n+1}^2) = \frac{1}{8}y_n^4+\frac{5}{64}y_n^8+\frac{15}{256}y_n^{12}+\cdots \quad (12)$$

となることに注意しておく．

π の値を求めるには，式 (6) の a_{n+1} が求まればよいので，そのためには a_n と式 (11)，式 (12) の値だけを計算すればよいことになる．

(i) $10^{-N} \le y_n^4 < 10^{-N/2}$ のとき

このときは，$y_n^8 < 10^{-N}$ となるので，式 (11) において，y_n^8以下の項を無視すると，

$$(1+y_{n+1})^4 \approx 1+\frac{1}{2}y_n^4 \quad (13)$$

となり，式 (12) は，

$$y_{n+1}(1+y_{n+1}+y_{n+1}^2) \approx \frac{1}{8}y_n^4 \quad (14)$$

と近似できるので，この場合は多倍長数どうしの乗算および自乗計算，そして多倍長数の逆4乗根や逆数計算が不要になっていることが分かる．

(ii) $10^{-N/2} \le y_n^4 < 10^{-N/3}$ のとき

この場合は (i) の場合と同様に考えると，式 (11) は，

$$(1+y_{n+1})^4 \approx 1+\frac{1}{2}y_n^4+\frac{11}{32}y_n^8 \quad (15)$$

となり，式 (12) は，

$$y_{n+1}(1+y_{n+1}+y_{n+1}^2) \approx \frac{1}{8}y_n^4+\frac{5}{64}y_n^8 \quad (16)$$

と近似できるので，この場合は多倍長数の逆4乗根や逆数計算が不要になっていることが分かる．

(iii) $10^{-N/3} \le y_n^4 < 10^{-N/4}$ のとき

この場合も，これまでと同様にして式 (11) は，

$$(1+y_{n+1})^4 \approx 1+\frac{1}{2}y_n^4+\frac{11}{32}y_n^8+\frac{17}{64}y_n^{12} \quad (17)$$

となり，式 (12) は，

$$y_{n+1}(1+y_{n+1}+y_{n+1}^2) \approx \frac{1}{8}y_n^4+\frac{5}{64}y_n^8+\frac{15}{256}y_n^{12} \quad (18)$$

と近似できるので，この場合も多倍長数の逆4乗根や逆数計算が不要になっていることが分かる．

4. まとめ

本論文では，多数桁の円周率を計算する際に使われることが多いガウス-ルジャンドルの公式，およびボールウェインの4次の収束の公式を，多倍長数の乗算回数の削減により改良する手法について述べた．

参考文献

1) Salamin, E.: Computation of π Using Arithmetic-Geometric Mean, *Math. Comp.*, Vol.30, pp.565–570 (1976).
2) Brent, R.P.: Fast Multiple-Precision Evaluation of Elementary functions, *J. ACM*, Vol.23, pp.242–251 (1976).
3) Knuth, D.E.: *The Art of Computer Programming, Vol.2: Seminumerical Algorithms*, Addison-Wesley, Reading, MA (1981).
4) 高橋大介，金田康正：多倍長平方根の高速計算法，情報処理学会研究報告 95-HPC-58, pp.51–56 (1995).
5) Borwein, J.M. and Borwein, P.B.: The Arithmetic-Geometric Mean and Fast Computation of Elementary Functions, *SIAM Rev.*, Vol.26, pp.351–365 (1984).
6) Bailey, D.H.: The Computation of π to 29,360,000 Decimal Digits Using Borwein's Quartically Convergent Algorithm, *Math. Comp.*, Vol.50, pp.283–296 (1988).
7) 高橋大介，金田康正：円周率—高速計算法と統計性 (3)，情報処理学会第37回プログラミングシンポジウム報告集，pp.73–84 (1996).
8) Kanada, Y.: Vectorization of Multiple-Precision Arithmetic Program and 201,326,000 Decimal Digits of π Calculation, *Supercomputing 88: Volume II, Science and Applications*, pp.117–128, IEEE Computer Society Press (1989).
9) 二宮市三：多倍長高速演算システム— π とメルセンヌ素数，第26回数値解析シンポジウム講演予稿集，pp.47–50 (1997).

(平成9年6月20日受付)
(平成9年9月10日採録)