Journal of Computational and Applied Mathematics 236 (2012) 1961–1975

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Precise and fast computation of a general incomplete elliptic integral of third kind by half and double argument transformations

Toshio Fukushima

National Astronomical Observatory of Japan, 2-21-1, Ohsawa, Mitaka, Tokyo 181-8588, Japan

ARTICLE INFO

Article history:
Received 16 July 2011
Received in revised form 4 November 2011

Keywords:
Double argument transformation
Legendre's incomplete elliptic integral
Jacobian elliptic function
Half argument transformation

ABSTRACT

This is a continuation of our works to compute the incomplete elliptic integrals of the first and second kind (Fukushima (2010, 2011) [21,23]). We developed a method to compute an associate incomplete elliptic integral of the third kind, $J(\varphi, n|m) \equiv [\Pi(\varphi, n|m) - F(\varphi|m)]/n$, by the half argument formulas of the sine and cosine amplitude functions and the double argument transformation of the integral. The relative errors of $J(\varphi, n|m)$ computed by the new method are sufficiently small as less than 20 machine epsilons. Meanwhile, the simplicity of the adopted algorithm makes the new method run 1.5 to 3.7 times faster than Carlson's duplication method. The combination of the new method and that to compute simultaneously two other associate incomplete elliptic integrals of the second kind, $B(\varphi|m) \equiv [E(\varphi|m) - (1-m)F(\varphi|m)]/m$ and $D(\varphi|m) \equiv [F(\varphi|m) - E(\varphi|m)]/m$, which we established recently [23], enables a precise and fast computation of arbitrary linear combination of Legendre's incomplete elliptic integrals of all three kinds, $F(\varphi|m)$, $E(\varphi|m)$, and $\Pi(\varphi, n|m)$. These new procedures share the same device, the half argument transformations, while the double argument transformation of $J(\varphi, n|m)$ includes those of $B(\varphi|m)$ and $D(\varphi|m)$ as its sub component. As a result, the simultaneous computation of the three associate integrals is significantly faster than computing them separately. In fact, our combined procedure is 2.7 to 5.9 times faster than the combination of Carlson's duplication method to compute R_D and R_J.

1. Introduction

1.1. Legendre's incomplete elliptic integral of third kind

We write Legendre's incomplete elliptic integral of the third kind as

$$\Pi(\varphi, n|m) \equiv \int_0^{\varphi} \frac{d\theta}{\left(1 - n\sin^2\theta\right)\sqrt{1 - m\sin^2\theta}}, \tag{1}$$

where φ is the amplitude, n is the characteristic, and m is the parameter [1]. Its major references are [2–4]. Among them, useful are Chapters 19 and 22 of [4] and their web site, http://dlmf.nist.gov/.

There are significant differences in the definition and notation of the input arguments, the separators among them, and the order of input arguments in literature [5]. This is quite confusing for non-experts. In the present article, we follow the standard sign convention of n and the notations of [5] except for the order of arguments.

E-mail address: Toshio.Fukushima@nao.ac.jp.

0377-0427/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2011.11.007

The integral $\Pi(\varphi, n|m)$ is not so frequently used in mathematical physics and engineering as Legendre's two other standard forms of the incomplete elliptic integrals:

$$F(\varphi|m) \equiv \int_0^{\varphi} \frac{d\theta}{\sqrt{1 - m\sin^2\theta}}, \qquad E(\varphi|m) \equiv \int_0^{\varphi} \sqrt{1 - m\sin^2\theta}\, d\theta. \tag{2}$$

Nevertheless, $\Pi(\varphi, n|m)$ appears in describing the gravitational and/or electromagnetic field associated with scalar and/or vector potential of a simple distribution such as annular disks with finite thickness. An example is the magnetic field caused by a thick coil [6].

Also, the integral plays a key role in describing the torque-free rotation of a triaxial rigid body. Refer to Appendix C of [7]. This motion serves a platform to study general rotational motions of a celestial body [8] and of a molecule [9]. Further, the integral is used in describing the periodic solution of a certain class of nonlinear Schrödinger equation [10].

Without doubt, $\Pi(\varphi, n|m)$ is the mostly complicated special function among the elliptic integrals and functions since it is a function of three variables [11]. In fact, Legendre's complete elliptic integrals of the first and second kind, $K(m) \equiv F(\pi/2|m)$ and $E(m) \equiv E(\pi/2|m)$, are single variable functions. Meanwhile, not only $F(\varphi|m)$ and $E(\varphi|m)$ but also Legendre's complete elliptic integral of the third kind, $\Pi(n|m) \equiv \Pi(\pi/2, n|m)$, and Jacobian elliptic functions, $\mathrm{sn}(u|m)$, $\mathrm{cn}(u|m)$, $\mathrm{dn}(u|m)$, and $\mathrm{am}(u|m)$, are bivariate functions.

On the other hand, $F(\varphi|m)$ and $E(\varphi|m)$ are computable from $\Pi(\varphi, n|m)$ and elementary functions:

$$F(\varphi|m) = \Pi(\varphi, 0|m), \tag{3}$$

$$E(\varphi|m) = \frac{\sin\varphi\cos\varphi}{\sqrt{1 - m\sin^2\varphi}} + m_c\Pi(\varphi, m|m), \tag{4}$$

$$E(\varphi|m) = (\tan\varphi)\sqrt{1 - m\sin^2\varphi} + m_c\left[\Pi(\varphi, 0|m) - \Pi(\varphi, 1|m)\right], \tag{5}$$

where $m_c \equiv 1 - m$ is the complementary parameter. Refer to formula 111.06 of [2] and formula 19.6.13 of [4]. In this sense, $\Pi(\varphi, n|m)$ can be thought as a generator of any incomplete elliptic integral.

1.2. Associate incomplete elliptic integral of the third kind

It is rare to see $\Pi(\varphi, n|m)$ alone in practical applications [12,6]. Usually, it appears together with $F(\varphi|m)$ and/or $E(\varphi|m)$. For example, the rotation angle of a triaxial rigid body is described by a linear combination of $F(\varphi|m)$ and $\Pi(\varphi, n|m)$ [12]. Since $\Pi(\varphi, 0|m) = F(\varphi|m)$, the pair of $F(\varphi|m)$ and $\Pi(\varphi, n|m)$ is not suitable as basic components of such a linear combination when $|n|$ is small. The typical values of $|n|$ in the case of celestial bodies in the solar system is as small as 10^{-7} or so [13]. Then, in order to avoid the loss of significant digits for small values of $|n|$, we introduce a linear combination of these two integrals in place of $\Pi(\varphi, n|m)$ itself:

$$J(\varphi, n|m) \equiv \frac{\Pi(\varphi, n|m) - F(\varphi|m)}{n} = \int_0^{\varphi} \frac{\sin^2\theta d\theta}{\left(1 - n\sin^2\theta\right)\sqrt{1 - m\sin^2\theta}}. \tag{6}$$

We call it an associate incomplete elliptic integral of the third kind. As will be seen below, $J(\varphi, n|m)$ is tightly related with Carlson's R_J [4].

If $F(\varphi|m)$ and $J(\varphi, n|m)$ are known, $\Pi(\varphi, n|m)$ is computable from them without suffering from the smallness of $|n|$ as

$$\Pi(\varphi, n|m) = F(\varphi|m) + nJ(\varphi, n|m). \tag{7}$$

This device is the same as a recipe to resolve the round-off error problems caused by the smallness of $|m|$ in case of $F(\varphi|m)$ and $E(\varphi|m)$ [14]. Indeed, $F(\varphi|m)$ and $E(\varphi|m)$ are computable from their variants named the associate incomplete elliptic integrals of the second kind [15,16]:

$$B(\varphi|m) \equiv \int_0^{\varphi} \frac{\cos^2\theta d\theta}{\sqrt{1 - m\sin^2\theta}}, \qquad D(\varphi|m) \equiv \int_0^{\varphi} \frac{\sin^2\theta d\theta}{\sqrt{1 - m\sin^2\theta}}, \tag{8}$$

such that

$$F(\varphi|m) = B(\varphi|m) + D(\varphi|m), \qquad E(\varphi|m) = B(\varphi|m) + m_c D(\varphi|m). \tag{9}$$

Summarizing these, we learn that a general incomplete elliptic integral is expressed as a linear combination of $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$ as

$$\alpha F(\varphi|m) + \beta E(\varphi|m) + \gamma\Pi(\varphi, n|m) = (\alpha + \beta + \gamma)B(\varphi|m) + (\alpha + \beta m_c + \gamma)D(\varphi|m) + \gamma nJ(\varphi, n|m). \tag{10}$$

The expression based on $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$ is more appropriate than that based on $F(\varphi|m)$, $E(\varphi|m)$, and $\Pi(\varphi, n|m)$ when $|m|$ and/or $|n|$ are small.

