乗積公式と積分による q-gamma 関数の
精度保証付き数値計算

○金泉大介 *，丸野健一 †

1 はじめに

可積分系などの数理物理の世界には，多様な特殊関数が住んでいるが, それらの中には性質が十分に理解されていないものも多くある．多様な特殊関数の性質を探索する手段の一つとして，様々な力学系の問題に適用され強力なツールとなりつつある精度保証付き数値計算が考えられる．

精度保証付き数値計算とは近似値の計算をすると同時に計算結果の (数学的に) 厳密な誤差評価も行う数値計算のことである [1]. 計算する際は数を区間に置き換えて計算し, 真値を含む区間を結果として出力する. これにより真値が含まれる区間を知ることができるだけでなく, 区間の幅から計算に混入した誤差を把握できる.

本講演では，可積分系などの数理物理で現れる様々な特殊関数（つまり可積分な微分方程式または差分方程式の解）の精度保証付き数値計算法の確立を目指し，q-特殊関数の精度保証付き数値計算についての研究結果を報告する．これまで様々な特殊関数の精度保証付き数値計算法が開発されてきたが [2, 3, 4], 可積分系などの数理物理で現れる q-特殊関数の精度保証付き数値計算法はまだない. q-特殊関数の精度保証付き数値計算法を確立するため, 可積分系で現れる q-gamma 関数の精度保証付き数値計算法の研究を行なった. 本講演では我々が以前提案した交代級数の性質を用いる方法を振り返ると共に, 新たに乗積公式を用いる方法と積分を用いる方法を提案する.

2 精度保証付き数値計算

精度保証付き数値計算では" 区間演算" という技術によって数値計算の際に生じる誤差を把握している [1]. 区間演算を行う際は数を閉区間に置き換えて下記のルールに従い計算している. ($\bar{x}$ が上限, $\underline{x}$ が下限,$[x] = [\underline{x}, \bar{x}]$ とする)

- 加算: $[x] + [y] = [\underline{x} + \underline{y}, \bar{x} + \bar{y}]$
- 減算: $[x] - [y] = [\underline{x} - \bar{y}, \bar{x} - \underline{y}]$
- 乗算: $[x] \times [y] = [\min(\underline{xy}, \underline{x}\bar{y}, \underline{y}\bar{x}, \bar{x}\bar{y}), \max(\underline{xy}, \underline{x}\bar{y}, \underline{y}\bar{x}, \bar{x}\bar{y})]$
- 除算: $[x]/[y] = [\min(\underline{x}/\underline{y}, \underline{x}/\bar{y}, \bar{x}/\underline{y}, \bar{x}\bar{y}), \max(\underline{xy}, \underline{x}\bar{y}, \underline{y}\bar{x}, \bar{x}/\bar{y})]$
  (ただし $[y]$ は 0 を含まない区間とする)

精度保証付き数値計算では主に次の誤差を扱う.

*早稲田大学大学院基幹理工学研究科数学応用数理専攻 (〒 169-8555　東京都新宿区大久保 3-4-1), E-mail: daisuke15@asagi.waseda.jp）

†早稲田大学理工学術院 (〒 169-8555　東京都新宿区大久保 3-4-1), E-mail: kmaruno@waseda.jp）

- 丸め誤差: 実数を計算機で表現可能な数に置き換えたことによる誤差
- 打切り誤差: 無限回やる操作を有限で打ち切ったことによる誤差
- 離散化誤差: 数理モデルを多項式や漸化式に変換した際の誤差

ただし, モデル誤差 (数理モデルそのものの誤差) は扱わない.

3 q-解析学と q-特殊関数

q 類似とは, ある量, 関数または等式にパラメータ q を入れて一般化したものであり, q 類似は $q \to 1$ としたとき元の量, 関数または等式に戻る. 例えば, 自然数 n に対して

$$\frac{1-q^n}{1-q} = 1 + q + q^2 + \cdots + q^{n-1}$$

を考える時,

$$\lim_{q\to 1} \frac{1-q^n}{1-q} = \lim_{q\to 1}(1 + q + q^2 + \cdots + q^{n-1}) = n$$

となるので, $\frac{1-q^n}{1-q}$ は自然数 n の q 類似である. よって, 自然数 n の q 類似を

$$[n]_q := \frac{1-q^n}{1-q} \tag{1}$$

と書く. これを用いて, q-階乗

$$[n]_q! := [n]_q [n-1]_q \cdots [2]_q [1]_q \tag{2}$$

や, q-二項係数

$$\begin{bmatrix} n \\ k \end{bmatrix}_q := \frac{[n]_q [n-1]_q \cdots [n-k]_q}{[k]_q} \tag{3}$$

