Hand-to-hand combat with thousand-digit integrals

David H. Bailey* Jonathan M. Borwein†

November 4, 2010

Abstract

In this paper we describe numerical investigations of definite integrals that arise by considering the moments of multi-step uniform random walks in the plane, together with a closely related class of integrals involving the elliptic functions K, K', E and E'. We find that in many cases such integrals can be "experimentally" evaluated in closed form or that intriguing linear relations exist within a class of similar integrals. Discovering these identities and relations often requires the evaluation of integrals to extreme precision, combined with large-scale runs of the "PSLQ" integer relation algorithm. This paper presents details of the techniques used in these calculations and mentions some of the many difficulties that can arise.

1 Introduction

In previous studies by the present authors and a number of collaborators [BBC2006, BBC2007, BBC2007a, BBC2010], we have evaluated integrals to extreme precision, identified these numerical values in analytic terms, then (in most cases) subsequently proved the resulting experimentally discovered formulas. In some cases, we have had

*Lawrence Berkeley National Laboratory, Berkeley, CA 94720, dhbailey@lbl.gov. Supported in part by the Director, Office of Computational and Technology Research, Division of Mathematical, Information, and Computational Sciences of the U.S. Department of Energy, under contract number DE-AC02-05CH11231.

†Centre for Computer Assisted Research Mathematics and its Applications (CARMA), University of Newcastle, Callaghan, NSW 2308, Australia, jonathan.borwein@newcastle.edu.au. Supported in part by the Australian Research Council.

difficulty finding analytic evaluations for individual integrals, but nonetheless discovered (and subsequently proved) intriguing linear relations *within* a class of integrals. These studies have underscored the enormous power of the experimental method in this arena, although many challenges (both mathematical and computational) must be overcome to obtain reliable results.

2 Research problems

We present in this paper two new applications of this methodology, which, as it turns out, are closely related at a basic mathematical level: (a) multi-step uniform random walks in the plane, and (b) the theory of moments of elliptic integral functions. In (a), earlier studies by one of the present authors and several other collaborators have discovered some very interesting formulas for the resulting integrals, but it is clear that much remains to be done. In (b), earlier studies have found some interesting relations among sets of integrals. In both of these applications, we have found that very high-precision numerical values (at least several hundred digits, and in some cases several thousand digits) are required to provide high levels of numerical confidence. Even so, as we will see below, some issues remain unresolved, because even with well over 1000-digit precision, we still cannot definitively answer some questions. We suspect that many marvelous results remain hidden due to the inadequacy of our tools, which from a perspective ten years hence will doubtless appear to be pathetically underpowered.

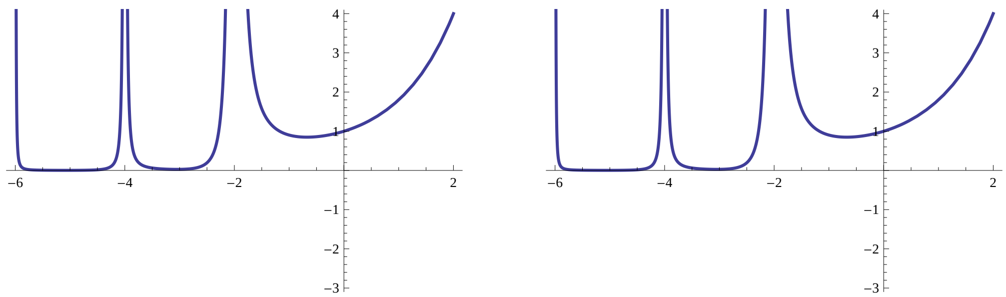
2.1 Ramble integrals

Continuing research commenced in [BNSW2009, BSW2010], for complex s, we consider the n-dimensional integral

$$W_n(s) := \int_{[0,1]^n} \left| \sum_{k=1}^{n} e^{2\pi x_k i}, \right|^s \mathrm{d}x \tag{1}$$

which occurs in the theory of uniform random walk integrals in the plane, where at each step a unit-step is taken in a random direction. Integrals such as (1) are the s-th moment of the distance to the origin after n steps. The study of such walks largely originated with Karl Pearson more than a century ago [Pear1905, Pear1905b, Pear1906], and in his honor we will call these *ramble integrals*. Such integrals can be studied by a mixture of analytic, combinatoric, algebraic and probabilistic methods and provide both numeric and symbolic computation challenges. Nearly all of the results cited in [BNSW2009, BSW2010] were initially discovered by experimental explorations.

For $n \geqslant 3$, the integral (1) is well-defined and analytic for $\operatorname{Re} s > -2$. Its analytic continuation to the complex plane features poles at certain negative integers [BNSW2009]. Figure 1 shows the continuations of W_3 and W_4 on the negative real axis. W_3 has poles at negative even integers. The graphs suggest that the functions are zero at negative odd integers, but they are not.



(a) W_3 (b) W_4

Figure 1: W_3, W_4 analytically continued to the real line.

It is easy to determine that $W_1(s) = 1$, and $W_2(s) = \binom{s}{s/2}$. Also, it is proven in [BNSW2009] that, for k a nonnegative integer,

$$W_3(k) \quad = \quad \operatorname{Re}\, {}_3F_2\left({\frac{1}{2}, -\frac{k}{2}, -\frac{k}{2} \atop 1, 1} \middle| 4 \right), \tag{2}$$

where F denotes the generalized hypergeometric function. The following specific values were also established in [BNSW2009]:

$$W_3(1) \quad = \quad \frac{4\sqrt{3}}{3}\left({}_3F_2\left({-\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2} \atop 1, 1} \middle| \frac{1}{4} \right) - \frac{1}{\pi} \right) + \frac{\sqrt{3}}{24}\, {}_3F_2\left({\frac{1}{2}, \frac{1}{2}, \frac{1}{2} \atop 2, 2} \middle| \frac{1}{4} \right) \tag{3}$$

$$= \quad \frac{3}{16}\frac{2^{1/3}}{\pi^4}\Gamma^6\left(\frac{1}{3}\right) + \frac{27}{4}\frac{2^{2/3}}{\pi^4}\Gamma^6\left(\frac{2}{3}\right), \tag{4}$$

$$W_3(-1) \quad = \quad 2\sqrt{3}\,\frac{K^2\left(k_3\right)}{\pi^2} = \frac{3}{16}\,\frac{2^{1/3}}{\pi^4}\,\Gamma^6\left(\frac{1}{3}\right) = \frac{2^{\frac{1}{3}}}{4\pi^2}\beta^2\left(\frac{1}{3}\right), \tag{5}$$

where $\beta(x) = B(x, x)$ is a central Beta-function value [Borw1987]. Similar expressions can be given for W_3 evaluated at odd integers.

When s is an even positive integer, the moments $W_n(s)$ take explicit integer values:

$$W_n(2k) \quad = \quad \sum_{a_1+\cdots+a_n=k} \binom{k}{a_1,\ldots,a_n}^2. \tag{6}$$

Based on these results and others, the following conjecture was first presented in [BNSW2009]. A complete resolution when $n = 2$ is given in [BSWZ2010], and progress was made on the general case in [BSW2010].