Further, $J(\varphi, n|m)$ is useful in computing the partial derivatives of $\Pi(\varphi, n|m)$ with respect to n or m when $|n|$ and/or $|m|$ is small. In fact, the derivatives are expressed as

$$\left(\frac{\partial \Pi(\varphi, n|m)}{\partial m}\right)_{\varphi,n} = \frac{J(\varphi, n|m) - J(\varphi, m|m)}{n - m}, \tag{11}$$

$$\left(\frac{\partial \Pi(\varphi, n|m)}{\partial n}\right)_{\varphi,m} = \frac{1}{2(n-m)(1-n)}\left[nB(\varphi|m) + (n-m)D(\varphi|m) + \left(n^2 - m\right)J(\varphi, n|m) - \frac{n\sin\varphi\cos\varphi\sqrt{1 - m\sin^2\varphi}}{1 - n\sin^2\varphi}\right]. \tag{12}$$

These are rewriting of formula 710.02 and 733.00 of [2] in terms of the associate integrals. See also formula 19.4.7 of [4]. These expressions contain no small divisors, m nor n.

1.3. Outline of the present article

The representative methods to compute $J(\varphi, n|m)$ are Bulirsch's procedures based on Gauss and Bartky transformations and Carlson's duplication method to compute R_J, one of his symmetric integrals. From the practical viewpoint, however, both methods are inappropriate due to either cancellation errors or large computational amount. Refer to our review later in Section 2.

Required by practical needs to implement recently-developed formulations to study rotational motions of a general triaxial rigid body numerically [13,12] and analytically [8] and to compute gravitational acceleration due to uniform ring or disk without cancellation problems [17], we have investigated precise and fast algorithms to compute elliptic functions and integrals [18–23,14].

Among them, we learn that the combination of half and double argument formulas is effective in computing $F(\varphi|m)$ [21]. In fact, our routine to compute $F(\varphi|m)$ is 1.2–1.6 times faster than Bulirsch's `el1` and 1.9–2.2 times faster than Carlson's duplication method to compute R_F. The same idea is successfully applied to the simultaneous calculation of $B(\varphi|m)$ and $D(\varphi|m)$ [23]. Indeed, our procedure to compute $B(\varphi|m)$ and $D(\varphi|m)$ simultaneously runs 3.1–5.9 times faster than Carlson's duplication method to compute R_D. In this article, we apply this approach to compute $J(\varphi, n|m)$.

We first review the existing formulations in Section 2. Then, we describe the new method in Section 3 and compare its cost and performance with those of Bulirsch's and Carlson's procedures in Section 4.

2. Existing formulations to compute $J(\varphi, n|m)$

There are four type of formulations to compute Legendre's incomplete elliptic integrals of the third kind; (i) the method based on the series expansion of the integrals, which are obtained by various kinds of binomial expansions of the integrand [2,22], (ii) the approach using the theta functions [1], which are usually expressed as series expansion in terms of Jacobi's nome, q, (iii) the procedure derived from the Gauss or Bartky transformations [15,24–26,18], and (iv) the formula utilizing the duplication theorems on Carlson's symmetric form of elliptic integrals [27–30].

The first method seems to be inefficient since it is too slow to converge for general values of input arguments [2]. Recently, we developed a new method of series expansion of general elliptic integrals [22]. Although its full examination as a fast computing procedure is not yet completed, preliminary trials indicate that it runs a little slower than the third method. Meanwhile, the second approach was once regarded as the standard until the appearance of the method using the Bartky transformation [25].

On the other hand, the last formulation is mostly popular since it is free from the cancellation problems found in the original form of the third one. However, this defect was overcome by the generalization of the third approach conducted by us [18]. Consequently, we shall concentrate on the last two formulations.

Hereafter, we assume that the input arguments are in their standard domain: $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. If not, we can transform them to satisfy the condition by utilizing the special value formulas and some transformations given in Appendix.

2.1. Bulirsch's procedures

When φ is restricted as $0 < \varphi < \pi/2$, the associate integral, $J(\varphi, n|m)$, is described by elliptic integrals introduced in [15, 24–26]:

$$\texttt{el1}\,(x, k_c) \equiv \int_0^{\tan^{-1}x} \frac{d\theta}{\sqrt{\cos^2\theta + k_c^2\sin^2\theta}}, \tag{13}$$

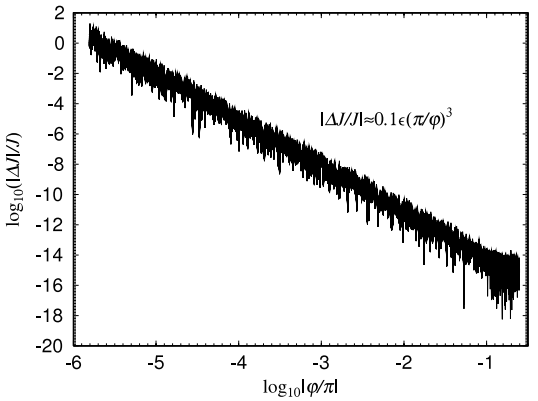

Fig. 1. Round-off errors of $J(\varphi, n|m)$ computed by `el`. Shown are the relative errors of $J(\varphi, n|m)$ computed by `el`, our extension of Bulirsch's `el3` [18], in the double precision environment. The errors are measured as the difference from the results of the quadruple precision calculation using Carlson's duplication method to compute R_J and plotted as functions of φ in a log–log manner. Overlapped are the error curves with various values of n and m in their standard domain, $0 < n < 1$ and $0 < m < 1$.

$$\texttt{el3}\,(x, k_c, p) \equiv \int_0^{\tan^{-1} x} \frac{d\theta}{\left(\cos^2\theta + p\sin^2\theta\right)\sqrt{\cos^2\theta + k_c^2\sin^2\theta}}, \tag{14}$$

as

$$J(\varphi, n|m) = \frac{\texttt{el3}\,(t, k_c, n_c) - \texttt{el1}\,(t, k_c)}{n}, \tag{15}$$

where $t \equiv \tan\varphi$, $k_c \equiv \sqrt{1-m}$ is the complementary modulus, and $n_c \equiv 1 - n$ is the complementary characteristic.

This expression faces cancellation problems when $|n|$ is small. Thus, Bulirsch proposed to generalize `el3` [18] into

$$\texttt{el}\,(x, k_c, p, a, b) \equiv \int_0^{\tan^{-1} x} \frac{\left(a\cos^2\theta + b\sin^2\theta\right)d\theta}{\left(\cos^2\theta + p\sin^2\theta\right)\sqrt{\cos^2\theta + k_c^2\sin^2\theta}}, \tag{16}$$

since "it would include elliptic integrals of all three kinds as special cases" as

$$\lambda F(\varphi|m) + \mu E(\varphi|m) = \texttt{el}\,(x, k_c, 1, \lambda+\mu, \lambda+\mu m_c)\,, \tag{17}$$

$$\lambda F(\varphi|m) + \mu \Pi(\varphi, n|m) = \texttt{el}\,(x, k_c, n_c, \lambda+\mu, \lambda n_c+\mu)\,, \tag{18}$$

where λ and μ are arbitrary constants. Later, we established its algorithm by extending that of `el3` [18]. Then, $J(\varphi, n|m)$ is computed from `el` as

$$J(\varphi, n|m) = \begin{cases} \texttt{el}\,(\tan\varphi, k_c, n_c, 0, 1) & (0 \le \varphi < \pi/4) \\ \texttt{el}\,(1/\tan\varphi_c, k_c, n_c, 0, 1) & (0 < \varphi_c \le \pi/4)\,, \end{cases} \tag{19}$$

where $\varphi_c \equiv \pi/2 - \varphi$ is the complementary amplitude. This expression contains no dangerous divisor, n. Thus, the cancellation problem with respect to n is positively resolved.

Nevertheless, the extended procedure, `el`, suffers loss of precision when φ is small. See Fig. 1 showing that the relative errors of $J(\varphi, n|m)$ computed by `el` are roughly in proportion to $1/\varphi^3$ independently on the values of m and n. This is a negative legacy of `el2`, Bulirsch's generalization of the incomplete elliptic integrals of the first and second kind. Refer to Fig. 1 and Appendix A of [23].

2.2. Carlson's formulation

On the other hand, $J(\varphi, n|m)$ is rewritten by a fundamental symmetric elliptic integral introduced in [27,29,30]:

$$R_J(x, y, z, p) \equiv \frac{3}{2}\int_0^{\infty} \frac{dt}{(t+p)\sqrt{(t+x)(t+y)(t+z)}}, \tag{20}$$

as

$$J(\varphi, n|m) = \begin{cases} s^3 R_J\left(1-s^2, 1-ms^2, 1, 1-ns^2\right)/3 & (0 \le \varphi < \pi/4) \\ \left(\sqrt{1-c^2}\right)^3 R_J\left(c^2, m_c+mc^2, 1, n_c+nc^2\right)/3 & (0 < \varphi_c \le \pi/4)\,, \end{cases} \tag{21}$$

where $s \equiv \sin\varphi$ and $c \equiv \sin\varphi_c$. This expression is entirely free from cancellation problems. Also it produces no round-off errors when $\varphi \sim 0$ and/or $\varphi \sim \pi/2$.

