# Experimental Evaluation of Euler Sums

David H. Bailey, Jonathan M. Borwein and Roland Girgensohn

**CONTENTS**



Euler expressed certain sums of the form

$$\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)(k+1)^{-n},$$

where $m$ and $n$ are positive integers, in terms of the Riemann zeta function. In [Borwein et al. 1993], Euler's results were extended to a significantly larger class of sums of this type, including sums with alternating signs.

This research was facilitated by numerical computations using an algorithm that can determine, with high confidence, whether or not a particular numerical value can be expressed as a rational linear combination of several given constants. The present paper presents the numerical techniques used in these computations and lists many of the experimental results that have been obtained.

## 1. INTRODUCTION

In response to a letter from Goldbach, Euler considered sums of the form

$$\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)(k+1)^{-n},$$

and was able to give explicit values for certain of these sums in terms of the Riemann zeta function $\zeta(t)=\sum_{k=1}^{\infty}k^{-t}$. For example, Euler found an explicit formula for the case $m=1$, $n\geq 2$. Little has been done on this problem in the intervening years (see [Berndt 1985] for some references).

In April 1993, Enrico Au-Yeung, an undergraduate at the University of Waterloo, brought to the attention of one of us the curious fact that

$$\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2}+\cdots+\frac{1}{k}\Bigr)^2 k^{-2}=4.59987\ldots$$
$$\approx\frac{17}{4}\zeta(4)=\frac{17\pi^4}{360},$$

Borwein was supported by NSERC and the Shrum Endowment at Simon Fraser University.

Girgensohn was supported by a DFG fellowship.


1058-6458/96 $0.50 per page

based on a computation to 500,000 terms. This author's reaction was to compute the value of this constant to a higher level of precision in order to dispel this conjecture. Surprisingly, a computation to 30 and later to 100 decimal digits still affirmed it. (Unknown to us at that time, de Doelder [1991] had proved a related result from which the above identity follows.)

Intrigued by this empirical result, we computed numerical values for several of these and similar sums, which we have termed Euler sums. We then analyzed these values by a technique, presented below, that permits one to determine, with a high level of confidence, whether a numerical value can be expressed as a rational linear combination of several given constants. These efforts produced even more empirical evaluations, suggesting broad patterns and general conjectures. Ultimately we found proofs for many of these experimental results.

The classes of Euler sums that we will consider are listed in Table 1. Explicit evaluations of some of the constants in these classes are presented with proofs in [Borwein and Borwein 1994] and [Borwein et al. 1994]. Table 2 contains a summary of these results, including some already known to Euler. Results for alternating sums are also given in [Borwein et al. 1994].

Variants of the sums defined in Table 1 can be evaluated by using these results. For example, for all $m \geq 1$ and $n \geq 2$ one can write

$$\begin{aligned}
\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)k^{-n}
&= \sum_{k=0}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{(k+1)^m}\Bigr)(k+1)^{-n}\\
&= \sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)(k+1)^{-n}+\sum_{k=1}^{\infty}k^{-m-n}\\
&= \sigma_h(m,n)+\zeta(m+n).
\end{aligned}$$

| | |
|---|---|
| $s_h(m,n)=\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2}+\cdots+\frac{1}{k}\Bigr)^m(k+1)^{-n}$ | for $m\geq 1$, $n\geq 2$ |
| $s_a(m,n)=\sum_{k=1}^{\infty}\Bigl(1-\frac{1}{2}+\cdots+\frac{(-1)^{k+1}}{k}\Bigr)^m(k+1)^{-n}$ | for $m\geq 1$, $n\geq 2$ |
| $a_h(m,n)=\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2}+\cdots+\frac{1}{k}\Bigr)^m(-1)^{k+1}(k+1)^{-n}$ | for $m\geq 1$, $n\geq 1$ |
| $a_a(m,n)=\sum_{k=1}^{\infty}\Bigl(1-\frac{1}{2}+\cdots+\frac{(-1)^{k+1}}{k}\Bigr)^m(-1)^{k+1}(k+1)^{-n}$ | for $m\geq 1$, $n\geq 1$ |
| $\sigma_h(m,n)=\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)(k+1)^{-n}$ | for $m\geq 1$, $n\geq 2$ |
| $\sigma_a(m,n)=\sum_{k=1}^{\infty}\Bigl(1-\frac{1}{2^m}+\cdots+\frac{(-1)^{k+1}}{k^m}\Bigr)(k+1)^{-n}$ | for $m\geq 1$, $n\geq 2$ |
| $\alpha_h(m,n)=\sum_{k=1}^{\infty}\Bigl(1+\frac{1}{2^m}+\cdots+\frac{1}{k^m}\Bigr)(-1)^{k+1}(k+1)^{-n}$ | for $m\geq 1$, $n\geq 1$ |
| $\alpha_a(m,n)=\sum_{k=1}^{\infty}\Bigl(1-\frac{1}{2^m}+\cdots+\frac{(-1)^{k+1}}{k^m}\Bigr)(-1)^{k+1}(k+1)^{-n}$ | for $m\geq 1$, $n\geq 1$ |

