The Calculation and Application of the Partial Derivatives of the Generalized Hypergeometric Function

Aijuan Li, Fen Qin and Huizeng Qin

Abstract—**In this paper, a algorithm of the partial derivatives of the generalized hypergeometric function ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$ is obtained, where $\tilde{a}=\{a_1,a_2,\ldots,a_p\},\tilde{b}=\{b_1,b_2,\ldots,b_q\}$. Moreover, we compare some algorithms of calculating the partial derivatives of ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$. Numerical examples show the algorithm given in this paper improves the precision and accelerates the calculation of partial derivatives of the generalized hypergeometric function. Furthermore, we obtain some applications of the partial derivatives of the generalized hypergeometric function in calculating improper integrals and the derivative of the special function, such as the derivatives with respect to the order of Bessel function etc. The accuracy and the speed of calculating the improper integrals are improved by numerical examples.**

Index Terms—**hypergeometric function; Pochhammer symbol; improper integral; partial derivatives.**

I. Introduction

THE generalized hypergeometric function ${}_pF_q\left(a_1,a_2,\ldots,a_p;b_1,b_2,\ldots,b_q;z\right)$ can be defined by the following power series in [1]

$$\begin{aligned}&{}_pF_q\left(a_1,a_2,\ldots,a_p;b_1,b_2,\ldots,b_q;z\right)\\&=\sum_{k=0}^{\infty}A_k\frac{z^k}{k!}=\sum_{k=0}^{\infty}\frac{(a_1)_k(a_2)_k\cdots(a_p)_k z^k}{(b_1)_k(b_2)_k\cdots(b_q)_k k!},\end{aligned}\tag{1}$$

where $(x)_n$ is a Pochhammer symbol, *i.e.*, $(x)_n=x(x+1)\cdots(x+n-1)$, A_k is the abbreviation of $\frac{(a_1)_k(a_2)_k\cdots(a_p)_k}{(b_1)_k(b_2)_k\cdots(b_q)_k}$ and $b_j\neq 0,-1,-2,\ldots(j=1,2,\ldots,q)$. For convenience, ${}_pF_q\left(a_1,a_2,\ldots,a_p;b_1,b_2,\ldots,b_q;z\right)$ is written as ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$, where $\tilde{a}=\{a_1,a_2,\ldots,a_p\}$ and $\tilde{b}=\{b_1,b_2,\ldots,b_q\}$. The generalized hypergeometric function often occurs in a wide variety of problems in theoretical physics, applied mathematics, statistics and engineering sciences in [2-5]. For example, in quantum mechanics, the solution of the Schrödinger equation for some systems is expressed in terms of function ${}_2F_1(a,b;c;z)$, when solving the Pöschl-Teller, Wood-Saxon or Hulthén potentials in [3].

The *n*th derivatives with respect to the variable z of ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$ have been expressed in a compact form in [1]. However, the derivatives of ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$ with respect to the first or second parameter have been less discussed because the representation formulation are relatively difficult and complicated. In some cases, the parameters $a_i,b_i(i=1,2,\ldots,p;j=1,2,\ldots,q)$ of ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$ have played an important role in physical problems rather than the variable z. Thus, it is essential to study the functions ${}_pF_q\left(\tilde{a};\tilde{b};z\right)$ as a function of a_i or $b_j(i=1,2,\ldots,p;j=1,2,\ldots,q)$, rather than z in [2]. For the generalized hypergeometric function and many other special functions, people pay more attention to the higher order partial derivatives with respect to the parameters in [2,4-9]. For example, the authors considered some properties of the partial derivatives for ${}_1F_1\left(a;b;z\right)$ and ${}_2F_1\left(a;b;z\right)$ in [4,5]. Moreover, the authors extended these results of [4,5] in [2]. In the following, the authors give some results of [2]. Let

$$G_{a_i}^{(n)}=\frac{\partial^n}{\partial a_i^n}\left[{}_pF_q\left(\tilde{a};\tilde{b};z\right)\right],H_{b_j}^{(n)}=\frac{\partial^n}{\partial b_j^n}\left[{}_pF_q\left(\tilde{a};\tilde{b};z\right)\right],\quad i=1,2,\ldots,p;\quad j=1,2,\ldots,q.\tag{2}$$

there are the following results:

$$G_{a_1}^{(n)}=\frac{A_nz^n}{(a_1)_n}\times{}_p\Theta_q^{(n)}\left(\begin{matrix}1,1,\ldots,1|a_1,a_1+1,\ldots,a_1+n,\ldots,a_p+n\\a_1+1,\ldots,a_1+n|n+1,b_1+n,\ldots,b_q+n\end{matrix};z,\ldots,z\right)\tag{3}$$

and

$$H_{b_1}^{(n)}=\frac{n!(-1)^nA_1z}{b_1^n}\times{}_p\Theta_q^{(n)}\left(\begin{matrix}1,1,\ldots,1|b_1,b_1,\ldots,b_1,a_1+1,\ldots,a_p+1\\b_1+1,\ldots,b_1+1|2,b_1+1,\ldots,b_q+1\end{matrix};z,\ldots,z\right),\tag{4}$$

where

$$\begin{aligned}&{}_p\Theta_q^{(n)}\left(\begin{matrix}\alpha_1,\alpha_2,\ldots,\alpha_{n+1}|\beta_1,\beta_2,\ldots,\beta_{n+p}\\\gamma_1,\gamma_2,\ldots,\gamma_n|\delta_1,\delta_2,\ldots,\delta_{q+1}\end{matrix};x_1,x_2,\ldots,x_{n+1}\right)\\&=\sum_{m_1=0}^{\infty}\cdots\sum_{m_{n+1}=0}^{\infty}(\alpha_1)_{m_1}(\alpha_2)_{m_2}\cdots(\alpha_{n+1})_{m_{n+1}}\\&\times\frac{(\beta_1)_{m_1}(\beta_2)_{m_1+m_2}\cdots(\beta_n)_{m_1+m_2+\ldots+m_n}}{(\gamma_1)_{m_1}(\gamma_2)_{m_1+m_2}\cdots(\gamma_n)_{m_1+m_2+\ldots+m_n}}\\&\times\frac{(\beta_{n+1})_{m_1+m_2+\ldots+m_{n+1}}\cdots(\beta_{n+p})_{m_1+m_2+\ldots+m_{n+1}}x_1^{m_1}\cdots x_{n+1}^{m_{n+1}}}{(\delta_1)_{m_1+m_2+\ldots+m_{n+1}}\cdots(\delta_{q+1})_{m_1+m_2+\ldots+m_{n+1}}m_1!\ldots m_{n+1}!}.\end{aligned}\tag{5}$$

we can see that (5) is an hypergeometric function in $n+1$ variables.

Moreover, for (3)-(5), the following forms were given in

Manuscript received February 1, 2020. This work was supported by National Natural Science Foundation of China under Grant No. 61379009 and No. 61771010

Aijuan Li is with Mathematics and Statistics, Shandong University of Technology, Zibo, Shandong, 255049, P. R. China.