However, the numerical evaluation of R_J by the duplication method [27–29] consumes a considerable amount of computational time. In fact, the CPU time of its improved algorithm [29,30] runs around 25–80 times more than that of an elementary special function such as `atan` as will be shown later in Table 4. This is significant computational labor if we consider its frequent usage.

3. Method

3.1. Strategy

Let us explain our method to compute the associate incomplete elliptic integral of the third kind, $J(\varphi, n|m)$, when the amplitude, φ, the characteristic, n, and the parameter, m, are in their standard domain: $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. Refer to Appendix for the procedure to reduce the integral of arbitrary input arguments into the standard case.

In order to compute the integral, we (i) use repeatedly the half argument formulas of the sine and cosine amplitude functions so as to reduce the magnitude of φ while keeping n and m unchanged, (ii) calculate an approximate value of the integral based on its Maclaurin series expansion formula, and (iii) recover the integral value corresponding to the original φ by successive applications of the double argument formula of the integral. The resulting method is of the same linear convergence property as Carlson's algorithm based on the duplication theorem and yet is much simpler than it.

On the other hand, the first stage of the method, namely the half argument transformations, are common to other methods of ours to compute (i) $F(\varphi|m)$ solely [21], and (ii) $B(\varphi|m)$ and $D(\varphi|m)$ simultaneously [23]. Also the last stage of the present method, i.e. the double argument formula of $J(\varphi, n|m)$, includes the double argument formulas of $F(\varphi|m)$, $B(\varphi|m)$, and $D(\varphi|m)$ as its sub components. Thus, the new procedure to compute $J(\varphi, n|m)$ contains most of those of $F(\varphi|m)$, $B(\varphi|m)$, and $D(\varphi|m)$ except for the second stage, the Maclaurin series expansion of the integrals for small amplitude. Therefore, the CPU time of simultaneous computation of $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$ is significantly smaller than the sum of CPU times of their separate computation as will be shown later in Section 4.2. This gain in the speed of simultaneous computation is not achieved by the existing formulations of Bulirsch and Carlson.

3.2. Selection rule

Let us be more specific. In a similar manner as we compute $B(\varphi|m)$ and $D(\varphi|m)$ [23], we select one of the following four expressions to compute $J(\varphi, n|m)$ depending on the values of φ and m:

$$J(\varphi, n|m) = \begin{cases} J_s(s, n|m), & (\text{if } \varphi < \varphi_S) \\ J(n|m) - J_s(z, n|m) - T(t_s, h), & (\text{elseif } z^2 < y_S) \\ J_c(c, n|m), & (\text{elseif } c < w) \\ J(n|m) - J_c(w, n|m) - T(t_c, h), & (\text{otherwise}) \end{cases} \tag{22}$$

where

$$s \equiv \sin\varphi, \qquad c \equiv \cos\varphi = \sin\varphi_c, \qquad z \equiv \frac{c}{\sqrt{m_c + mc^2}}, \qquad w \equiv \sqrt{\frac{m_c}{1 - ms^2}}\, s.$$

$$h \equiv nn_c(n - m), \qquad t_s \equiv \frac{sz}{n_c}, \qquad t_c \equiv \frac{\sqrt{\left(1 - c^2\right)\left(1 - w^2\right)}}{n_c}, \tag{23}$$

and

$$J_s(s, n|m) \equiv J\left(\sin^{-1} s, n|m\right), \qquad J_c(c, n|m) \equiv J\left(\cos^{-1} c, n|m\right), \tag{24}$$

are the abbreviations of the integral with different form of the first input argument when $0 < \varphi < \pi/2$. In the above, t_s and t_c is the same variable but with different expressions.

The newly introduced function $T(t, h)$ is defined as

$$T(t, h) \equiv t \sum_{j=0}^{\infty} \frac{\left(-ht^2\right)^j}{2j + 1} = \begin{cases} \tan^{-1}\left(t\sqrt{h}\right)/\sqrt{h} & (h > 0) \\ t & (h = 0) \\ \tanh^{-1}\left(t\sqrt{-h}\right)/\sqrt{-h} & (h < 0). \end{cases} \tag{25}$$

When the hyperbolic arc tangent is not available in the mathematical library, we may use the logarithm instead:

$$T(t, h) = \frac{1}{2\sqrt{-h}} \log\left(\frac{1 + t\sqrt{-h}}{1 - t\sqrt{-h}}\right) \quad (h < 0). \tag{26}$$

The function, $T(t, h)$, is a function to treat universally the arctangent and the hyperbolic arctangent functions [19]. It is related with one of Carlson's fundamental symmetric integrals given in formula 19.3.17 of [4]:

$$R_C(x, y) \equiv \frac{1}{2}\int_0^\infty \frac{dt}{(t+y)\sqrt{t+x}}, \tag{27}$$

such that

$$T(t, h) = R_C\left(t^{-2}, h + t^{-2}\right). \tag{28}$$

Although $T(t, h)$ is essentially the same as $R_C(x, y)$, we prefer using $T(t, h)$ because the simplicity of its Maclaurin series expansion with respect to $z \equiv -ht^2$ leads to its faster computation as will be shown later in Section 3.7 and in Table 5.

On the other hand, y_S and φ_S are constants to govern the above selection rules. We chose y_S as 0.95 and 0.9 in the single and double precision environments, respectively. The latter is computed from the former as $\varphi_S \equiv \sin^{-1}\sqrt{y_S}$ although its exact evaluation is not necessary for the test purpose. Then, we set φ_S as 1.345 and 1.249 in the single and double precision environments, respectively. The discussion to adopt these values are given in [19].

Meanwhile, an efficient procedure to compute $J(n|m)$ is Bulirsch's `cel` [15]:

$$J(n|m) = \texttt{cel}\,(k_c, n_c, 0, 1)\,. \tag{29}$$

The second and last expressions of (22) are derived from the special addition formulas of $F(\varphi|m)$ and $\Pi(\varphi, n|m)$, the second and third[1] parts of formula 117.01 of [2]. See also formulas 19.11.7, 19.11.10, and 19.11.11 of [4].

3.3. Half and double argument transformations

The computation of $J_s(s, n|m)$ is the core part of this formulation. We conduct it by (i) successive applications of the half argument transformation with respect to $y \equiv s^2$ so as to decrease y while keeping n and m the same, (ii) the evaluation of the integral for the decreased y by its truncated Maclaurin series expansion with respect to s, and (iii) the recovery of the integral value for the original s by repeating the application of the double argument transformation of the integral. Similarly, we calculate $J_c(c, n|m)$ by (i) applying successively the half argument transformation with respect to $x \equiv c^2$ so as to increase x while letting n and m unchanged, (ii) calling the routine to compute $J_s(s, n|m)$ described above when the increased x is sufficiently large, and (iii) obtaining the integral value for the original c by repeating the application of the double argument transformation.

Let us be more specific. Consider evaluating the associate integral, $J_0 \equiv J_s\,(s_0, n|m)$. We construct a sequence of y by starting from $y_0 \equiv s_0^2$ and applying successively the half argument transformation to them as

$$y_{i+1} = \frac{y_i}{(1+c_i)\,(1+d_i)}, \quad (i = 0, 1, \ldots), \tag{30}$$

where

$$c_i \equiv \sqrt{1-y_i}, \qquad d_i \equiv \sqrt{1-my_i}. \tag{31}$$

This is a rewriting of the half argument formula of $\mathrm{sn}(u|m)$ given as the first part of formula 124.02 of [2]. See also formula 22.6.19 of [4]. The formula provides a decreasing sequence of y_i since the divisor $(1+c_i)\,(1+d_i)$ is greater than unity. Refer to [21,23]. The intermediate values, y_i, and the products, $c_i d_i$, are stored for later use. The sequence of the transformation is terminated when $y_i < y_B$ where y_B is the critical value set as 0.05909 and 0.01622 in the single and double precision environments, respectively. Refer to Section 3.5 later on the determination of y_B.

Denote by I the index at the termination. Then, we approximate the corresponding integral value, $J_I \equiv J_s\,(s_I|m)$, by its truncated Maclaurin series expansion as

$$J_I \approx s_I \sum_{\ell=0}^{L} J_\ell(n|m) y_I^\ell, \tag{32}$$

where $s_I \equiv \sqrt{y_I}$ and the order of approximate polynomials, L, is 6 and 9 in the single and double precision environments, respectively. We chose these orders since they lead to minimum CPU time. Refer to [21] for the manner of the choice in case of $F(\varphi|m)$. The computation of the expansion coefficients, $J_\ell(n|m)$, will be described in Section 3.4 later.

