ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/346309410

Precise and fast computation of Lambert W function by piecewise minimax rational function approximation with variable transformation

Preprint · November 2020

DOI: 10.13140/RG.2.2.30264.37128

CITATIONS
4

READS
720

1 author:

Toshio Fukushima
National Astronomical Observatory of Japan
778 PUBLICATIONS **4,050** CITATIONS

SEE PROFILE

All content following this page was uploaded by Toshio Fukushima on 25 November 2020.

The user has requested enhancement of the downloaded file.

Precise and fast computation of Lambert W function by piecewise minimax rational function approximation with variable transformation

Toshio Fukushima

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

Abstract

We developed new low- and high-precision procedures for computing the primary and secondary branches of the Lambert W functions, $W_0(z)$ and $W_{-1}(z)$. The procedures for $W_0(z)$ are conditional switches of rational functions of transformed variables: (i) $\ln(z)$ when z is sufficiently large, and (ii) $\sqrt{z+1/e}$ otherwise. Also, the procedures for $W_{-1}(z)$ are similar switches of rational functions of additional variables: (iii) $\ln(-z)$ when $|z|$ is sufficiently small, and (iv) $-z/\left(\sqrt{1/e}+\sqrt{z+1/e}\right)$ otherwise. Numerical experiments confirmed that the new low- and high-precision procedures are of 24- and 50-bit accuracies, respectively, for an arbitrary value of z representable as a double-precision floating-point number. In average, the new high-precision procedures for $W_0(z)$ and $W_{-1}(z)$ run 1.8 and 1.7 times faster than the existing fastest procedures (Fukushima 2013) and the new low-precision procedures are 15 and 41% faster than the new high-precision ones.

Keywords: Lambert W function; minimax approximation; piecewise rational function; variable transformation

1. Introduction

The Lambert W function, $W(z)$, is defined as the solution of a transcendental equation [1, § 4.13]:

$$f(W) \equiv We^W - z = 0. \tag{1}$$

Email address: Toshio.Fukushima@nao.ac.jp (Toshio Fukushima)

It has many branches as a complex-valued function [2]. However, only two of them are meant for real numbers: the primary and the secondary ones denoted by $W_0(z)$ and $W_{-1}(z)$, respectively.

A comprehensive explanation of the function is found in the literature [3]. Also, the approximation of $W_0(z)$ is well studied [4, 5]. In short, their key properties are summarized below:

$$-\infty < W_{-1}(z) \leq W_{-1}(-1/e) = -1 = W_0(-1/e) \leq W_0(z) < +\infty, \qquad (2)$$

See Figs 1 and 2 for their functional behavior.

Although the range of the function value is infinitely wide in principle, its practically-meaningful restriction is not so in the IEEE745 double-precision environment as that of the logarithm function as

$$W_0(z) \leq W_0\left(\omega_{\mathrm{D}}\right) \approx +703.227, \;\; W_{-1}(z) \geq W_{-1}\left(-\eta_{\mathrm{D}}\right) \approx -714.196, \qquad (3)$$

where ω_{D} and η_{D} are the maximum finite and minimum normal values of representable double-precision floating-point numbers [6] defined as

$$\omega_{\mathrm{D}} \equiv 2^{1024} - 2^{971} \approx 1.798 \times 10^{308}, \;\; \eta_{\mathrm{D}} \equiv 2^{-1022} \approx 2.225 \times 10^{-308}. \qquad (4)$$

The function has a long history dating back to the days of Lambert and Euler [7, 8, 9]. Since then, it has been used in various fields of science and technology [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28]. Indeed, a conference devoted to the function itself was held in 2016.

There exist several works on their numerical computation [29, 30]. Among them, remarkable achievements are (i) that for $W_0(z)$ when $z \geq 0$ [31], (ii) that for $W_{-1}(z)$ [32], and (iii) those for both branches [33, 34, 35]. Very recently proposed were (iv) an approach using the power series extension method (PSEM) [36] and (v) an algorithm suitable for arbitrary-precision arithmetics [37]. However, the PSEM realization for $W_{-1}(z)$ has a severe loss of information when the argument z is close to its branch point, $-1/e \approx -0.368$, say when $z < -0.35$, and therefore when $W \to -1/e - 0$ as illustrated in Fig. 3. Also, all of the existing procedures except the TOMS algorithm 743 with the modified input argument, $z_c \equiv z + 1/e$, suffer from a significant increase of relative errors in a central region, say when $|W + 1| < 0.1$ as indicated in Figs 3 and 4. Furthermore, even this exceptional method experiences a precision degradation in a couple of regions outside the central region, namely when $-1.4 < W < -1.02$ and $-0.98 < W < -0.8$, respectively as shown in Fig 5. On the other hand, the CPU time of these existing

Table 1: CPU time comparison: $W_0(z)$. Listed are the averaged CPU time computing $W_0(z)$ by various methods for a practical function value interval, $-1 < W \leq +20$. The unit of the CPU times is nanosecond at a commercial PC with the Intel Core i9-9980HK CPU running at 2.40 to 5.00 GHz clock. All the computer programs were coded in Fortran 90 and compiled by the Intel Visual Fortran Compiler 19.1, namely the so-called `ifort 19.1`, with the maximum optimization flags. Then, the CPU times are averaged over $2^{30} \approx 1.07 \times 10^9$ executions for the arguments the function value of which are evenly spaced in the function value interval. Compared are six procedures: (i) the TOMS algorithm 743 with z as the input argument [33], (ii) the method of Veberic [34], (iii) the previous method of ours [35] denoted by `Fukushima`, (iv) the method of Vazquez-Leal et al. [36] denoted by `PSEM`, (v) the new low-precision procedure denoted by `low`, and (vi) the new high-precision procedure denoted by `high`. As a reference, added is the averaged CPU time evaluating $\ln(1+|z|)$ by using the Intel standard mathematical library. We did not include the TOMS algorithm 443 [31] since it works only when $z > 0$, and therefore $W > 0$.

procedure	ns
low	12.8
high	14.7
log	25.1
Fukushima	26.6
PSEM	58.8
Veberic	67.0
TOMS743	99.9

Table 2: CPU time comparison: $W_{-1}(z)$. Same as Table 1 but evaluating $W_{-1}(z)$ for a practical function value interval, $-10 \le W < -1$. This time, added is the method of Chapeau-Blondeau & Monir [32].

procedure	ns
low	14.8
high	20.9
log	24.2
Fukushima	35.2
PSEM	52.9
TOMS743	62.4
Veberic	113.0
Chapeau-Brondeau & Monir	120.5

procedures is not so small when compared with that of the logarithm function evaluation as listed in Tables 1 and 2 if noted a fact that the typical interval of function values in the practical applications is $-10 \le W \le 20$. Furthermore, only a limited accuracy, say 3 to 6 effective digits, is usually required in practical problems. In this sense, the double-precision accuracy is too much and the single-precision accuracy is far more than enough in many cases. These issues are problematic if considering the frequent and/or low-precision use of the function in the actual applications. To improve the current situation, we developed new low- and high-precision procedures for computing $W_0(z)$ and $W_{-1}(z)$ by utilizing the minimax rational function approximation [39, 40, 41, 42, 43]. The new procedures return the answers with the 24- and 50-bit accuracies, respectively, for all input arguments representable by double-precision floating-point numbers. See Figs 3, 4, and 6.

Also, the new procedures run significantly faster than the existing methods as already listed in Tables 1 and 2. Indeed, the averaged CPU time of the new high-precision procedure for practical arguments is significantly smaller than that of a single evaluation of the logarithm function provided by the standard mathematical library.

We notice that this success was enabled by making appropriate variable transformations before the approximation as we experienced in approximating the complete elliptic integrals [41]. Below, we explain the new method

in §2 and present the results of numerical experiments in §3.

2. Method

2.1. Minimax approximation

We evaluate the primary and secondary branches of the Lambert W functions, $W_0(z)$ and $W_{-1}(z)$, by a conditional switch of rational functions of transformed variables:

$$W_0(z) = \begin{cases} X_k(x), & (z_{k-1} \le z < z_k, \;\; k = 1, 2, \cdots, 17)\,, \\ U_k(u), & (z_{k-1} \le z < z_k, \;\; k = 18, 19)\,, \end{cases} \tag{5}$$

$$W_{-1}(z) = \begin{cases} Y_k(y), & (z_{k+1} \le z < z_k, \;\; k = -1, -2, \cdots, -7)\,, \\ V_k(v), & (z_{k+1} \le z < z_k, \;\; k = -8, -9, -10)\,, \end{cases} \tag{6}$$

where x, u, y, and v are new variables calculated as

$$x \equiv \sqrt{z - z_0}, \;\; u \equiv \ln z, \;\; y \equiv -z/\left(x + x_0\right), \;\; v \equiv \ln(-z), \tag{7}$$

where z_0 and x_0 are constants defined and numerically approximated as

$$z_0 \equiv -1/e \approx -0.36787944117144232160, \tag{8}$$

$$x_0 \equiv \sqrt{-z_0} \approx 0.60653065971263342360. \tag{9}$$

In the above, z_k are critical arguments listed in Tables 3 and 4, $X_k(x)$, $U_k(u)$, $Y_k(y)$, or $V_k(v)$ are rational functions of type (N, M) expressed as

$$X_k(x) \equiv \frac{\sum_{n=0}^{N} P_{kn} x^n}{\sum_{m=0}^{M} Q_{km} x^m}, \;\; U_k(u) \equiv \frac{\sum_{n=0}^{N} P_{kn} u^n}{\sum_{m=0}^{M} Q_{km} u^m}, \tag{10}$$

$$Y_k(y) \equiv \frac{\sum_{n=0}^{N} P_{kn} y^n}{\sum_{m=0}^{M} Q_{km} y^m}, \;\; V_k(v) \equiv \frac{\sum_{n=0}^{N} P_{kn} v^n}{\sum_{m=0}^{M} Q_{km} v^m}, \tag{11}$$

where P_{kn} and Q_{km} are the numerical coefficients provided in Tables 5 through 9 for the low-precision procedure, and Tables 10 through 16 for the high-precision procedure, respectively, while Q_{k0} is set to unity for the normalization as

$$Q_{k0} \equiv 1. \tag{12}$$

The type of the rational functions in the low-precision procedure is $(3, 3)$ except for $X_1(x)$ where the type is $(4, 3)$ and that in the high-precision procedures is $(7, 7)$ except for $X_1(x)$ where the type is $(8, 7)$.

Table 3: Critical arguments: $W_0(z)$. Shown are z_k, the critical arguments to switch the approximation forms computing $W_0(z)$, in the new low- and high-precision procedures. The numbers of displayed digits are a few more than enough to avoid unnecessary round-off errors in the implementation into computer codes. Note that both of z_{19} in the low- and high-precision procedures are larger than the largest representable double-precision floating-point number, $\omega_{\rm D} \equiv 2^{1024} - 2^{971} \approx 1.798 \times 10^{308}$.

k	low	high
1	+2.008217812	+2.1820144653320312500
2	+3.053914211E+1	+4.3246045021497925573E+1
3	+3.716698437E+2	+5.9808565427761132714E+2
4	+4.705918954E+3	+8.0491241056345904686E+3
5	+6.464079736E+4	+1.1112458624177664276E+5
6	+9.656490309E+5	+1.5870426133287885398E+6
7	+1.559333423E+7	+2.3414708033996018338E+7
8	+2.702564028E+8	+3.5576474271222021108E+8
9	+4.995018740E+9	+5.5501716292484833443E+9
10	+9.791115442E+10	+8.8674704839289895890E+10
11	+2.025975386E+12	+1.4477791865269224022E+12
12	+4.407744425E+13	+2.4111458632511484051E+13
13	+1.004838215E+15	+4.0897036442600808776E+14
14	+2.393255260E+16	+7.0555901476789968723E+15
15	+5.939799660E+17	+1.2366607557976727250E+17
16	+1.532693859E+19	+2.1999373487930999771E+18
17	+4.103565940E+20	+3.9685392198344016155E+19
18	+2.172370661E+141	+1.4127075145274652069E+104
19	+1.986615351E+2222	+2.8134195736211426913E+618

Table 4: Critical arguments: $W_{-1}(z)$. Same as Table 3 but for $W_{-1}(z)$. Note that both of $|z_{-10}|$ in the low- and high-precision procedures are smaller than the least positive representable double-precision floating-point number, $\eta_{\mathrm{D}} \equiv 2^{-1022} \approx 2.225 \times 10^{-308}$.

k	low	high
-1	$-2.072937776\mathrm{E}{-1}$	$-1.8872688282289434049\mathrm{E}{-1}$
-2	$-7.150770508\mathrm{E}{-2}$	$-6.0497597226958343647\mathrm{E}{-2}$
-3	$-2.070441262\mathrm{E}{-2}$	$-1.7105334740676008194\mathrm{E}{-2}$
-4	$-5.480012945\mathrm{E}{-3}$	$-4.5954962127943706433\mathrm{E}{-3}$
-5	$-1.367466989\mathrm{E}{-3}$	$-1.2001610672197724173\mathrm{E}{-3}$
-6	$-3.261422673\mathrm{E}{-4}$	$-3.0728805932191499844\mathrm{E}{-4}$
-7	$-7.490661204\mathrm{E}{-5}$	$-7.7447159838062184354\mathrm{E}{-5}$
-8	$-1.096244453\mathrm{E}{-19}$	$-4.5808119698158173174\mathrm{E}{-17}$
-9	$-2.509609930\mathrm{E}{-136}$	$-6.1073672236594792982\mathrm{E}{-79}$
-10	$-2.023167727\mathrm{E}{-2155}$	$-2.3703540064502081009\mathrm{E}{-453}$

2.2. Background

The variable transformation leading to the expressions of x, u, and v was hinted from the Taylor series expansion around the branch point [5]

$$W_0(z) = -1 + \sqrt{2e}x - \frac{3e}{2}x^2 + \cdots, \quad W_{-1}(z) = -1 - \sqrt{2e}x - \frac{3e}{2}x^2 - \cdots, \tag{13}$$

and the asymptotic expansions [1, Eqs (4.13.9) and (4.13.10)] expressed as

$$W_0(z) = u - \ln u + \cdots, \quad W_{-1}(z) = v - \ln(-v) + \cdots. \tag{14}$$

Meanwhile, the direct usage of x in approximating $W_{-1}(z)$ experiences a significant loss of effective digits when $z \approx 0$ as will be explained later. To avoid this information loss, we introdued additional new variable y as an offset value of x as

$$y \equiv x - x_0, \tag{15}$$

such that $y \to 0$ when $z \to 0$. Since this defining form of y suffers from the round-off errors, we rewrite it into the cancellation-free form as

$$y = -z/\left(x + x_0\right), \tag{16}$$

Table 5: Coefficients of rational functions: low-precision procedure for $W_0(z)$, part 1. Listed are the coefficients of rational function used in the new low-precision procedure, P_{kn} and Q_{kn} when $k = 1$ through 7. The argument of the rational functions is $x \equiv \sqrt{z + 1/e}$ such that $W_0(z) = \left(\sum_n P_{kn} x^n\right) / \left(\sum_n Q_{kn} x^n\right)$. The numbers of displayed digits are a few more than enough so as to avoid unnecessary round-off errors in the implementation into computer codes.