Conjecture 1. *For positive integers n and complex s,*

$$W_{2n}(s) \overset{?}{=} \sum_{j\geqslant 0} \binom{s/2}{j}^2 W_{2n-1}(s-2j). \tag{7}$$

It was shown in 1906 by Kluyver, see [BNSW2009], that the probability that an n-step walk ends up within a disc of radius α is given by

$$P_n(\alpha) \quad = \quad \alpha \int_0^\infty J_1(\alpha x) J_0^n(x)\,\mathrm{d}x. \tag{8}$$

Using this result, David Broadhurst [Broad2009] established

$$W_n(s) \quad = \quad 2^{s+1-k}\frac{\Gamma(1+\frac{s}{2})}{\Gamma(k-\frac{s}{2})}\int_0^\infty x^{2k-s-1}\left(-\frac{1}{x}\frac{\mathrm{d}}{\mathrm{d}x}\right)^k J_0^n(x)\,\mathrm{d}x, \tag{9}$$

which is valid whenever $2k > s > -\frac{n}{2}$. Here $J_\nu(z)$ denotes the *Bessel function of the first kind* [AS1970, Ch. 9]. This representation allowed Broadhurst to provide confirmation of Conjecture 1 to 50 places for $n \leqslant 13$.

Reasoning from the above results and applying a number of other results, the authors of [BSW2010] deduced a large number of intriguing results, including the following items (here Cl denotes the *Clausen* function, and γ denotes the Euler-

Mascheroni constant $= 0.5772156649015\ldots$):

$$\operatorname{Res}_{-2}(W_3) = \frac{8+12W_3'(0)-4W_3'(2)}{9} = \frac{2}{\sqrt{3}\pi} \tag{10}$$

$$\begin{aligned} W_3'(0) &= \frac{1}{2}\int_0^1\int_0^1 \log(4\sin(\pi y)\cos(2\pi x)+3-2\cos(2\pi y))\,\mathrm{d}x\,\mathrm{d}y \\ &= \int_{1/6}^{5/6}\log(2\sin(\pi y))\,\mathrm{d}y = \frac{1}{\pi}\,\operatorname{Cl}\left(\frac{\pi}{3}\right) \end{aligned} \tag{11}$$

$$W_3'(2) = 2+\frac{3}{\pi}\operatorname{Cl}\left(\frac{\pi}{3}\right)-\frac{3\sqrt{3}}{2\pi} \tag{12}$$

$$\begin{aligned} W_4'(0) &= \frac{3}{8\pi^2}\int_0^\pi\int_0^\pi \log\left(3+2\,\cos x+2\,\cos y+2\,\cos(x-y)\right)\,\mathrm{d}x\,\mathrm{d}y \\ &= \frac{7}{2}\frac{\zeta(3)}{\pi^2} \end{aligned} \tag{13}$$

$$\operatorname{Res}_{-2}(W_5) = \frac{16+1140W_5'(0)-804W_5'(2)+64W_5'(4)}{225} \tag{14}$$

$$= \frac{\sqrt{5/3}}{\pi}\,{}_2F_1\left(\left.{\frac{1}{3},\frac{2}{3} \atop 1}\right|\frac{1-\sqrt{5}}{2}\right)^2 \tag{15}$$

$$\operatorname{Res}_{-4}(W_5) = \frac{26\operatorname{Res}_{-2}(W_5)-16-20W_5'(0)+4W_5'(2)}{225} \tag{16}$$

$$W_n'(0) = \log(2)-\gamma-\int_0^1\left(J_0^n(x)-1\right)\frac{\mathrm{d}x}{x}-\int_1^\infty J_0^n(x)\frac{\mathrm{d}x}{x} \tag{17}$$

$$= \log(2)-\gamma-n\int_0^\infty \log(x)J_0^{n-1}(x)J_1(x)\,\mathrm{d}x \tag{18}$$

$$W_n''(0) = n\int_0^\infty\left(\log\left(\frac{2}{x}\right)-\gamma\right)^2 J_0^{n-1}(x)J_1(x)\,\mathrm{d}x \tag{19}$$

$$W_n'(-1) = (\log 2-\gamma)W_n(-1)-\int_0^\infty \log(x)J_0^n(x)\,\mathrm{d}x \tag{20}$$

$$W_n'(1) = \int_0^\infty \frac{n}{x}J_0^{n-1}(x)J_1(x)\left(1-\gamma-\log(2x)\right)\,\mathrm{d}x \tag{21}$$

While these and other evaluations have been confirmed, and all — most recently (15) in [BSWZ2010] — are now proven, there is continuing interest in extending these results to larger specific values of n and s.

2.2 Elliptical function integrals

In another very recent study of moments of elliptic integral functions, we examined integrals of the form

$$I(n_0, n_1, n_2, n_3, n_4) \;:=\; \int_0^1 x^{n_0} K^{n_1}(x) K'^{n_2}(x) E^{n_3}(x) E'^{n_4}(x)\, \mathrm{d}x, \tag{22}$$

where the elliptic functions K, E and their complementary versions are given by:

$$\begin{aligned} K(x) &:= \int_0^1 \frac{\mathrm{d}t}{\sqrt{(1-t^2)(1-x^2t^2)}} \\ K'(x) &:= K(\sqrt{1-x^2}) \\ E(x) &:= \int_0^1 \frac{\sqrt{1-x^2t^2}}{\sqrt{1-t^2}}\, \mathrm{d}t \\ E'(x) &:= E(\sqrt{1-x^2}). \end{aligned} \tag{23}$$

The motivation for this study was the discovery in [BSW2010] of the following elliptic integral representations for the moments of a 4-step walk. We found that

$$W_4(-1) \;=\; \frac{8}{\pi^3} \int_0^1 K^2(x)\, \mathrm{d}x \tag{24}$$

$$W_4(1) \;=\; \frac{96}{\pi^3} \int_0^1 E'(x) K'(x)\, \mathrm{d}x - 8\, W_4(-1), \tag{25}$$

and, inter alia, that

$$2\int_0^1 K(x)^2 \mathrm{d}x \;=\; \int_0^1 K'(x)^2 \mathrm{d}x \;=\; \left(\frac{\pi}{2}\right)^4 {}_7F_6\left(\left. {\frac{5}{4}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2} \atop \frac{1}{4}, 1, 1, 1, 1, 1} \right| 1 \right), \tag{26}$$

while

$$2\int_0^1 K'(x) E'(x)\ \mathrm{d}x = \int_0^1 (1-x^2) K'(x)^2\ \mathrm{d}x. \tag{27}$$

These last two equations were unfamiliar to us and our colleagues, and so we determined to see what other such relations held. This investigation has led us to the need for systematic exploration, via high-precision numerical computation, of a large collection of these integrals.