Finally, we obtain J_0 by conducting the double argument transformation of $J(\varphi, n|m)$ repeatedly as

$$J_{i-1} = 2J_i + T\,(t_i, h)\,, \quad (i = I, I-1, \ldots, 1), \tag{33}$$

where

$$t_i \equiv \frac{s_{i-1}y_i}{1 - n\,(y_{i-1} - c_{i-1}d_{i-1}y_i)}, \tag{34}$$

and

$$s_{i-1} \equiv \sqrt{y_{i-1}}. \tag{35}$$

[1] There is a typo in the sign of the term corresponding to $T(t, h)$.

These formulas are derived from the addition theorems of $F(\varphi|m)$ and $\Pi(\varphi, n|m)$ given in the first part of formula 116.01 and formulas 116.02 and 116.03 of [2]. See also formulas 19.11.1 and 19.11.3 through 19.11.6 of [4].

We note that the double argument transformation of $J(\varphi, n|m)$ includes those of $F(\varphi|m)$, $B(\varphi|m)$, and $D(\varphi|m)$ as its sub component:

$$F_{i-1} = 2F_i, \qquad B_{i-1} = 2B_i - s_{i-1}y_i, \qquad D_{i-1} = 2D_i + s_{i-1}y_i, \quad (i = I, I-1, \ldots, 1), \tag{36}$$

where F_i, B_i, and D_i are the intermediate integral values corresponding to J_i. See (21) of [21] and (16) of [23]. This fact implies that the simultaneous execution of double argument transformation of all these three associate integrals is achieved with a small increase in computational labor.

The above algorithm faces cancellation problems when y_0 and/or m are close to unity. This occurs in the computation of small factors, c_0 and/or d_0. In that case, we use another main variable, $x \equiv 1-y$, and replace the half argument transformation of y with that of x:

$$x_{i+1} = \frac{c_i + d_i}{1 + d_i}, \quad (i = 0, 1, \ldots) \tag{37}$$

where

$$c_i \equiv \sqrt{x_i}, \qquad d_i \equiv \sqrt{m_c + m x_i}, \tag{38}$$

this time. This is a rewriting of the half argument formula of $\mathrm{cn}(u|m)$ given in the second part of formula 124.02 of [2]. See also formula 22.6.20 of [4].

At any rate, it leads to an increasing sequence of x_i. If x_i becomes sufficiently large by repeated usage of this transformation, we shift to y_i by the translation $y_i = 1 - x_i$. We set $x_S \equiv 1 - y_S$ as the critical value of x_i to shift to the computation in terms of y_i. A dangerous divisor n appears nowhere in the above procedure. This assures the robustness of the present algorithm against small values of $|n|$.

3.4. Maclaurin series expansion of $J_s(s, n|m)$

Assume that y becomes sufficiently small, say $y < y_B$ where y_B will be discussed later in Section 3.5. Then, we evaluate the associate integral by its truncated Maclaurin series expansion with respect to $s \equiv \sqrt{y}$ as

$$J_s(s, n|m) \approx s \sum_{\ell=1}^{L} J_\ell(n|m) y^\ell, \tag{39}$$

where L is the order of truncated series expansion, which will be specified in Section 3.5, and the expansion coefficients, $J_\ell(n|m)$, are ℓ-th order bivariate polynomials of n and m:

$$J_\ell(n|m) \equiv \sum_{k=0}^{\ell-1} \sum_{j=0}^{k-1} J_{\ell kj} n^j m^k, \tag{40}$$

as

$$J_1(n|m) = \frac{1}{3}, \qquad J_2(n|m) = \frac{1 + 2n + m}{10}, \qquad J_3(n|m) = \frac{3 + 4n + 8n^2 + (2 + 4n)m + 3m^2}{56}, \ldots. \tag{41}$$

The coefficients, $J_{\ell kj}$, satisfy the symmetry condition

$$J_{\ell kj} = J_{\ell, \ell-1-k-j, j}, \tag{42}$$

such as $J_{320} = J_{300}, J_{421} = J_{401}, J_{622} = J_{612}$, etc. Their non-trivial values are explicitly given for ℓ up to 9 in Table 1. These are obtained by Mathematica [5] by issuing a command such as

```
Series[(EllipticPi[n, ArcSin[s], m] − EllipticF[ArcSin[s], m])/n, {s, 0, 19}].
```

All the coefficients are positive definite, and therefore, no cancellation problems occur in the triple summation since y, n, and m are all positive definite. As a result, the coefficients are bound as

$$J_\ell(0|0) = J_{\ell 00} \le J_\ell(n|m) \le J_\ell(1|1), \tag{43}$$

where the upper bound is explicitly given as

$$J_\ell(1|1) = \frac{\ell}{2\ell + 1}, \tag{44}$$

since it is obtained by the Maclaurin series expansion of an elementary function

$$J(\varphi, 1|1) = \frac{1}{2}\left[\frac{\sin\varphi}{\cos^2\varphi} - \tanh^{-1}(\sin\varphi)\right]. \tag{45}$$

Refer to (A.8).

Table 1
Polynomial coefficients of series expansion of $J_s(s, n|m) \equiv J\left(\sin^{-1} s, n|m\right)$ in $y \equiv s^2$, n, and m. Shown are the numerators and the common denominator of the polynomial coefficients, $J_{\ell kj}$, appearing in the Maclaurin series expansion of the associate incomplete elliptic integral of the third kind with respect to y, n, and m as $J_s(s, n|m) \approx s\sum_{\ell=1}^{L}\sum_{k=0}^{\ell-1}\sum_{j=0}^{k-1} J_{\ell kj} n^j m^k y^\ell$. The coefficients satisfy the symmetry relation, $J_{\ell kj} = J_{\ell,\ell-1-k-j,j}$, such as $J_{210} = J_{200}$, $J_{320} = J_{300}$, and $J_{311} = J_{301}$. Then, only non-trivial coefficients are listed in the increasing order of ℓ and k, and then in the increasing order of j separated by commas. Examples are $J_{100} = 1/3$, $J_{200} = 1/10$, $J_{211} = 2/10$, $J_{300} = 3/56$, $J_{301} = 4/56$, $J_{302} = 8/56$, and $J_{310} = 2/56$.

ℓ	k	Numerators ($j = 0, 1, \ldots$)	Denom.
1	0	1	3
2	0	1, 2	10
3	0	3, 4, 8	56
	1	2	
4	0	5, 6, 8, 16	144
	1	3, 4	
5	0	35, 40, 48, 64, 128	1 408
	1	20, 24, 32	
	2	18	
6	0	63, 70, 80, 96, 128, 256	3 328
	1	35, 40, 48, 64	
	2	30, 36	
7	0	231, 252, 280, 320, 384, 512, 1024	15 360
	1	126, 140, 160, 192, 256	
	2	105, 120, 144	
	3	100	
8	0	429, 462, 504, 560, 640, 768, 1024, 2048	34 816
	1	231, 252, 280, 320, 384, 512	
	2	189, 210, 240, 288	
	3	175, 200	
9	0	6435, 6864, 7392, 8064, 8960, 10 240, 12 288, 16 384, 32 768	622 592
	1	3432, 3696, 4032, 4480, 5120, 6144, 8192	
	2	2772, 3024, 3360, 3840, 4608	
	3	2520, 2800, 3200	
	4	2450	

3.5. Criterion to terminate half argument transformations

Let us examine the condition to terminate the half argument transformation in order to move to the evaluation of the truncated Maclaurin series of $J_s(s, n|m)$. When we truncate the series expansion so as to include up to the term $\ell = L$ as (39), the relative magnitude of the leading error term becomes

$$\frac{J_{L+1}(n|m)s^{2L+3}}{J_1(n|m)s^3} = 3J_{L+1}(n|m)y^L, \tag{46}$$

since $J_1(n|m) = s^3/3$ independently on the value of n and m. By equating this quantity with the machine epsilon, ϵ, and solving it with respect to y, we obtain a critical value in terms of y as a function of n and m for the order L as

$$y_C(n|m) \equiv \left(\frac{\epsilon}{3J_{L+1}(n|m)}\right)^{1/L}. \tag{47}$$

This is a complicated function of n and m. Its exact evaluation requires a significant amount of computational time.

For the purpose to suppress efficiently the truncation error below a certain level, we need a fast procedure to compute a lower bound of $y_C(n|m)$. Then, we replace it by the global lower bound with respect to n and m in the standard domain, namely

$$y_B(L) \equiv y_C(1|1) = \left[\left(\frac{2L+3}{3L+3}\right)\epsilon\right]^{1/L}, \tag{48}$$

where we used the expression, (44). The numerical values of $y_B(L)$ for various values of L are listed in Table 2. In conclusion, if $y < y_B(L)$, then the truncated series expansion up to the L-th term is expected to approximate the integral with a relative error less than ϵ.