k	n	P_{kn}	Q_{kn}
1	0	−9.999999404E−1	+1
	1	+5.573005216E−2	+2.275906560
	2	+2.126973249	+1.367597014
	3	+8.135112368E−1	+1.861582345E−1
	4	+1.632488015E−2	
2	0	−9.855197091E−1	+1
	1	+1.077497573	+1.186101404
	2	+8.717510307E−1	+2.499629843E−1
	3	+5.435272861E−2	+6.881368649E−3
3	0	−7.623971135E−1	+1
	1	+1.231773161	+5.793862150E−1
	2	+2.434244711E−1	+4.660142774E−2
	3	+4.320601394E−3	+4.351281757E−4
4	0	+8.580124743E−2	+1
	1	+8.253979810E−1	+2.133807682E−1
	2	+3.978196076E−2	+5.462672040E−3
	3	+1.878557893E−4	+1.544953448E−5
5	0	+1.621924538	+1
	1	+3.886914513E−1	+6.521946074E−2
	2	+4.575064268E−3	+4.788276079E−4
	3	+5.538467215E−6	+3.809482815E−7
6	0	+3.621899609	+1
	1	+1.488464675E−1	+1.798565932E−2
	2	+4.246962241E−4	+3.544644976E−5
	3	+1.279017971E−7	+7.506249296E−9
7	0	+5.907336974	+1
	1	+5.005365359E−2	+4.655899002E−3
	2	+3.407214863E−5	+2.344944586E−6
	3	+2.481206469E−9	+1.263142996E−10

Table 6: Coefficients of rational functions: low-precision procedure for $W_0(z)$, part 2. Same as Table 5 but when $k = 8$ through 14.

k	n	P_{kn}	Q_{kn}
8	0	+8.382600585	+1
	1	+1.536034648E−2	+1.150742322E−3
	2	+2.443338440E−6	+1.422142847E−7
	3	+4.185680326E−11	+1.873917203E−12
9	0	+1.099667480E+1	+1
	1	+4.394213890E−3	+2.738375768E−4
	2	+1.596666535E−7	+8.015706232E−9
	3	+6.266538284E−13	+2.495698216E−14
10	0	+1.371983380E+1	+1
	1	+1.187444381E−3	+6.305637242E−5
	2	+9.630338120E−9	+4.235876603E−10
	3	+8.443452423E−15	+3.020540501E−16
11	0	+1.653311948E+1	+1
	1	+3.058312575E−4	+1.409916121E−5
	2	+5.411294663E−10	+2.112109541E−11
	3	+1.034713033E−16	+3.352692716E−18
12	0	+1.942351926E+1	+1
	1	+7.555926976E−5	+3.069209279E−6
	2	+2.853002312E−11	+9.986661305E−13
	3	+1.162962710E−18	+3.437671712E−20
13	0	+2.238157605E+1	+1
	1	+1.799472403E−5	+6.518396281E−7
	2	+1.419487642E−12	+4.495866571E−14
	3	+1.207110515E−20	+3.275542925E−22
14	0	+2.540010542E+1	+1
	1	+4.146737839E−6	+1.352980136E−7
	2	+6.696269722E−14	+1.933608179E−15
	3	+1.163790516E−22	+2.914939620E−24

Table 7: Coefficients of rational functions: low-precision procedure for $W_0(z)$, part 3. Same as Table 5 but when $k = 15$ through 19. Note that the argument of the rational functions is altered as $u \equiv \ln z$ when $k = 18$ and 19 such that $W_0(z) = \left(\sum_n P_{kn} u^n\right) / \left(\sum_n Q_{kn} u^n\right)$.

k	n	P_{kn}	Q_{kn}
15	0	+2.847345563E+1	+1
	1	+9.274682469E−7	+2.748648970E−8
	2	+3.006899016E−15	+7.967898707E−17
	3	+1.047355759E−24	+2.433166637E−26
16	0	+3.159705544E+1	+1
	1	+2.018422528E−7	+5.472394513E−9
	2	+1.289578820E−16	+3.153772918E−18
	3	+8.836117471E−27	+1.912203513E−28
17	0	+3.476712449E+1	+1
	1	+4.283079924E−8	+1.068930113E−9
	2	+5.297588412E−18	+1.201669906E−19
	3	+7.014551539E−29	+1.419524481E−30
18	0	−6.070237337E−1	+1
	1	+6.982871632E−1	+7.904842997E−2
	2	+7.579513508E−2	+5.176099090E−4
	3	+5.166925608E−4	−4.243840393E−10
19	0	−3.132005603	+1
	1	+9.488946573E−1	+8.365681868E−3
	2	+8.317815296E−3	+5.559715494E−6
	3	+5.558784816E−6	−3.7481535833E−14

Table 8: Coefficients of rational functions: low-precision procedure for $W_{-1}(z)$, part 1. Same as Table 5 but for $W_{-1}(z)$ when $k = -1$ through -7. The argument of the rational functions is changed to $y \equiv -z/\left(\sqrt{1/e} + \sqrt{z + 1/e}\right)$ such that $W_0(z) = \left(\sum_n P_{kn} y^n\right) / \left(\sum_n Q_{kn} y^n\right)$.

k	n	P_{kn}	Q_{kn}
−1	0	−6.383722823	+1
	1	−7.496865326E+1	+2.429583695E+1
	2	−1.971482155E+1	+6.411246061E+1
	3	+7.067732667E+1	+1.799449737E+1
−2	0	−7.723328481	+1
	1	−3.524846910E+2	+7.768124259E+1
	2	−1.242008890E+3	+6.485643121E+2
	3	+1.171647596E+3	+5.667015498E+2
−3	0	−9.137773142	+1
	1	−1.644724479E+3	+2.723752614E+2
	2	−2.810509610E+4	+7.929224261E+3
	3	+3.896079810E+3	+2.398012286E+4
−4	0	−1.060338824E+1	+1
	1	−7.733348521E+3	+1.021793857E+3
	2	−5.754824071E+5	+1.113002292E+5
	3	−2.154552604E+6	+1.261425640E+6
−5	0	−1.210869927E+1	+1
	1	−3.689653511E+4	+4.044975306E+3
	2	−1.183112672E+7	+1.741827762E+6
	3	−2.756583081E+8	+7.843690738E+7
−6	0	−1.364676194E+1	+1
	1	−1.790861159E+5	+1.67438266E+4
	2	−2.508463494E+8	+2.98096509E+7
	3	−2.934370049E+10	+5.57395148E+9
−7	0	−1.521295814E+1	+1
	1	−8.849546880E+5	+7.20092555E+4
	2	−5.529815438E+9	+5.50590078E+8
	3	−3.093418744E+12	+4.43248949E+11

Table 9: Coefficients of rational functions: low-precision procedure for $W_{-1}(z)$, part 2. Same as Table 8 when $k = -8$ through -10. Here the argument of the rational functions is $v \equiv \ln(-z)$ such that $W_0(z) = \left(\sum_n P_{kn} v^n\right) / \left(\sum_n Q_{kn} v^n\right)$.

k	n	P_{kn}	Q_{kn}
-8	0	$-3.240116318\mathrm{E}{-2}$	$+1$
	1	$+2.028194214$	$-4.500427444\mathrm{E}{-1}$
	2	$-5.275243124\mathrm{E}{-1}$	$+1.715470575\mathrm{E}{-2}$
	3	$+1.734029477\mathrm{E}{-2}$	$-5.243819620\mathrm{E}{-7}$
-9	0	-1.441124660	$+1$
	1	$+1.281926964$	$-7.200087372\mathrm{E}{-2}$
	2	$-7.497935611\mathrm{E}{-2}$	$+4.754893299\mathrm{E}{-4}$
	3	$+4.763630916\mathrm{E}{-4}$	$-4.171497925\mathrm{E}{-10}$
-10	0	-3.310876091	$+1$
	1	$+1.050067881$	$-8.189272743\mathrm{E}{-3}$
	2	$-8.236749582\mathrm{E}{-3}$	$+5.528007601\mathrm{E}{-6}$
	3	$+5.528956159\mathrm{E}{-6}$	$-3.922277308\mathrm{E}{-14}$

by utilizing the identity formula, $(x - x_0)(x + x_0) = x^2 - x_0^2 = -z$.

The global minimax rational function approximation determining z_k was performed by following the previous methods of ours [39, 40, 41, 42, 43]. More precisely speaking, we sequentially maximize z_k starting from z_0 such that the maximum error of the k-th rational function determined by the local minimax approximation, namely that in the interval, $[z_{k-1}, z_k]$ or $[z_{k+1}, z_k]$, are almost the same as the single- or double-precision machine epsilons, respectively.

Each of the local minimax approximation determining the coefficients of the k-th rational function, P_{kn} and Q_{kn}, was conducted by utilizing the Mathematica [44] extensively. Actually, the approximation was executed by its command `MiniMaxApproximation` where $W_0(z)$ and $W_{-1}(z)$ are realized by the commands `ProductLog[z]` and `ProductLog[-1,z]`, respectively.

As for the type of rational functions, (N, M), we first tried the case $N = M = 3$ for the low-precision approximation, and $N = M = 7$ in the high-precision one. For the case of $X_1(x)$, however, naive usage of the command `MiniMaxApproximation` fails because the function has a zero in

Table 10: Coefficients of rational functions: high-precision procedure for $W_0(z)$, part 1. Same as Table 5 but for the new high-precision procedure when $k = 1$ through 4.

k	n	P_{kn}	Q_{kn}
1	0	−9.9999999999999988900E−1	+1
	1	−2.7399668668203659304	+5.0716108484174280050
	2	+2.6164207726990399347E−2	+9.9868388183545283370
	3	+6.3709168078949009170	+9.6607551922078869080
	4	+7.1013286517854026680	+4.7943728991336119052
	5	+2.9800826783006852573	+1.1629703477704522300
	6	+4.8819596813789865043E−1	+1.1849462500733755233E−1
	7	+2.3753035787333611915E−2	+3.4326525132402226488E−3
	8	+7.7365760093772431000E−5	
2	0	−9.9997801800578916749E−1	+1
	1	−7.0415751590483602272E−1	+3.0356026828085410884
	2	+2.1232260832802529071	+3.1434530151286777057
	3	+2.3896760702935718341	+1.3723156566592447275
	4	+7.7765311805029175244E−1	+2.5844697415744211142E−1
	5	+8.9686698993644741433E−2	+1.9551162251819044265E−2
	6	+3.3062485753746403559E−3	+4.8775933244530123101E−4
	7	+2.5106760479132851033E−5	+2.3165116841073152717E−6
3	0	−9.8967420337273506393E−1	+1
	1	+5.9587680606394382748E−1	+1.6959402394626198052
	2	+1.4225083018151943148	+8.0968573415500900896E−1
	3	+4.4882889168323809798E−1	+1.4002034999817021955E−1
	4	+4.4504943332390033511E−2	+9.3571878493790164480E−3
	5	+1.5218794835419578554E−3	+2.3251487593389773464E−4
	6	+1.6072263556502220023E−5	+1.8060170751502988645E−6
	7	+3.3723373020306510843E−8	+2.5750667337015924224E−9
4	0	−7.7316491997206225517E−1	+1
	1	+1.1391333504296703783	+7.4657287456514418083E−1
	2	+4.3116117255217074492E−1	+1.2629777033419350576E−1
	3	+3.5773078319037507449E−2	+6.9741512959563184881E−3
	4	+9.6441640580559092740E−4	+1.4089339244355354892E−4
	5	+8.9723854598675864757E−6	+1.0257432883152943078E−6
	6	+2.5623503144117723217E−8	+2.2902687190119230940E−9
	7	+1.4348813778416631453E−11	+9.2794231013264501664E−13

Table 11: Coefficients of rational functions: high-precision procedure for $W_0(z)$, part 2. Same as Table 10 when $k = 5$ through 8.

k	n	P_{kn}	Q_{kn}
5	0	+1.200710167155368843 0E−1	+1
	1	+8.335264082991282289 6E−1	+2.539673884561912663 0E−1
	2	+7.014277591694833758 2E−2	+1.283923890733031739 3E−2
	3	+1.484635798547512484 9E−3	+2.027537563251099737 1E−4
	4	+1.047875736611015529 0E−5	+1.148295607344914138 4E−6
	5	+2.571589298707103852 7E−8	+2.318837060567426364 7E−9
	6	+1.938421447960647474 9E−11	+1.427199416574256341 9E−12
	7	+2.844704903913940965 2E−15	+1.588483694239479696 1E−16
6	0	+1.7221104439937710112	+1
	1	+3.991959428648427560 5E−1	+7.400743811802054300 8E−2
	2	+7.988554014068502893 7E−3	+1.033350150669774054 5E−3
	3	+4.288974225325792054 1E−5	+4.436085803572750850 6E−6
	4	+7.814682818052986498 1E−8	+6.782291231637104157 0E−9
	5	+4.981963876435468235 9E−11	+3.683435670763949202 1E−12
	6	+9.765088971426529460 6E−15	+6.083615956026604116 8E−16
	7	+3.705299728172172443 9E−19	+1.814986933598122531 6E−20
7	0	+3.7529314023434544256	+1
	1	+1.549134269035780652 5E−1	+2.011298533885444355 5E−2
	2	+7.566314067590078450 5E−4	+7.471228615483014176 8E−5
	3	+1.027160923596997905 9E−6	+8.480059800369383746 9E−8
	4	+4.785324767593006615 0E−10	+3.418242413037691176 2E−11
	5	+7.832804077027547441 0E−14	+4.886625913969095789 9E−15
	6	+3.943303375839103665 3E−18	+2.122337362683463417 8E−19
	7	+3.823286220566028397 8E−23	+1.664298567126058251 5E−24
8	0	+6.0196542055606555577	+1
	1	+5.349667284179786476 2E−2	+5.280968370423337167 5E−3
	2	+6.434084927531650151 9E−5	+5.102050121938955808 2E−6
	3	+2.196909010009596748 5E−8	+1.501831229227083210 3E−9
	4	+2.592798893703306107 0E−12	+1.567770663641318837 9E−13
	5	+1.077919816180152730 8E−16	+5.799204123891187836 1E−18
	6	+1.378042409101789830 1E−21	+6.513317077032078025 9E−23
	7	+3.376897315074255280 2E−27	+1.320508013921340607 1E−28

Table 12: Coefficients of rational functions: high-precision procedure for $W_0(z)$, part 3. Same as Table 10 when $k = 9$ through 12.

k	n	P_{kn}	Q_{kn}
9	0	+8.4280268500989701597	+1
	1	+1.7155758546279713315E−2	+1.3572006754595300315E−3
	2	+5.0836620669829321508E−6	+3.3535243481426203694E−7
	3	+4.3354903691832581802E−10	+2.5206969246421264128E−11
	4	+1.2841017145645583385E−14	+6.7136226273060530496E−16
	5	+1.3419106769745885927E−19	+6.3324226680854686574E−21
	6	+4.3101698455492225750E−25	+1.8128167400013774194E−26
	7	+2.6422433422088187549E−31	+9.3662030058136796889E−33
10	0	+1.0931063230472498189E+1	+1
	1	+5.2224234540245532982E−3	+3.432870255119757797E−4
	2	+3.7996105711810129682E−7	+2.139535151853844476E−8
	3	+8.0305793533410355824E−12	+4.0524170186631594159E−13
	4	+5.9139785627090605866E−17	+2.7181424315335710420E−18
	5	+1.5382020359533028724E−22	+6.4538986638355490894E−24
	6	+1.2288944126268109432E−28	+4.6494613785888987942E−30
	7	+1.8665089270660122398E−35	+6.0442024367299387616E−37
11	0	+1.3502943080893871412E+1	+1
	1	+1.5284636506346264572E−3	+8.570151287908946255E−5
	2	+2.7156967358262346166E−8	+1.3311244435752691563E−9
	3	+1.4110394051242161772E−13	+6.2788924440385347269E−15
	4	+2.5605734311219728461E−19	+1.0483788152252204824E−20
	5	+1.6421293724425337463E−25	+6.1943499966249160886E−27
	6	+3.2324944691435843553E−32	+1.1101567860340917294E−33
	7	+1.2054662641251783155E−39	+3.5897381128308962590E−41
12	0	+1.6128076167439014775E+1	+1
	1	+4.3360385176467069131E−4	+2.1154255263102938752E−5
	2	+1.8696403871820916466E−9	+8.1006115442323280538E−11
	3	+2.3691795766901486045E−15	+9.4155986022169905738E−17
	4	+1.0503191826963154893E−21	+3.8725127902295302254E−23
	5	+1.6461927573606764263E−28	+5.6344651115570565066E−30
	6	+7.9138276083474522931E−36	+2.4860951084210029191E−37
	7	+7.1845890343701668760E−44	+1.9788304737427787405E−45