Fen Qin is with Raintree Systems.

Huizeng Qin is with Mathematics and Statistics, Shandong University of Technology, Zibo, Shandong, 255049, P. R. China. Huizeng Qin is the corresponding author. (e-mail: qin_hz@163.com(H.Z.Qin))

[6]

$$G_{a_1}^{(n)} = n! \sum_{m_0=0}^{\infty} \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} \cdots \times \sum_{m_n=0}^{\infty} \frac{z^{\widetilde{m}_n+n} \prod_{i=1}^{p} (a_i)_{\widetilde{m}_n+n}}{(\widetilde{m}_n+n)! \prod_{j=0}^{n-1} (a_1+\sum_{u=0}^{j} m_u+j) \prod_{j=1}^{q} (b_j)_{\widetilde{m}_n+n}} \quad (6)$$

and

$$H_{b_1}^{(n)} = (-1)^n n! \sum_{m_0=0}^{\infty} \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} \cdots \times \sum_{m_n=0}^{\infty} \frac{\prod_{i=1}^{p} (a_i)_{\widetilde{m}_n+1} z^{\widetilde{m}_n+1}}{\prod_{j=0}^{n-1} (b_1+\sum_{u=0}^{j} m_u) \prod_{j=1}^{q} (b_j)_{\widetilde{m}_n+1} (\widetilde{m}_n+1)!} \quad (7)$$

where $\widetilde{m}_n = m_0 + m_1 + m_2 + \ldots + m_n$.

From (3)-(7), we notice that if $\sum_{m_0=0}^{\infty} \sum_{m_1=0}^{\infty} \sum_{m_2=0}^{\infty} \cdots \sum_{m_n=0}^{\infty}$ is replaced by the part sum $\sum_{m_0=0}^{N} \sum_{m_1=0}^{N} \sum_{m_2=0}^{N} \cdots \sum_{m_n=0}^{N}$, then the magnitude of calculating $G_{a_1}^{(n)}$ and $H_{b_1}^{(n)}$ is on the order of N^{n+1}. Hence, from the point of view of calculation, it is inadvisable to calculate $G_{a_1}^{(n)}$ and $H_{b_1}^{(n)}$ according to (3), (4) or (6), (7). The aim of this paper is to obtain a fast algorithm for $G_{a_i}^{(n)}$ and $H_{b_j}^{(n)} (i = 1, 2, \ldots, p; j = 1, 2, \ldots, q)$.

This paper is organized as follows. In Section 2, we give some algorithms of the partial derivatives of ${}_pF_q(\tilde{a}; \tilde{b}; z)$ with respect to a_i or $b_j (i = 1, 2, \ldots, p; j = 1, 2, \ldots, q)$. In Section 3, we obtain the fast algorithms of the partial derivatives of ${}_pF_q(\tilde{a}; \tilde{b}; z)$ with respect to a_i or $b_j (i = 1, 2, \ldots, p; j = 1, 2, \ldots, q)$. Moreover, numerical examples are given. In Section 4, we obtain some applications of the partial derivatives of ${}_pF_q(\tilde{a}; \tilde{b}; z)$ in calculating improper integrals and the derivative of the special function, such as the derivatives with respect to the order of Bessel function etc. The conclusion is given in the last section of the paper.

II. Some algorithms of the partial derivatives of ${}_pF_q(\tilde{a}; \tilde{b}; z)$ with respect to a_i or b_j for $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, q$

In this section, we consider the calculation for

$$ {}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a}; \tilde{b}; z) = \frac{\partial^{m_1+\ldots+m_p+n_1+\ldots+n_q}}{\partial^{m_1} a_1 \ldots \partial^{m_p} a_p \partial^{n_1} b_1 \ldots \partial^{n_q} b_q} {}_pF_q(\tilde{a}; \tilde{b}; z). \quad (8)$$

By (1), we have

$$ {}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a}; \tilde{b}; z) = \sum_{k=\widetilde{m}}^{\infty} \frac{z^k}{k!} \prod_{i=1}^{p} A_{k,m_i}(a_i) \prod_{j=1}^{q} B_{k,n_j}(b_j), \quad (9)$$

where

$$\begin{aligned} A_{k,m}(x) &= \frac{d^m}{dx^m} (x)_k , \\ B_{k,n}(x) &= \frac{d^n}{dx^n} \frac{1}{(x)_k}, \\ \widetilde{m} &= \max\{m_i, i = 1, 2, \ldots, p\}. \end{aligned} \quad (10)$$

Note that ${}_pF_q^{(0,\ldots,0;0,\ldots,0)}(\tilde{a}; \tilde{b}; z) = {}_pF_q(\tilde{a}; \tilde{b}; z)$. According to (9), the key is to give the fast algorithm of $A_{k,m}(x), B_{k,n}(x)$.

By the following identities

$$\begin{aligned} (x)_n &= \sum_{m=0}^{n} (-1)^{n-m} \begin{bmatrix} n \\ m \end{bmatrix} x^m, \\ \frac{1}{(x)_n} &= \frac{1}{(n-1)!} \sum_{m=0}^{n-1} \binom{n-1}{m} \frac{(-1)^m}{x+m}, \end{aligned} \quad (11)$$

the following algorithms were given in [10]

$$A_{k,n}(x) = \begin{cases} (x)_k, & n = 0, \\ \sum_{m=n}^{k} (-1)^{k-m} \begin{bmatrix} k \\ m \end{bmatrix} (m-n+1)_n x^{m-n}, & n \le k, \\ 0, & n > k. \end{cases} \quad (12)$$

and

$$B_{k,n}(x) = \begin{cases} \frac{1}{(x)_k}, & n = 0, \\ \frac{(-1)^n n!}{(k-1)!} \sum_{m=0}^{k-1} \binom{k-1}{m} \frac{(-1)^m}{(x+m)^{n+1}}, & n > 0. \end{cases} \quad (13)$$

where $\begin{bmatrix} k \\ m \end{bmatrix}$ is the Stirling numbers of the first kind and $\binom{n}{m}$ is a binomial coefficient, *i.e.*, $\binom{n}{m} = \frac{n!}{m!(n-m)!}$.

By (9), (12) and (13), we have

$$\begin{aligned} G_{a_1}^{(n)} &= \sum_{k=n}^{\infty} \frac{(a_2)_k \cdots (a_p)_k z^k}{(b_1)_k (b_2)_k \cdots (b_q)_k k!} \times \sum_{m=n}^{k} (-1)^{k-m} \begin{bmatrix} k \\ m \end{bmatrix} (m-n+1)_n a_1^{m-n} \\ &= n! \sum_{k=n}^{\infty} \frac{(a_2)_k \cdots (a_p)_k z^k}{(b_1)_k (b_2)_k \cdots (b_q)_k k!} \times \sum_{m=n}^{k} (-1)^{k-m} \begin{bmatrix} k \\ m \end{bmatrix} \binom{m}{n} a_1^{m-n}, \end{aligned} \quad (14)$$

and

$$H_{b_1}^{(n)} = (-1)^n n! \sum_{k=1}^{\infty} \frac{(a_1)_k (a_2)_k \cdots (a_p)_k z^k}{(b_2)_k \cdots (b_q)_k (k-1)!} \times \sum_{m=0}^{k-1} \binom{k-1}{m} \frac{(-1)^m}{(b_1+m)^{n+1}}. \quad (15)$$

If $G_{a_1}^{(n)}$ and $H_{b_1}^{(n)}$ are calculated according to (14) and (15), then the amount of computation is on the order of N^2. Comparing to the results of (3),(4) or (6),(7), there is a substantial improvement from the point of view of calculation.