Let us consider determining the optimal order of truncation. The number of terms in $J_\ell(n|m)$ increases quadratically with respect to ℓ. Then the total number of terms of the truncated Maclaurin series increases cubically with respect to L. Thus, if L is too large, the CPU time to evaluate the truncated Maclaurin series becomes huge, and therefore the total CPU time will increase. On the other hand, if L is too small, then the corresponding critical value, $y_B(L)$, becomes tiny. As a result, the

Table 2
Critical values required to truncate the series expansion of $J_s(s, n|m) \equiv J\left(\sin^{-1} s, n|m\right)$. The critical value in terms of $y \equiv s^2$ is defined as a function of L, the order of truncation, as $y_B(L) \equiv [\epsilon(2L+3)/(3L+3)]^{1/L}$. Also shown are the ratio in percent of the cases when $y_B(L-1) \le y < y_B(L)$ for the reduced y, which is obtained by repeating the half argument transformations starting from $y_0 \equiv \sin^2 \varphi$. The statistics are taken for a uniform set of the input arguments in their standard domain, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$.

L	Single precision		Double precision	
	$y_B(L)$	Ratio (%)	$y_B(L)$	Ratio (%)
2	2.153E−04	1.5	9.292E−08	0.0
3	3.549E−03	4.8	4.366E−06	0.2
4	1.445E−02	6.7	9.499E−05	0.8
5	3.363E−02	54.8	6.036E−04	1.6
6	5.909E−02	32.2	2.072E−03	2.3
7			5.004E−03	17.0
8			9.696E−03	46.3
9			1.622E−02	31.8

number of half argument transformations increases. Thus, the total CPU time becomes large. These tendencies are opposite. Then, there must be an optimum value of L in the sense to minimize the total CPU time. After some preliminary tests, we confirmed that the optimal value of L is 6 and 9 in the single and double precision environments, respectively. These do coincide with the results of the case of the first and the second incomplete integrals [21,23].

3.6. Efficient evaluation of truncated Maclaurin series expansion

As we noted already, the total number of terms of the truncated Maclaurin series expansion increases cubically with respect to the order of truncation, L. Therefore, even if the chance is small, it is meaningful to design the computer program to decrease the order of truncated series to be used in the actual computation as much as possible. We learn that such a chance is not small by taking the statistics on the actual values of y_I satisfying the criterion, $y_I < y_B(L)$. Table 2 also shows the ratios of occasion where y_I satisfies the condition $y_I < y_B(L)$ for a lower value of L. For example, in case of the single precision computation, the necessity to evaluate the full series where $L = 6$ is roughly one third. Then, we design the program to calculate the truncated Maclaurin series as a long sequence of `if-then-else` structure. This trick significantly reduces the total CPU time of the new method.

3.7. Fast evaluation of $T(t, h)$

Let us consider to speed up another key part of the new formulation: the evaluation of the newly introduced function, $T(t, h)$. Of course, it can be computed by using Carlson's R_C or by calling the arc tangent or logarithm function in the standard mathematical library. However, they require some non-negligible amount of computation as will be shown later in Table 5. Therefore, if the magnitude of its main argument, $z \equiv -ht^2$, is sufficiently small, the evaluation by means of series expansion is rather fast. By conducting a similar research as we did in Section 3.5, we obtained an analytic expression of the critical value of z as a function of the order of truncated series, L:

$$z_C(L) \equiv [(2L+1)\epsilon]^{1/L} \tag{49}$$

which are shown in Table 3. In other words, if $|z| < z_C(L)$, then the relative error of the L-th order truncated Maclaurin series becomes less than the machine epsilon.

One may think that the chance to use the series expansion is small. However, we find that the truth is the opposite. Table 3 also lists the ratio in percentage of the cases when $z_C(L-1) \le |z| < z_C(L)$ as well as those when calling `atan` or `log` becomes faster in the actual computation of $J(\varphi, n|m)$. The statistics shown are the results of counting for the set of three input arguments uniformly distributed in their standard domain, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. The table indicates that the computation is mostly done by low-order series expansions. The necessity to call arc tangent or logarithm functions are less than a few percent.

This is mainly because the function $T(t, h)$ is called during the process of double argument transformation. The value of the sine amplitude, s, starts from a small value and increases exponentially. Therefore, t, which is roughly in proportion to s^3, also starts from a much smaller value and increases exponentially. As a result, the distribution of t is rather uniform in a logarithmic scale. Another reason is the smallness of $|h|$. For the values of n and m in their standard domain, $0 < n < 1$ and $0 < m < 1$, the magnitude of $h \equiv n(1-n)(n-m)$ is bound as $|h| < 4/27$. See Fig. 2 for its behavior as a function of n for various values of m. At any rate, this fact can be used to an acceleration of the computation of $T(t, h)$.

Based on these observations, let us design the program to compute $T(t, h)$ by using maximally its truncated Maclaurin series. We experimentally learn that the averaged computational amounts of the expressions using `atan` and `log` are around 21 and 27 FLOPS, respectively. Since the evaluation of an L-th order polynomial by Horner's method costs $2L$ FLOPS, we use the truncated series as long as (1) $L \le 10$ or (2) $10 < L \le 13$ and $h < 0$. Otherwise, we use the expression in terms

Table 3
Critical values to truncate the series expansion of $T(t, h)$. Listed are the critical values of the main variable, $z \equiv -ht^2$, used in truncating the Maclaurin series expansion of $T(t, h)$. The critical value is defined as a function of L, the order of truncation, as $z_C(L) \equiv [(2L+1)\epsilon]^{1/L}$. Also shown are the ratio in percent of the cases when $z_C(L-1) \le |z| < z_C(L)$ for the pairs of t and h encountered during the double argument transformations of $J(\varphi, n|m)$. The statistics are taken for a uniform set of the input arguments of $J(\varphi, n|m)$ in their standard domain, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. Added also are the ratio of `atan` and `log`, where the former is called when $z > z_C(10)$ while the latter is used when $z < -z_C(13)$.

L	Single precision		Double precision	
	$z_C(L)$	Ratio (%)	$z_C(L)$	Ratio (%)
2	5.459E−04	72.1	2.356E−08	16.5
3	7.472E−03	16.6	9.193E−06	43.0
4	2.706E−02	6.6	1.777E−04	16.4
5	5.798E−02	2.7	1.040E−03	8.4
6	9.583E−02	1.1	3.361E−03	4.7
7	1.367E−01	0.5	7.740E−03	3.3
8	1.781E−01	0.2	1.443E−02	2.3
9	2.184E−01	0.1	2.340E−02	1.6
10	2.568E−01	0.0	3.441E−02	1.0
11			4.713E−02	0.3
12			6.122E−02	0.1
13			7.635E−02	0.1
atan		0.1		2.0
log		0.0		0.3

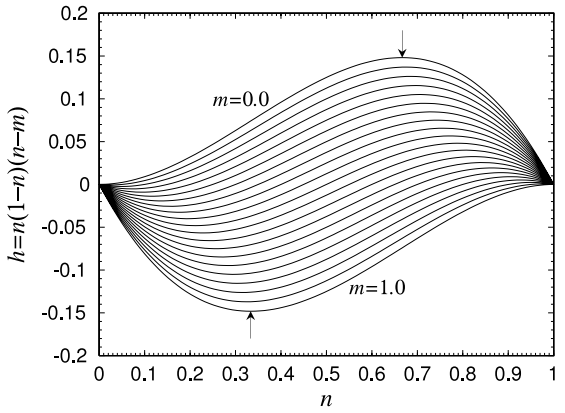

Fig. 2. Factor h. Plotted are the factor $h \equiv n(1-n)(n-m)$ as a function of n for various values of m in the domain $0 \le m \le 1$. The maximum absolute value is $4/27$ which is reached when $m = 0$ and $n = 2/3$ or $m = 1$ and $n = 1/3$ as shown by arrows.

of arc tangent or logarithm functions depending on the sign of h. This is simply achieved by means of a long sequence of `if-then-else` control. As a result, the accelerated program runs 2.9–4.5 times faster than the simple switched call of arc tangent and logarithm functions.

4. Comparison with existing methods

Let us compare the computational cost and performance of the new method described in Section 3 with that of the existing methods explained in Section 2. All the computation codes used in the numerical experiments described below were (i) written in Fortran 77/90, (ii) compiled by the Intel Visual Fortran Composer XE 2011 update 3 with level 3 optimization, and (iii) executed on a PC with an Intel Core i7 processor running at 2.80 GHz under Windows XP.