Table 13: Coefficients of rational functions: high-precision procedure for $W_0(z)$, part 4. Same as Table 10 when $k = 13$ through 16.

k	n	P_{kn}	Q_{kn}
13	0	+1.87963011055344866O4E+1	+1
	1	+1.1989443339646469157E−4	+5.1691031988359922329E−6
	2	+1.2463377528676863250E−10	+4.8325571823313711932E−12
	3	+3.8219456858010368172E−17	+1.3707888746916928107E−18
	4	+4.1055693930252083265E−24	+1.3754560850024480337E−25
	5	+1.5595231456048464246E−31	+4.8811882975661805184E−33
	6	+1.8157173553077986962E−39	+5.2518641828170201894E−41
	7	+3.9807997764326166245E−48	+1.0192119593134756440E−49
14	0	+2.1500582830667332906E+1	+1
	1	+3.2441943237735273768E−5	+1.2515317642433850197E−6
	2	+8.0764963416837559148E−12	+2.8310314214817074806E−13
	3	+5.9488445506122883523E−19	+1.9423666416123637998E−20
	4	+1.5364106187215861531E−26	+4.7128616004157359714E−28
	5	+1.4033231297002386995E−34	+4.0433347391839945960E−36
	6	+3.9259872712305770430E−43	+1.0515141443831187271E−44
	7	+2.0629086382257737517E−52	+4.9316490935436927307E−54
15	0	+2.4235812532416977267E+1	+1
	1	+8.6161505995776802509E−6	+3.004676184474947987E−7
	2	+5.1033431561868273692E−13	+1.630910427085546322 3E−14
	3	+8.9642393665849638164E−21	+2.6842271030298931329E−22
	4	+5.5254364181097420777E−29	+1.5619672632458881195E−30
	5	+1.2045072724050605792E−37	+3.2131689030397984274E−39
	6	+8.0372997176526840184E−47	+2.0032396245307684134E−48
	7	+1.0049140812146492611E−56	+2.2520274554676331938E−58
16	0	+2.6998134347987436511E+1	+1
	1	+2.2512257767572285866E−6	+7.1572676370907573898E−8
	2	+3.1521230759866963941E−14	+9.2500506091115760826E−16
	3	+1.3114035719790631541E−22	+3.6239819582787573031E−24
	4	+1.9156784033962366146E−31	+5.0187712493800424118E−33
	5	+9.8967003053444799163E−41	+2.4565861988218069039E−42
	6	+1.5640423898448433548E−50	+3.6435658433991660284E−52
	7	+4.6216193040664872606E−61	+9.7432490640155346004E−63

Table 14: Coefficients of rational functions: high-precision procedure for $W_0(z)$, part 5. Same as Table 10 when $k = 17$ through 19. Note that the argument of the rational functions is altered as $u \equiv \ln z$ when $k = 18$ and 19.

k	n	P_{kn}	Q_{kn}
17	0	+2.9784546702831970770E+1	+1
	1	+5.7971764392171329944E−7	+1.6924463180469706372E−8
	2	+1.9069872792601950808E−15	+5.170393431125454011 1E−17
	3	+1.8668700870858763312E−24	+4.7871532721560069095E−26
	4	+6.4200510953370940075E−34	+1.5664405832545149368E−35
	5	+7.8076624650818968559E−44	+1.8113137982381331398E−45
	6	+2.9029638696956315654E−54	+6.3454150289495419529E−56
	7	+2.0141870458566179853E−65	+4.0072964025244397967E−67
18	0	+7.4413499460126776143E−1	+1
	1	+4.1403243618005911160E−1	+3.3487811067467010907E−1
	2	+2.6012564166773416170E−1	+2.3756834394570626395E−2
	3	+2.1450457095960295520E−2	+5.4225633008907735160E−4
	4	+5.1872377264705907577E−4	+4.4378980052579623037E−6
	5	+4.3574693568319975996E−6	+1.2436585497668099330E−8
	6	+1.2363066058921706716E−8	+9.0225825867631852215E−12
	7	+9.0194147766309957537E−12	−4.2057836270109716654E−19
19	0	−6.1514412812729761526E−1	+1
	1	+6.7979310133630936580E−1	+9.7300263710401439315E−2
	2	+8.9685353704585808963E−2	+1.6103672748442058651E−3
	3	+1.5644941483989379249E−3	+7.8247741003077000012E−6
	4	+7.7349901878176351162E−6	+1.2949261308971345209E−8
	5	+1.2891647546699435229E−8	+7.0986911219342827130E−12
	6	+7.0890325988973812656E−12	+9.8426285042227044979E−16
	7	+9.8419790334279711453E−16	−1.5960147252606055352E−24

Table 15: Coefficients of rational functions: high-precision procedure for $W_{-1}(z)$, part 1. Same as Table 8 but for the new high-precision procedure when $k = -1$ through -4.

k	n	P_{kn}	Q_{kn}
-1	0	-8.2253155264446844854	$+1$
	1	$-8.132070673200148717 8$E+2	$+1.463631516166956765 9$E+2
	2	$-1.527011323767850900 0$E+4	$+3.912476137253924071 2$E+3
	3	$-7.997158508967414923 7$E+4	$+3.191269374975484746 0$E+4
	4	$-1.036675421580837651 1$E+5	$+9.244129371710861952 7$E+4
	5	$+4.228475550506125742 7$E+4	$+9.491873312047034616 5$E+4
	6	$+7.495352539760548488 4$E+4	$+2.953116540657174534 0$E+4
	7	$+1.055436914636673681 1$E+4	$+1.641680896033037098 7$E+3
-2	0	-9.6184127443354024295	$+1$
	1	$-3.557856904301800412 1$E+3	$+5.074052562852330080 1$E+2
	2	$-2.540155931128438104 3$E+5	$+4.685274715977787619 2$E+4
	3	$-5.392389363067063939 1$E+6	$+1.316830464009143629 7$E+6
	4	$-3.663825741753689679 8$E+7	$+1.311169069371241524 2$E+7
	5	$-6.148431948622696621 3$E+7	$+4.614211644525801519 5$E+7
	6	$+3.042169037744613445 1$E+7	$+4.898226895620883087 6$E+7
	7	$+3.972813905487932045 2$E+7	$+9.195910098798385512 2$E+6
-3	0	$-1.103848946229746638 8$E+1	$+1$
	1	$-1.557581288265661919 5$E+4	$+1.837077069301716681 8$E+3
	2	$-4.249294730489777343 3$E+6	$+6.128409758559509276 1$E+5
	3	$-3.517024593880342376 8$E+8	$+6.214918139846548303 7$E+7
	4	$-9.865916303661136464 0$E+9	$+2.230401131444308396 9$E+9
	5	$-8.619537230330500390 8$E+10	$+2.825423248527369802 1$E+10
	6	$-1.328633557402761600 0$E+11	$+1.077086663954315616 5$E+11
	7	$+1.598954643442066046 2$E+11	$+7.196469887604913199 2$E+10
-4	0	$-1.247440591639574605 2$E+1	$+1$
	1	$-6.818033557554377338 5$E+4	$+6.852581373443110097 1$E+3
	2	$-7.184659984562009327 8$E+7	$+8.515300102546654437 9$E+6
	3	$-2.314268822175918115 1$E+10	$+3.214602823968569465 5$E+9
	4	$-2.580137833794529513 0$E+12	$+4.292980741745319611 3$E+11
	5	$-9.518274816138631461 6$E+13	$+2.023438116163808435 9$E+13
	6	$-8.607325098621032176 6$E+14	$+2.869993326823392384 2$E+14
	7	$+1.404194185333996143 9$E+14	$+7.121013665152547709 6$E+14

Table 16: Coefficients of rational functions: high-precision procedure for $W_{-1}(z)$, part 2. Same as Table 15 when $k = -5$ through -7.

k	n	P_{kn}	Q_{kn}
−5	0	−1.3921651376890072595E+1	+1
	1	−2.9878956482388065526E+5	+2.6154955236499142433E+4
	2	−1.2313019937322092334E+9	+1.2393087277442041494E+8
	3	−1.5556149081899508970E+12	+1.7832922702470761113E+11
	4	−6.8685341106772708734E+14	+9.0772608163810850446E+13
	5	−1.0290616275933266835E+17	+1.6314734740054252741E+16
	6	−4.1404683701619648471E+18	+8.8371323861233504533E+17
	7	−1.4423309998006368397E+19	+8.4166620643385013384E+18
−6	0	−1.5377894224591557534E+1	+1
	1	−1.3122312005096979952E+6	+1.0171286771760620046E+5
	2	−2.1408157022111737888E+10	+1.8728545945050381188E+9
	3	−1.0718287431557811808E+14	+1.0469617416664402757E+13
	4	−1.8849353524027734456E+17	+2.0704349060120443049E+16
	5	−1.1394858607309311995E+20	+1.4464907902386074496E+19
	6	−1.9261555088729141590E+22	+3.0510432205608900949E+21
	7	−3.9978452086676901296E+23	+1.1397589139790739717E+23
−7	0	−1.6841701411264981596E+1	+1
	1	−5.7790823257577138416E+6	+401820.46666230725328E+5
	2	−3.7757230791256404116E+11	+2.9211518136900492046E+10
	3	−7.5712133742589860941E+15	+6.4456135373410289079E+14
	4	−5.3479338916011465685E+19	+5.0311809576499530281E+18
	5	−1.3082711732297865476E+23	+1.3879041239716289478E+22
	6	−9.1462777004521427440E+25	+1.1575146167513516225E+25
	7	−8.9602768119263629340E+27	+1.7199220185947756654E+27

Table 17: Coefficients of rational functions: high-precision procedure for $W_{-1}(z)$, part 3. Same as Table 15 when $k = -8$ through -10. Here the argument of the rational functions is $v \equiv \ln(-z)$.

k	n	P_{kn}	Q_{kn}
−8	0	−2.0836260384016439265	+1
	1	+1.6122436242271495710	+2.3699648912703015610
	2	+5.4464264959637207619	−2.1249449707404812847
	3	−3.0886331128317160105	+3.8480980098588483913E−1
	4	+4.6107829155370137880E−1	−2.1720009380176605969E−2
	5	−2.3553839118456381330E−2	+3.9405862890608636876E−4
	6	+4.0538904170253404780E−4	−1.7909312066865957905E−6
	7	−1.7948156922516825458E−6	+3.1153673308133671452E−12
−9	0	+1.6045383766570541409E−1	+1
	1	+2.2214182524461514029	−7.0254996087870332289E−1
	2	−9.4119662492050892971E−1	+8.0974347786703195026E−2
	3	+9.1921523818747869300E−2	−2.7469850029563153939E−3
	4	−2.9069760533171663224E−3	+3.1943362385183657062E−5
	5	+3.2707247990255961149E−5	−1.2390620687321666439E−7
	6	−1.2486672336889893018E−7	+1.2241636115168201999E−10
	7	+1.2247438279861785291E−10	−1.0275718020546765400E−17
−10	0	−1.2742179703075440564	+1
	1	+1.3696658805421383765	−1.1420006474152465694E−1
	2	−1.2519345387558783223E−1	+2.4285233832122595942E−3
	3	+2.5155722460763844737E−3	−1.5520907512751723152E−5
	4	−1.5748033750499977208E−5	+3.4120534760396002260E−8
	5	+3.4316085386913786410E−8	−2.4981056186450274587E−11
	6	−2.5025242885340438533E−11	+4.6419768093059706079E−15
	7	+4.6423885014099583351E−15	−1.3608713936942602985E−23

the considered interval as

$$W_0\left(z_0\right) = -1 < W_0(0) = 0 < W_0\left(z_1\right), \tag{17}$$

since $z_0 < 0 < z_1$. Therefore, we (i) modified the function to be approximated from $W_0(z)$ to $W_0(z)/z$, (ii) obtained the minimax rational function approximation of the modified function, and (iii) multiplied z to the approximated rational function. As a result, the type of the final form of the rational function $X_1(x)$ is slightly changed as $(8, 7)$ and $(4, 3)$, respectively. This modification significantly alters the error curves around $W \approx 0$ as will be shown later.

2.3. Reduction of round-off errors

As already indicated in Figs 3 and 4, difficult is the precise computation of $W_0(z)$ and $W_{-1}(z)$ near the branch point, namely when $z \approx -1/e$, and therefore $W \approx -1$. This is mainly because the derivatives of the function with respect to z are inversely proportional to some powers of $W + 1$ as

$$\frac{dW}{dz} = \frac{W}{z(1+W)}, \quad \frac{d^2W}{dz^2} = \frac{-W^2(2+W)}{z^2(1+W)^3}, \tag{18}$$

where the derivative formulae are correct for both branches. See Figs 1 and 2 again. Consequently, the magnitude of the derivatives become huge when $z \to -1/e$ and $W \to -1$ and therefore, all of the iterative methods to solve the defining equation, Eq. (1), such as the Newton method and Halley's method [38] fail to converge.

Also, this situation was not so effectively improved by using the Taylor series expansion around the branch point [35]. This phenomenon was caused by the round-off effects in calculating x or its analog from the given value of z. We confirmed this fact from a comparison of the results of the same procedure evaluated in the double- and quadruple-precision environments.

Let us return to the new procedures. The new low-precision procedures do not suffer such precision degradation since they are evaluated in the double-precision environment. Similarly, the new high-precision procedures are accurate *if* evaluated in the quadruple-precision environment. Nonetheless, it is not practical to use quadruple-precision computation because of its relatively-high computational cost.