**TABLE 1.** Definitions of the Euler sums considered.

$$
\begin{aligned}
s_h(2,2) &= \tfrac{3}{2}\,\zeta(4) + \tfrac{1}{2}\,\zeta^2(2) = \tfrac{11}{360}\pi^4\\
s_h(2,4) &= \tfrac{2}{3}\,\zeta(6) - \tfrac{1}{3}\,\zeta(2)\,\zeta(4) + \tfrac{1}{3}\,\zeta^3(2) - \zeta^2(3) = \tfrac{37}{22680}\pi^6 - \zeta^2(3)\\
\sigma_h(2,2) &= \tfrac{1}{2}\,\zeta^2(2) - \tfrac{1}{2}\,\zeta(4) = \tfrac{1}{120}\pi^4\\
\sigma_h(2,4) &= -6\,\zeta(6) + \tfrac{8}{3}\,\zeta(2)\,\zeta(4) + \zeta^2(3) = \zeta^2(3) - \tfrac{4}{2835}\pi^6
\end{aligned}
$$

$$s_h(1,n) = \sigma_h(1,n) = \tfrac{1}{2}n\,\zeta(n+1) - \tfrac{1}{2}\sum_{k=1}^{n-2}\zeta(n-k)\,\zeta(k+1)$$

$$
\begin{aligned}
s_h(2,n) = {}& \tfrac{1}{3}n(n+1)\,\zeta(n+2) + \zeta(2)\,\zeta(n) - \tfrac{1}{2}n\sum_{k=0}^{n-2}\zeta(n-k)\,\zeta(k+2)\\
&+ \tfrac{1}{3}\sum_{k=2}^{n-2}\zeta(n-k)\sum_{j=1}^{k-1}\zeta(j+1)\,\zeta(k+1-j) + \sigma_h(2,n)
\end{aligned}
$$

$$\sigma_h(2,\,2n-1) = -\tfrac{1}{2}2n^2 + n + 1\,\zeta(2n+1) + \zeta(2)\,\zeta(2n-1) + \sum_{k=1}^{n-1}2k\,\zeta(2k+1)\,\zeta(2n-2k)$$

$$
\begin{aligned}
s_h(2,\,2n-1) = {}& \tfrac{1}{6}(2n^2-7n-3)\,\zeta(2n+1) + \zeta(2)\,\zeta(2n-1) - \tfrac{1}{2}\sum_{k=1}^{n-2}(2k-1)\,\zeta(2n-1-2k)\,\zeta(2k+2)\\
&+ \tfrac{1}{3}\sum_{k=1}^{n-2}\zeta(2k+1)\sum_{j=1}^{n-2-k}\zeta(2j+1)\,\zeta(2n-1-2k-2j)
\end{aligned}
$$

for $m$ odd, $n$ even:

$$\sigma_h(m,n) = \tfrac{1}{2}\left(\tbinom{m+n}{m} - 1\right)\zeta(m+n) + \zeta(m)\,\zeta(n) - \sum_{j=1}^{m+n}\left(\tbinom{2j-2}{m-1} + \tbinom{2j-2}{n-1}\right)\zeta(2j-1)\,\zeta(m+n-2j+1)$$

for $m$ even, $n$ odd:

$$\sigma_h(m,n) = -\tfrac{1}{2}\left(\tbinom{m+n}{m} + 1\right)\zeta(m+n) + \sum_{j=1}^{m+n}\left(\tbinom{2j-2}{m-1} + \tbinom{2j-2}{n-1}\right)\zeta(2j-1)\,\zeta(m+n-2j+1)$$