を定めることができる. ここで, $n, N \in \mathbb{N}$ に対して, $\begin{bmatrix} N+n \\ n \end{bmatrix}_q$ を q について展開して,

$$\begin{bmatrix} N+n \\ n \end{bmatrix}_q = \sum_k c(n, N, k) q^k \tag{4}$$

と表わすと, 係数 $c(n, N, k)$ は, "$k \in \mathbb{N}$ を高々n 個の N 以下の自然数の和で表す場合の数" という組み合わせ論的な意味合いを持ち, q-二項係数はある種の母関数になっている [9]. この q-二項係数について q-二項定理

$$(1+z)(1+zq)\cdots(1+zq^{n-1}) = \sum_{k=0}^{n} q^{\frac{k(k-1)}{2}} \begin{bmatrix} n \\ k \end{bmatrix}_q z^k \tag{5}$$

が成り立つ. これらは組み合せ論において頻繁に登場する.

q-階乗や q-二項係数を用いるといろいろな関数の q 類似を考えることができる. 例えば, q-指数関数

$$e_q(z) = \sum_{k=0}^{\infty} \frac{z^k}{[k]_q!} \tag{6}$$

や q-三角関数

$$\cos_q(z) = \sum_{k=0}^{\infty}(-1)^k \frac{z^{2k}}{[2k]_q!} \tag{7}$$

$$\sin_q(z) = \sum_{k=0}^{\infty}(-1)^k \frac{z^{2k+1}}{[2k+1]_q!} \tag{8}$$

がある. 初等関数の q 類似以外にも, 特殊関数の q 類似である q-特殊関数も考えることができる. これらの関数の q 類似たちを舞台にして q-微分 (Jackson 微分)

$$D_q f(x) := \frac{f(qx) - f(x)}{qx - x} \tag{9}$$

や q-積分 (Jackson 積分)

$$\int_0^1 f(x)\mathrm{d}_q x := (1-q)x\sum_{k=0}^{\infty} q^k f(q^k x) \tag{10}$$

を導入し, 関数の q 類似の性質を調べるのが q-解析学である [5, 6]. q-特殊関数は次の q-Pochhammer 記号

$$(z;q)_n := \prod_{k=0}^{n-1}(1-zq^k), \quad (z;q)_\infty := \lim_{n\to\infty}(z;q)_n, \quad z\in\mathbb{C}, \quad |q|<1 \tag{11}$$

と超幾何関数の q 類似である q-超幾何関数

$$ {}_r\phi_s(\alpha_1,\cdots,\alpha_r;\beta_1,\cdots,\beta_s;q,z) := \sum_{n=0}^{\infty} \frac{\prod_{i=1}^{r}(\alpha_i;q)_n \left[(-1)^n q^{\frac{n(n-1)}{2}}\right]^l z^n}{\prod_{j=1}^{s}(\beta_j;q)_n (q;q)_n}, \quad l = 1+s-r. \tag{12}$$

を用いて定義されることが多い. 通常の Pochhammer 記号では $(z)_\infty$ は発散してしまうため定義できないが, $(z;q)_\infty$ は定義できる. 特殊関数の q 類似では q 類似する前にはなかったものも現れるが, そのことが q-特殊関数の世界を豊穣にしているのである. q-Pochhammer 記号と q-超幾何関数を用いて定義される q-特殊関数として Bessel 関数の q 類似である

$$J_\nu^{(1)}(x;q) = \frac{(q^{\nu+1};q)_\infty}{(q;q)_\infty}\left(\frac{x}{2}\right)^\nu {}_2\phi_1\left(0,0;q^{\nu+1};q,-\frac{x^2}{4}\right),\ (|x|<2), \tag{13}$$

$$J_\nu^{(2)}(x;q) = \frac{(q^{\nu+1};q)_\infty}{(q;q)_\infty}\left(\frac{x}{2}\right)^\nu {}_0\phi_1\left(-;q^{\nu+1};q,-\frac{q^{\nu+1}x^2}{4}\right),\ (x\in\mathbb{C}), \tag{14}$$

$$J_\nu^{(3)}(x;q) = \frac{(q^{\nu+1};q)_\infty}{(q;q)_\infty}x^\nu\ {}_1\phi_1(0;q^{\nu+1};q,qx^2),\ (x\in\mathbb{C}). \tag{15}$$