3 High-precision numerical integration

In several previous studies [BBC2006, BBC2007, BBC2007a, BBC2010], we have relied heavily on an experimental approach, wherein we compute integrals to very high precision and then use tools such as the PSLQ algorithm [FBA1999, BB2000] to identify the resulting numerical values. But in contrast to the majority of the integrals we computed in the earlier studies, integrals such as

$$\int_0^\infty \log(x) J_0^{n-1}(x) J_1(x)\,\mathrm{d}x\,\mathrm{d}t, \tag{28}$$

which we mentioned in (18) above, present considerable challenges.

3.1 Gaussian quadrature

In our previous studies, we have used either Gaussian quadrature or the "tanh-sinh" quadrature scheme due to Takahasi and Mori [TM1974]. Gaussian quadrature approximates $\int_{-1}^1 f(x)\,\mathrm{d}x$ as the sum $\sum_{0\le j<N} w_j f(x_j)$, where the abscissas x_j are the roots of the N-th degree Legendre polynomial $P_N(x)$ on $[-1,1]$, and the weights w_j are

$$w_j \quad := \quad \frac{-2}{(N+1)P_N'(x_j)P_{N+1}(x_j)},$$

see [Atkin1993, pg. 187]. The abscissas and weights are independent of $f(x)$.

In our high-precision implementations, we compute an individual abscissa by using a Newton iteration root-finding algorithm with a dynamic precision scheme. The starting value for x_j in these Newton iterations is given by $\cos[\pi(j-1/4)/(N+1/2)]$, which may be calculated using ordinary 64-bit floating-point arithmetic [Press1986, pg. 125]. We compute the Legendre polynomial function values using an N-long iteration of the recurrence $P_0(x) = 0,\ P_1(x) = 1$ and

$$(k+1)P_{k+1}(x) \quad = \quad (2k+1)xP_k(x) - kP_{k-1}(x)$$

for $k \ge 2$. The derivative is computed as $P_N'(x) = N(xP_N(x) - P_{N-1}(x))/(x^2-1)$. Full details are given in [BLJ2005].

For completely regular integrand functions, Gaussian quadrature is typically the fastest quadrature scheme for numerical integration, although it behaves rather poorly in other cases, such as when the integrand function has a singularity at an endpoint. One disadvantage of Gaussian quadrature is that the cost of computing a set of abscissas and weights increases quadratically with N (assuming constant precision level). For many problems of interest, the precision level also increases roughly linearly with N, so this means that the full computational cost typically increases as N^4 or $N^3 \log N$, depending on whether or not FFT-based multiplication is used. Thus for very high precision integral evaluations, the cost of computing abscissas and weights is typically hundreds of times greater than the cost of actually calculating a definite integral using these abscissas and weights. Fortunately, the abscissas and weights for a given N and precision level can be computed once, then saved in memory or on disk.

3.2 Doubly-exponential quadrature

The tanh-sinh quadrature scheme is an example of the class of doubly-exponential schemes. They are based on the observation, rooted in the *Euler-Maclaurin summation* formula, that for certain bell-shaped integrands (namely those where the function and all higher derivatives rapidly approach zero at the endpoints of the interval), a simple block-function or trapezoidal approximation to the integral is remarkably accurate [Atkin1993, pg. 180]. This principle is exploited in the tanh-sinh scheme by transforming the integral of a given function $f(x)$ on a finite interval such as $[-1, 1]$ to an integral on $(-\infty, \infty)$, by using the change of variable $x = g(t)$, where $g(t) = \tanh(\pi/2 \cdot \sinh t)$. The function $g(t)$ has the property that $g(x) \to 1$ as $x \to \infty$ and $g(x) \to -1$ as $x \to -\infty$, and also that $g'(x)$ and all higher derivatives rapidly approach zero for large positive and negative arguments. Thus one can write, for $h > 0$,

$$\int_{-1}^{1} f(x)\,\mathrm{d}x \;=\; \int_{-\infty}^{\infty} f(g(t))g'(t)\,\mathrm{d}t \;\approx\; h \sum_{j=-N}^{N} w_j f(x_j), \tag{29}$$

where the abscissas x_j and weights w_j are given by

$$\begin{aligned} x_j &= g(hj) \;=\; \tanh\left(\pi/2 \cdot \sinh(hj)\right) \\ w_j &= g'(hj) \;=\; \pi/2 \cdot \cosh(hj) / \cosh\left(\pi/2 \cdot \sinh(hj)\right)^2, \end{aligned} \tag{30}$$

and where N is chosen large enough that terms beyond N (positive or negative) are smaller than the "epsilon" of the numeric precision being used. Full details are given in [BLJ2005].

The tanh-sinh algorithm has two key advantages over classical schemes such as Gaussian quadrature. First of all, tanh-sinh often can be applied even when $f(x)$ has an infinite derivative or an integrable singularity at one or both endpoints. Also, the

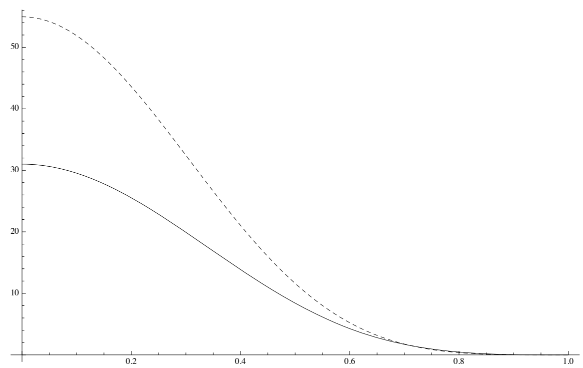


Figure 2: Plot of $f(x) = \sin^p(\pi x)\zeta(p, x)$ on $[0, 1]$ for $p = 3$ (solid) and $p = 3.5$ (dashed).

cost of computing abscissas and weights increases only *linearly* with N (assuming constant precision level). Even if the precision level is assumed to increase linearly with N, this still leaves a N^3 or $N^2 \log N$ scaling factor. Thus for very high precision levels, the cost of computing abscissas and weights for tanh-sinh quadrature is many times less than that of Gaussian quadrature. On the other hand, the actual computation of an integral using tanh-sinh is typically more costly than Gaussian quadrature when the integrand function is regular.

It is not always obvious, just by visual inspection of the integrand function, to determine whether a given integrand function is regular. Consider the function $f(x) = \sin^p(\pi x)\zeta(p, x)$ for various p. When $p = 3$, for instance, this function and all of its higher derivatives are regular on the integral $[0, 1]$. But when $p = 3.5$, even though the plot of the function itself looks entirely unremarkable, the plot of its fourth derivative has severe blow-up singularities at 0 and 1. This is illustrated in Figures 2 and 3. Because of these singularities in the fourth and higher derivatives, Gaussian quadrature gives very poor results for this integrand function when $p = 3.5$, even though it works quite well when $p = 3$. The tanh-sinh scheme, in contrast, is not bothered by these singularities and gives high-precision results in both cases (although it is slower than Gaussian quadrature for the $p = 3$ case).