**TABLE 2.** Some explicit evaluations of Euler sums

Similarly, let $h_k = \sum_{j=1}^{k} j^{-1}$ and $h_0 = 0$. Then, for all $n \geq 2$,

$$
\begin{aligned}
&\sum_{k=1}^{\infty}\Bigl(1 + \frac{1}{2} + \cdots + \frac{1}{k}\Bigr)^2 k^{-n}\\
&\qquad = \sum_{k=0}^{\infty} h_{k+1}^2 (k+1)^{-n}\\
&\qquad = \sum_{k=0}^{\infty}\Bigl(h_k + \frac{1}{k+1}\Bigr)^2 (k+1)^{-n}\\
&\qquad = \sum_{k=0}^{\infty} h_k^2 (k+1)^{-n} + 2\sum_{k=0}^{\infty} h_k (k+1)^{-n-1}\\
&\qquad\qquad + \sum_{k=0}^{\infty}(k+1)^{-n-2}\\
&\qquad = s_h(2,n) + 2s_h(1,n+1) + \zeta(n+2).
\end{aligned}
$$

## 2. NUMERICAL TECHNIQUES

It is not easy to naïvely compute numerical values of any of these Euler sums to high precision. Straightforward evaluation using the defining formulas, to some upper limit feasible on present-day computers, yields only about eight digits accuracy. Because the integer relation detection algorithm described in Section 4 requires much higher precision to obtain reliable results, more advanced techniques must be employed.

We present here a reasonably straightforward method that is generally applicable to all Euler sums discussed in this paper. It involves the compound application of the Euler–Maclaurin summation formula [Abramowitz and Stegun 1972, p. 806; Atkinson 1989, p. 289; Knuth 1973, p. 108], which can be stated as follows. Suppose $f(t)$ has at least $2p+2$ continuous derivatives on $(a,b)$. Let $D$ be the differentiation operator, let $B_k$ denote the $k$-th Bernoulli number, and let $B_k(\cdot)$ denote the $k$-th Bernoulli polynomial. Then

$$\sum_{j=a}^{b} f(j) = \int_a^b f(t)\,dt + \tfrac{1}{2}(f(a)+f(b)) + \sum_{j=1}^{p} \frac{B_{2j}}{(2j)!}(D^{2j-1}f(b) - D^{2j-1}f(a)) + R_p(a,b), \tag{2.1}$$

where the remainder $R_p(a,b)$ is given [Atkinson 1989, p. 289] by

$$R_p(a,b) = \frac{-1}{(2p+2)!}\int_a^b B_{2p+2}(t-[t])D^{2p+2}f(t)\,dt.$$

We will start by demonstrating the method in the computation of $s_h(m,n)$. Let $h(k) = \sum_{j=1}^{k} j^{-1}$ and $f(t) = t^{-1}$. Note that $\lim_{k\to\infty}(h(k) - \ln k)$ is Euler's constant $\gamma$. By (2.1),

$$h(k) = \ln k + \tfrac{1}{2} + \tfrac{1}{2}k^{-1} + \sum_{j=1}^{p} \frac{B_{2j}}{2jk^{2j}} - \sum_{j=1}^{p} \frac{B_{2j}}{2j} + R_p(1,k).$$

Since $|B_{2k}(t)| \le |B_{2k}|$ for all $k$ and for $|t| \le 1$ [Abramowitz and Stegun 1972, p. 805], the remainder $R_p(1,k)$ has a well-defined limit $R_p(1,\infty)$ as $k$ approaches infinity. We therefore have $\gamma = \frac{1}{2} - \sum_{j=1}^{p} \frac{B_{2j}}{2j} + R_p(1,\infty)$, which gives

$$h(k) = \gamma + \ln k + \frac{1}{2k} + \sum_{j=1}^{p} \frac{B_{2j}}{2jk^{2j}} - R_p(k,\infty).$$

Now

$$|R_p(k,\infty)| = \left|\int_k^\infty B_{2p+2}(t-[t])t^{-2p-3}\,dt\right| \le \frac{|B_{2p+2}|}{(2p+2)k^{2p+2}},$$

so that the remainder in the expression for $h(k)$ is no greater than the first term omitted in the summation. We can then write, for example,

$$h(k) = \gamma + \ln k + \frac{1}{2k} - \frac{1}{12k^2} + \frac{1}{120k^4} - \frac{1}{252k^6} + \frac{1}{240k^8} - \frac{1}{132k^{10}} + \frac{691}{32760k^{12}} - \frac{1}{12k^{14}} + \frac{3617}{8160k^{16}} + O(k^{-18}).$$

We will use $\bar{h}(k)$ to denote this particular approximation (without the error term). It is an unfortunate fact that $\bar{h}$ cannot be extended to a valid infinite series. The difficulty is that for any fixed $k$, the Bernoulli coefficients eventually become very large and the series diverges. On the other hand, it is clear that for any fixed number of terms, approximations such as $\bar{h}$ become ever more accurate as $k$ increases to infinity.