が挙げられる. 上から順に Jackson の第 1 種, 第 2 種 q-Bessel 関数, Hahn-Exton の q-Bessel 関数とよばれている.

q-特殊関数は Euler による自然数の分割に関する研究などで初めて現れ [8, 9], 19 世紀から Jacobi らによって q-解析学の観点から研究されるようになった. これらの時代には q-特殊関数の数学的背景は不明であったが, 1980 年代に Drinfeld-Jimbo によって量子群の理論が導入され [10, 11, 12], q-特殊関数と q-解析学の本質が解明された. その後, 直交多項式を体系的にとらえる図式である Askey scheme [13] 等, q-特殊関数を統一的に扱う枠組みが出

来上がった. q-解析学は様々な関数の q 類似の性質を理解するための道具であるだけでなく, ”q の世界” 以外の場所（例えば数理物理）でもしばしば登場し有益な結果をもたらしてくれるのである.

q-gamma 関数は Jackson によって提案された gamma 関数の q 類似であり, 以下のように定義される [5, 6, 7]:

$$\Gamma_q(x) := (1-q)^{1-x}\frac{(q;q)_\infty}{(q^x;q)_\infty}, \quad 0<q<1. \tag{16}$$

q-gamma 関数は関数方程式

$$\Gamma_q(x+1) = [x]_q\Gamma_q(x), \quad \Gamma_q(1)=1 \tag{17}$$

を満たす. q-gamma 関数の定義から, q-Pochhammer 記号 $(a;q)_\infty$ を精度保証付き数値計算できれば, q-gamma 関数の精度保証付き数値計算ができるということが分かる.

4 q-gamma 関数の精度保証付き数値計算

4.1 乗積公式による q-gamma 関数の精度保証付き数値計算

q-Pochhammer 記号 $(z;q)_\infty$ の精度保証付き数値計算法として我々は以下の 2 つを開発した:

- 先行研究: $(z;q)_\infty$ を交代級数に変換して打切り誤差を評価する方法 [14]
- 今回提案する手法: $(z;q)_\infty$ の近似と誤差半径を利用する方法

4.1.1 $(z;q)_\infty$ に関する先行研究

q-Pochhammer 記号については次が成り立つ.

定理 1. (Karpelevich) [15] $|z|<1$ のとき次が成り立つ:

$$\frac{(z;q)_\infty}{(q;q)_\infty} = 1\Big/\sum_{n=0}^{\infty}\frac{(-1)^n q^{n(n+1)/2}}{(q;q)_n(1-zq^n)}. \tag{18}$$

定理 2. [14]

$$(q;q)_\infty = \sum_{n=0}^{\infty}(-1)^n(1-q^{2n+1})q^{n(3n+1)/2}. \tag{19}$$

定理 2 は Euler による五角数定理 [5, 6] の別表現である.

$-1<z<1$ に制限したうえで $\frac{(z;q)_\infty}{(q;q)_\infty}$ と $(q;q)_\infty$ を交代級数に変形した後, 次の定理により打切り誤差を評価する.

定理 3. [16] 数列 $\{p_n\}_{n=0}^\infty$ が $\lim_{n\to\infty} p_n = 0$ を満たす単調減少な正数列ならば, 交代級数 $\sum_{n=0}^\infty(-1)^n p_n$ は収束する.

系 4. [16] $s := \sum_{n=0}^\infty(-1)^n p_n$, $s_N := \sum_{n=0}^N(-1)^n p_n$ とおくと, $|s-s_N| \le p_{N+1}$ が成り立つ.

こうして $-1<z<1$ の時に限り $(z;q)_\infty$ を精度保証付き数値計算できるようになった. 提案手法では $z\in\mathbb{C}$ のときに $(z;q)_\infty$ を精度保証付き数値計算する方法を扱う.

4.1.2 $(z;q)_\infty$ に関する提案手法

$(z;q)_\infty$ の精度保証付き数値計算では以下の定理 5, 6 が使える.

定理 5. [17]
$z \in \mathbb{C}$, $0<q<1$ とする. ある $n \in \mathbb{N}$ に対して $0 < \frac{|z|q^n}{1-q} < \frac{1}{2}$ であるとき, 以下が成り立つ:

$$\frac{(z;q)_\infty}{(z;q)_n} = (zq^n;q)_\infty = 1 + r(z;n), \quad |r(z;n)| \le \frac{2|z|q^n}{1-q}. \tag{20}$$

定理 5 は $(z;q)_n \times$ (中心 1, 半径 $r(z;n)$ の近傍) という形の評価である.
定理 6 は $(z;q)_\infty$ の近似と誤差半径を与える定理である.