For $A_{k,m}(x), B_{k,n}(x)$ we can use the following method to calculate them.

Use the following results of [12,13]

$$\frac{d^r}{dx^r} e^{f(x)} = e^{f(x)} Y_r \left(f^{(1)}(x), f^{(2)}(x), \ldots, f^{(r)}(x) \right), \quad (16)$$

where the (exponential) complete Bell polynomials may be defined by $Y_0 = 1$ and for

$$Y_r (x_1, x_2, \ldots, x_r) = \sum_{\pi(r)} \frac{r!}{k_1! k_2! \ldots k_r!} \left(\frac{x_1}{1!}\right)^{k_1} \left(\frac{x_2}{2!}\right)^{k_2} \cdots \left(\frac{x_r}{r!}\right)^{k_r}, \quad (17)$$

where the sum is taken over all partitions $\pi(r)$ of r, i.e. over all sets of integers k_j such that

$$k_1 + 2k_2 + \ldots + r k_r = r. \quad (18)$$

Let $f(x) = \ln(x)_k$ and $f(x) = \ln \frac{1}{(x)_k} = -\ln(x)_k$, respectively, by (16) and the following formula

$$(\ln(x)_k)^{(n)} = H_{k,n}(x), \quad (19)$$

where

$$H_{k,i}(x) = (-1)^{i-1}(i-1)!\sum_{l=0}^{k-1}\frac{1}{(x+l)^i}, i=1,2,\ldots,n, \tag{20}$$

we have

$$A_{k,n}(x) = (x)_k Y_n\left(H_{k,1}(x), H_{k,2}(x), \ldots, H_{k,n}(x)\right) \tag{21}$$

and

$$B_{k,n}(x) = \frac{1}{(x)_k} Y_n\left(-H_{k,1}(x), -H_{k,2}(x), \ldots, -H_{k,n}(x)\right), \tag{22}$$

so

$${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z) = \sum_{k=\widetilde{m}}^{\infty} Y_{k,n,p,q}(\tilde{a};\tilde{b})\frac{A_k z^k}{k!}, \tag{23}$$

where

$$\begin{aligned} &Y_{k,n,p,q}(\tilde{a};\tilde{b}) \\ &= \prod_{i=1}^{p} Y_{m_i}\left(H_{k,1}(a_i), H_{k,2}(a_i), \ldots, H_{k,m_i}(a_i)\right) \\ &\cdot \prod_{j=1}^{q} Y_{n_j}\left(-H_{k,1}(b_j), -H_{k,2}(b_j), \ldots, -H_{k,n_j}(b_j)\right). \end{aligned} \tag{24}$$

According to (23), the calculation of $Y_{k,n,p,q}(\tilde{a};\tilde{b})$ is increased in calculating the ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$. Note that $H_{k+1,i}(x) = H_{k,i}(x) + \frac{(-1)^{i-1}(i-1)!}{(x+k)^i}$, the amounts of $H_{k,1}(a_i), H_{k,2}(a_i), \ldots, H_{k,m_i}(a_i)(i=1,2,\ldots,p)$ and $H_{k,1}(b_j), H_{k,2}(b_j), \ldots, H_{k,n_j}(b_j)$ $(j=1,2,\ldots,n)$ are the order of k. However, with the increase of n, the calculation of $Y_n\left(H_{k,1}(x), H_{k,2}(x), \ldots, H_{k,n}(x)\right)$ is quite time-consuming. Moreover, with the increase of $\widetilde{m}$, the amount of $Y_{k,n,p,q}(\tilde{a};\tilde{b})$ is more time-consuming. The algorithm (23) of calculating ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$ is not the best algorithm. Therefore, we need to obtain the fast algorithm of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$.

In the following, we obtain some recursive formulas of $A_{k,n}(x)$ and $B_{k,n}(x)$ by $(x)_k = (x+k-1)(x)_{k-1}$.

Lemma 2.1 For the integer $n \geq 0$, $A_{k,n}(x)$ and $B_{k,n}(x)$ are calculated by the following recursive formulas:

$$A_{k,n}(x) = (x+k-1)A_{k-1,n}(x) + nA_{k-1,n-1}(x), \tag{25}$$

and

$$B_{k,n}(x) = \frac{B_{k-1,n}(x) - nB_{k,n-1}(x)}{x+k-1}, \tag{26}$$

for $k=1,2,\ldots$, where

$$\begin{gathered} A_{k,0}(x) = (x)_k, \\ A_{k,n}(x) = \begin{cases} 0, k<n \\ n!, k=n \end{cases}, B_{k,0}(x) = \frac{1}{(x)_k}, \\ A_{0,0}(x)=1,\ B_{0,0}(x)=1, \\ A_{0,n}(x)=0,\ B_{0,n}(x)=0,\ n>0. \end{gathered} \tag{27}$$

Obviously, (10) is calculated according to (25)-(27), amount of calculation for (9) is on the order of N. For the algorithm (25), (27) and the algorithm (21), (22) we give the following numerical results in Mathematica.

$H[hx\_, hk\_, hi\_] := -(-1)\hat{}hi(hi-1)!\sum_{ll=0}^{hk-1}\frac{1}{(hx+ll)^{hi}}(**)$
$dn=16; x=11/7; k=46; Prec=32;$

$Timing[N[Pochhammer[x,k]BellY[Table$

$[\{1, H[x,k,kk]\},\{kk,dn\}]], Prec]](*(21)*)$

$Timing[N[D[Pochhammer[xx,k],\{xx,dn\}]$
$/.xx->x, Prec]](*Symbol\ derivation*)$

$Timing[N[pochhammerDs[x,k,dn]$
$[[k+1, dn+1]], Prec]](*(25)*)$

$Timing[N[Pochhammer[x,k]BellY[Table$
$[\{1, H[x,k,kk]\},\{kk,dn\}]], Prec]](*(22)*)$

$Timing[N[D[Pochhammer[xx,k],\{xx,dn\}]$
$/.xx->x, Prec]](*Symbol\ derivation*)$
$Timing[N[pochhammerDs[x,k,dn]$
$[[k+1, dn+1]], Prec]](*(26)*)$
$\{0.046875, 1.4632950207573880737267438563108*10^{64}\}$
$\{0.187500, 1.4632950207573880737267438563108*10^{64}\}$
$\{0.015625, 1.4632950207573880737267438563108*10^{64}\}$
$\{0.046875, 3.1043057295614322605242801522030*10^{-34}\}$
$\{18.625000, 3.1043057295614322605242801522030*10^{-34}\}$
$\{0.015625, 3.1043057295614322605242801522030*10^{-34}\}$

Numerical calculation show that the algorithms of (25), (26) are superior to the algorithm of (21),(22).