Now consider

$$s_h(m,n) = \sum_{k=1}^{\infty}\Big(1+\frac{1}{2}+\cdots+\frac{1}{k}\Big)^m (k+1)^{-n}.$$

Let $c$ be a large integer, and let

$$g(t) = \bar{h}^m(t)(t+1)^{-n}.$$

Applying (2.1) again, we can write

$$\begin{aligned} s_h(m,n) &= \sum_{k=1}^{c}\Big(1+\frac{1}{2}+\cdots+\frac{1}{k}\Big)^m (k+1)^{-n} + \sum_{k=c+1}^{\infty}\Big(1+\frac{1}{2}+\cdots+\frac{1}{k}\Big)^m (k+1)^{-n} \\ &= \sum_{k=1}^{c} h^m(k)(k+1)^{-n} + \int_{c+1}^{\infty} g(t)\,dt + \tfrac{1}{2}g(c+1) - \sum_{k=1}^{9}\frac{B_{2k}}{(2k)!}D^{2k-1}g(c+1) + O(c^{-18}). \end{aligned} \tag{2.2}$$

This formula suggests the following computational scheme. First evaluate explicitly the sum

$$\sum_{k=1}^{c} h^m(k)(k+1)^{-n}$$

for $c = 10^8$, using a numeric working precision of 150 digits. Then perform the symbolic integration and differentiation steps indicated in (2.2). Finally, evaluate the resulting expression, again using a working precision of 150 digits. The final result should be equal to $s_h(m,n)$ to approximately 135 significant digits.

The difficulty and cost of performing the requisite symbolic integration and differentiation operations can be greatly reduced by approximating $g(t)$ as follows: expand $\bar{h}^m(t)$, the numerator of $g(t)$, into a sum of individual terms; write $(1+t)^{-n}$ as $t^{-n}(1+t^{-1})^{-n}$; expand $(1+t^{-1})^{-n}$ using the binomial theorem to 18 terms; multiply together the resulting numerator and denominator expressions; and finally omit all terms in which the exponent of $t^{-1}$ is greater than 18. The result is a linear sum of terms of the form $t^{-p}\ln^q(t)$ for modest-sized integers $p$ and $q$.

Of course, even more accurate results can be obtained by utilizing more terms in the Euler–Maclaurin expansions, although the cost of the required symbolic manipulation correspondingly increases. The determination of the optimal balance between numeric and symbolic calculations, and of the number of Euler–Maclaurin terms (at both steps) required for various levels of precision, is an interesting problem in its own right. However, we found that only minor tuning of the above scheme, based on simple timing and accuracy experiments, sufficed for the cases we studied.

For alternating Euler sums, the scheme is a bit more complicated. Euler–Maclaurin summation works badly for alternating series, because oscillations lead to large high-order derivatives. The solution is to effectively split the summation into two, one positive and one negative. Consider, for instance, the sum $s_a(m,n)$. We can write

$$\begin{aligned} s_a(m,n) &= \sum_{k=1}^{\infty}\Big(1-\frac{1}{2}+\cdots+\frac{(-1)^{k+1}}{k}\Big)^m (k+1)^{-n} \\ &= \sum_{k=1}^{\infty}\Big(\sum_{j=1}^{k}\frac{1}{2j-1}-\sum_{j=1}^{k-1}\frac{1}{2j}\Big)^m \frac{1}{(2k)^n} \\ &\quad + \sum_{k=1}^{\infty}\Big(\sum_{j=1}^{k}\frac{1}{2j-1}-\sum_{j=1}^{k}\frac{1}{2j}\Big)^m \frac{1}{(2k+1)^n} \\ &= \sum_{k=1}^{\infty}\Big(r_k+\frac{1}{2k}\Big)^m\frac{1}{(2k)^n}+\sum_{k=1}^{\infty}\frac{r_k^m}{(2k+1)^n}, \end{aligned}$$

where

$$r_k = \sum_{j=1}^{k}\frac{1}{2j-1}-\sum_{j=1}^{k}\frac{1}{2j} = \sum_{j=1}^{k}\frac{1}{2j(2j-1)}.$$

The Euler–Maclaurin formula (2.1) can then be applied first to obtain a highly accurate approximation to $r_k$, and then to evaluate the two remaining outer summations.