定理 6. [18] $z \in \mathbb{C}$, $0<q<1$ とし, $N \in \mathbb{N}$ を十分大なる数とすると以下が成り立つ:

$$(z;q)_\infty - T_{m-1,N}(z) = (z;q)_N \sum_{j=0}^{\infty} d_{m+j}(zq^N)^{m+j}, \tag{21}$$

$$d_k = \frac{qq^2\cdots q^{k-1}(1-q)^k}{(1-q)(1-q^2)\cdots(1-q^k)} \le q^{k(k-1)/2}, \tag{22}$$

$$T_{m,N}(z) = (z;q)_N \sum_{k=0}^{m} d_k(zq^N)^k. \tag{23}$$

系 7. 定理 6 で $|zq^N| < 1$ を仮定すれば以下が成り立つ:

$$|(z;q)_\infty - T_{m-1,N}(z)| \le \left| (z;q)_N \frac{|zq^N|^m q^{m(m-1)/2}}{1-|zq^N|} \right|. \tag{24}$$

数値実験を通して, 定理 5, 6 を使ったときの違いを見てみよう. なお実験環境は OS: Ubuntu14.04LTS, コンパイラー: gcc version 4.8.4, CPU: Intel Xeon(R) CPU E3-1241 v3 @ 3.50GHz × 8, メモリ: 15.6GB の端末であり, kv 0.4.42[2] を使用している.

$q=0.1, z=15$ で実験を行い, 以下の結果を得た.
精度保証の結果 (定理 5): [5.8509835632983984,5.8509835632985006]
精度保証の結果 (定理 6): [5.8509835632983975,5.8509835632985015]
この場合は 2 つの手法に大きな違いはない. 今度は z を変えずに $q=0.9$ としてみる.
精度保証の結果 (定理 5): [411.89219387077093,418.50918051346912]
精度保証の結果 (定理 6): [415.2006871920434,415.2006871919541]
定理 5 を使う方が区間幅が広がってしまう. これは定理 5 で $q \to 1$ としたとき, 定理の仮定が満たされるように $n \to \infty$ となって計算量が増えてしまうためである. $q \to 1$ のときは定理 6 を使って計算を行うことにする.

4.1.3 q-gamma 関数の乗積公式

定理 6 より $q \to 1$ としたときに q-Pochhammer 記号を精度保証付き数値計算できるようになったが, $q \to 1, x \to \infty$ としたときに q-gamma 関数が精度保証付き数値計算できるようになったわけではない. $(1-q)^{1-x}$ または $\frac{1}{(q^z;q)_\infty}$ でゼロ除算が発生してしまうからである. この場合の対策には q-gamma 関数の乗積公式を用いる.

定理 8. [5]

$$\begin{aligned}\Gamma_q(nx)\Gamma_r(1/n)\Gamma_r(2/n)\cdots\Gamma_r((n-1)/n) \\ = \left(\frac{1-q^n}{1-q}\right)^{nx-1} \Gamma_r(x)\Gamma_r(x+1/n)\cdots\Gamma_r(x+(n-1)/n), \quad r=q^n.\end{aligned} \tag{25}$$

q^n とすることで q を 1 から遠ざけて, x/n とすることで x を小さくしている.

定理 8 により q-gamma 関数を精度保証付き数値計算するプログラムを C++で自作し, 改良前の計算結果を比較した. $z = -100 + 100\mathrm{i}, q = 0.99$ として実験した.
改良前: ゼロ除算発生
改良後 ($n = 20$): ([3.520226006915545e-275,3.5245441358982552e-275])+
([-1.741771549163063e-274,-1.7413396208989371e-274])i
計算に成功していることから, 改良が成功しているといえる.

4.2 積分による q-gamma 関数の精度保証付き数値計算

ここでは q-gamma 関数の持つ次の積分表示 [19] を扱う:

$$\frac{1}{\Gamma_q(z)} = \frac{\sin(\pi z)}{\pi} \int_0^\infty \frac{t^{-z}\mathrm{d}t}{(-t(1-q);q)_\infty}. \tag{26}$$

積分による q-gamma 関数の精度保証付き数値計算を行う前に, 先行研究である gamma 関数の精度保証付き数値計算法について見ていく. 積分による gamma 関数の精度保証付き数値計算は次の流れで行われた [20]:

$$\Gamma(z) = \int_0^\infty x^{z-1} e^{-x}\, dx.$$

積分表示を実部と虚部に分割する.