4.1. Computational precision

First, we investigate the computational errors. Fig. 3 shows the relative errors of $J(\varphi, n|m)$ computed by `rj`, Carlson's duplication method to compute R_J [27], and by `elj`, the new procedure described in Section 3. We omit the errors of `el`, the generalization of Bulirsch's routines [18], since they would be scaled out for small φ. The figure depicts the errors in the double precision environment as functions of φ in its standard domain, $0 < \varphi < \pi/2$. Overlapped are the results for various values of n and m in their standard domain, $0 < n < 1$ and $0 < m < 1$, namely for the cases $n = 0.01, 0.02, \ldots, 0.99$ and $m = 0.01, 0.02, \ldots, 0.99$.

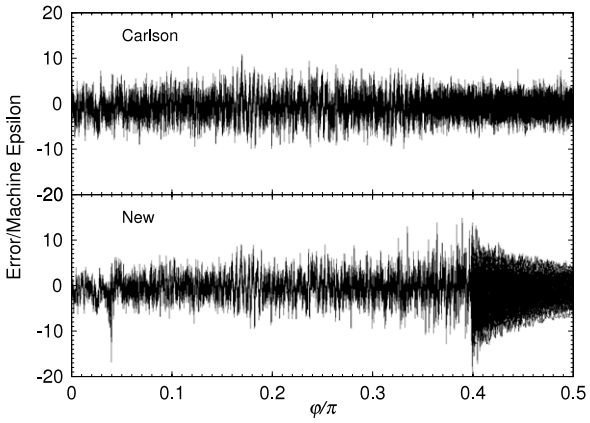

Fig. 3. Amplitude dependence of relative errors of associate incomplete integral of third kind. Shown are the relative errors of $J(\varphi, n|m)$ in the double precision environment obtained by (i) `rj`, Carlson's duplication method to compute R_J [27], and (ii) `elj`, the new procedure explained in Section 3. The errors are plotted as functions of φ in its standard domain, $0 < \varphi < \pi/2$. Overlapped are the results for various values of n and m in the standard domain, $0 < n < 1$ and $0 < m < 1$. A change of manner of errors around $\varphi/\pi \sim 0.4$ in the new method is due to the switch structure in its base algorithm shown in (22).

Table 4
Averaged CPU times to compute $J(\varphi, n|m)$. Listed are the averaged CPU times to compute $J(\varphi, n|m)$ by the new and existing methods. They are obtained as simple means of the CPU times for n and m in their standard domain, $0 < n < 1$ and $0 < m < 1$. The unit of CPU time is a nano second on a PC with an Intel Core i7 processor running at 2.80 GHz.

Method	Procedures	$0 < \varphi < \pi/2$		$-\infty < \varphi < \infty$	
		Single	Double	Single	Double
Duplication	`rj`	192	582	412	1416
Bartky transf.	`el`, `cel`	250	397	499	506
New	`elj`, `cel`	131	298	162	384

The errors are (i) measured as the differences from the quadruple precision results obtained by `qrj`, the quadruple precision extension of `rj`, (ii) normalized by the magnitude of the integrals, (iii) scaled by the machine epsilon, and (iv) illustrated as functions of φ. We confirmed that the errors of `qrj` are far less than the double precision machine epsilon by random check with the results of 40-digits computation by Mathematica [5]. A change of manner of errors around $\varphi/\pi \sim 0.4$ in the new method is due to the switch structure in its base algorithm shown in (22). At any rate, we conclude that the relative errors of `rj` and `elj` are satisfactorily small.

4.2. Computational speed

Next, we measure the CPU times. Table 4 compares the averaged CPU times to compute $J(\varphi, n|m)$ by the new and existing methods in the single and double precision environments for two sets of the amplitude: $0 < \varphi < \pi/2$ and $-\infty < \varphi < \infty$. Although the routine `el` based on Bartky transformation suffers significant cancellation for small values of φ, we added its results for the purpose of reference.

The listed CPU times are those uniformly averaged with respect to φ, n, and m. Actually, in the case $0 < \varphi < \pi/2$, we measured the CPU times as simple means of the results for $1023 \times 1023 \times 1023$ equally-spaced grid points in the standard domain, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$.

The complete integral, $J(n|m)$, is required in calculating the incomplete integral, $J(\varphi, n|m)$, for an arbitrary value of φ. Refer to Appendix A.1. Then, in the case $-\infty < \varphi < \infty$, we measured the total CPU times to compute $J(n|m)$ first and $J(\varphi, n|m)$ next. Since the new procedure partially uses the value of $J(n|m)$, the total CPU time becomes somewhat smaller than the simple sum of that of $J(n|m)$ and that of $J(\varphi, n|m)$. This saving of CPU time is not expected in the existing procedures because of the difference in nature of the algorithms adopted.

The table clearly shows that the new method runs roughly 1.5–3.7 times faster than `rj`, Carlson's duplication method to compute R_J, and 1.3–3.1 times faster than `el`, the generalization of Bulirsch's routines, `el1`, `el2`, and `el3`, developed by us [18] although `el` suffers cancellation problems when φ is small. Since the listed CPU times are in nano seconds on a computer running at 2.8 GHz, the number of FLOPS are directly obtained by multiplying by 2.8. For example, `rj` in the double precision costs $582 \times 2.80 \approx 1630$ FLOPS. This is around 80 times that of a typical elementary function such as `atan`.

The superiority of the new formulation owes to (i) the simplicity of the half and double argument transformations, (ii) the effectiveness of the adopted selection rule, (iii) the minimization of the number of terms in evaluating the truncated

Table 5
Effect of acceleration to compute $T(t, h)$. Listed are the ratio of averaged CPU times to compute $J(\varphi, n|m)$ by the new formulation with different methods to compute $T(t, h)$. Compared methods are (i) the translation in terms of Carlson's R_C written in (28), (ii) a simple switch between `atan` and `log` described in the definition of $T(t, h)$, (25) and (26), and (iii) the new method using Maclaurin series expansion maximally and explained in Section 3.7. Shown are the CPU times of the first two methods divided by those of the last one in the single and double precision environments. The CPU times are averaged for φ, n, and m in their standard domain, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$.

Method	Procedures	Single	Double
Duplication	`rc`	1.85	1.75
Definition	`atan`, `log`	1.43	1.47

Table 6
Averaged CPU times to compute $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$ simultaneously. Same as Table 4 but for the simultaneous computation of three basic incomplete elliptic integrals, $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$.

Method	Procedures	$0 < \varphi < \pi/2$		$-\infty < \varphi < \infty$	
		Single	Double	Single	Double
Duplication	`rd`,`rj`	466	1211	966	2857
Bartky transf.	`el2`, `el`, `cel2`, `cel`	298	856	614	1135
New	`elbdj`, `celbd`, `cel`	176	383	218	486

Maclaurin series of the integral, and (iv) the acceleration of evaluation of $T(t, h)$ by the maximum usage of its Maclaurin series expansions.

Among them, we learn that the last effect is significant. See Table 5 showing the difference in CPU time of the new formulation due to the difference in the methods to compute $T(t, h)$: (i) that using Carlson's R_C, (ii) a simple switch of the arc tangent and logarithm functions, and (iii) the method using Maclaurin series expansion maximally. It is obvious that the new method to compute $T(t, h)$ accelerates the new formulation to evaluate $J(\varphi, n|m)$ by a factor of 1.4 than that using its defining formula calling `atan` and `log` and by a factor of 1.8 than that using Carlson's duplication method to compute R_C.

As we noted in Section 1.2, a sole computation of $J(\varphi, n|m)$ is rather rare. Then, we prepared Table 6 comparing the CPU time to compute $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$ simultaneously. This time, we used the combined procedure `elbdj` in place of the pair of `elj` and `elbd`, the routine we developed to compute $B(\varphi|m)$ and $D(\varphi|m)$ simultaneously [23]. This is because (i) all the procedures share the same main engine, the half argument transformation of Jacobian elliptic functions, and (ii) the double argument transformations of $B(\varphi|m)$ and $D(\varphi|m)$ can be implemented as a sub component of that of $J(\varphi, n|m)$ as we remarked in Section 3.3. Therefore, the CPU time of the combined procedure is significantly smaller than the sum of those to compute them separately.

On the other hand, neither Carlson's duplication theorem nor Bartky transformations of Bulirsch's procedures have this property to save CPU times in the case of the simultaneous computation. As a result, the ratio of CPU times between the new and existing procedures becomes significantly larger: (i) 2.7–5.9 for the pair of `rd` and `rj`, Carlson's duplication methods to compute R_D and R_J, respectively, and (ii) 1.7–2.8 for Bulirsch's procedures and its extension by us, `el2`, `el`, `cel2`, and `cel`, though the latter group suffers serious information loss for small amplitudes. Compare Table 6 with the pair of Table 1 of [23] and Table 4.