Another approach for alternating Euler sums is to apply the Boole summation formula [Borwein et al. 1989], which deals specifically with alternating sums.

## 3. EXPERIMENTAL SETUP AND OPTIMIZATIONS

We have performed many computations of the type described. The integrations and differentiations required in (2.2) can be handled by a symbolic mathematics package such as Maple [Char et al. 1991] or Mathematica [Wolfram 1991]. The explicit summation of the first $c$ terms, as indicated in (2.2), could be performed by utilizing the multiple precision facility in the Maple or Mathematica packages. However, we found that the MPFUN multiple precision package and translator developed by one of us [Bailey] was significantly faster for this purpose.

Whatever software is used, this explicit summation is a very expensive operation. For example, the evaluation of $s_h(3,4)$ to $10^8$ terms, using MPFUN with 150-digit precision arithmetic, requires twenty hours on a Crimson Silicon Graphics workstation. Such runs can be made, but clearly this is pressing the capabilities of current workstations.

MPFUN is available on vector supercomputers such as those manufactured by Cray Research, Inc. However, for the modest precision levels typical of these problems (100 to 200 decimal digits), the resulting vector lengths are too short to yield the high performance these systems are capable of. On the other hand, multiple precision calculations of this type are well suited for RISC processors, because they are well behaved in cache memory systems.

These considerations suggest that, in principle, a highly parallel computer based on RISC processors could be effectively employed for computing these explicit sums. However, at first glance these computations appear not to possess any significant opportunity for parallelism, since evidently both the inner and outer sums must be simultaneously accumulated.

Fortunately, there are algorithms that efficiently exploit parallelism. The basic idea is simple: suppose, say, that we wish to compute $s_h(m,n)$ using (2.2), and thus need to sum $h^m(k)(k+1)^{-n}$ up to $k=c$. If the parallel system has $P$ processor nodes, and $c=JP$, each processor stores a length-$J$ run of values of $k^{-1}$, then adds these values together and combines the sum with the accumulated sum from the preceding processors. More precisely:

**Algorithm 1 (Parallel Summation).** Suppose given $P$ parallel processor nodes, each with an associated array $r_p$ of length $J+1$ and storage cells $h_p$ and $s_p$, for $1 \le p \le P$. All floating-point storage and arithmetic is assumed to be done in multiple precision. We compute

$$S = \sum_{k=1}^{JP} h^m(k)(k+1)^{-n}$$

as follows:

- Initialize by setting $H \leftarrow 0$ and $S \leftarrow 0$.
- For $p \leftarrow 1$ to $P$, in parallel:
  - Set $h_p \leftarrow 0$ and $s_p \leftarrow 0$.
  - For $j \leftarrow 1$ to $J$, set $r_{p,j} \leftarrow 1/k$ (where $k = j+(p-1)J$), and $h_p \leftarrow h_p + r_{p,j}$.
  - Set $r_{p,J+1} \leftarrow 1/(1+pJ)$.
- For $p \leftarrow 1$ to $P$, sequentially:
  - Set $H' \leftarrow H$, $H \leftarrow H + h_p$, and $h_p \leftarrow H'$.
- For $p \leftarrow 1$ to $P$, in parallel:
  - For $j \leftarrow 1$ to $J$:
    - Set $h_p \leftarrow h_p + r_{p,j}$ and $s_p \leftarrow s_p + h_p^m r_{p,j+1}^n$.
- For $p \leftarrow 1$ to $P$, sequentially:
  - Set $S \leftarrow S + s_p$.

Note that, as stated, the algorithm maintains all the $c = JP$ inverses at once in the array $r$. In a typical computation one may have $c = 10^8$, and the required amount of memory per node may not be available on some highly parallel computers. This difficulty can be remedied by setting up an outer loop: we write $c = IJP$ and run the algorithm successively for each block $k \in [(i-1)JP+1,\, iJP]$, where $1 \le i \le I$. (Naturally, $H$ and $S$ are maintained from one iteration to the next instead of being initialized to 0, and the value assigned to $r_{p,j}$ is $1/k$ where

$$k = j + (p-1)J + (i-1)JP.)$$

Upon completion of $I$ iterations, $S$ is the desired overall sum.