We experienced similar round-off problems in approximating the complete elliptic integrals around their logarithmic singularity [40, 41]. To reduce the

Table 18: Coefficients of rational function: high-precision procedure for $W_{-1}(z)$ near the branch point. Same as Table 15 when $k = -1$ but expressed as a rational function of $x \equiv \sqrt{z + 1/e}$.

k	n	P_{kn}	Q_{kn}
-1	0	-1.0000000000000001110	$+1$
	1	$+4.2963016178777127009$	-6.6279455994747624059
	2	-4.0991407924007457612	$+1.7740962374121397994\text{E}{+1}$
	3	-6.8442842200833309724	$-2.4446872319343475890\text{E}{+1}$
	4	$+1.7084773793345271001\text{E}{+1}$	$+1.8249006287190617068\text{E}{+1}$
	5	$-1.3015133123886661124\text{E}{+1}$	-7.0580758756624790550
	6	$+3.9303608629539851049$	$+1.1978786762794003545$
	7	$-3.4636746512247457319\text{E}{-1}$	$-5.3875778140352599789\text{E}{-2}$

resulting computational errors, we changed the input argument from z to $z_c \equiv z+1/e$ as adopted by the TOMS algorithm 743 [33]. The transformation from z_c to $x = \sqrt{z_c}$ is free from cancellation, and therefore this trick works quite well for $W_0(z)$ for the whole argument interval. Nevertheless, its naive application to $W_{-1}(z)$ faces with a significant increase of the round-off errors when z increases, say when $z > -0.354$ and therefore $W < -1.30$ as viewed in Fig. 7. This is because the reverse transformation from z_c to z suffers from a loss of information when $z \approx 0$. In such cases, we adopted an offset value of x as the main variable such that its computation from z is sufficiently precise. This is the reason why we chose $y \equiv x - \sqrt{1/e}$ as the main variable in computing $W_{-1}(z)$ and compute y in a cancellation-free form as $y = -z/\left(x + \sqrt{1/e}\right)$. We confirmed that the selection of y as the main variable is appropriate for the second and later intervals, namely when $z > z_{-1}$. In the first interval where $z_0 \le z \le z_{-1}$, however, another care is needed. Indeed, the rational function approximation using $Y_{-1}(y)$ causes a significant increase of round-off errors when $z < -0.354$, and therefore $W > -1.30$ as already indicated in Fig. 7. Thus, we introduced an additional conditional switch in

the high-precision procedure computing $W_{-1}(z)$ as

$$W_{-1}(z) = \begin{cases} X_{-1}(x), & \left(z_0 \le z < z_{-1/2}\right), \\ Y_{-1}(y), & \left(z_{-1/2} \le z < z_{-1}\right), \\ Y_k(y), & \left(z_{k+1} \le z < z_k, \;\; k = -2, -3, \cdots, -7\right), \\ V_k(v), & \left(z_{k+1} \le z < z_k, \;\; k = -8, -9, -10\right), \end{cases} \tag{19}$$

where $z_{-1/2}$ is the additional critical argument defined as

$$z_{-1/2} \equiv -1.30 \exp(-1.30) \approx -0.3542913309442164, \tag{20}$$

and $X_{-1}(x)$ is a rational function the coefficients of which are listed in Table 18. Note that $X_{-1}(x)$ and $Y_{-1}(y)$ are exactly the same if the roud-off errors are sufficiently small as in the quadruple-precision environment. At any rate, the resulting new high-precision procedure for $W_{-1}(z)$ maintains the 50-bit accuracy as already illustrated in Figs 3, 4, 6, and 7.

3. Numerical experiments

Let us examine the computational performance of the newly developed procedures. We begin with computational errors. Figs 8 through 11 illustrate the relative errors of the new low-precision procedures for a wide range of function values, $-1 \le W_0(z) \le 703.227$ and $-714.196 \le W_{-1}(z) \le -1$, and therefore, for a similarly wide range of argument values, $-1/e \le z < 1.798 \times 10^{+308}$ for $W_0(z)$ and $-1/e \le z \le -4.813 \times 10^{-308}$ for $W_{-1}(z)$, respectively. Note that this argument range mostly covers the double-precision floating-point numbers.

In drawing the figures, we evaluate the low-precision procedures in the double-precision environment to reduce the round-off effects significantly. As a result, the plotted error curves exhibit an almost global minimax nature of the procedures. In practice, the double-precision computation is as fast as the single-precision one nowadays. Therefore, we may evaluate the low-precision procedure in the double-precision environment.

Similarly, Figs 12 through 15 show the error curves of the high-precision procedures evaluated in the quadruple-precision environment. Again, obvious is the global minimax feature of the procedure. Nevertheless, the quadruple-precision computation is significantly time-consuming, say 20–40 times more than the double-precision computation. Therefore, it is inappropriate to evaluate the high-precision procedure in the quadruple-precision environment.

Then, we prepared Figs 16 through 19 indicating the relative errors of the high-precision procedures evaluated in the double-precision environment. As anticipated, the errors do increase by the round-off effects. At any rate, the high-precision procedures are sufficiently accurate in the sense that the maximum relative errors are eight double-precision machine epsilons at most.

Let us move on to the aspect of the computational cost. Already Tables 1 and 2 illustrated the averaged CPU time computing $W_0(z)$ and $W_{-1}(z)$, respectively. To examine the argument dependence of the CPU time, we prepared Figs 20 and 21 plotting the CPU time as a function of W.

In the case of $W_0(z)$, the compared methods are seven: (i) the TOMS algorithm 443 [31] which works only when $W > 0$, (ii) the TOMS algorithm 743 [33], (iii) the method of Veberic [34], (iv) the previous method of ours [35], (v) the method of Vazquez-Leal et al. [36], (vi) the new low-precision procedure, and (vii) the new high-precision procedure. On the other hand, in the case of $W_{-1}(z)$, the compared methods are also seven: (i) the TOMS algorithm 743 [33], (ii) the method of Chapeau-Blondeau & Monir [32], (iii) the method of Veberic [34], (iv) the previous method of ours [35], (v) the method of Vazquez-Leal et al. [36], (vi) the new low-precision procedure, and (vii) the new high-precision procedure.

Actually, the CPU times are measured at a commercial PC with the Intel Core i9-9980HK CPU running at 2.40–5.00 GHz clock under the Windows 10 OS. All the computer programs were coded in Fortran 90 and compiled by the Intel Visual Fortran Compiler 19.1, or the so-called `ifort 19.1`, with the maximum optimization flags. They were executed in a single-core mode and the total CPU time of $2^{26} \approx 6.71 \times 10^7$ executions for slightly different arguments were measured by the Fortran intrinsic subroutine, `date_and_time`, and averaged. Then, to avoid the influence of the system interruption in the multi-task operation, we regarded the minimum value among 10 independent runs of the time measurements as the CPU time. Finally, we plotted the adopted values as a function of W after normalized by that of the single execution of the exponential function provided by the Intel mathematical library, which was 8.14 ns at the computer used.

At any rate, the figures indicate the smallness of the CPU times of the new procedures. In a relatively wide region $(-1, +35]$, the new high-precision procedure computing $W_0(z)$ is 1.11 to 2.00 times faster than the previous double-precision procedure of ours [35], which is the fastest among the existing methods [31, 33, 34, 36]. Similarly, in a region $[-11, -1)$, the new

high-precision procedure computing $W_{-1}(z)$ is 1.13 to 2.08 times faster than the previous double-precision procedure of ours [35], which is the fastest among the existing methods [33, 32, 34, 36].

On the other hand, the new low-precision procedure computing $W_0(z)$ is 1.10 to 1.70 times faster than the high-precision one computing $W_0(z)$ and the new low-precision procedure computing $W_{-1}(z)$ is 1.17 to 2.14 times faster than the high-precision one computing $W_{-1}(z)$.

4. Conclusion

To obtain a precise and/or fast procedure computing $W_0(z)$ and $W_{-1}(z)$, the primary and secondary branches of the Lambert W functions, we developed new low- and high-precision procedures. They are conditional switches of rational functions of four new variables,

$$x \equiv \sqrt{z + 1/e}, \quad y \equiv -z/\left(x + \sqrt{1/e}\right), \quad u \equiv \ln z, \quad v \equiv \ln(-z). \tag{21}$$

Numerical experiments confirmed that the new low- and high-precision procedures are of 24- and 50-bit accuracies, respectively, for arbitrary value of z representable as a double-precision floating-point number, namely when $-1/e \le z < 1.798 \times 10^{+308}$ for $W_0(z)$ and when $-1/e \le z < -4.841 \times 10^{-324}$ for $W_{-1}(z)$, respectively.

For the practical arguments corresponding to the function value range as $-10 \le W_{-1}(z) < -1$ and $-1 < W_0(z) \le 20$, the new high-precision procedures for $W_0(z)$ and $W_{-1}(z)$ run 1.81 and 1.68 times faster than the previous double-precision procedure of ours [35], which was the fastest among the existing procedures, respectively. On the other hand, the new low-precision procedures for $W_0(z)$ and $W_{-1}(z)$ are 15 and 41% faster than the new high-precision ones, respectively.

A Fortran 90 test program package of the new procedures, `xlambw.txt`, including its test driver and a sample output is freely available at the author's WEB site:

https://www.researchgate.net/profile/Toshio_Fukushima

References

[1] Olver, F.W.J., Lozier, D.W., Boisvert, R.F., & Clark C.W. (eds.), *NIST Handbook of Mathematical Functions*, Cambridge Univ. Press, Cambridge (2010).

[2] Jeffrey, D.J., Hare, D.E.G., & Corless, R.M., *Unwinding the branches of the Lambert W function*, Math. Scientist, 21 (1996) 1-7.

[3] Corless, R.M., Gonnet, G.H., Hare, D.E.G., Jeffrey, D.J., & Knuth, D.E., *On the Lambert W function*, Adv. Comput. Math., 5 (1996) 329-359.

[4] Boyd, J.P., *Global approximations to the principal real-valued branch of the Lambert W-function*, Appl. Math. Lett., 11 (1998) 27-31 (1998).

[5] Iacono, R. & Boyd, J.P., *New approximations to the principal real-valued branch of the Lambert W-function*, Adv. Comp. Math., 43 (2017) 1403-1436.

[6] Kahan, W., *Lecture notes on the status of IEEE Standard 754 for binary floating-point arithmetic*, EECS Dept., UC Berkeley, (1997). `https://people.eecs.berkeley.edu/~wkahan/ieee754status/IEEE754.PDF`

[7] Borwein, J.M. & Corless, R.M., *Emerging tools for experimental mathematics*, Amer. Math. Monthly, 106 (1999) 889-909.

[8] Hayes, B., *Why W?*, Amer. Scientists, 93 (2005), 104-108.

[9] Brito, P.B., Fabiao, F., & Staubyn, A., *Euler, Lambert, and the Lambert W-function today*, Math. Scientist, 33 (2008) 127-133.

[10] Belgacem, C.H., Fnaiech, M., *Solution for the critical thickness models of dislocation generation in epitaxial thin films using the Lambert W function*, J. Mater. Sci., 46 (2011) 1913-1915.

[11] Molli, M., Venkataramaniah, K., Valluri, S.R., *The polylogarithm and the Lambert W functions in thermoelectrics*, Can. J. Phys., 89 (2011) 1171-1178.

[12] Brkić, D., *Lambert W function in hydraulic problems*, Math. Balkan., 26 (2012) 285-292.

[13] Dubinov, A.E., Senilov, L.A., *Generalized Bohm criterion for a multicomponent plasma*, Tech. Phys., 57 (2012), 1090-1094.

[14] Chatzigeorgiou, I., *Bounds on the Lambert function and their application to the outage analysis of user cooperation*, IEEE Comm. Lett., 17 (2013) 1505-1508.

[15] Houari, A., *Additional applications of the Lambert W function in physics*, Eur. J. Phys., 34 (2013) 695-702.

[16] Jaisson, D., *Simple formula for the wave number of the Goubau line*, Electromag., 34 (2014) 85-91.

[17] Jordan, P.M., *A note on the Lambert W-function: applications in the mathematical and physical sciences*, Cont. Math., 618 (2014) 247-263.

[18] Sharma, S., Shokeen, P., Saini, B., Chetna, S.S., Kashyap, J., Guliani, R., Sharma, S., Khanna, U.M., Jain, A., & Kapoor, A., *Exact analytical solutions of the parameters of different generation real solar cells using Lambert W-function: a review article*, Int. J. Renew. Energy, 4 (2014) 155-194.

[19] Bhamidi, S., Steele, J.M., Zaman, T., *Twitter event networks and the Superstar model*, Ann. Appl. Probab., 25 (2015) 2462-2502.

[20] Franca, G., LeClair, A., *Transcendental equations satisfied by the individual zeros of Riemann zeta, Dirichlet and modular L-functions*, Comm. Num. Theor. Phys., 9 (2015) 1-50.

[21] Gasull, A., Jolis, M., Utzet, F., *On the norming constants for normal maxima*, J. Math. Anal. Appl., 422 (2015) 376-396.

[22] Belgacem, C.H., *Explicit solution for critical thickness of semicircular misfit dislocation loops in strained semiconductors heterostructures*, Silicon, 8 (2016) 397-399.

[23] Roberts, K., Valluri, S.R., *Tutorial: The quantum finite square well and the Lambert W function*, Canad. J. Phys., 95 (2016) 105-110.

[24] Pakes, A.G., *The Lambert W function, Nuttall's integral, and the Lambert law*, Stat. Prob. Lett., 119 (2018) 53-60.

[25] Saillenfest, M., Tabone, B., Behar, E., *Solar wind dynamics around a comet- the paradigmatic inverse-square-law model*, Astron. Astrophys., 617 (2018) A99.

[26] Belkic, D., *All the trinomial roots, their powers and logarithms from the Lambert series, Bell polynomials and Fox-Wright function: illustration for genome multiplicity in survival of irradiated cells*, J. Math. Chem., 57 (2019) 59-106.

[27] Saha, S., Bamba, K., *The Lambert W function: A newcomer in the Cosmology class?*, Zeitschr. Naturf., A 75 (2019), 23-27.

[28] Tekam, R.B.W., Kengne, J., Kenmoe, G.D., *High frequency Colpitts' oscillator: A simple configuration for chaos generation*, Chaos Solit. Fract., 126 (2019) 351-360.

[29] Wright, E.M., *Solution of the equation* $ze^z = a$, Bull. Amer. Math. Soc., 65 (1959) 89-93.

[30] Siewert, C.E. & Burniston, E.E., *Exact analytical solutions of* $ze^z = a$, J. Math. Anal. Appl., 43 (1973) 626-632.

[31] Fritsch, F.N., Shafer, R.E., & Crowley, W.P., *Algorithm 443: Solution of the transcendental equation* $we^w = x$, Commun. ACM 16 (1973) 123-124.

[32] Chapeau-Blondeau, F. & Monir, A., *Evaluation of the Lambert W-function and application to generation of generalized Gaussian noise with exponent 1/2*, IEEE Trans. Signal Process., 50 (2002) 2160-2165.

[33] Barry, D.A., Barry, S.J., & Culligan-Hensley, P.J., *Algorithm 743: WAPR-a fortran routine for calculating real values of the W-function*, ACM Trans. on Math. Softw. (TOMS), 21 (1995) 172-181.

[34] Veberic, D., *Lambert W-function for applications in physics*, Comput. Phys. Comm., 183 (2012) 2622-2628.

[35] Fukushima, T., *Precise and fast computation of Lambert W-functions without transcendental function evaluations*, J. Comp. Appl. Math., 244 (2013) 77-89.

[36] Vazquez-Leal, H., Sandoval-Hernandez, M.A., Garcia-Gervacio, J.L., Herrera-May, A.L., & Filobello-Nino, U.A., *PSEM approximations for both branches of Lambert W function with applications*, Discr. Dyn. Nat. Soc., 2019 (2019) 8267951.

[37] Johansson, F., *Computing the Lambert W function in arbitrary-precision complex interval arithmetic*, Numer. Alg., 83 (2020) 221-242.

[38] Householder, A., *The Numerical Treatment of a Single Nonlinear Equation*, McGraw-Hill, New York, 1970.

[39] Fukushima, T., *Fast computation of complete elliptic integrals and Jacobian elliptic functions*, Cele. Mech. Dyn. Astron., 105 (2009) 305-328.

[40] Fukushima, T., *Precise and fast computation of general complete elliptic integral of second kind*, Math. Comp., 80 (2011) 1725-1743.

[41] Fukushima, T., *Precise and fast computation of complete elliptic integrals by piecewise minimax rational function approximation*, J. Comp. Appl. Math., 282 (2015) 71-76.

[42] Fukushima, T., *Precise and fast computation of inverse Fermi-Dirac integral of order 1/2 by minimax rational function approximation*, Appl. Math. Comp., 259 (2015) 698-707.