4 Numerical integration of oscillatory functions

Both Gaussian quadrature and the tanh-sinh scheme are defined for the finite interval $[-1, 1]$. Functions on other finite intervals can be integrated simply by an appropriate

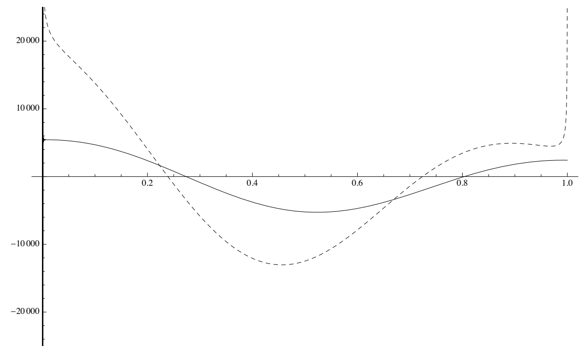


Figure 3: Plot of fourth derivative of $f(x) = \sin^p(\pi x)\zeta(p, x)$ on $[0, 1]$ for $p = 3$ (solid) and $p = 3.5$ (dashed).

linear scaling of the abscissas. Functions on a semi-infinite interval such as $[0, \infty)$ can be handled by a simple transformation such as:

$$\int_0^\infty f(t)\,\mathrm{d}t \quad = \quad \int_0^1 f(t)\,\mathrm{d}t + \int_0^1 \frac{f(1/t)\,\mathrm{d}t}{t^2}. \tag{31}$$

Unfortunately, due to the oscillatory nature of Bessel functions, neither Gaussian quadrature nor tanh-sinh quadrature is effective for high-precision integration of functions involving Bessel functions over a half infinite interval.

Some integrals involving oscillatory integrands with trigonometric functions, such as $\int_0^\infty (1/x \sin x)^p\,\mathrm{d}x$, can be efficiently evaluated by applying a clever technique introduced by Ooura and Mori in 1991 [Ooura1991]. For other oscillatory integrands, such as those that arose in a previous study of p-norms of the sinc function, it is possible to convert the integral into an infinite summation [BB2010]. But neither this scheme nor the Ooura scheme is effective for integrating expressions with Bessel functions.

4.1 The Sidi mW algorithm

The approach that we have found effective for Bessel functions is known as the Sidi mW extrapolation algorithm, as described in a 1994 paper by Lucas and Stone [LS1994], which in turn is based on two earlier papers by Sidi [Sidi1982, Sidi1988].

This algorithm evaluates an integral on a semi-infinite integral, such as $\int_a^\infty f(x)\,\mathrm{d}x$, as follows. Initialize

$$\begin{aligned}
x_0 &= \pi \\
S_0 &= \int_a^{x_0} f(x)\,\mathrm{d}x \\
T_0 &= \int_{x_0}^{x_0+\pi} f(x)\,\mathrm{d}x \\
M_{0,-1} &= S_0/T_0 \\
N_{0,-1} &= 1/T_0.
\end{aligned} \tag{32}$$

Then iterate the following calculations, for $t = 0, 1, 2, \cdots$, until successive values of W_t are equal to within a specified tolerance:

$$\begin{aligned}
x_{t+1} &= (t+2)\pi \\
S_{t+1} &= S_t + T_t \;=\; \int_0^{x_{t+1}} f(x)\,\mathrm{d}x \\
T_{t+1} &= \int_{x_{t+1}}^{x_{t+1}+\pi} f(x)\,\mathrm{d}x, \\
M_{t+1,-1} &= S_{t+1}/T_{t+1} \\
N_{t+1,-1} &= 1/T_{t+1},
\end{aligned} \tag{33}$$

and, for $s = t, t-1, t-2, \cdots, 2, 1, 0$,

$$\begin{aligned}
M_{s,t-s} &= \frac{M_{s,t-s-1} - M_{s+1,t-s-1}}{1/x_s - 1/x_{t+1}} \\
N_{s,t-s} &= \frac{N_{s,t-s-1} - N_{s+1,t-s-1}}{1/x_s - 1/x_{t+1}},
\end{aligned} \tag{34}$$

and, finally,

$$W_t \;=\; \frac{M_{0,t}}{N_{0,t}}. \tag{35}$$

Successive values of W_t are the successive best estimates of the integral $\int_a^\infty f(x)\,\mathrm{d}x$.

From a computer programming perspective, it should be pointed out that it is not necessary to store the entire M and N arrays in memory. As we will mention in more detail below, this is a crucial consideration in the actual computations described in this paper. Instead, one can define the one-dimensional arrays $\hat{M}_t$ and $\hat{N}_t$ for $t = 0, 1, \cdots, L$, where L is some maximum anticipated iteration count. Then the algorithm above can be performed as stated with $\hat{M}$ and $\hat{N}$ in place of M and N, ignoring the second subscripts, where the formulas in (34) are replaced by

$$\begin{aligned}\hat{M}_s &\leftarrow \frac{\hat{M}_s - \hat{M}_{s+1}}{1/x_s - 1/x_{t+1}} \\ \hat{N}_s &\leftarrow \frac{\hat{N}_s - \hat{N}_{s+1,}}{1/x_s - 1/x_{t+1}}.\end{aligned} \tag{36}$$

These formulas must be performed for indices s in reverse order: $s = t, t-1, t-2, \cdots, 2, 1, 0$.

Note that in the above, we are taking the abscissas $x_t = (t+1)\pi$. A spacing of π is convenient for integrals involving Bessel functions, but another spacing might be better for other functions. In some other formulations of this technique, x_t are taken to be the successive zeroes of the function $f(x)$.

5 Computation of Bessel function integrals

As we mentioned above, one of our goals in this study was to compute the ramble integrals $W_n'(0)$ to very high precision, for various integers n, not only so that we can verify analytic results but also so that we can we experimentally "recognize" these numerical values using PSLQ or similar tools (see the next section). Such derivative values also yield logarithmic Mahler measures; see [BSWZ2010].

We found that the Sidi mW scheme described in the previous section was an excellent solution for the case where n is an odd integer. We applied this scheme to formulas (17) and (18) of Section 1, which we re-list here:

$$W_n'(0) = \log(2) - \gamma - \int_0^1 (J_0^n(x) - 1)\,\frac{\mathrm{d}x}{x} - \int_1^\infty J_0^n(x)\frac{\mathrm{d}x}{x} \tag{37}$$

$$= \log(2) - \gamma - n\int_0^\infty \log(x) J_0^{n-1}(x) J_1(x)\,\mathrm{d}x \tag{38}$$

The first integral of (37) is on a finite interval and can be evaluated by tanh-sinh. The second integral of (37) involves an oscillatory integrand with Bessel functions on a semi-infinite interval. Here we took $a = 1$ in the initialization of the Sidi mW scheme (i.e., in (32)). For the integral in (38), which also involves an oscillatory function on a semi-infinite interval, we took $a = 0$ in the initialization. In both of these cases, we used tanh-sinh quadrature to compute the two integrals in (32), since the integrand of the first integral in particular is singular at zero. For all the remaining integrals (33), we employed Gaussian quadrature, since these integrals are completely regular. However, this required the computation of a complete set of 1000-digit values of Gaussian abscissas and weights, for the abscissa-weight set sizes $N = 6, 12, 24, \cdots, 3 \cdot 2^{11}$, which computation required over one hour run time.