This algorithm has been implemented on an Intel Paragon parallel computer at NASA Ames Research Center, using MPFUN. For $m=3$ and $n=4$ in 150-digit precision arithmetic, with $P=128$, $J=2^{13}$ and $I=100$ (so that $c=104,857,600$), the program took only 971 seconds. This is 110 times faster than the time required by a straightforward serial algorithm that exercises only one node of the Paragon, and 40 times faster than the same algorithm running on one processor of a Cray YMP, using MPFUN tuned for the Cray. It may be possible, by reorganizing the computation, to achieve higher performance on the Cray; thus caution should be exercised when interpreting this last figure. But these results nonetheless confirm that Algorithm 1, running on a highly parallel RISC supercomputer, is a highly efficient and cost-effective solution to the problem of computing the explicit sums required in (2.2).

## 4. INTEGER RELATION DETECTION ALGORITHMS

Let $x = (x_1, x_2, \ldots, x_n)$ be a vector of real numbers. We say that $x$ possesses an integer relation if there exist integers $a_i$ not all zero such that

$$a_1x_1 + a_2x_2 + \cdots + a_nx_n = 0.$$

By an integer relation algorithm we mean an algorithm that is guaranteed (provided the computer implementation has sufficient numeric precision) to recover the vector of integers $a_i$, if it exists, or to produce bounds within which no integer relation can exist.

The problem of finding integer relations among a set of real numbers was first studied by Euclid, who gave an iterative algorithm which, when applied to two real numbers, either terminates, yielding an exact relation, or produces an infinite sequence of approximate relations. The generalization of this problem for $n > 2$ has been attempted by Euler, Jacobi, Poincaré, Minkowski, Perron, Brun, and Bernstein, among others. However, none of their algorithms has been proved to work for $n > 3$, and numerous counterexamples have been found.

The first integer relation algorithm with the desired properties mentioned above was discovered by Ferguson and Forcade [1979]. In the intervening years a number of other integer relation algorithms have been discovered, including a variant of the original algorithm [Ferguson 1987], the LLL algorithm [Lenstra et al. 1982], the HJLS algorithm [Hastad et al. 1988], which is based on LLL, and the PSOS algorithm [Bailey and Ferguson 1989].

Recently a new algorithm, known as the PSLQ algorithm, was developed by Ferguson and one of us [Ferguson and Bailey 1991]. It appears to combine some of the best features separately possessed by previous algorithms, including fast run times, numerical stability, numerical efficiency (that is, it successfully recovers a relation when the input is known to only limited precision), and a guaranteed completion in a polynomially bounded number of iterations. We present here a simplified but equivalent version of PSLQ. The proof of the algorithm and notes for efficient implementations are given in [Ferguson and Bailey 1991].

**Algorithm 2 (PSLQ).** Let $x$ be the the input real vector of length $n$, and let nint denote the nearest integer function (for exact half-integer values, define nint to be the integer with greater absolute value). Let $\gamma = \sqrt{4/3}$. Let $A$ and $B$ be $n \times n$ matrices, and $H$ an $n \times (n-1)$ matrix.

**Initialization**

- Set $A$ and $B$ to the identity.
- For $k \leftarrow 1$ to $n$, compute $s_k \leftarrow \sqrt{\sum_{j=k}^{n} x_j^2}$.
- For $k \leftarrow 1$ to $n$, set $y_k \leftarrow x_k/s_1$ and $s_k \leftarrow s_k/s_1$.
- (Initialize $H$) For $i \leftarrow 1$ to $n$:
  - for $j \leftarrow i+1$ to $n-1$, set $H_{ij} \leftarrow 0$;
  - if $i \le n-1$, set $H_{ii} \leftarrow s_{i+1}/s_i$;
  - for $j \leftarrow 1$ to $i-1$, set $H_{ij} \leftarrow -y_iy_j/(s_js_{j+1})$.
- (Perform full reduction on $H$ while updating $y$, $A$, $B$) For $i \leftarrow 2$ to $n$:
  - for $j \leftarrow i-1$ to 1 by $-1$:
    - set $t \leftarrow \mathrm{nint}(H_{ij}/H_{jj})$ and $y_j \leftarrow y_j + ty_i$;
    - for $k \leftarrow 1$ to $j$, set $H_{ik} \leftarrow H_{ik} - tH_{jk}$;
    - for $k \leftarrow 1$ to $n$, set $A_{ik} \leftarrow A_{ik} - tA_{jk}$ and $B_{kj} \leftarrow B_{kj} + tB_{ki}$.