[43] Fukushima, T., *Precise and fast computation of Fermi-Dirac integral of integer and half integer order by piecewise minimax rational approximation*, Appl. Math. Comp., 259 (2015) 708-729.

[44] Wolfram, S., *The Mathematica Book, 5th ed.*, Wolfram Research Inc./Cambridge Univ. Press, Cambridge, 2003.

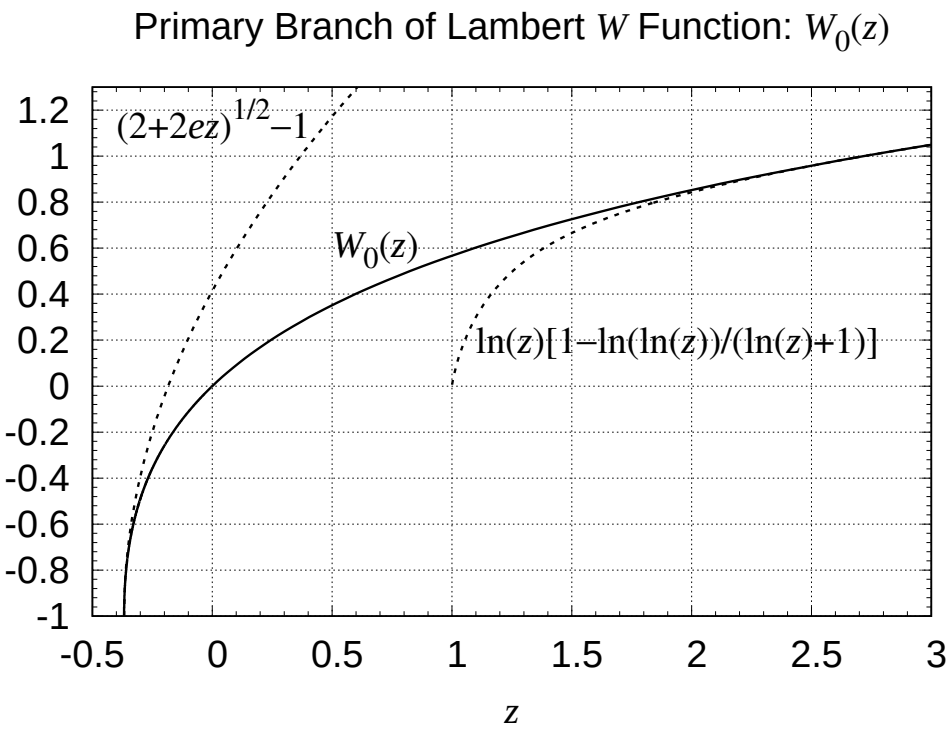

Figure 1: Primary branch of Lambert W function: $W_0(z)$. Displayed is the curve of the primary branch of the Lambert W function, $W_0(z)$, for the argument interval, $-1/e \le z \le 3$. Added are its two approximations indicated by broken curves: (i) $-1+\sqrt{2+2ez}$, a two-term truncation of the Taylor series expansion around the branch point, $z = -1/e$, and (ii) $\ln(z)[1 - \ln\{\ln(z)\} / \{\ln(z)+1\}]$, one Newton improvement applied to an asymptotic solution, $W_0(z) \approx \ln(z)$.

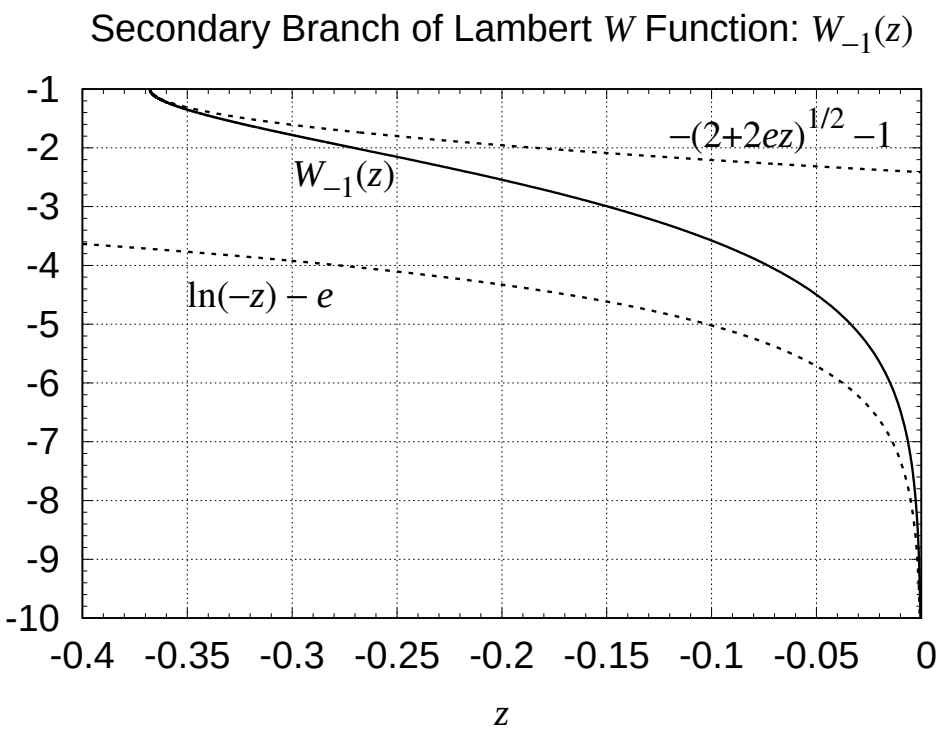

Figure 2: Secondary branch of Lambert W function: $W_{-1}(z)$. Same as Fig. 1 but for the secondary branch of the Lambert W function, $W_{-1}(z)$, for the argument interval, $-1/e \le z < 0$. This time, added are its two approximations: (i) $-1-\sqrt{2+2ez}$, a two-term truncation of the Taylor series expansion around the branch point, $z=-1/e$, and (ii) $\ln(-z)-e$, one Newton improvement applied to an asymptotic solution, $W_{-1}(z) \approx \ln(-z)$.

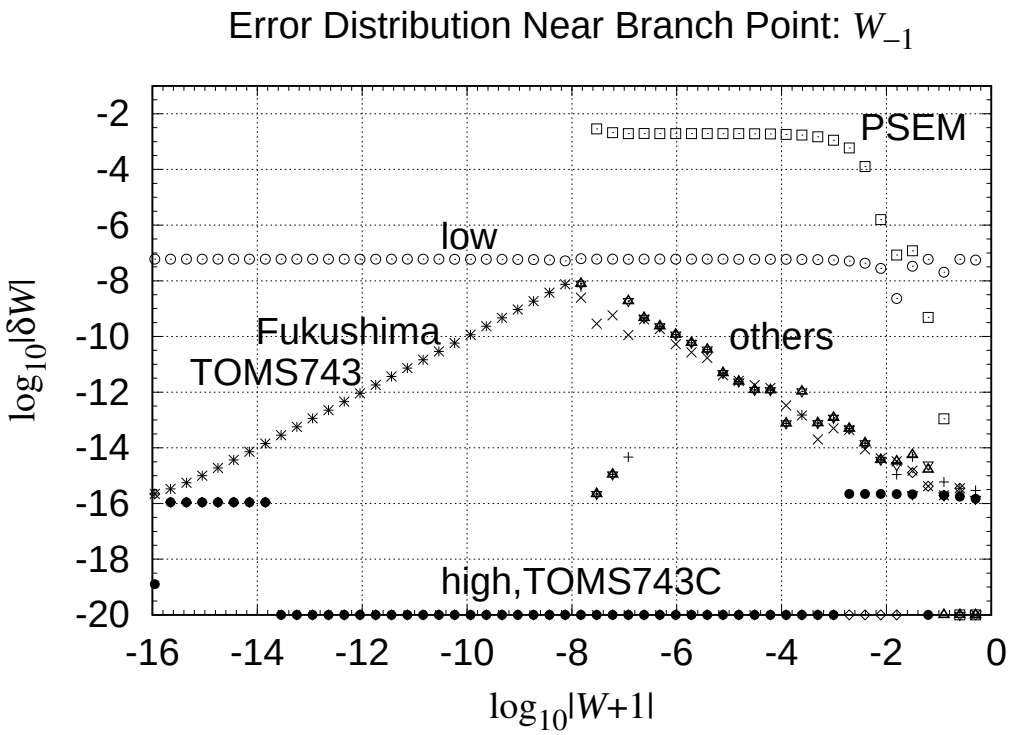

Figure 3: Error distribution near branch point: $W_{-1}(z)$. Shown are the errors of $W_{-1}(z)$ computed by various numerical procedures when z is near the branch point, $z = -1/e$, and therefore $W_{-1}(z)$ is close to its maximum value, -1. Plotted are the relative errors of the function value, $\delta W(z) \equiv W_{-1}(z)/W_{\rm ref}(z) - 1$, as a function of the function value when $-2 \le W_{-1}(z) < -1$. Here the reference value $W_{\rm ref}(z)$ is computed by repeated usage of Halley's method [38] to solve the defining equation in the quadruple-precision environment. Compared are eight procedures: (i) the TOMS algorithm 743 [33] with the argument z as the input variable, which is denoted by `TOMS743` and indicated by an x-mark, $\times$, (ii) the same TOMS algorithm 743 but with the modified input argument $z_c \equiv z + 1/e$, which is denoted by `TOMS743C` and indicated by an open diamond, $\Diamond$, (iii) the method of Chapeau-Blondeau & Monir (2002) [32] indicated by a nabla, ∇, (iv) the method of Veberic (2012) [34] indicated by a triangle, $\triangle$, (v) the previous double-precision procedure of ours [35] denoted by `Fukushima` and indicated by a plus sign, $+$, (vi) the method of Vazquez-Leal et al. (2019) [36] denoted by `PSEM` and indicated by a square box, $\square$, (vii) the new low-precision procedure denoted by `low` and indicated by an open circle, $\bigcirc$, and (viii) the new high-precision procedure denoted by `high` and indicated by a filled circle. The method of Vazquez-Leal et al. (2019) suffers from a significant increase of the errors when $z < -0.36$, then $W_{-1}(z) < -1.21$, and therefore $\log_{10}|W+1| < -0.68$. Also, the methods of Chapeau-Blondeau & Monir (2002) [32], Veberic (2012) [34], and Vazquez-Leal et al. (2019) fail to return the answer when $|W+1| < 10^{-8}$.

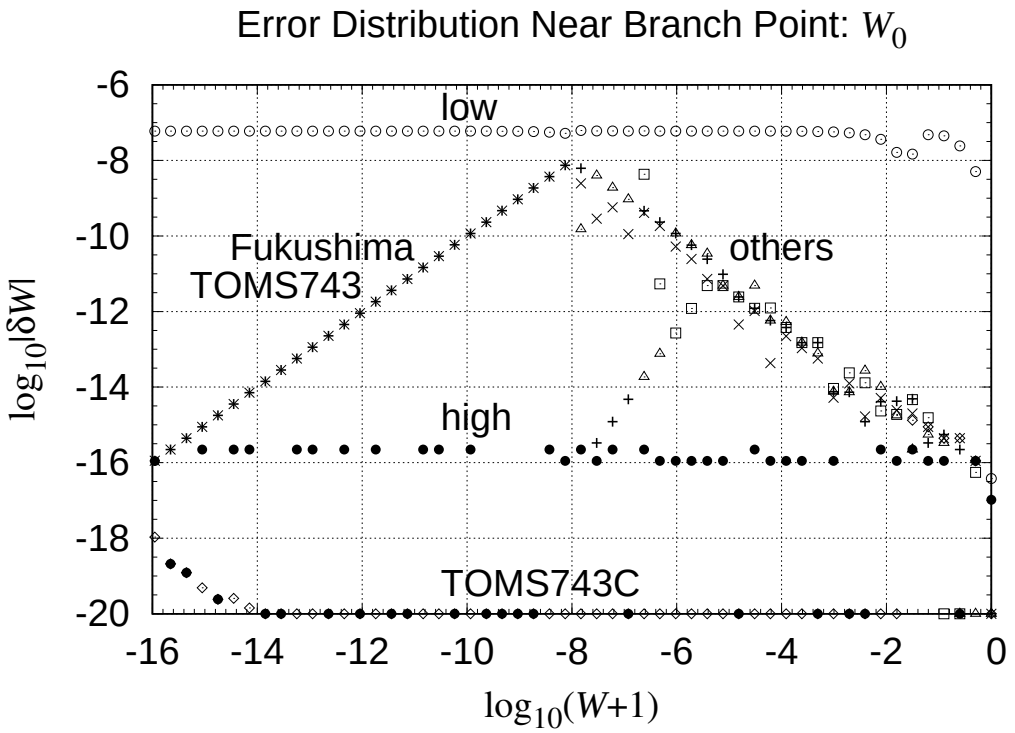

Figure 4: Error distribution near branch point: $W_0(z)$. Same as Fig. 3 but for $W_0(z)$ when $-1 < W_0(z) \leq 0$. This time, we removed the method of Chapeau-Blondeau & Monir (2002) [32] because it is effective only for $W_{-1}(z)$. The methods of Veberic (2012) [34] and Vazquez-Leal et al. (2019) fail to return the answer when $|W + 1| < 10^{-8}$. Again, the new low- and high-precision procedures maintain the 24- and 50-bit accuracies, respectively.

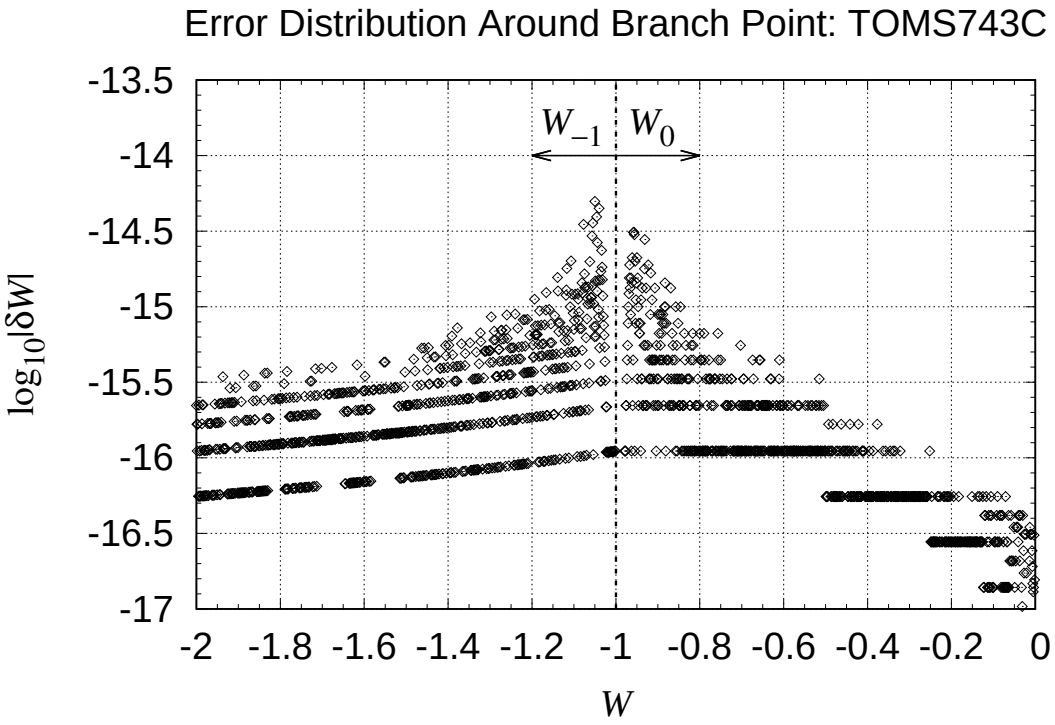

Figure 5: Error distribution around branch point: TOMS743C. Same as the errors of the TOMS algorithm 743 with the modified argument as the input variable [33] displayed in Figs 3 and 4 but combined for $W_0(z)$ and $W_{-1}(z)$ together and plotted as a function of W in the combined interval, $-2 \leq W \leq 0$. Although the relative errors in the central region, namely when $-1.02 \leq W \leq -0.98$, are suppressed down to the level of the double-precision machine epsilon, $\varepsilon_{\rm D} \equiv 2^{-53} \approx 1.11 \times 10^{-16}$, the maximum error outside the region is as large as around 5.0×10^{-15}. Thus, the TOMS algorithm 743 with the modified argument as the input variable is of the 47-bit accuracy around the branch point.