↓

積分区間を $[0,1], [1,T], [T,\infty]$ の 3 つに分けて積分計算を行う ($T > 1$).

↓

$[0,1]$ では中身を級数に直して項別積分し, $[1,T]$ では kv ライブラリ [2] の精度保証付き数値積分パッケージを使う.

↓

$[T,\infty]$ では厳密に積分値の求められる関数で被積分関数を評価する.

この方法を参考に, 積分による q-gamma 関数の精度保証付き数値計算を行う.

q-gamma 関数の積分表示を実部と虚部に分けると次のようになる ($z = x + \mathrm{i}y$):

$$\frac{1}{\Gamma_q(z)} = \frac{\sin(\pi z)}{\pi} \left(\int_0^\infty \frac{t^{-x}\cos(y\log t)\mathrm{d}t}{(-t(1-q);q)_\infty} - \mathrm{i} \int_0^\infty \frac{t^{-x}\sin(y\log t)\mathrm{d}t}{(-t(1-q);q)_\infty} \right). \tag{27}$$

実部と虚部で積分区間を $[0,1], [1,T], [T,\infty]$ の 3 つに分けて積分計算を行う ($T > 1$). ここからは実部の積分計算について見ていく (虚部も同様). また, $1 \le x \le 2$ を仮定する. 等式

$$\Gamma_q(z+1) = [z]_q \, \Gamma_q(z) \tag{28}$$

より, 他の範囲にある x を精度保証付き数値計算できるため $1 \le x \le 2$ を仮定してこの範囲での評価を考えれば良い.

[0,1] では次の定理を用いる.

定理 9. (Euler)[5, 6]

$$\sum_{n=0}^\infty \frac{z^n}{(q;q)_n} = \frac{1}{(z;q)_\infty}, \quad |z| < 1. \tag{29}$$

定理 9 を使って項別積分, 部分積分すると,

$$\int_0^1 \frac{t^{-x}\cos(y\log t)}{(-t(1-q);q)_\infty}\mathrm{d}t = \sum_{n=0}^{\infty}\frac{(-(1-q))^n}{(q;q)_n}\frac{n-x+1}{(n-x+1)^2+y^2} \tag{30}$$

となる. $T(n)=\frac{(-(1-q))^n}{(q;q)_n}\frac{n-x+1}{(n-x+1)^2+y^2}$ とおくと公比は $n\geq N$ のとき,

$$\begin{aligned} |T(n+1)/T(n)| &= \frac{1-q}{1-q^{n+1}}\frac{n-x+2}{n-x+1}\frac{(n-x+1)^2+y^2}{(n-x+2)^2+y^2} \\ &\leq \frac{1-q}{1-q^{N+1}}\left(1+\frac{1}{N-x+1}\right) =: D \end{aligned} \tag{31}$$

となる. $D<1$ のとき,

$$\left|\sum_{n=N}^{\infty}T(n)\right| \leq (\text{初項 } T(N),\ \text{公比 } D \text{ の等比数列の和}) \leq \frac{T(N)}{1-D}$$

となる. こうして級数の打ち切り誤差を評価した.

$[1,T]$ で積分する際は被積分関数:

$$\frac{t^{-x}\cos(y\log t)}{(-t(1-q);q)_\infty} \tag{32}$$

にある q-Pochhammer 記号を積分しやすい形に変形する. 変形には次の定理を使う.

定理 10. [17] 定理 5 と同じ仮定で以下が成り立つ:

$$\frac{(z;q)_n}{(z;q)_\infty} = \frac{1}{(zq^n;q)_\infty} = 1+r(z;n), \quad |r(z;n)| \leq \frac{2|z|q^n}{1-q}. \tag{33}$$

被積分関数は

$$\frac{t^{-x}\cos(y\log t)}{(-t(1-q);q)_\infty} = \frac{t^{-x}\cos(y\log t)}{(-t(1-q);q)_n}\left(1\pm\frac{2|t(1-q)|q^n}{1-q}\right) \tag{34}$$

と変形できる. 変形後に現れる $n\in\mathbb{N}$ は定理 9 の仮定を満たすように取る. 変形後の積分には C++による精度保証付き数値計算ライブラリである"kv ライブラリ"[2] に組み込まれている精度保証付き数値積分パッケージを使う.