All of these computations, except where noted specifically below, were performed using the ARPREC arbitrary precision computation software [BHLT2002], which is freely available from the first author's website. ARPREC consists of a set of C++ routines that perform basic arithmetic routines and numerous transcendental functions, together with C++ and Fortran-90 translation modules. These translation modules permit one to perform a calculation, such as Gaussian quadrature with Sidi mW extrapolation, merely by changing the type statements of those variables that one wishes to be treated as high precision. Then when one or more of these variables appears in an expression, the compiler generates references to the appropriate low-level arbitrary precision library routines.

To compute Bessel functions, we used the following formulas [AS1970, Sec 9.1, 9.2]:

$$J_n(z) \;=\; \left(\frac{z}{2}\right)^n \sum_{n=0}^{\infty} \frac{(-z^2/4)^k}{k!\Gamma(n+k+1)} \tag{39}$$

and

$$\begin{aligned} J_n(z) &= \sqrt{\frac{2}{\pi z}}\,[P_n(z,\infty)\cos\chi - Q_n(z,\infty)\sin\chi] \\ P_n(z,D) &:= 1 + \sum_{k=1}^{D} (-1)^k \frac{(\mu-1)(\mu-3^2)\cdots(\mu-(4k-1)^2)}{(2k)!(8z)^{2k}} \\ Q_n(z,D) &:= \sum_{k=0}^{D} (-1)^k \frac{(\mu-1)(\mu-3^2)\cdots(\mu-(4k+1)^2)}{(2k+1)!(8z)^{2k+1}}, \end{aligned} \tag{40}$$

where $\chi := z - (n/2 + 1/4)\pi$, $\mu := 4n^2$ and $P_n(z, D), Q_n(z, D)$ give genuine asymptotic expansions. The first formula (39) for $J_n(z)$ is used for modest-sized values of z, whereas the second set of formulas (40) is used for large values. In our implementation, we use the first formula for $|z| < 1.25d$, where d is the number of digits of precision, and the second for larger values. One other possibility for the Bessel function is a technique described in [BBC2008], which uses the error function to produce a uniformly convergent formula for the Bessel functions that avoids the need for different schemes in different regions. This may be worthy of further research if even higher precision is required.

5.1 Odd values of $W_n'(0)$

We found that based on our implementations of these algorithms, formula (17) is faster than (18) for computing $W_n'(0)$. Using formula (17), we were able to compute $W_n'(0)$, for $n = 3, 5, 7, \cdots, 17$, up to 1000-digit accuracy, although in each case hundreds of iterations of the Sidi mW scheme (and, thus, hundreds of individual quadratures) were required. We believe our final results are accurate to 1000-digit precision, because after several hundred iterations, the extrapolated integral results agreed to this level. Run times (in seconds), iteration counts and 30-digit numerical values are shown in Table 1. These calculations were performed on an eight-core 2 Ghz Apple workstation. The full 1000-digit values are available from the authors.

5.2 Even values of $W_n'(0)$

For even n, we found that even the extrapolation scheme described in the previous section was not able to produce a highly accurate value of the respective integrals. We do not yet have a satisfactory explanation for this, but doubtless it is due to the behavior of high-order derivatives of these integrand functions.

However, we were able to compute modestly high precision (80-digit) numerical values in *Maple* by applying the asymptotic formulas for the Bessel function mentioned above (39). In particular, we used (17) and (39) to write

$$\begin{aligned} W_n'(0) \approx & - \int_0^1 \left(J_0^n(x) - 1\right) \frac{\mathrm{d}x}{x} + \log(2) - \gamma - \int_1^C J_0^n(x) \frac{\mathrm{d}x}{x} \\ & - \int_C^\infty \left\{ \sqrt{\frac{2}{\pi x}} \left[P_0(x, D) \cos(x - \pi/4) - Q_0(x, D) \sin(x - \pi/4) \right] \right\}^n \frac{\mathrm{d}x}{x}. \end{aligned} \tag{41}$$

For $n = 4, 6, 8, 10$ we set $C = 1000$ and $D = 100$, which is based on a rough Stirling estimate of the error in truncation after D terms. We applied (31) to the final integral on an infinite interval and then applied a Gaussian rule to all integrals. Each computation took of the order of 20 minutes on a modern laptop, with no serious attempt to optimize the code. Fifty-digit numerical values[1] of $W_n'(0)$ for even integers n are given in Table 2.

We are quite confident in these numerical results, since we have checked them in several ways. For instance, the result of (41) for $n = 6$ matched to 80-digit precision

[1] For $n = 10$, attempts to use variants of (41) to higher precision than 50 places using *Maple* exhausted the available memory on the system being employed.

n	Precision	Iterations	Time	30-digit numerical values
3	200	159	123	0.323065947219450514093636510723 8...
	400	320	2046	
	1000	802	106860	
5	200	159	249	0.544412561752185585195878062745 0...
	400	319	2052	
	1000	801	106860	
7	200	157	249	0.702926292476967266787823944395 2...
	400	318	2050	
	1000	800	106860	
9	200	156	248	0.824156239532388694820522824849 6...
	400	317	2120	
	1000	799	106800	
11	200	155	247	0.921850886732653697565891527970 3...
	400	316	4123	
	1000	796	213480	
13	200	154	246	1.003583530489320110604453874320 8...
	400	314	4113	
	1000	796	213540	
15	200	152	245	1.073826217256856036184252781500 3...
	400	313	4096	
	1000	795	213480	
17	200	151	244	1.135410703767411072953239250042 9...
	400	312	4104	
	1000	794	213360	

Table 1: Precision levels (in digits), Sidi mW iteration counts, run times (in seconds) and approximate numerical values for $W_n'(0)$ calculations, for odd integers n.

n	50-digit numerical values
4	0.42627839881750579092352142659616687305800676962 96...
6	0.62731707483690980718358664940461715251475543810 78...
8	0.76683108806961275701405175616677786312599958330 27...
10	0.87532865811448453408495821795044968468699477558 65...

Table 2: Approximate numerical values of $W_n'(0)$, for even integers n.

a computation based on a conjecture due to Villegas [BSWZ2010],[2] which, when cast in our terms, is:

$$W_6'(0) \stackrel{?}{=} \left(\frac{3}{\pi^2}\right)^3 \int_0^\infty \eta^2(e^{-t})\eta^2(e^{-2t})\eta^2(e^{-3t})\eta^2(e^{-6t})\, t^4 \,\mathrm{d}t, \tag{42}$$

where

$$\eta(q) := q^{1/24} \prod_{n\geqslant 1}(1-q^n) = q^{1/24} \sum_{n=-\infty}^{\infty} (-1)^n q^{n(3n+1)/2}. \tag{43}$$

Similarly, for $n = 4$ our 80-digit result agrees to full precision with the closed form given in (13). Moreover, for odd $n \leq 9$, formula (41) was confirmed to 80-digit precision by comparing with the 1000-digit results obtained using the scheme of subsection 5.1. Finally, the exact form (11) was used to verify the value of $W_3'(0)$, computed as described in subsection 5.1, to 1000-digit precision.