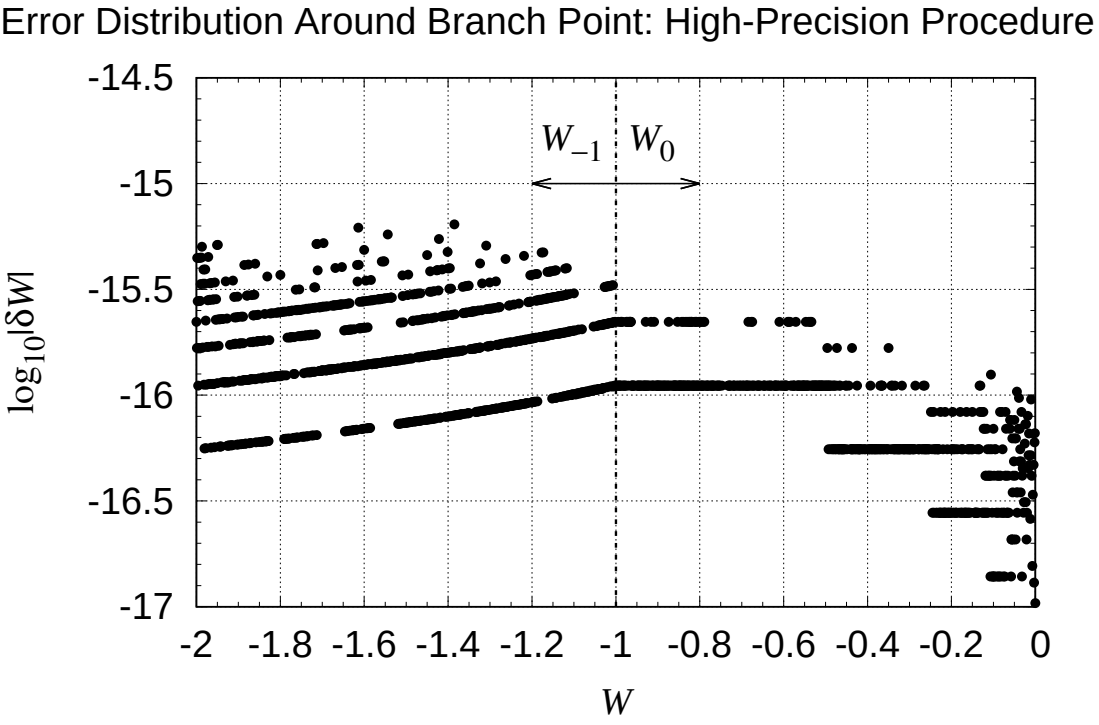

Figure 6: Error distribution around branch point: new high-precision procedures. Same as Fig. 5 but for the new high-precision procedures described in the main text. The maximum magnitude of the relative errors is roughly 6.4×10^{-16}. In this sense, the new high-precision procedures are said to be of the 50-bit accuracy around the branch point.

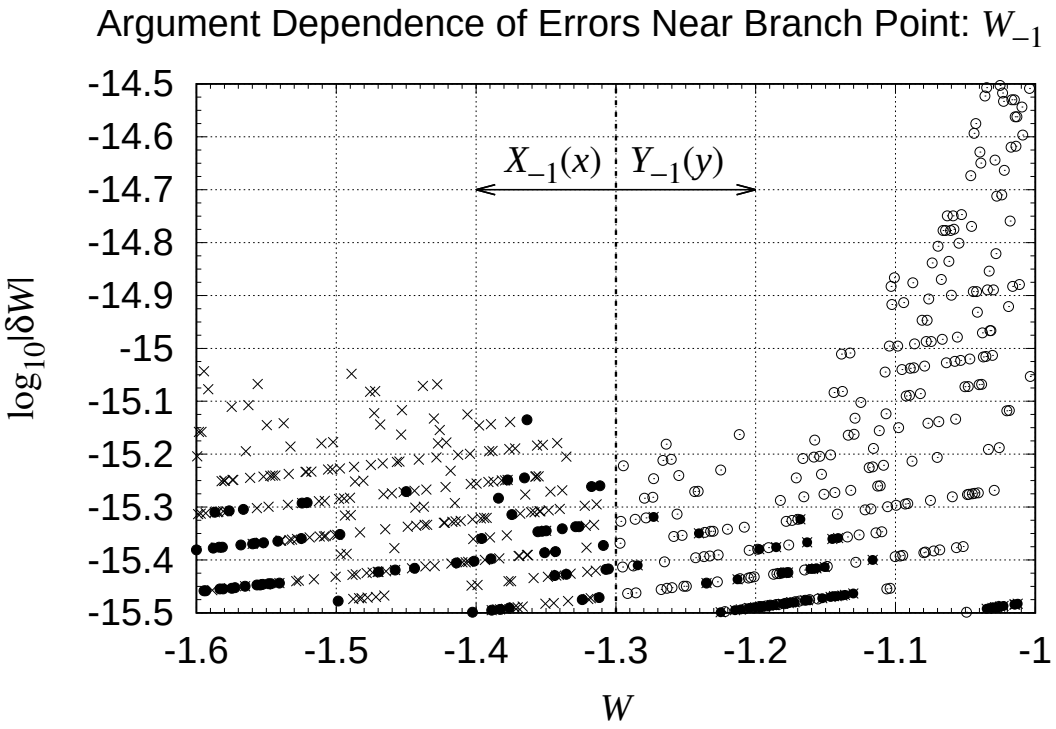

Figure 7: Argument dependence of errors near branch point: $W_{-1}(z)$. Same as a part of Fig. 6 but for $W_{-1}(z)$ in a narrower interval as $-1.6 \leq W < -1$. This time, added are the errors obtained by using two rational function approximations of different arguments: $X_{-1}(x)$ indicated by an x-mark, $\times$, and $Y_{-1}(y)$ indicated by an open circle, $\bigcirc$. Since $X_{-1}(x)$ is erroneous when $W < -1.3$ while $Y_{-1}(y)$ is so when $W > -1.3$, we adopted their conditional switch as a part of the high-precision procedure indicated by a filled circle.

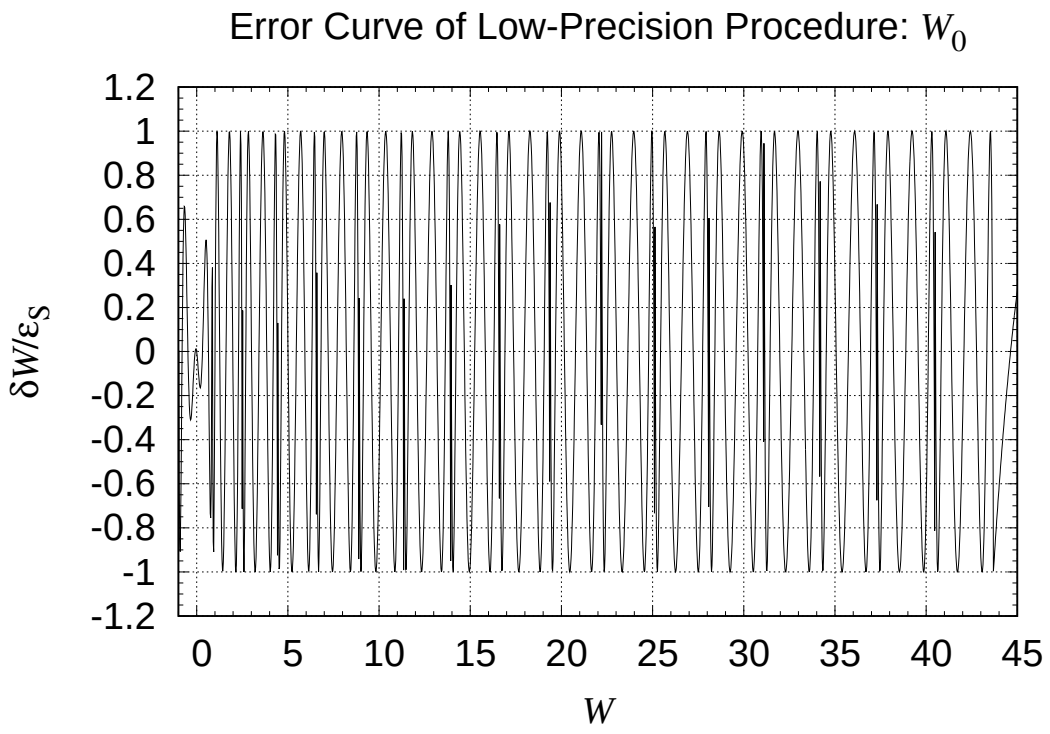

Figure 8: Error curve of low-precision procedure: $W_0(z)$. Shown are the relative errors of the primary branch of the Lambert W function, $W_0(z)$, computed by the new low-precision procedure described in the main text. To reduce the round-off errors, we conducted all the computation in the double-precision environment. Then, the errors were measured as the difference from the new high-precision procedure. Finally, the measured errors were scaled by the single-precision machine epsilon, $\varepsilon_S \equiv 2^{-24} \approx 5.96 \times 10^{-8}$ and plotted as a function of the function value for its interval, $-1 \leq W_0(z) \leq 45$, which approximately corresponds to the argument interval, $-0.368 < z < 1.572 \times 10^{+21}$. Obvious is the minimax nature of the new procedure.

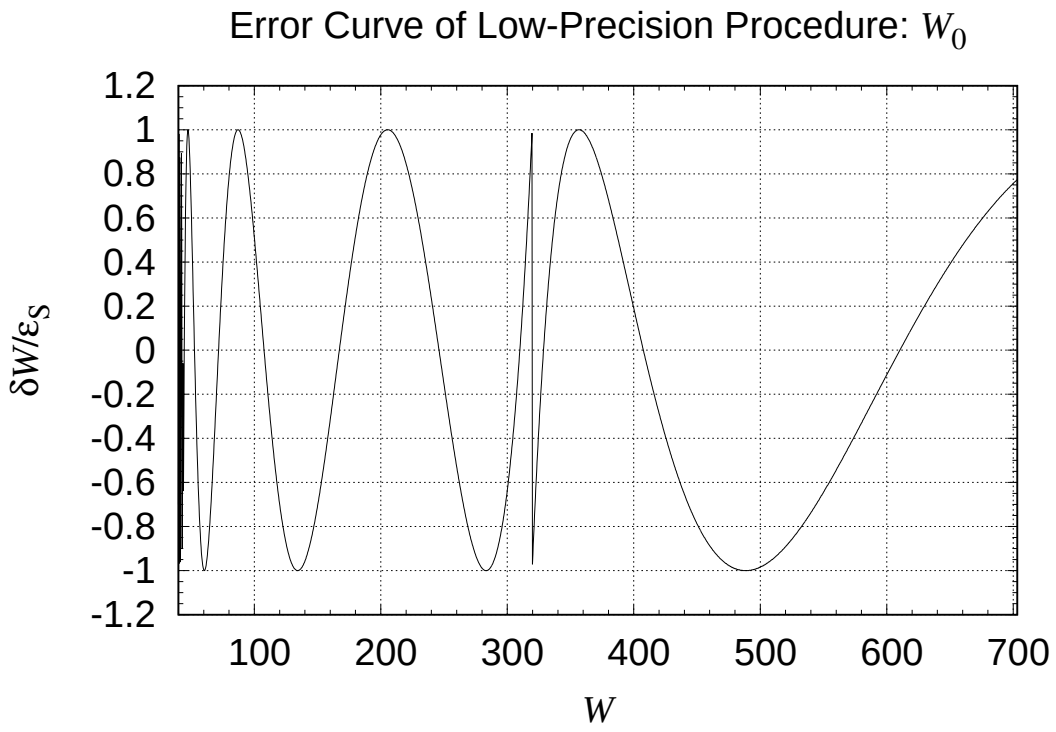

Figure 9: Error curve of low-precision procedure: $W_0(z)$, large arguments. Same as Fig. 8 but when $40 \leq W_0(z) \leq 703.227$, which approximately corresponds to the argument interval, $9.415 \times 10^{+18} < z < 1.798 \times 10^{+308}$.

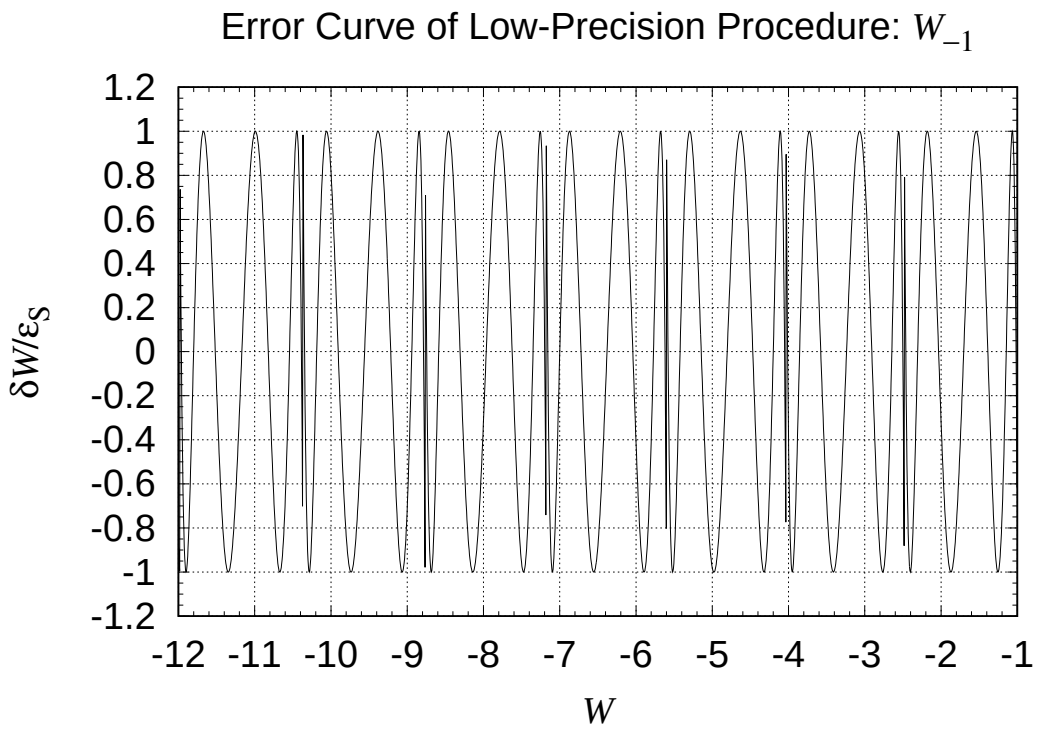

Figure 10: Error curve of low-precision procedure: $W_{-1}(z)$. Same as Fig. 8 but for $W_{-1}(z)$ when $-12 \leq W_{-1}(z) \leq -1$, which approximately corresponds to the argument interval, $-0.368 < z < -7.373 \times 10^{-5}$.