5. Conclusion

We developed a new method to compute the associate incomplete elliptic integral of the third kind, $J(\varphi, n|m)$, for the standard domain of input arguments, $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. The computation of the integral for other input arguments are finally reduced to this standard case by means of the special value formulas and various transformations described in Appendix.

The key techniques we used are (i) the combination of half argument formulas of the sine and cosine amplitude functions, (ii) the truncated Maclaurin series expansion of the integral with respect to the sine amplitude, and (iii) the double argument formula of the integral. The new procedure is sufficiently precise in the sense that the maximum relative errors are less than 20 machine epsilons.

Meanwhile, it is significantly faster then the existing procedures: 1.5–3.7 times faster than Carlson's `rj` based on the duplication theorem of the symmetric elliptic integrals. This is greatly owed to the acceleration of a procedure to evaluate $T(t, h) = R_C\left(t^{-2}, h + t^{-2}\right)$, an auxiliary function needed in the double argument formula, by maximally using its Maclaurin series expansion.

As a consequence, the simultaneous computation of all three associate incomplete integrals, $B(\varphi|m)$, $D(\varphi|m)$, and $J(\varphi, n|m)$, is significantly accelerated by our combined procedure, `elbdj`, with the help of our `celbd` [22], that compute three associate complete integrals, $B(m)$ and $D(m)$, as well as Bulirsch's `cel` [26] to calculate $J(n|m)$. In fact, its CPU time is

2.7–5.9 times less than Carlson's `rd` and `rj`. Therefore, the new formulation establishes a precise and fast method to compute arbitrary linear combination of Legendre's incomplete elliptic integrals of all three kinds, $F(\varphi|m)$, $E(\varphi|m)$, and $\Pi(\varphi, n|m)$.

The Fortran programs of `elj` and `elbdj` are available from the author upon request.

Acknowledgments

The author thanks the anonymous referees for their valuable advice to improve the quality of the present article.

Appendix. Domain reduction of input arguments

The associate incomplete elliptic integral of the third kind, $J(\varphi, n|m)$, is real-valued for (i) arbitrary values of φ if $m < 1$ and $n < 1$, and (ii) the values of φ satisfying the condition $|\varphi| < \sin^{-1}\sqrt{1/m}$ if $m \geq 1$ and $n < m$. When $n \geq 1/\sqrt{\sin|\varphi|}$, the integrand has a pole at $\theta = \pm\sin^{-1}\sqrt{1/n}$. In this case, we understand that the integral is defined by its principal value [4].

Under these conditions and understanding, we will explain below a series of procedures to reduce the domain of the amplitude, φ, the characteristic, n, and the parameter, m, into their standard ones: $0 < \varphi < \pi/2$, $0 < n < 1$, and $0 < m < 1$. First, for arbitrary n and m, we will reduce the domain of φ such that $0 < \varphi < \pi/2$ in Appendix A.1. Then, under the condition $0 < \varphi < \pi/2$ and $m\sin^2\varphi \leq 1$, we will reduce the domain of m for arbitrary value of n such that $0 < m < 1$ in Appendix A.2. Finally, under the conditions $0 < \varphi < \pi/2$ and $0 < m < 1$, we will reduce the domain of n such that $0 < n < 1$ in Appendix A.3.

All the formulas shown below are numerically confirmed by using Mathematica, version 5.1 [5].

A.1. Reduction of amplitude

We begin with the domain reduction of φ. If $\varphi = 0$ or $\varphi = \pi/2$, the integral is expressed as

$$J(0, n|m) = 0, \qquad J(\pi/2, n|m) = J(n|m), \tag{A.1}$$

where $J(n|m)$ is the associate complete elliptic integral of the third kind. Its precise and fast computation is realized by Bulirsch's `cel` [25,26].

Next, we assume that $|\varphi| > \pi/2$. This implies that $m < 1$. Then, we transform φ so as to lie in $(-\pi/2, \pi/2)$ by utilizing the amplitude modulus transformation:

$$J(j\pi \pm \varphi, n|m) = 2jJ(n|m) \pm J(\varphi, n|m), \tag{A.2}$$

where j is an integer such that $|\varphi - j\pi| < \pi/2$. This relation is derived from those of $F(\varphi|m)$ and $\Pi(\varphi, n|m)$ given as formulas 113.01 and 113.03 of [2].

Finally, if $\varphi < 0$, we make φ positive by the negative amplitude transformation:

$$J(\varphi, n|m) = -J(-\varphi, n|m), \tag{A.3}$$

which is easily proven by definition. In conclusion, we modify φ so as to satisfy the condition, $0 < \varphi < \pi/2$, for arbitrary n and m.

A.2. Reduction of parameter

Next, under the condition $0 < \varphi < \pi/2$ and $m\sin^2\varphi \leq 1$, consider the reduction with respect to the parameter, m, for arbitrary n.

First, if $m = 0$ and $n \neq 0$, the integral is written in terms of elementary functions as

$$J(\varphi, n|0) = [T(\tan\varphi, n_c) - \varphi]/n, \tag{A.4}$$

where $T(t, h)$ is the function introduced in (25) of the main text. This is a unified rewriting of the second and third parts of formula 111.01 of [2]. See also the second part of formula 19.6.7 and the first part of formula 19.6.12 of [4].

Second, if $m = n = 0$, the integral is expressed by another elementary functions as

$$J(\varphi, 0|0) = D(\varphi|0) = \frac{\varphi}{2} - \frac{\sin 2\varphi}{4}. \tag{A.5}$$

See equation (8.6) of [14]. It is also obtained by taking the limit $n \to 0$ of the Maclaurin expansion of (A.4) with respect to n.

Third, if $m = 1$ and $n \neq 1$, the integral reduces to another form of elementary functions as

$$J(\varphi, n|1) = \left[\tanh^{-1}(\sin\varphi) - T(\sin\varphi, -n)\right]/n_c. \tag{A.6}$$

This is calculated from the second and third parts of formula 111.04 of [2]. See also formula 19.6.8 and the second part of formula 19.6.12 of [4]. Some mathematical function libraries in an ordinary PC contain no implementation of $\tanh^{-1}(s)$. In that case, the function is rewritten in terms of another elementary function as

$$\tanh^{-1}(s) = \frac{1}{2}\log\left|\frac{1+s}{1-s}\right|. \tag{A.7}$$

Fourth, if $m = n = 1$, the integral is also written in terms of elementary functions as

$$J(\varphi, 1|1) = \frac{1}{2}\left[\frac{\sin\varphi}{\cos^2\varphi} - \tanh^{-1}(\sin\varphi)\right]. \tag{A.8}$$

This is derived from formula 2.526.11 of [31] and the second part of formula 111.04 of [2]. See also formula 19.6.8 and the third part of formula 19.6.12 of [4].

Fifth, if $1 < m \leq 1/\sin^2\varphi$, we reduce the domain of m into the standard one, $0 < m < 1$, by the reciprocal parameter transformation[2]:

$$J(\varphi, n|m) = m_R\sqrt{m_R}J(\varphi_R, n_R|m_R), \tag{A.9}$$

where φ_R, n_R, and m_R are defined as

$$\varphi_R \equiv \sin^{-1}\left[\left(\sqrt{m}\right)\sin\varphi\right], \qquad n_R \equiv n/m, \qquad m_R \equiv 1/m. \tag{A.10}$$

This is computed from the first and fourth parts of formula 162.02 of [2]. Also refer to the first and third parts of formula 19.7.4 of [4]. The transformed parameter is in the standard domain as $0 < m_R < 1$. On the other hand, the transformed amplitude, φ_R, is uniquely determined since $|\sqrt{m}\sin\varphi| \leq 1$, and therefore, the arc sine function is real-valued.

Finally, if $m < 0$, we reduce the domain of m into the standard one, $0 < m < 1$, by the negative parameter transformation[3]:

$$J(\varphi, n|m) = m_N\sqrt{m_N}J(\varphi_N, n_N|m_N), \tag{A.11}$$

where φ_N, n_N, and m_N are defined as

$$\varphi_N \equiv \sin^{-1}\left[\left(\sqrt{\frac{m_c}{1-m\sin^2\varphi}}\right)\sin\varphi\right], \qquad n_N \equiv \frac{n-m}{m_c}, \qquad m_N \equiv \frac{-m}{m_c}. \tag{A.12}$$

This is derived from the first and third parts of formula 160.02 of [2]. Also refer to the first and third parts of formula 19.7.5 as well as formula 19.7.6 of [4]. The transformed parameter is in the standard domain as $0 < m_N < 1$. Meanwhile, the transformed amplitude, φ_N, is well-defined since the absolute value of the argument of the arc sine function is less than unity.