Along this line, we also confirmed, to 600-digit precision, the following conjecture from [BSWZ2010]:

$$W_5'(0) \stackrel{?}{=} \left(\frac{15}{4\pi^2}\right)^{5/2} \int_0^\infty \left\{\eta^3(e^{-3t})\eta^3(e^{-5t}) + \eta^3(e^{-t})\eta^3(e^{-15t})\right\} t^3 \,\mathrm{d}t. \tag{44}$$

6 Computation of elliptical function integrals

As we briefly mentioned above, our research on ramble integrals led us to examine moments of elliptic integral functions of the form

$$I(n_0, n_1, n_2, n_3, n_4) := \int_0^1 x^{n_0} K^{n_1}(x) K'^{n_2}(x) E^{n_3}(x) E'^{n_4}(x)\,\mathrm{d}x, \tag{45}$$

where the elliptic functions K, E and their complementary versions are given by:

$$\begin{aligned} K(x) &:= \int_0^1 \frac{\mathrm{d}t}{\sqrt{(1-t^2)(1-x^2t^2)}} \\ K'(x) &:= K(\sqrt{1-x^2}) \\ E(x) &:= \int_0^1 \frac{\sqrt{1-x^2t^2}}{\sqrt{1-t^2}}\,\mathrm{d}t \\ E'(x) &:= E(\sqrt{1-x^2}). \end{aligned} \tag{46}$$

[2] $W_n'(0) = \mu(1 + \sum_{k=1}^{n-1} x_k)$ is the logarithmic Mahler measure of the polynomial on the right. For $n = 5, 6$ it is conjectured to agree with the eta integrals displayed in (44) and (42) respectively.

To better understand these integrals, we computed a large number of them (4389 individual integrals in total) to extreme precision: 1500-digit precision in most cases, with some to 3000-digit precision. We employed the tanh-sinh scheme here, because most of these integrands have singularities at the endpoints. In contrast to the random walk integrals, each of which required many hours run time, most of the 1500-digit integrals were dispatched in less than one minute, and most of the 3000-digit integrals were dispatched in less than 10 minutes. But with so many integrals to be computed, over 300 CPU-hours of computation were required in total.

We then discovered, using the PSLQ integer relation algorithm [FBA1999], thousands of intriguing relations between these numerical values. Given an input vector of real numbers $(x_1, x_2, \cdots, x_n)$, the PSLQ algorithm finds integers $(a_1, a_2, \cdots, a_n)$, not all zero, such that $a_1 x_1 + a_2 x_2 + \cdots + a_n x_n = 0$, or determines that there is no such vector of integers whose Euclidean norm is less than a certain bound. Several variants of PSLQ, even more efficient than the original algorithm, are described in [BB2000]. In order to obtain numerically meaningful results with PSLQ or any other integer relation scheme, very high precision arithmetic (typically several hundred or a few thousand digits) is required.

As a single example of our results, when we examined the set of all integrals (45) with $n_0 \le D_1 = 4$ and $n_1 + n_2 + n_3 + n_4 = D_2 = 3$ (a total of 100 integrals), we found, using PSLQ, that all such integrals can be expressed in terms of a "basis" set of eight integrals, which are:

$$\begin{aligned} &\int_0^1 K^3(x)\,\mathrm{d}x, && \int_0^1 x^2 K^3(x)\,\mathrm{d}x, \\ &\int_0^1 x^3 K^3(x)\,\mathrm{d}x, && \int_0^1 x^4 K^3(x)\,\mathrm{d}x, \\ &\int_0^1 x^2 K^2(x) K'(x)\,\mathrm{d}x, && \int_0^1 K(x)E(x)K'(x)\,\mathrm{d}x, \\ &\int_0^1 x K(x)E(x)K'(x)\,\mathrm{d}x, && \int_0^1 E(x)K'^2(x)\,\mathrm{d}x. \end{aligned} \tag{47}$$

In other words, we found that the other 92 integrals in the set defined by $D_1 = 4$ and $D_2 = 3$ can each be expressed in terms of an integer linear combination of these eight integrals. Among the 92 experimental relations we discovered for this set are:

$$\begin{aligned}
81\int_0^1 x^3K^2(x)E(x)\,\mathrm{d}x &\stackrel{?}{=} -6\int_0^1 K^3(x)\,\mathrm{d}x - 24\int_0^1 x^2K^3(x)\,\mathrm{d}x \\
&+51\int_0^1 x^3K^3(x)\,\mathrm{d}x + 32\int_0^1 x^4K^3(x)\,\mathrm{d}x \qquad (48)\\
-243\int_0^1 x^3K(x)E(x)K'(x)\,\mathrm{d}x &\stackrel{?}{=} -59\int_0^1 K^3(x)\,\mathrm{d}x + 468\int_0^1 x^2K^3(x)\,\mathrm{d}x \\
&+156\int_0^1 x^3K^3(x)\,\mathrm{d}x - 624\int_0^1 x^4K^3(x)\,\mathrm{d}x - 135\int_0^1 xK(x)E(x)K'(x)\,\mathrm{d}x \qquad (49)\\
-20736\int_0^1 x^4E^2(x)K'(x)\,\mathrm{d}x &\stackrel{?}{=} 3901\int_0^1 K^3(x)\,\mathrm{d}x - 3852\int_0^1 x^2K^3(x)\,\mathrm{d}x \\
&-1284\int_0^1 x^3K^3(x)\,\mathrm{d}x + 5136\int x^4K^3(x)\,\mathrm{d}x - 2592\int_0^1 x^2K^2(x)K'(x)\,\mathrm{d}x \\
&-972\int_0^1 K(x)E(x)K'(x)\,\mathrm{d}x - 8316\int_0^1 xK(x)E(x)K'(x)\,\mathrm{d}x. \qquad (50)
\end{aligned}$$

6.1 Numerical results

The tabulations in Table 3 summarize very briefly the many results that we have found to date, the full details of which, including high-precision numerical values, are available from the authors if desired. The parameters D_1 and D_2 are, as defined above, the upper limit on the power of x, and the sum of the powers of K, K', E, E', respectively. The column labeled "Relations" gives the number of integer relations found in the set of integrals defined by the parameters D_1 and D_2. The column labeled "Basis" gives the size of the basis for this set of integrals. The column labeled "Total" gives the total number of integrals in the set (which is the sum of the two previous columns). The column labeled "Precision" gives the numeric precision that was used both to compute the integrals in the given set and to analyze them using PSLQ. The column labled "Basis norm bound" gives the exclusion bound found by PSLQ for the given basis set—in other words, PSLQ certified that there is no integer relation among the elements of the experimentally determined basis set whose Euclidean norm is less than the stated bound. The column labeled "Max relation norm" gives the largest Euclidean norm of any of the relations that were found in the entire set of integrals defined by D_1 and D_2.