$x\in[1,2]$ と定理 5 より $[T,\infty]$ での被積分関数の評価は次のようになる:

$$\frac{t^{-x}\cos(y\log t)}{(-t(1-q);q)_\infty} \in \left[-\frac{1-2tq^n/(1-q)}{t^2(1+t(1-q))^n}, \frac{1+2tq^n/(1-q)}{t(1+t(1-q)q^{n-1})^n}\right]. \tag{35}$$

上限と下限をそれぞれ不定積分すると[1]次のようになる:

$$\begin{aligned} &\int \frac{1+2tq^n/(1-q)}{t(1+t(1-q)q^{n-1})^n}\mathrm{d}t \\ &= \left(1-(q-1)tq^{n-1}\right)^{-n}\left(\frac{2\left(tq^{n+1}-tq^n-q\right)}{(n-1)(q-1)^2} - \frac{\left(\frac{q^{1-n}}{t-qt}+1\right)^n {}_2F_1\left(n,n;n+1;\frac{q^{1-n}}{(q-1)t}\right)}{n}\right), \end{aligned} \tag{36}$$

$$\begin{aligned} &\int -\frac{1-2tq^n/(1-q)}{t^2(1+t(1-q))^n}\mathrm{d}t \\ &= (-qt+t+1)^{-n}\left(\frac{1}{t-qt}+1\right)^n\left(2(n+1)tq^n\,{}_2F_1\left(n,n;n+1;\frac{1}{(q-1)t}\right)\right. \\ &\quad \left.+n(q-1){}_2F_1\left(n,n+1;n+2;\frac{1}{(q-1)t}\right)\right)/\{n(n+1)(q-1)t\}. \end{aligned} \tag{37}$$

[1]不定積分には Mathematica の `Integrate` を用いた.

4.3 数値実験 (積分)

q-gamma 関数を積分によって精度保証付き数値計算するプログラムを C++で自作し, 数値実験を行った. 実験には C++による精度保証付き数値計算ライブラリである"kv ライブラリ"[2] を使用している. $x = 1.2 + \mathrm{i}, q = 0.1$ として実験を行い, 以下の結果を得た.
精度保証の結果 (積分):([0.48702750125220128,2.2700419350688224])+
([-0.46939808387927973,1.1606835821327298])i
得られた結果の区間幅が広がっている. 被積分関数をより厳しく評価していく必要がある.

参考文献

[1] 大石進一 (2000): 精度保証付き数値計算, コロナ社.

[2] M. Kashiwagi: kv - a C++ Library for Verified Numerical Computation, `http://verifiedby.me/kv/index-e.html`

[3] 大石進一 (2008): 電子情報通信学会技術研究報告. NLP, 非線形問題, 108, 55-57.

[4] N. Yamamoto and N. Matsuda (2005): Trans. Jap. Soc. Indust. Appl. Math., 15, 347-359.

[5] G. Gasper and M. Rahman (2004): Basic Hypergeometric Series, Cambridge University Press.

[6] G. Andrews, A. Askey and R. Roy (1999): Special Functions, Cambridge University Press.

[7] R. Askey (1978): Appl. Anal., 8, 125-141.

[8] 堀田良之, 渡辺敬一, 庄司俊明, 三町勝久 (2004): 群論の進化, 代数学百科 I, 朝倉書店.

[9] G. Andrews and K. Eriksson (2004): Integer Partitions, Cambridge University Press.

[10] V. G. Drinfeld (1986): Proc. Intern. Congr. Math., 798-820.

[11] M. Jimbo (1985): Lett. Math. Phys. , 10, 63-69.

[12] M. Jimbo (1986): Lett. Math. Phys. , 11, 247-252.

[13] R. Koekoek and R. Swarttouw (1996): arXiv preprint math/9602214.

[14] 金泉 大介, 丸野 健一: q-Bessel 関数の精度保証付き数値計算, 日本応用数理学会第 13 回研究部会連合発表会, 2017 年 3 月 6 日.

[15] M. Olshanetsky and V. Rogov (1995): arXiv preprint q-alg/9509013.

[16] 杉浦光夫 (1980): 解析入門 I, 東京大学出版会.

[17] R. Zhang (2008): Adv. Math., 217, 1588-1613.

[18] B. Gabutti and G. Allasia (2008): Numer. Algorithms, 49, 159-168.

[19] M. Ismail (1981): SIAM J. Math. Anal., 12, 454-468.

[20] 橋本崇希, 柏木雅英: 複素 Gamma 関数の精度保証付き数値計算, 日本応用数理学会年会, 武蔵野大学, 2017 年 9 月 6-8 日.