No other incomplete elliptic integrals appear in the last two transformation formulas of $J(\varphi, n|m)$. In this sense, they are self-contained,[4] and therefore, simpler than those of $\Pi(\varphi, n|m)$. At any rate, we can reduce the domain of m into the standard one, $0 < m < 1$, by using the above special value formulas and transformations.

A.3. Reduction of characteristics

Finally, we reduce the domain of characteristic, n, under the condition that $0 < \varphi < 1$ and $0 < m < 1$. First, if $n = 0$, the integral reduces to an associate incomplete elliptic integral of the second kind

$$J(\varphi, 0|m) = D(\varphi|m). \tag{A.13}$$

This is simply derived from the definition, (6).

Next, if $n = 1$, the integral reduces to the incomplete elliptic integral of the second kind as

$$J(\varphi, 1|m) = \left[(\tan\varphi)\sqrt{1-m\sin^2\varphi} - E(\varphi|m)\right]/m_c. \tag{A.14}$$

This is given in the second part of formula 111.06 of [2].

---

2 The classic name is the reciprocal modulus transformation.

3 The classic name is the imaginary modulus transformation.

4 The same thing is said for the pair of $B(\varphi|m)$ and $D(\varphi|m)$ as shown in [22].

Third, if $n > 1$, we reduce the domain of n into a sub domain of the standard one, $0 < n < m$, by the first characteristic transformation as

$$J(\varphi, n|m) = \left[-F(\varphi|m) + T(t_1, h_1) - n_1 J(\varphi, n_1|m)\right]/n, \tag{A.15}$$

where

$$t_1 \equiv \frac{\tan\varphi}{\sqrt{1 - m\sin^2\varphi}}, \qquad h_1 \equiv \frac{n_c(n-m)}{n}, \qquad n_1 \equiv \frac{m}{n}. \tag{A.16}$$

This is a universal rewriting of the first part of formula 117.01 and the second part of formula 117.02 of [2]. See also formula 19.7.8 of [4]. The transformed characteristic, n_1, satisfies the condition, $0 < n_1 < m$, since $0 < m < 1$ and $n > 1$. Also, the transformed factor, h_1, is negative definite. Then, the function $T(t_1, h_1)$ can be expressed by the hyperbolic arc tangent. However, we prefer the present form since it is more suitable in implementing the computation code. This is because the series expansion of $T(t, h)$ runs faster than evaluating the arc tangent or the hyperbolic arc tangent when $|h|t^2$ is sufficiently small, say less than 0.1.

Finally, if $n < 0$, we reduce the domain of n into another sub domain of the standard one, $m < n < 1$, by the second characteristic transformation as

$$J(\varphi, n|m) = \left[F(\varphi|m) - T(t_2, h_2) - (m_c/n_c) J(\varphi, n_2|m)\right]/n_c, \tag{A.17}$$

where

$$t_2 \equiv \frac{\sin\varphi\cos\varphi}{\sqrt{1 - m\sin^2\varphi}}, \qquad h_2 \equiv \frac{-n(m-n)}{n_c}, \qquad n_2 \equiv \frac{m-n}{n_c}. \tag{A.18}$$

This is a universal rewriting of the first part of formula 117.01 and the second part of formula 117.03 of [2]. See also formula 19.7.9 of [4]. The transformed characteristic, n_2, satisfies the condition, $m < n_2 < 1$, since $0 < m < 1$ and $n > 1$. Also, the transformed factor, h_2, is positive definite. Then, the function $T(t_2, h_2)$ can be expressed by the arc tangent. Nevertheless, we prefer the present form due to the same reason in the previous case.

Summarizing the above, we can reduce the domain of n into the standard one, $0 < n < 1$, when $0 < \varphi < \pi/2$ and $0 < m < 1$.

References

[1] M. Abramowitz, I.A. Stegun (Eds.), Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 10th Printing, National Bureau of Standards, Washington, 1964, (Chapter 17) 1972.
[2] P.F. Byrd, M.D. Friedman, Handbook on Elliptic Integrals for Engineers and Physicists, 2nd ed., Springer-Verlag, Berlin, 1971.
[3] K.B. Oldham, J. Myland, J. Spanier, An Atlas of Functions, 2nd ed., Hemisphere Publ., Wash., DC, 2009 (Chapters 61, 62, and 63).
[4] F.W.J. Olver, D.W. Lozier, R.F. Boisvert, C.W. Clark (Eds.), NIST Handbook of Mathematical Functions, Cambridge Univ. Press, Cambridge, 2010 (Chapters 19 and 22).
[5] S. Wolfram, The Mathematica Book, 5th ed., Wolfram Research Inc., Cambridge Univ. Press, Cambridge, 2003.
[6] B. Delinchant, F. Wurtz, J.-P. Yonnet, J.-L. Coulomb, Interaction between ring-shaped permanent magnets with symbolic gradients: application to magnetic bearing system optimization, IEEE Trans. Magn. 47 (2011) 1418–1421.
[7] T. Fukushima, Simple, regular, and efficient numerical integration of rotational motion, Astron. J. 135 (2008) 2298–2322.
[8] T. Fukushima, Canonical and universal elements of rotational motion of triaxial rigid body, Astron. J. 136 (2008) 1728–1735.
[9] E.C. Celledoni, N. Säfström, Efficient time-symmetric simulation of torqued rigid bodies using Jacobi elliptic functions, J. Phys. A 39 (19) (2006) 5463–5478.
[10] K.W. Chowa, T.W. Ng, Periodic solutions of a derivative nonlinear Schrödinger equation: elliptic integrals of the third kind, J. Comput. Appl. Math. 235 (2011) 3825–3830.
[11] E.T. Whittaker, G.N. Watson, A Course of Modern Analysis, 4th ed., Cambridge Univ. Press, Cambridge, 1958.
[12] T. Fukushima, Efficient solution of initial-value problem of torque-free rotation, Astron. J. 137 (2009) 210–218.
[13] T. Fukushima, Gaussian element formulation of short-axis-mode rotation of a rigid body, Astron. J. 136 (2008) 649–653.
[14] T. Fukushima, Series expansion of symmetric elliptic integrals, Math. Comp. (2011) (in printing), online-first S0025-5718(2011)02531-7.
[15] R. Bulirsch, Numerical computation of elliptic integrals and elliptic functions, Numer. Math. 7 (1965) 78–90.
[16] S.-D. Lin, L.-F. Chang, H.M. Srivastava, A certain class of incomplete elliptic integrals and associated definite integrals, Appl. Math. Comput. 213 (2009) 1176–1184.
[17] T. Fukushima, Precise computation of acceleration due to uniform ring or disk, Celest. Mech. Dyn. Astron. 108 (2010) 339–356.
[18] T. Fukushima, H. Ishizaki, Numerical computation of incomplete elliptic integrals of a general form, Celest. Mech. Dyn. Astron. 59 (1994) 237–251.
[19] T. Fukushima, Fast computation of Jacobian elliptic functions and incomplete elliptic integrals for constant values of elliptic parameter and elliptic characteristic, Celest. Mech. Dyn. Astron. 105 (2009) 245–260.
[20] T. Fukushima, Fast computation of complete elliptic integrals and Jacobian elliptic functions, Celest. Mech. Dyn. Astron. 105 (2009) 305–328.
[21] T. Fukushima, Fast computation of incomplete elliptic integral of first kind by half argument transformation, Numer. Math. 116 (2010) 687–719.
[22] T. Fukushima, Precise and fast computation of general complete elliptic integral of second kind, Math. Comp. 80 (2011) 1725–1743.
[23] T. Fukushima, Precise and fast computation of a general incomplete elliptic integral of second kind by half and double argument transformations, J. Comput. Appl. Math. 235 (2011) 4140–4148.
[24] R. Bulirsch, Numerical computation of elliptic integrals and elliptic functions II, Numer. Math. 7 (1965) 353–354.
[25] R. Bulirsch, An extension of the Bartky-transformation to incomplete elliptic integrals of the third kind, Numer. Math. 13 (1969) 266–284.
[26] R. Bulirsch, Numerical computation of elliptic integrals and elliptic functions III, Numer. Math. 13 (1969) 305–315.
[27] B.C. Carlson, Computing elliptic integrals by duplication, Numer. Math. 33 (1979) 1–16.
[28] B.C. Carlson, E.M. Notis, Algorithm 577. Algorithms for incomplete elliptic integrals, ACM Trans. Math. Software 7 (1981) 398–403.
[29] B.C. Carlson, Numerical computation of real or complex elliptic integrals, Numer. Algorithms 10 (1995) 13–26.
[30] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes: The Art of Scientific Computing, 3rd ed., Cambridge Univ. Press, Cambridge, 2007.
[31] A. Jeffrey, D. Zwillinger (Eds.), Gradshteyn and Ryzhik's Table of Integrals, Series, and Products, seventh ed., Academic Press, New York, 2007.