D_1	D_2	Relations	Basis	Total	Precision	Basis norm bound	Max relation norm
0	1	1	3	4	1500	1.582082×10^{298}	2.236068×10^{0}
1	1	5	3	8	1500	2.155768×10^{297}	3.605551×10^{0}
2	1	9	3	12	1500	2.155768×10^{297}	5.916080×10^{0}
3	1	13	3	16	1500	2.155768×10^{297}	1.679286×10^{1}
4	1	17	3	20	1500	2.155768×10^{297}	6.592420×10^{1}
5	1	21	3	24	1500	2.155768×10^{297}	2.419628×10^{2}
0	2	4	6	10	1500	5.609665×10^{261}	2.109502×10^{1}
1	2	12	8	20	1500	4.877336×10^{196}	5.744563×10^{0}
2	2	22	8	30	1500	6.109876×10^{195}	2.293469×10^{1}
3	2	32	8	40	1500	6.109876×10^{195}	2.293469×10^{1}
4	2	42	8	50	1500	6.109876×10^{195}	1.639153×10^{3}
5	2	52	8	60	1500	6.109876×10^{195}	2.428260×10^{3}
0	3	14	6	20	1500	3.871282×10^{262}	2.664001×10^{2}
1	3	34	6	40	1500	2.164052×10^{261}	8.960469×10^{1}
2	3	52	8	60	1500	1.496420×10^{197}	9.666276×10^{2}
3	3	72	8	80	1500	2.829003×10^{196}	2.291372×10^{3}
4	3	92	8	100	1500	8.853827×10^{195}	5.860112×10^{3}
5	3	112	8	120	1500	8.853827×10^{195}	9.240898×10^{4}
0	4	20	15	35	1500	2.689124×10^{104}	1.963656×10^{4}
1	4	53	17	70	1500	6.195547×10^{91}	2.186030×10^{3}
2	4	88	17	105	1500	4.059577×10^{91}	2.970026×10^{4}
3	4	121	19	140	1500	8.856138×10^{81}	5.658994×10^{5}
4	4	156	19	175	1500	2.759846×10^{82}	5.571466×10^{6}
5	4	191	19	210	1500	1.663418×10^{82}	1.857555×10^{5}
0	5	45	11	56	1500	1.256977×10^{142}	1.061532×10^{5}
1	5	101	11	112	1500	2.602478×10^{142}	1.025453×10^{5}
2	5	155	13	168	1500	2.151577×10^{120}	3.953731×10^{5}
3	5	211	13	224	1500	1.314945×10^{120}	3.728547×10^{5}
4	5	265	15	280	1500	5.040597×10^{104}	8.658997×10^{6}
5	5	321	15	336	1500	4.186191×10^{104}	3.954175×10^{11}
0	6	56	28	84	3000	2.958413×10^{105}	1.748907×10^{6}
1	6	138	30	168	3000	2.018080×10^{98}	2.219430×10^{6}
2	6	222	30	252	3000	3.089318×10^{98}	6.301251×10^{8}
3	6	304	32	336	3000	1.324953×10^{92}	2.929549×10^{10}
4	6	388	32	420	3000	9.312061×10^{91}	6.168516×10^{12}
5	6	470	34	504	3000	6.616755×10^{86}	7.199329×10^{13}

Table 3: Summary of relations found in elliptic integral study

In this table, it is worth comparing the last two columns. In each case, our confidence that we have identified a true Z-linearly independent basis set is founded, experimentally speaking, on our observation that the basis norm bound (in the next-to-last column) is, in most cases, enormously larger than the corresponding maximum relation norm (in the last column), suggesting that it is exceedingly unlikely that there are any additional relations among the integrals in the basis set. Indeed, from this data, it appears fairly clear that the basis sets we found are, almost certainly, true basis sets for all cases with $D_2 \leq 5$. For the $D_2 = 6$ cases, however, based on our original 1500-digit calculations, this conclusion was less compelling, since, for instance, when $D_1 = 5$ and $D_2 = 6$, the basis norm bound was "only" 2.243721×10^{45}, which was uncomfortably close to the maximum relation norm, namely 7.199329×10^{13}. Thus we re-ran the last set of results, namely the six $D_2 = 6$ cases, with 3000-digit precision, in order to obtain stronger basis norm bounds. These 3000-digit computations, which are listed at the end of Table 3, were nearly ten times as expensive as the same computations with 1500-digit precision (which are not listed), and required a total of 252 CPU-hours run time. This was more than three times the total of all other cases combined.

In all of the computer runs that were performed in this study, over 99% of the run time was for quadrature calculations. By comparison, the PSLQ runs performed on the resulting numerical values to find underlying relations ran very fast.

These tables have been carefully analyzed and many proofs and extensions have been provided by James Wan in a recent study [Wan2010]. For example, he evaluates the moments $\int_0^1 x^n K'(x)E'(x)\,\mathrm{d}x$ as ${}_7F_6$ hypergeometric functions, as in (26), with similar formulas for the moments of ${K'}^2$ and ${E'}^2$. However, as seems to be more and more the case as experimental computational tools improve, our ability to discover outstrips our ability to prove.

7 Computational experience

In the process of computing these integrals, we encountered numerous difficulties, which, in our experience, are entirely typical of the challenge of computing and analyzing extreme precision numerical values. Some of these difficulties include:

1. Difficulties in computing the integrand function to very high precision, as rapidly as possible, over a wide range of arguments. Computing with 1000-digit precision magnifies run times by at least a factor of 1000, compared with standard 64-bit computations, so highly efficient algorithms are essential.

2. Difficulties in managing working precision level. For example, in the computation of the Bessel function using the first of the formulas above (39), we found that it was necessary to employ *double* the internal working precision (i.e., 2000-digit arithmetic) for this part, because of severe precision loss when positive and negative terms are summed. In our experience, numerical anomalies of this sort that are minor nuisances with, say, double precision arithmetic often are magnified to monumental levels in extreme precision calculations.

3. Difficulties in managing memory. For example, until we discovered the "trick" to avoid storing the entire matrix in (36), we had frequent problems with system errors, because the total size of our arrays was 229 Gbyte (when the Sidi mW iteration limit $L = 10,000$) or 2.2 Gbyte (when $L = 1000$), whereas our system (an 8-core Apple workstation) only had 8 Gbyte real DRAM memory. Even a 2 Gbyte memory requirement causes problems when trying to run eight such calculations simultaneously on eight cores, or when running a single job in parallel using the Message Passing Interface (MPI), since data arrays are replicated in each of the eight tasks.