We consider the computation of $\prod_{i=1}^{p} A_{k,m_i}(a_i)\prod_{j=1}^{q} B_{k,n_j}(b_j)$. If only multiplication and division are considered, then the calculation of the partial sum of series (1) is $(p+1+\lambda(q+1))N$, where λ is the time-consuming ratio of division and multiplication. By (23) and (24), number of calculations of the partial sum of series (9) is

$$N_{|m|,|n|} = (p+1+|m|+\lambda(q+1+|n|))N, \tag{28}$$

where $|m| = \sum_{k=1}^{p} m_k, |n| = \sum_{k=1}^{q} n_k$. In this way, there is the same amount of calculation as the algorithm of the generalized hypergeometric function ${}_pF_q(\tilde{a};\tilde{b};z)$ itself (see Table III and Table IV in section 3).

In the following, we consider a fast algorithm of the partial derivatives of ${}_pF_q(\tilde{a};\tilde{b};z)$ with respect to a_i or b_j for $i=1,2,\ldots,p$ and $j=1,2,\ldots,q$.

III. The fast algorithm of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$

By (9), we notice that the key to the fast computation of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$ is the fast calculation of $A_{k,m_i}(a_i)$ and $B_{k,n_j}(b_j)$ for $i=1,2,\ldots,p$ and $j=1,2,\ldots,q$.

According to (16)-(18), we give the following algorithm.

Algorithm of $A_{k,m}(x)$ and $B_{k,n}(y)$ For non-negative integers m,n and $x \neq 0,-1,-2,\ldots$, there are the following recursive formulas:

$$\begin{aligned} &A_{0,0}(x)=1, A_{0,l}(x)=0, \quad l=1,2,\ldots,m, \\ &A_{k,0}(x)=(x)_k, \\ &A_{k,l}(x)=(x+k-1)A_{k-1,l}(x)+lA_{k-1,l-1}(x), \\ &l=1,2,\ldots,m. \end{aligned} \tag{29}$$

and

$$\begin{array}{l} B_{0,0}(y)=1, \quad B_{0,l}(y)=0, l=1,2,\ldots n, \\ B_{k,0}(y)=\frac{1}{(y)_k}, \\ B_{k,l}(y)=\frac{1}{y+k-1}\left(B_{k-1,l}(y)-lB_{k,l-1}(y)\right), \\ l=1,2,\ldots,n. \end{array} \tag{30}$$

For $A_{k,m}(x)$, setting $x=a_i, m=m_i (i=1,2,\ldots,p)$, $A_{k,m_i}(a_i)(l=0,1,\ldots,m_i, i=1,2,\ldots,p)$ is calculated by (29). For $B_{k,n}(y)$, setting $y=b_j, n=n_j(j=1,2,\ldots,q)$, $B_{k,n_j}(b_j)(l=0,1,\ldots,n_j, j=1,2,\ldots,q)$ is calculated by (30), where $B_{k,n_j}(b_j)=B_{k,n_j}(b_j)$.

Algorithm of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$

$$\begin{array}{l} z_0=1, z_{l+1}=\frac{z_l z}{l}, \quad l=1,2,\ldots. \\ S_0=\left\{\begin{array}{l} 1, \ \ m_1+\cdots+m_p+n_1+\cdots+n_q=0, \\ 0, \ \ m_1+\cdots+m_p+n_1+\cdots+n_q>0. \end{array}\right. \\ s=z_{l+1}\prod\limits_{i=1}^{p}A_{l,m_i}(a_i)\prod\limits_{j=1}^{q}B_{l,n_j}(b_j), S_{l+1}=S_l+s, \\ l=0,1,\ldots,N, \end{array} \tag{31}$$

where

$$N=\max\{l: z_{l+1}\prod_{i=1}^{p}A_{l,m_i}(a_i)\prod_{j=1}^{q}B_{l,n_j}(b_j)>10^{-prec}\}. \tag{32}$$

For N we can approximate calculations in the following way:

$$N=\left[\left|\frac{prec\ln 10}{\ln|z|}\right|\right]+1, \tag{33}$$

for $p=q+1, |z|<1$, and N satisfies the following conditions

$$\left|\left(\frac{z}{(n!)^{q+1-p}}\right)^n\right|\approx\left|\left(\frac{e^{q+1-p}z}{n^{q+1-p}}\right)^n\right|<10^{-prec}, \tag{34}$$

or

$$n\left((q+1-p)(1-\ln n)+\ln|z|\right)+prec\ln 10\approx 0, \tag{35}$$

for $p<q+1$, where $prec$ is the precision.

By an approximate Newton iterative method, N can be determined as follows

$$\begin{array}{l} N=\left[\frac{n_0(q+1-p)+prec\ln 10}{(q+1-p)\ln n_0}\right]+1, \\ n_0=\max\{e^{\ q+1-p}\sqrt{|10z|}, prec\}. \end{array} \tag{36}$$

In Mathematica, $HypergeometricPFQ[a,b,z]$ is used to calculate ${}_pF_q(\tilde{a};\tilde{b};z)$. However, there is no special internal function for calculating ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$. It can be calculated by symbolic differentiation. For example, the following codes can be used to calculate ${}_pF_q^{(m_1,0,\ldots,0;n_1,0,\ldots,0)}(\tilde{a};\tilde{b};z)$

$$\begin{array}{l} Clear[tt]; Clear[ss]; aa=a[[1]]; \\ bb=b[[1]]; a[[1]]=tt; b[[1]]=ss; \\ N[D[D[HypergeometricPFQ[a,b,z],\{tt,da[[1]]\}], \\ \{ss,db[[1]]\}]/.tt->aa/.ss->bb, Prec] \end{array} \tag{37}$$

where $m_1=da[[1]], n_1=db[[1]]$.

In the following, we give the comparison between the algorithm (31) and (37) by numerical examples in Table I and Table II.