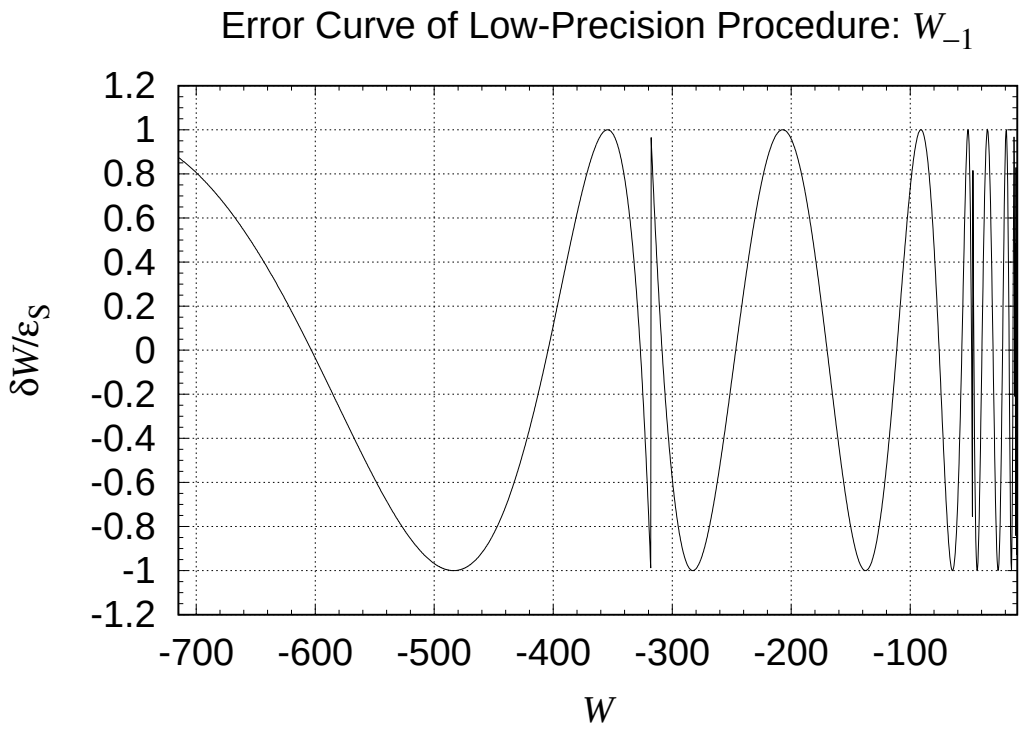

Figure 11: Error curve of low-precision procedure: $W_{-1}(z)$, large argument. Same as Fig. 10 but when $-714.196 \leq W \leq -10$, which approximately corresponds to the argument interval, $-4.540 \times 10^{-4} < z < -2.225 \times 10^{-308}$.

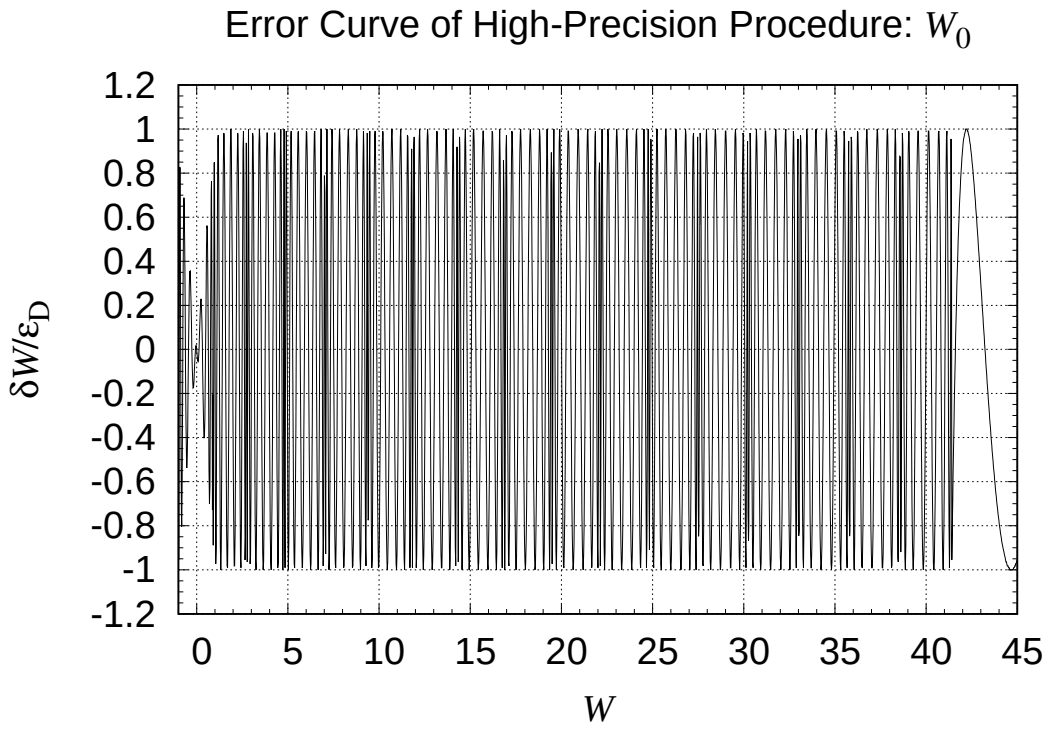

Figure 12: Error curve of high-precision procedure: $W_0(z)$. Same as Fig. 8 but for the new high-precision procedure described in the main text. To avoid the round-off effects, we evaluated the procedure in the quadruple precision arithmetics. Then, the errors were measured as the difference from the reference solution with the 34 digits accuracy, which was prepared by repeatedly applying Halley's method [38] to the obtained high-precision solution in the quadruple-precision environment, and scaled by the double-precision machine epsilon, $\varepsilon_{\rm D} \equiv 2^{-53} \approx 1.11 \times 10^{-16}$.

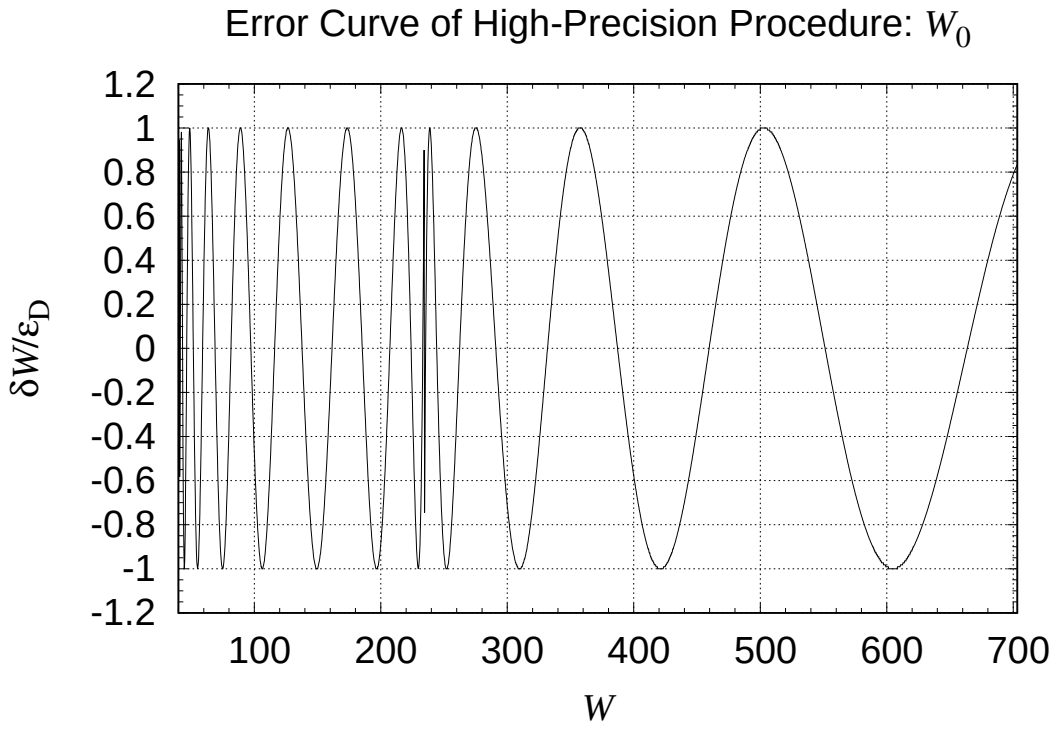

Figure 13: Error curve of high-precision procedure: $W_0(z)$, large arguments. Same as Fig. 12 but when $40 \leq W_0(z) \leq 703.227$.

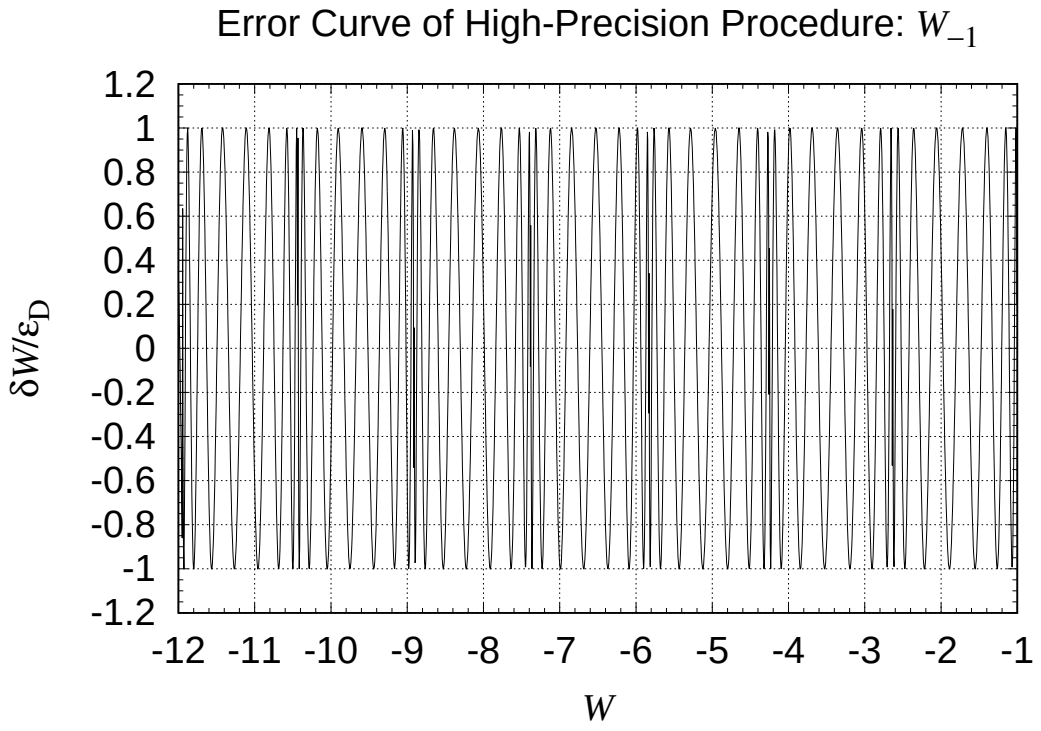

Figure 14: Error curve of high-precision procedure: $W_{-1}(z)$. Same as Fig. 12 but for $W_{-1}(z)$ when $-12 \leq W_{-1}(z) \leq -1$.

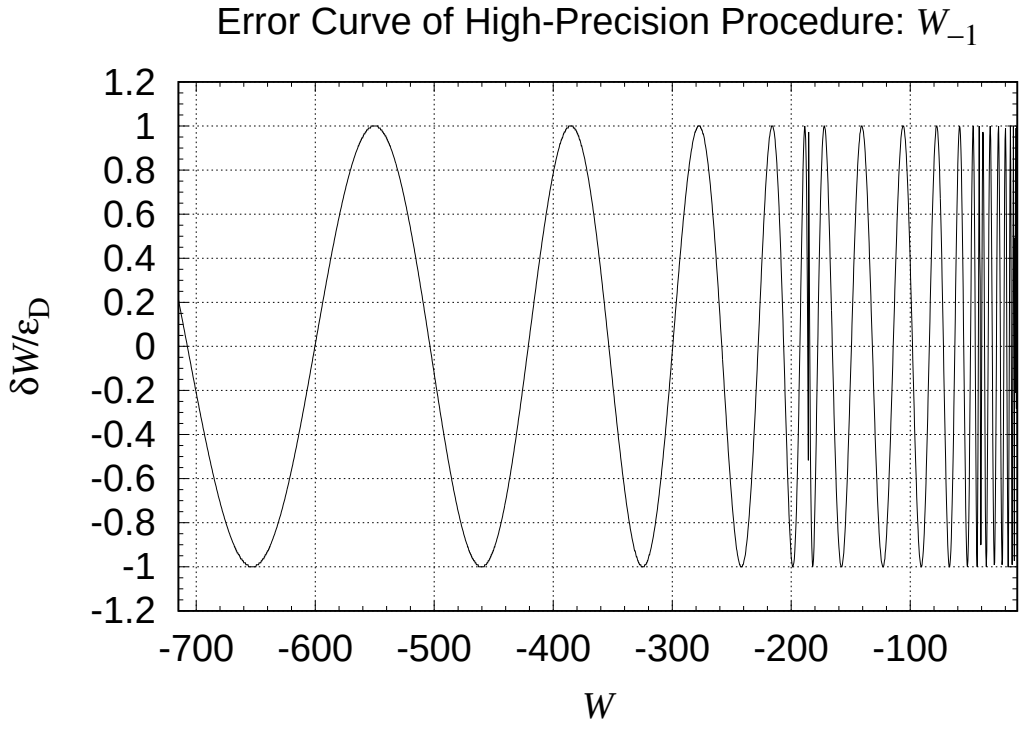

Figure 15: Error curve of high-precision procedure: $W_{-1}(z)$, nearly zero arguments. Same as Fig. 14 but when $-714.196 \leq W_{-1}(z) \leq -10$.

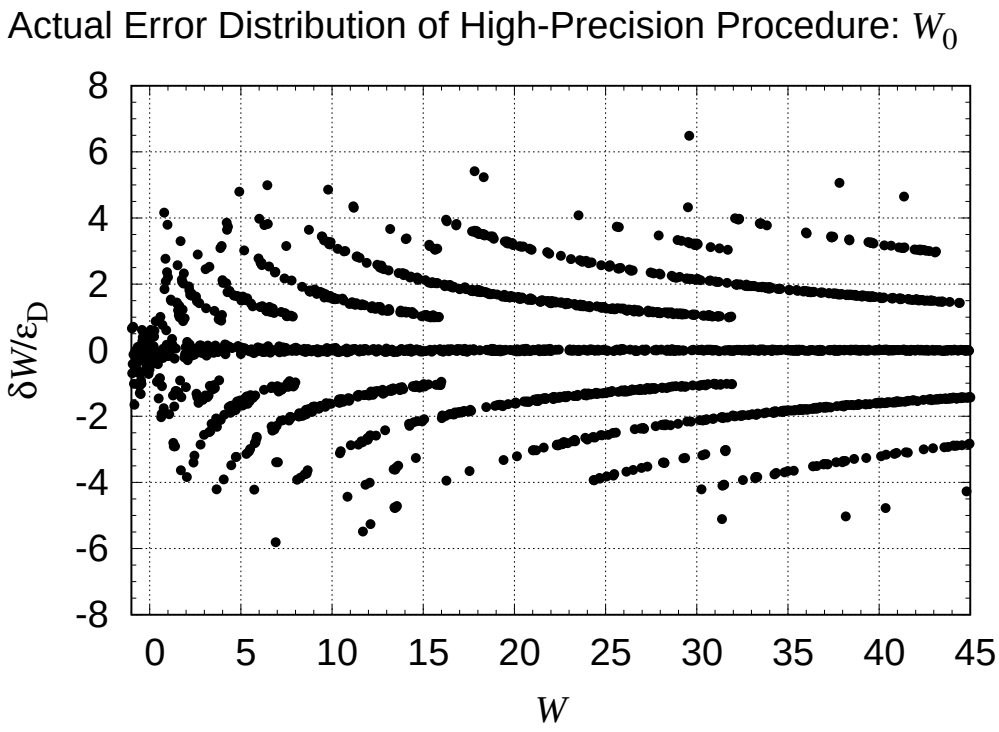

Figure 16: Actual error distribution of new high-precision procedure: $W_0(z)$. Same as Fig. 12 but when the new high-precision procedure was evaluated in the double-precision environment. The differences from Fig. 12 are all due to the round-off errors.