4. Enormous run times. As can be seen from Table 1, run times for $W_n'(0)$ calculations increase very rapidly with precision level. Run times for the 1000-digit calculations were as high as 59 hours, which is approximately 52 times greater than the 400-digit calculations, and up to 870 times greater than the 200-digit calculations. We converted our program to run in parallel, using MPI, but instead decided to run the eight calculations listed in Table 1 as eight jobs simultaneously on the eight cores of our system. However, it is clear that any future research in this area will need to employ highly parallel computer systems. Techniques for efficient parallel implementation of high-precision integration are described in [BB2008].

8 Conclusion

These results underscore the need for even more effective numerical algorithms and computational tools, and, yes, even higher precision. For example, to gain further confidence that the basis sets we have listed in Table 3 are true Z-linearly independent basis sets, it will be necessary to perform these computations with even higher numerical precision, so that we can obtain larger basis norm bounds. For that matter, extending the results in Table 3 to larger values of the parameters D_1 and D_2 will also require higher precision.

This seems to be almost a universal law of computation, not just in computational mathematics, but in most other arenas of scientific computation as well: whenever more advanced computing tools become available, researchers quickly fully utilize these facilities and ask for more. Thus, there will always be interest in more effective numerical algorithms, more efficient and easier-to-use mathematical software, and, of course, faster and more capacious computer hardware.

Fortunately, the prognosis for the future remains bright, both in light of the inexorable advance of Moore's Law and also in anticipation of expected improvements in computational software. If anything, the pace of advancement may quicken in the future. Ten years from now, we may wonder how we ever accomplished any research work with the primitive facilities that we employ today.

References

[AS1970] Milton Abramowitz and Irene Stegun, *A Handbook of Mathematical Functions*, U.S. National Bureau of Standards, Washington, DC, 1970.

[Atkin1993] Kendall E. Atkinson, *Elementary Numerical Analysis*, John Wiley and Sons, San Francisco,1993.

[BB2000] David H. Bailey and David J. Broadhurst, "Parallel Integer Relation Detection: Techniques and Applications," *Mathematics of Computation*, **70**, no. 236 (Oct 2000), pg. 1719–1736.

[BB2008] David H. Bailey and Jonathan M. Borwein, "Highly Parallel, High-Precision Numerical Integration," *International Journal of Computational Science and Engineering*, to appear, April 2008, available at http://crd.lbl.gov/~dhbailey/dhbpapers/quadparallel.pdf.

[BB2010] David H. Bailey and Jonathan M. Borwein, "Experimental Computation with Oscillatory Integrals," Tewodros Amdeberhan, Luis A. Medina and Victor H. Moll, ed., *Gems in Experimental Mathematics*, Contemporary Mathematics, American Mathematical Society, **517** (2010), pg. 25–40.

[BBC2006] David H. Bailey, Jonathan M. Borwein and Richard E. Crandall, "Integrals of the Ising Class," *Journal of Physics A: Mathematical and General*, **39** (2006), pg. 12271–12302.

[BBC2007] David H. Bailey, Jonathan M. Borwein and Richard E. Crandall, "Box Integrals," *Journal of Computational and Applied Mathematics*, **206** (2007), pg. 196–208.

[BBC2007a] David H. Bailey, David Borwein, Jonathan M. Borwein and Richard Crandall, "Hypergeometric Forms for Ising-Class Integrals," *Experimental Mathematics*, **16** (2007), no. 3, pg. 257–276.

[BBC2008] D. Borwein, J. M. Borwein and R.E. Crandall, "Effective Laguerre Asymptotics," *SIAM Journal on Numerical Analysis*, **6** (2008), pg. 3285-3312.

[BBC2010] David H. Bailey, Jonathan M. Borwein and Richard E. Crandall, "Advances in the Theory of Box Integrals," *Mathematics of Computation*, **79**, no. 271 (Jul 2010), pg. 1839–1866.

[BHLT2002] David H. Bailey, Yozo Hida, Xiaoye S. Li and Brandon Thompson, "ARPREC: An Arbitrary Precision Computation Package," manuscript, Sep 2002, available at http://crd.lbl.gov/~dhbailey/dhbpapers/arprec.pdf. The software itself is available at http://crd.lbl.gov/~dhbailey/mpdist.

[BLJ2005] David H. Bailey, Xiaoye S. Li and K. Jeyabalan, "A comparison of three high-precision quadrature schemes," *Experimental Mathematics*, **14** (2005), pg. 317–329.

[BNSW2009] Jonathan M. Borwein, Dirk Nuyens, Armin Straub, and James Wan, "Random walk integrals," preprint, Oct 2009, available at http://www.carma.newcastle.edu.au/~jb616/walks.pdf.

[BSW2010] Jonathan M. Borwein, Armin Straub, and James Wan, "Three-step and four-step random walk integrals," *Experimental Mathematics*, to appear Sept 2010, available at http://www.carma.newcastle.edu.au/~jb616/walks2.pdf.

[BSWZ2010] Jonathan M. Borwein, Armin Straub, James Wan and Wadim Zudilin, "Densities of short uniform random walks," preprint, Oct 2010, available at http://www.carma.newcastle.edu.au/~jb616/densities.pdf.

[Borw1987] Jonathan M. Borwein and Peter B. Borwein, *Pi and the AGM: A Study in Analytic Number Theory and Computational Complexity*, Wiley, San Francisco, 1987.

[Broad2009] David Broadhurst, "Bessel moments, random walks and Calabi-Yau equations," preprint, Nov 2009.

[FBA1999] Helaman R. P. Ferguson, David H. Bailey and Stephen Arno, "Analysis of PSLQ, An Integer Relation Finding Algorithm," *Mathematics of Computation*, **68**, no. 225 (Jan 1999), pg. 351–369.

[LS1994] S. K. Lucas and H. A. Stone, "Evaluating infinite integrals involving Bessel functions of arbitrary order," *Journal of Computational and Applied Mathematics*, **64** (1995), pg. 217–231.

[Ooura1991] T. Ooura and M. Mori, "Double exponential formulas for oscillatory functions over the half infinite interval," *Journal of Computational and Applied Mathematics*, **38** (1991), pg. 353–360.

[Pear1905] Karl Pearson, "The random walk," *Nature*, **72** (1905), pg. 294.

[Pear1905b] Karl Pearson, "The problem of the random walk," *Nature*, **72** (1905), pg. 342.

[Pear1906] Karl Pearson, *A Mathematical Theory of Random Migration*, Mathematical Contributions to the Theory of Evolution XV, Cambridge University Press, London, 1906.

[Press1986] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, Cambridge Univ. Press, 1986.

[Sidi1982] Avram Sidi, "The numerical evaluation of very oscillatory infinite integrals by extrapolation," *Mathematics of Computation*, **38** (1982), pg. 517–529.

[Sidi1988] Avram Sidi, "A user-friendly extrapolation method for oscillatory infinite integrals," *Mathematics of Computation*, **51** (1988), pg. 249–266.

[TM1974] H. Takahasi and M. Mori, "Double exponential formulas for numerical integration," *Publications of RIMS*, Kyoto University, **9** (1974), pg. 721–741.

[Wan2010] James Wan, "Moments of products of elliptic integrals," preprint, October 2010.