Table I The comparison of algorithm (31) and (37) for computing time

$\tilde{a}$, $\tilde{b}$, z	$\widetilde{M_1}$, $\widetilde{N_1}$	*Algorithm*	*prec, N*	*Time*
$\{\frac{1}{5},\frac{1}{2}\}$, $\{\frac{1}{3},\frac{3}{2}\}$, $\frac{5}{4}$	$(2,0)$, $(1,0)$	(37)	16,	0.187201
		(31)	16, 22	0.
		(37)	32,	0.171601
		(31)	32, 33	0.
		(37)	48,	0.312002
		(31)	48, 44	0.015600
$\{\frac{1}{5},\frac{1}{2}\}$, $\{\frac{1}{3},\frac{3}{2}\}$, $\frac{5}{4}$	$(3,0)$, $(2,0)$	(37)	16,	0.608404
		(31)	16, 22	0.015600
		(37)	32,	0.156001
		(31)	32, 33	0.
		(37)	48,	0.265202
		(31)	48, 44	0.015600
$\tilde{a}$, $\tilde{b}$, z	$\widetilde{M_2}$, $\widetilde{N_1}$	*Algorithm*	*prec, N*	*Time*
$\{\frac{1}{5},\frac{1}{2},\frac{5}{3}\}$, $\{\frac{1}{3},\frac{3}{7}\}$, $\frac{1}{4}$	$(1,0,0)$, $(2,0)$	(37)	16,	0.561604
		(31)	16, 26	0.015600
		(37)	32,	0.390002
		(31)	32, 53	0.015600
		(37)	48,	1.014007
		(31)	48, 79	0.015600
$\{\frac{1}{5},\frac{1}{2},\frac{5}{3}\}$, $\{\frac{1}{3},\frac{3}{7}\}$ $\frac{1}{4}$	$(2,0,0)$, $(2,0)$	(37)	16,	0.858005
		(31)	16, 26	0.
		(37)	32,	2.402415
		(31)	32, 53,	0.015600
		(37)	48,	0.795605
		(31)	48, 79	0.031200

Table II The comparison of algorithm (31) and(37) for function value

$\tilde{a}$, $\tilde{b}$, z	*Algorithm*	*function value*
$\{\frac{1}{5},\frac{1}{2}\}$, $\{\frac{1}{3},\frac{3}{2}\}$, $\frac{5}{4}$	(37)	−4.553114098298790
	(31)	−4.553114098298790
	(37)	−4.5531140982987902039695 040338230
	(31)	−4.5531140982987902039695 040338230
	(37)	−4.5531140982987902039695 0403382300417243044115698
	(31)	−4.5531140982987902039695 0403382300417243044115698
$\{\frac{1}{5},\frac{1}{2}\}$, $\{\frac{1}{3},\frac{3}{2}\}$, $\frac{5}{4}$	(37)	12.97967
	(31)	12.97967410124726
	(37)	*
	(31)	12.9796741012472602558419227 40789
	(37)	*
	(31)	12.9796741012472602558419227 407893695005363576462
$\tilde{a}$, $\tilde{b}$, z	*Algorithm*	*function value*
$\{\frac{1}{5},\frac{1}{2},\frac{5}{3}\}$, $\{\frac{1}{3},\frac{3}{7}\}$, $\frac{1}{4}$	(37)	45.78572314198492
	(31)	45.78572314198473
	(37)	45.7857231419849195416544 13452488
	(31)	45.7857231419849195416544 13452469
	(37)	45.7857231419849195416544 134524884351229872376496
	(31)	45.7857231419849195416544 134524884351229872376433
$\{\frac{1}{5},\frac{1}{2},\frac{5}{3}\}$, $\{\frac{1}{3},\frac{3}{7}\}$ $\frac{1}{4}$	(37)	34.41078647370002
	(31)	34.41078647369899
	(37)	34.41078647370
	(31)	34.4107864737000205825759 60801885
	(37)	*
	(31)	34.4107864737000205825759 608020127564112438092121

Where $\widetilde{M_1}=(m_1,m_2)$, $\widetilde{M_2}=(m_1,m_2,m_3)$ and $\widetilde{N_1}=$

(n_1, n_2). Seen from Table I and Table II, the algorithms (31) and (37) have almost the same accuracy for $m_1 + n_1 \le 3$. However, the computing speed of algorithm (31) is 20 times faster than algorithm (37). Moreover, when $m_1 + n_1 > 3$, the specified precision can't be achieved and the calculation can't be finished with the increase of precision by (37)(see the asterisk in Table II). However, the specified accuracy can be achieved by (31).

In order to illustrate the calculation of using (31) and the generalized hypergeometric function ${}_pF_q(\tilde{a};\tilde{b};z)$ have the same order of magnitude, we obtain the following results:

Table III The comparison of algorithm (31)and HypergeometricPFQ[a, b, z](PFQ)
$(\tilde{a} = \{\frac{1}{5}, \frac{1}{2}, \frac{4}{5}\}, \tilde{b} = \{-\frac{4}{3}, \frac{3}{2}, -\frac{11}{5}\}, z = \frac{-15}{4})$

(m_1, m_2, m_3), (n_1, n_2, n_3)	*Algorithm*	*prec*	*Time*
(2, 2, 2), (1, 2, 3)	(31)	16	0.015625
		32	0.015625
		64	0.031250
		128	0.046875
(0, 0, 0), (0, 0, 0)	(31)	16	0.0
		32	0.015625
		64	0.015625
		128	0.015625
(0, 0, 0), (0, 0, 0)	*PFQ*	16	0.03125
		32	0.046875
		64	0.046875
		128	0.046875

Table IV The comparison of algorithm (31) and HypergeometricPFQ[a, b, z](PFQ)
$(\tilde{a} = \{\frac{2}{5}, \frac{1}{2}, \frac{4}{5}, \frac{1}{7}\}, \tilde{b} = \{-\frac{4}{3}, \frac{3}{2}, -\frac{7}{5}\}, z = \frac{1}{3})$

(m_1, m_2, m_3, m_4), (n_1, n_2, n_3)	*Algorithm*	*prec*	*Time*
(2, 1, 1, 2), (1, 2, 3)	(31)	16	0.015625
		32	0.031250
		64	0.062500
		128	0.109375
(0, 0, 0, 0), (0, 0, 0)	(31)	16	0.015625
		32	0.015625
		64	0.015625
		128	0.046875
(0, 0, 0, 0), (0, 0, 0)	*PFQ*	16	0.046875
		32	0.062500
		64	0.062500
		128	0.062500

Seen from Table III and Table IV, the speed of calculating ${}_pF_q(\tilde{a};\tilde{b};z)$ by using (31) is faster than by using the command $N[HypergeometricPFQ[a, b, z], Prec]$. In fact, the traditional numerical calculation will be terminated when truncation error is less than the specified precision. However, the application of (36) can reduce the link so that the computational efficiency can be improved. According to numerical results of Table III and Table IV, ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$ and the ratio of the amount of calculation for the ${}_pF_q(\tilde{a};\tilde{b};z)$ are as follows:

$$\frac{N_{|m|,|n|}}{N_{|0|,|0|}} = \frac{(p+1+|m|+\lambda(q+1+|n|))}{(p+1+\lambda(q+1))} = 1 + \frac{|m|+\lambda|n|}{(p+1+\lambda(q+1))} \sim 3, \tag{38}$$

Furthermore, the calculation results are basically identical.

IV. Some applications of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$

In this section, we consider the applications of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$. Many special function can be expressed in ${}_pF_q(\tilde{a};\tilde{b};z)$. For example, incomplete Beta Function,Whittaker Functions,Parabolic Cylinder Function, Bateman's Function, Incomplete Gamma Functions, Coulomb Wave Functions, Bessel functions and associated functions, Kelvin's and Associated Functions, Struve's Functions, Lommel's Functions, Elliptic Functions and Integrals, etc(see [14]).