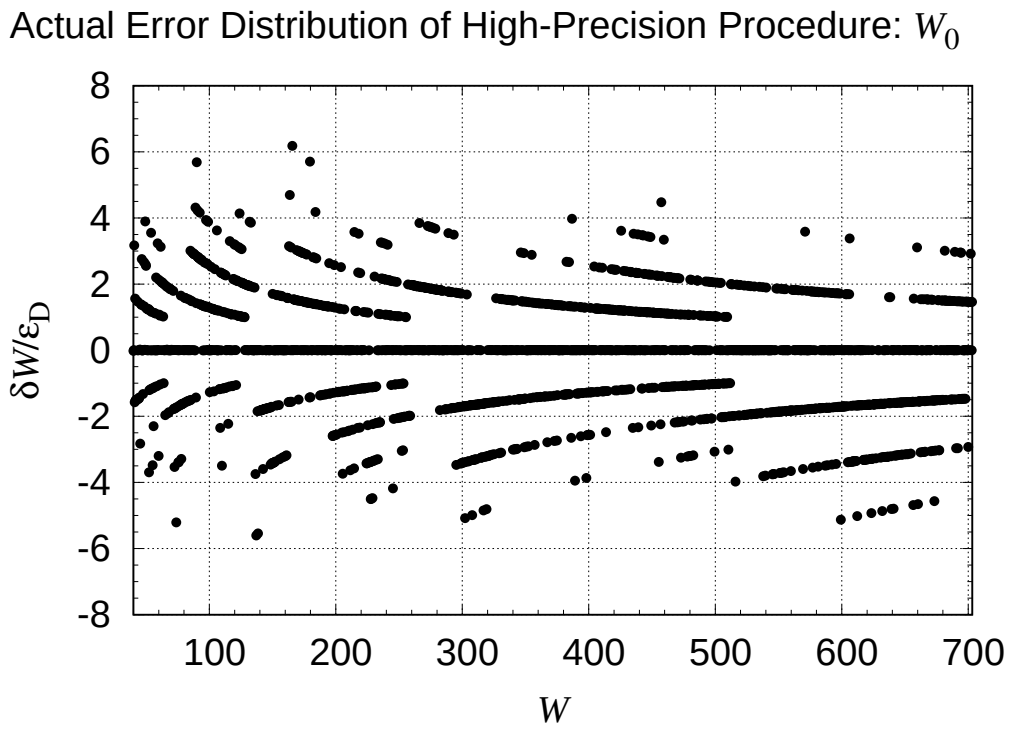

Figure 17: Actual error distribution of new high-precision procedure: $W_0(z)$, large arguments. Same as Fig. 13 but when the new high-precision procedure was evaluated in the double-precision environment.

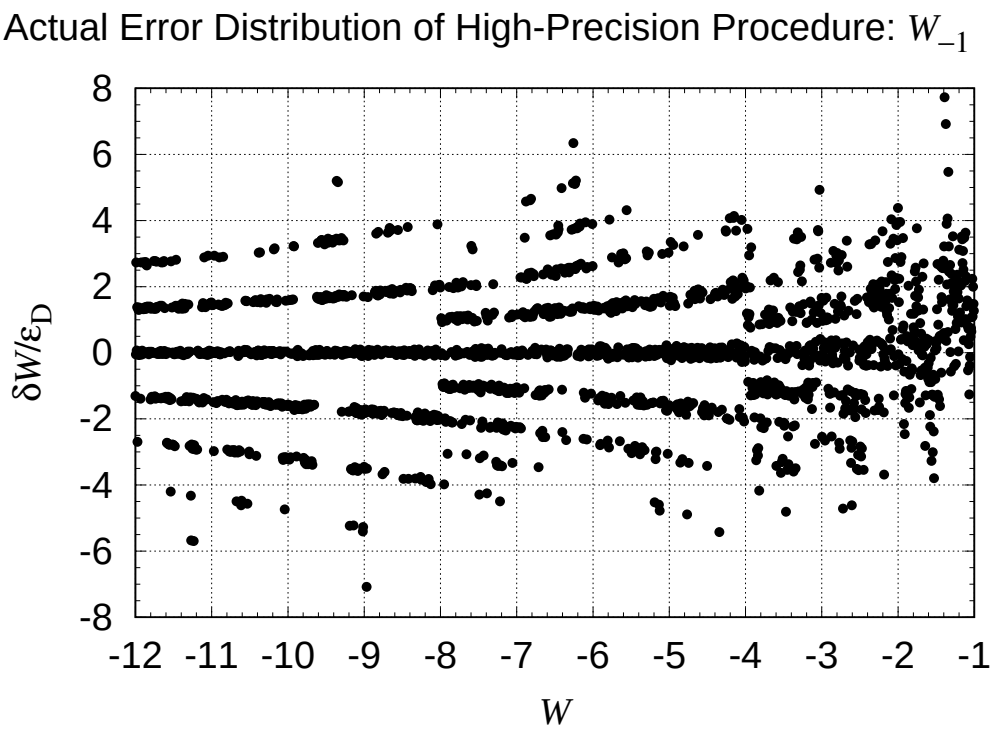

Figure 18: Actual error distribution of new high-precision procedure: $W_{-1}(z)$. Same as Fig. 14 but when the new high-precision procedure was evaluated in the double-precision environment.

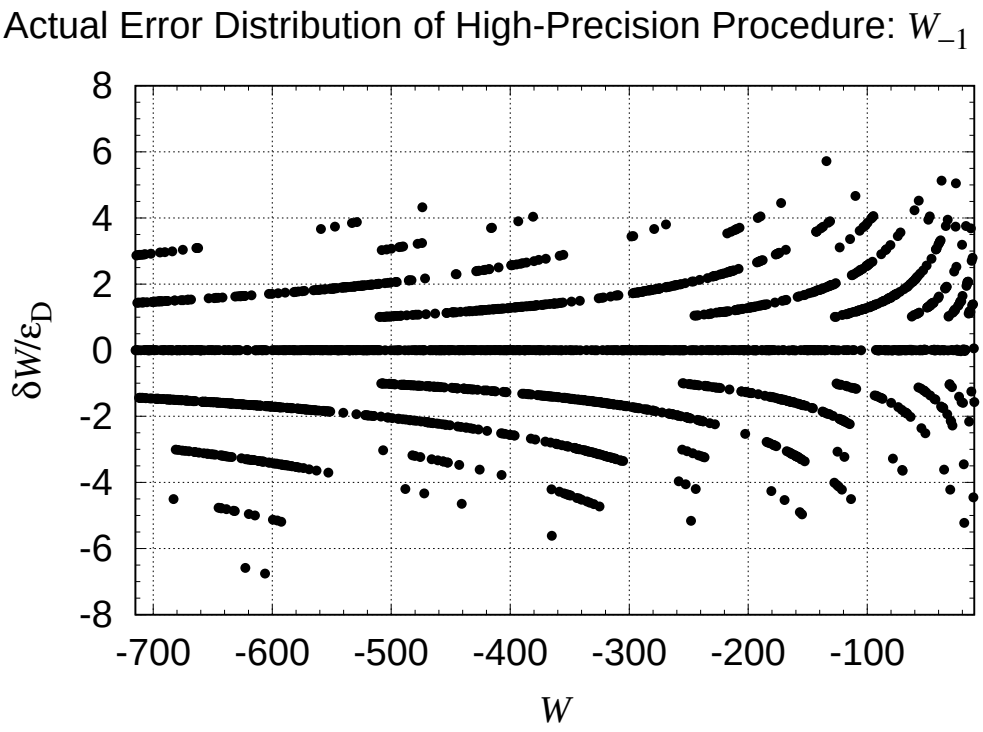

Figure 19: Actual error distribution of new high-precision procedure: $W_{-1}(z)$, nearly zero arguments. Same as Fig. 15 but when the new high-precision procedure was evaluated in the double-precision environment.

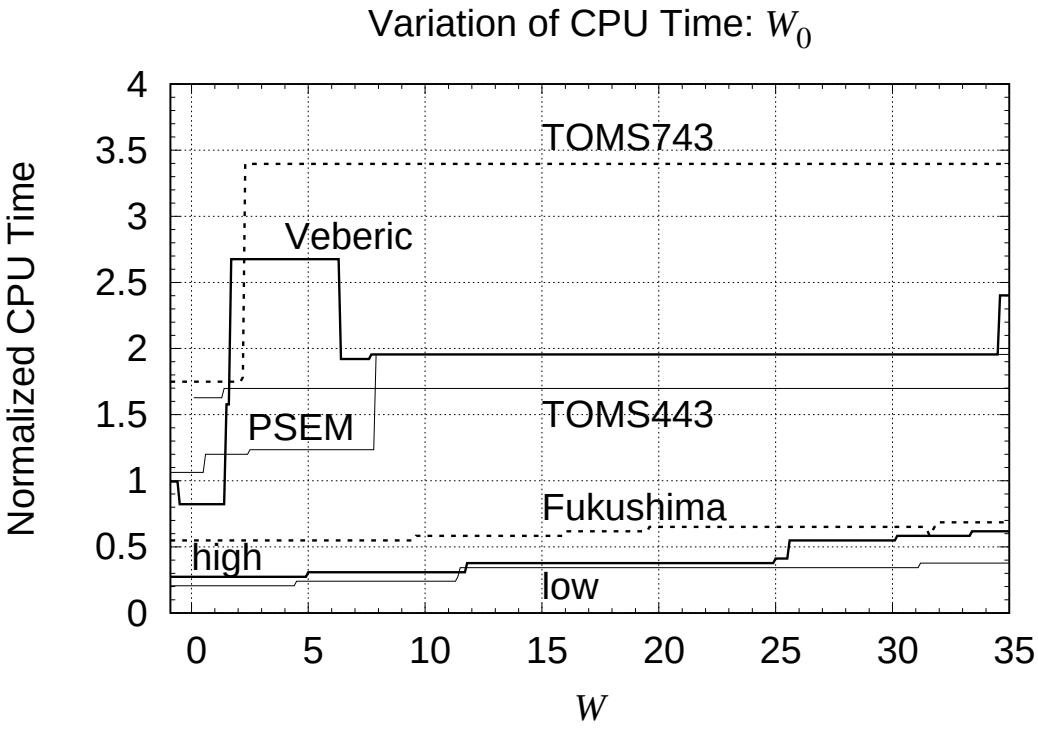

Figure 20: Variation of CPU time: $W_0(z)$. Shown is the variation of CPU time computing $W_0(z)$ by various procedures. All the computer programs were coded in Fortran 90 and compiled by the Intel Visual Fortran Compiler 19.1, or the so-called `ifort 19.1`, with the maximum optimization flags. Then, the CPU times are (i) measured at a commercial PC with the Intel Core i9-9980HK CPU, (ii) averaged over $2^{20} \approx 1.05 \times 10^6$ executions, (iii) normalized by that of the single execution of the logarithm function provided by the Intel mathematical library, and (iv) displayed as a function of the function value for the function value interval, $-1 < W \leq 35$. Compared are seven procedures: (i) the TOMS algorithm 443 [31] which works only when $W > 0$, denoted by `TOMS443`, and drawn by a thin curve, (ii) the TOMS algorithm 743 [33] denoted by `TOMS743` and drawn by a broken curve, (iii) the method of Veberic [34] denoted by `Veberic` and drawn by a thick curve, (iv) the previous method of ours [35] denoted by `Fukushima` and drawn by a broken curve, (v) the method of Vazquez-Leal et al. [36] denoted by `PSEM` and drawn by a thin curve, (vi) the new low-precision procedure denoted by `low` and drawn by a thin curve, and (vii) the new high-precision procedure denoted by `high` and drawn by a thick curve.

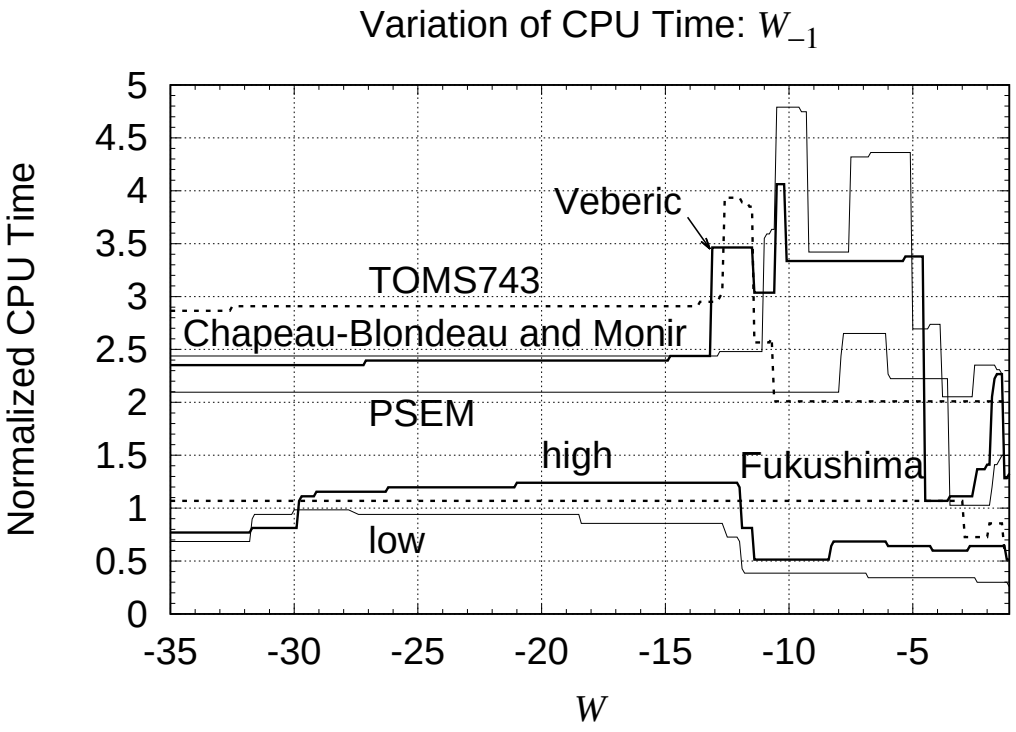

Figure 21: Variation of CPU time: $W_{-1}(z)$. Same as Fig. 20 but computing $W_{-1}(z)$ for the interval, $-35 \leq W < -1$. This time, the TOMS algorihm 443 was removed and the method of Chapeau-Blondeau & Monir [32] was added and drawn by a thin curve.

View publication stats