The Lommel function of the first kind $s_{\mu,\nu}$ is a particular solution of the inhomogeneous Bessel differential equation

$$x^2y''(x) + xy'(x) + (x^2 - \nu^2)y(x) = x^{\mu+1}, \tag{39}$$

and it can be expressed in terms of a hypergeometric series[14,15]

$$s_{\mu,\nu}(x) = \frac{x^{\mu+1}}{(\mu-\nu+1)(\mu+\nu+1)}{}_1F_2\left(1; \frac{\mu-\nu+3}{2}, \frac{\mu+\nu+3}{2}; -\frac{x^2}{4}\right). \tag{40}$$

By $\frac{d^k}{d\mu^k}x^{\mu+1} = x^{\mu+1}\ln^k x$, we have

$$s_{\mu,\nu,k}(x) = \frac{d^k}{d\mu^k}\left[\frac{x^{\mu+1}}{(\mu-\nu+1)(\mu+\nu+1)}{}_1F_2\left(1; \frac{\mu-\nu+3}{2}, \frac{\mu+\nu+3}{2}; -\frac{x^2}{4}\right)\right] \tag{41}$$

is a particular solution of the inhomogeneous Bessel differential equation

$$x^2y''(x) + xy'(x) + (x^2 - \nu^2)y(x) = x^{\mu+1}\ln^k x. \tag{42}$$

In the following, we consider the evaluation of integrals involving the generalized hypergeometric function.

Since the following formulas hold:

$$\int_0^1 \frac{t^{b-1}(1-t)^{c-1}}{(1-zt)^a}dt = B(b,c)_2F_1(a, b; c+b; z), \tag{43}$$

and

$$\int_0^1 e^{zt}t^{a-1}(1-t)^{b-1}dt = B(a,b)_1F_1(a; a+b; z), \tag{44}$$

we have

$$\int_0^1 \frac{t^{b-1}(1-t)^{c-1}}{(1-zt)^a}\ln^m\frac{1}{1-zt}\ln^n(1-t)dt = \sum_{k=0}^{n}\binom{n}{k}\frac{\partial^{n-k}}{\partial c^{n-k}}B(b,c)_2F_1^{(m,0;k)}(a, b; c+b; z), \tag{45}$$

and

$$\int_0^1 e^{zt}t^{a-1}(1-t)^{b-1}\ln^n(1-t)dt = \sum_{k=0}^{n}\binom{n}{k}\frac{\partial^{n-k}}{\partial b^{n-k}}B(a,b)_1F_1^{(0;k)}(a; a+b; z). \tag{46}$$

Moreover, the following results of [6] were given by formula (44):

$$\begin{aligned}&\int_{-1}^1 \frac{e^{i\omega x}\ln^m\frac{1+x}{2}\ln^n\frac{1-x}{2}}{(\frac{1+x}{2})^{1-a}(\frac{1-x}{2})^{1-b}}dx\\ &= 2e^{-i\omega}\frac{\partial^{m+n}}{\partial a^m\partial b^n}\left[B(a,b)_1F_1(a; a+b; 2i\omega)\right]\\ &= 2e^{-i\omega}\sum_{k=0}^{n}\binom{n}{k}\sum_{i=0}^{m}\binom{m}{i}\frac{\partial^{m+n-i-k}}{\partial a^{m-i}\partial b^{n-k}}B(a,b)\\ &\cdot\sum_{u=0}^{i}\binom{i}{u}{}_1F_1^{(u;i+k-u)}(a; a+b; 2i\omega),\end{aligned} \tag{47}$$

and

$$\begin{aligned}&\int_0^1 x^{a-1}J_l(rx)\ln^m x dx\\ &= \frac{r^l}{l!2^l}\frac{\partial^m}{\partial a^m}\frac{1}{a+l}\,{}_1F_2\left(\frac{a+l}{2}; \frac{a+l}{2}+1, l+1; -\frac{r^2}{4}\right)\\ &= \frac{r^l m!}{l!2^l}\sum_{i=0}^{m}\frac{(-1)^{m-i}}{i!(a+l)^{m-i+1}2^i}\\ &\cdot\sum_{u=0}^{i}\binom{i}{u}{}_1F_2^{(u;i-u,0)}\left(\frac{a+l}{2}; \frac{a+l}{2}+1, l+1; -\frac{r^2}{4}\right).\end{aligned} \tag{48}$$

The left integral of (45)-(48) can be calculated by the following numerical integration:

$$NIntegrate[\frac{t^{b-1}(1-t)^{c-1}}{(1-zt)^{a}}\ln[\frac{1}{1-zt}]^{m}\ln[1-t]^{n},\{t,0,1\},\\ WorkingPrecision->Prec] \tag{49}$$

$$NIntegrate[Exp[z*t]t^{a-1}(1-t)^{b-1}\ln[1-t]^{n},\{t,0,1\},\\ WorkingPrecision->Prec] \tag{50}$$

$$NIntegrate[Exp[omg*I*t]\frac{\ln[\frac{1+t}{2}]^{m}\ln[\frac{1-t}{2}]^{n}}{(\frac{1+t}{2})^{1-a}(\frac{1-t}{2})^{1-b}},\{t,-1,1\},\\ WorkingPrecision->Prec] \tag{51}$$

and

$$NIntegrate[t^{a}BesselJ[l,r*t]\ln[t]^{m},\{t,0,1\},\\ WorkingPrecision->Prec] \tag{52}$$

while the right of (45)-(48) can be calculated by the following codes:

$$Bmat=BetaD[b,c,0,n,1];\\ \sum_{k=0}^{n}Binomial[n,k]Bmat[[1,n-k+1]]\\ HpFqSeries[\{a,b\},\{c+b\},z,\{m,0\},\{k\}] \tag{53}$$

$$Bmat=BetaD[a,b,0,n,1];\\ \sum_{k=0}^{n}Binomial[n,k]Bmat[[1,n-k+1]]\\ HpFqSeries[\{a\},\{a+b\},z,\{0\},\{k\}] \tag{54}$$

$$Bmat=BetaD[1+a,1+b,m,n,1];\\ 2*Exp[-omg*I]\sum_{i=0}^{m}Binomial[m,i]\\ \sum_{k=0}^{n}Binomial[n,k]Bmat[[m-i+1,n-k+1]]\\ \sum_{v=0}^{i}Binomial[i,u]HpFqSeries[\{1+a\},\{2+a+b\}\\ 2*omg*I,\{u\},\{i-u+k\}] \tag{55}$$

and

$$\frac{r\hat{}l*m!(-1)^{m}}{Gamma[l+1]2^{l}}\sum_{i=0}^{m}\frac{(-2)^{-i}}{i!(a+l)^{m-i+1}}\sum_{u=0}^{i}Binomial[i,u]\\ HpFqSeries[\{\frac{1+a}{2}\{\frac{1+a}{2}+1,l+1\},\frac{-r^{2}}{2},\{u\},\{i-u,0\}] \tag{56}$$

Remark 4.1 For the calculation of $\frac{\partial^{i+j}}{\partial a^{i}\partial b^{j}}B(a,b)(i=0,1,\ldots,m;j=0,1,\ldots,n)$, we have given the recursion methods in [10,16].

In the following, we give the comparison between algorithm (49) and (53) for calculating (45) in Table V.

Table V The comparison of algorithm (49) and (53) for calculating (45)

a,b,c,z	m,n	$Algorithm$	$prec,Time$	rr
$\frac{1}{2},\frac{1}{3},\frac{1}{5},\frac{1}{4}$	$4,2$	(49) (53)	$16,0.0468$ $16,0.0156$	1.53×10^{-11} 2.76×10^{-13}
$\frac{1}{2},\frac{1}{3},\frac{1}{5},\frac{1}{4}$	$4,2$	(49) (53)	$32,0.2652$ $32,0.0312$	1.18×10^{-12} 2.76×10^{-29}
$\frac{1}{2},\frac{1}{3},\frac{1}{5},\frac{1}{4}$	$4,2$	(49) (53)	$64,0.4212$ $64,0.0468$	7.56×10^{-22} 3.40×10^{-61}

Similarly, we can obtain the following results by using different algorithms:

Table VI The comparison of algorithm (50) and (54) for calculating (46)

a,b,z	n	$Algorithm$	$prec,Time$	rr
$\frac{1}{3},\frac{1}{5},3+8i$	5	(50) (54)	$16,0.2340$ $16,0.0624$	$0.\times10^{-16}$ $0.\times10^{-17}$
$\frac{1}{3},\frac{1}{5},3+8i$	5	(50) (54)	$32,0.6552$ $32,0.0624$	1.73×10^{-13} $0.\times10^{-37}$
$\frac{1}{3},\frac{1}{5},3+8i$	5	(50) (54)	$64,0.6552$ $64,0.0936$	1.01×10^{-16} $0.\times10^{-77}$

Table VII The comparison of algorithm (51) and (55) for calculating (47)

a,b,ω	m,n	$Algorithm$	$prec,Time$	rr
$\frac{-7}{4},\frac{-11}{9},4$	$2,2$	(51) (55)	$16,0.2964$ $16,0.1716$	$6.\times10^{-15}$ $0.\times10^{-9}$
$\frac{-7}{4},\frac{-11}{9},4$	$2,2$	(51) (55)	$32,0.9516$ $32,0.1872$	4.25×10^{-14} $0.\times10^{-29}$
$\frac{-7}{4},\frac{-11}{9},4$	$2,2$	(51) (55)	$64,1.6536$ $64,0.2340$	3.80×10^{-18} $0.\times10^{-69}$

Table VIII The comparison of algorithm (52) and (56) for calculating (48)

a,r	m,l	$Algorithm$	$prec,Time$	rr
$\frac{-15}{4},\frac{1}{2}$	$3,4$	(52) (56)	$16,0.0468$ $16,0.0156$	$0.\times10^{-16}$ $0.\times10^{-20}$
$\frac{-15}{4},\frac{1}{2}$	$3,4$	(52) (56)	$32,0.1716$ $32,0.0312$	$0.\times10^{-32}$ $0.\times10^{-40}$
$\frac{-15}{4},\frac{1}{2}$	$3,4$	(52) (56)	$64,0.5148$ $64,0.0312$	$0.\times10^{-65}$ $0.\times10^{-80}$

In Table V-VIII, rr represents relative error. The left of (45)-(48) are improper integrals. Seen from Table V-VIII, using the numerical integration in Mathematica, the computation time is consuming and the accuracy can not be improved with the increase of specified accuracy. However, the accuracy of calculation can be obtained and the calculation speed is fast by $HpFqSeries[a,b,z,Dms,Dns]$.

Remark 4.2 Let $z=\lambda+i\omega$ in (46). By the real and imaginary parts of (46), we obtain the following integrals:

$$\int_0^1 e^{\lambda t}t^{a-1}(1-t)^{b-1}\ln^{n}(1-t)\cos\omega t dt\\ =Re\sum_{k=0}^{n}\left(\begin{array}{c}n\\k\end{array}\right)\frac{\partial^{n-k}}{\partial b^{n-k}}B(a,b)_1F_1^{(0;k)}(a;a+b;\lambda+i\omega), \tag{57}$$

and

$$\int_0^1 e^{\lambda t}t^{a-1}(1-t)^{b-1}\ln^{n}(1-t)\sin\omega t dt\\ =Im\sum_{k=0}^{n}\left(\begin{array}{c}n\\k\end{array}\right)\frac{\partial^{n-k}}{\partial b^{n-k}}B(a,b)_1F_1^{(0;k)}(a;a+b;\lambda+i\omega). \tag{58}$$

Moreover, in order to show the fast algorithm of ${}_pF_q^{(m_1,\ldots,m_p;n_1,\ldots,n_q)}(\tilde{a};\tilde{b};z)$ given in this paper, we give the results of [11]

$$\begin{aligned}&\frac{d}{da}{}_2F_1(n+1,a;a+b;z)\\&=\frac{bz}{(1-z)^{n+1-b}}\sum_{k=0}^{n}\frac{(1+b)_k z^k(1-z)^k}{k!(a+b+k)^2}P_{n-k}^{(k+1,b+k-n-1)}(1-2z)\\&\cdot{}_3F_2(s,s,b+k+1;s+1,s+1;z)|_{s=a+b+k},\end{aligned} \quad (59)$$

where
$P_n^{(m,l)}(z)=\frac{(-1)^n}{2^n n!(1-z)^m(1+z)^l}\frac{d^n}{dz^n}\left[(1-z)^{m+n}(1+z)^{l+n}\right]$
is the Jacobi polynomials. If the right side of (51) is calculated, it is better than (3). However, it is still relatively large with the increase of n. Thus, by using the following result:

$$\begin{aligned}&\frac{d}{da}{}_2F_1(n+1,a;a+b;z)\\&=\frac{d}{dt}{}_2F_1(n+1,t;a+b;z)|_{t=a}\\&+\frac{d}{dt}{}_2F_1(n+1,a;t;z)|_{t=a+b}\end{aligned} \quad (60)$$

we will obtain a faster computation. Moreover, numerical results of Table I-Table IV show that the algorithms presented in this paper are the fastest.

Finally, we consider the basic integral identities

$$\begin{aligned}&\int_0^1 t^{\rho-1}(1-t)^{\sigma-1}\ {}_2F_1(\alpha,\beta;\gamma;tz)dt\\&=B(\rho,\sigma){}_3F_2(\alpha,\beta,\rho;\gamma,\rho+\sigma;z),\end{aligned} \quad (61)$$

$Re\,(\rho)>0, Re(\sigma)>0, |\arg(1-z)|<\pi$. By (47) we can obtain the following results:

$$\begin{aligned}&\int_0^1 t^{\rho-1}(1-t)^{\sigma-1}\ln^n t\ln^m(1-t){}_2F_1(\alpha,\beta;\gamma;tz)dt\\&=\sum_{k=0}^{n}\binom{n}{k}\sum_{l=0}^{m}\binom{m}{l}\frac{\partial^{n-k+m-l}}{\partial\rho^{n-k}\partial\sigma^{m-l}}B(\rho,\sigma)\\&\times\sum_{u=0}^{k}\binom{k}{u}\ {}_3F_2^{(0,0,k-u;0,u+l)}(\alpha,\beta,\rho;\gamma,\rho+\sigma;z).\end{aligned} \quad (62)$$

Moreover, many special functions can be represented by the generalized hypergeometric function. For example, the Bessel function, incomplete gamma function, incomplete beta Function and the Legendre function can be represented by the generalized hypergeometric function. So the derivatives with respect to the order of Bessel function, the derivatives with respect to the parameters of the incomplete gamma functions, incomplete beta Function and the Legendre function are obtained by the partial derivatives of the generalized hypergeometric function. From the results of [17], we can obtain the expressions of the Bessel functions and the modified Bessel functions of the first kind

$$\left\{\begin{array}{c}J_\alpha(x)\\I_\alpha(x)\end{array}\right\}=\frac{(\frac{x}{2})^\alpha}{\Gamma(\alpha+1)}{}_0F_1(\alpha+1;\mp\frac{x^2}{4}). \quad (63)$$

Using Leibniz's derivation rule and the results of [18], we have

$$\begin{aligned}&\frac{\partial^m}{\partial\alpha^m}\left\{\begin{array}{c}J_\alpha(x)\\I_\alpha(x)\end{array}\right\}\\&=\frac{(\frac{x}{2})^\alpha}{\Gamma(\alpha+1)}\sum_{l=0}^{m}\binom{m}{l}P_{m-l,\alpha+1}^{-}(\frac{x}{2})\\&\times\frac{d^l}{ds^l}\left[{}_0F_1(s,\frac{\mp x^2}{4})\right]|_{s=\alpha+1},\end{aligned} \quad (64)$$

where

$$P_{l,\alpha}^{\pm}(x)=\sum_{u=0}^{l}\binom{l}{u}H_u^{\psi_\pm}(\alpha)\ln^{l-u}x. \quad (65)$$

and

$$\begin{aligned}&\psi_\pm(\alpha)=\pm\psi(\alpha), H_0^{\psi_\pm}(\alpha)=1,\\&H_j^{\psi_\pm}(\alpha)=\sum_{k=0}^{j-1}\binom{j-1}{k}\psi_\pm^{(k)}(\alpha)H_{j-1-k}^{\psi_\pm}(\alpha),\\&j=1,2,\cdots,m.\end{aligned} \quad (66)$$

By (64)-(66), We can obtain a quick calculation of $\frac{\partial^m}{\partial\alpha^m}\left\{\begin{array}{c}J_\alpha(x)\\I_\alpha(x)\end{array}\right\}$.

V. Conclusion

In this paper, a algorithm of the partial derivatives of the generalized hypergeometric function is obtained. Moreover, we compare some algorithms of calculating the partial derivatives of the generalized hypergeometric function. Numerical examples show the algorithm given in this paper improves the precision and accelerates the calculation of partial derivatives of the generalized hypergeometric function. Furthermore, we obtain some applications of the partial derivatives of the generalized hypergeometric function in calculating improper integrals and the derivative of the special function, such as the derivatives with respect to the order of Bessel function etc. The accuracy and the speed of calculating the improper integrals are improved by numerical examples.

References

[1] F. W. Olver, D. W. Lozier, R. F. Boisvert and C. W. Clark, *NIST Handbook of Mathematical Functions.* New York: Cambridge University Press, 2010.

[2] L. U. Ancarani and G. Gasaneo, "Derivatives of Any Order of the Hypergeometric Function ${}_pF_q(a_1,...,a_p;b_1,...,b_q;z)$ with Respect to the Parameters a_i and b_i ," *J. Phys. A: Math. Theor.*, vol. 43, no. 8, 2010.

[3] S. Flügge, *Practical Quantum Mechanics I,* Berlin: Springer, 1971.

[4] L. U. Ancaraniand and G. Gasaneo, "Derivatives of Any Order of the Gaussian Hypergeometric Function ${}_2F_1(a,b,c;z)$ with Respect to the Parameters a,b and c," *J. Phys. A: Math. Theor.*, vol. 42, no.39, 2009.

[5] L. U. Ancarani and G. Gasaneo, "Derivatives of Any Order of the Confluent Hypergeometric Function ${}_1F_1(a,b,z)$ with respect to the Parameter a or b, " *J. Math. Phys.*, vol. 49, no.6, 2008.

[6] H. Kang, C. An, "Differentiation Formulas of Some Hypergeometric Functions with Respect to all Parameters, " *Appl. Math. Comput.* vol. 258, pp.454-464, 2015.

[7] J. Froehlich, "Parameter Derivatives of the Jacoby Polynomials and the Gaussian Hypergeometric Function, " *Int. Trans. Spec. Funct.*,vol. 2, pp. 253-266, 1994.

[8] R. Szmytkowski, "A Note on Parameter Derivatives of Classical Orthogonal Polynomials, " *Mathematics.*, 2010.

[9] B. X. Fejzullahu, "Parameter Derivatives of the Generalized Hypergeometric Function, " *Int. Trans. Spec. Funct.*, vol. 28, no. 11, pp.781-788, 2017.

[10] Z. Sun, H. Qin and A. Li, "Extension of the Partial Derivatives of the Incomplete Beta Function for Complex Values, " *Appl. Math. Comput.*, vol. 275, pp.63-71, 2016.

[11] Y. A. Brychkov, *Handbook of Special Functions: Derivatives, Integrals, Series and Other Formulas,* CRC press, 2008.

[12] M. W. Coffey, "A Set of Identities For a Class of Alternating Binomal Sums Arising in Computing Applications," *Utilitas Mathematica*, vol. 76, pp.79-90, 2006.

[13] K. S. Kölbig, "The Complete Bell Polynomials for Certain Arguments in Terms of Stirling Numbers of the First Kind, " *J. Comput. Appl. Math.*, vol. 51, no. 1, pp. 113-116, 1994.

[14] A. M. Mathai and R. K. Saxena, *Generalized Hypergeometric Functions With Applications in Statistics and Physical Sciences.* Berlin: Springer, 1971.

[15] A. Baricz and S. Koumandos, "Turán-type Inequalities for Some Lommel Function of the First Kind," *Proceedings of the Edinburgh Mathematical Society,* vol. 59, pp. 569-579, 2016.

[16] A. Li and H. Qin, "Some Transformation Properties of the Incomplete Beta Function and Its Partial Derivatives ," *IAENG International Journal of Applied Mathematics,* vol. 49, no. 1, pp.109-113, 2019.

[17] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals Series and Products.* Burlington: Academic Press, 2007.

[18] A. Li and H Qin, " The Representations on the Partial Derivatives of the Extended, Generalized Gamma and Incomplete Gamma Functions and Their Applications, " *IAENG International Journal of Applied Mathematics,* vol. 47, no. 3, pp. 312-318, 2017.