**Main loop**

- Repeat until precision is exhausted or a relation has been detected (see termination test at the end):
  - Select $m$ such that $\gamma^i|H_{ii}|$ is maximal when $i = m$.
  - (Perform block reduction on $H$ while updating $y$, $A$, $B$) For $i \leftarrow m+1$ to $n$:
    - for $j \leftarrow \min(i-1, m+1)$ to 1 by $-1$:
      - set $t \leftarrow \mathrm{nint}(H_{ij}/H_{jj})$ and $y_j \leftarrow y_j + ty_i$;
      - for $k \leftarrow 1$ to $j$, set $H_{ik} \leftarrow H_{ik} - tH_{jk}$;
      - for $k \leftarrow 1$ to $n$, set $A_{ik} \leftarrow A_{ik} - tA_{jk}$ and $B_{kj} \leftarrow B_{kj} + tB_{ki}$.
  - Exchange entries $m$ and $m+1$ of $y$, corresponding rows of $A$ and $H$, and corresponding columns of $B$.
  - If $m \le n-2$, update $H$ as follows:

- Set $t_0 \leftarrow \sqrt{H_{mm}^2 + H_{m,m+1}^2}$, $t_1 \leftarrow H_{mm}/t_0$ and $t_2 \leftarrow H_{m,m+1}/t_0$.
- For $i \leftarrow m$ to $n$, set $t_3 \leftarrow H_{im}$, $t_4 \leftarrow H_{i,m+1}$, $H_{im} \leftarrow t_1 t_3 + t_2 t_4$, $H_{i,m+1} \leftarrow -t_2 t_3 + t_1 t_4$.

- (Norm bound) Set $M \leftarrow 1/\max_j |H_j|$, where $H_j$ denotes the $j$-th row of $H$. There can exist no relation vector whose Euclidean norm is less than $M$.
- (Termination test) If the largest entry of $A$ exceeds the level of numeric precision used, then precision is exhausted. If the smallest entry of the $y$ vector is less than the detection threshold, a relation has been detected and is given in the corresponding column of $B$.

Regarding the termination test, it sometimes happens that a relation is missed at the point of potential detection because the $y$ entry is not quite as small as the detection threshold being used (the threshold is typically set to the "epsilon" of the precision level). When this happens, however, one will note that the ratio of the smallest and largest $y$ vector entries is suddenly very small, provided sufficient numeric precision is being used.

The actual probability distribution of this ratio is not known for the PSLQ algorithm. Most likely, however, the probability of this ratio being less than $\varepsilon$ is closely approximated by a modest-sized constant times $\varepsilon$. This is because the entries of the vector $y$ are related to the iterates of the continued fraction algorithm, which are distributed according to the Kuzmin distribution [Knuth 1981, p. 346]. In a normal run of PSLQ, prior to the detection of a relation, this ratio is seldom smaller than 0.01. Thus if this ratio suddenly decreases to a very small value, such as $10^{-20}$, it is almost certain that a relation has been detected—one need only adjust the detection threshold for the algorithm to terminate properly and output the relation. When detection does occur, this ratio may be thought of as a "confidence level" of the detection.

In practice, the PSLQ algorithm is very effective in finding relations. For example, in tests described in [Ferguson and Bailey 1991], relations were detected for input vectors in dimensions up to $n = 82$, the coefficients of the relation having size up to $10^{14}$. As a general rule, one can expect to detect a relation with coefficients of size $10^m$ for an input vector of dimension $n$ if the input is known to somewhat greater than $mn$ digit precision, and provided that computations are performed using at least this level of numeric precision.

## 5. APPLICATIONS OF THE PSLQ ALGORITHM

Integer relation detection algorithms have a number of applications in computational mathematics. One is to find out whether or not a given constant $\alpha$, whose value can be computed to high precision, is algebraic of some degree $n$ or less. This is done by computing the vector $x = (1, \alpha, \alpha^2, \ldots, \alpha^n)$ to high precision and then applying an integer relation algorithm to the vector $x$. If a relation is found, this integer vector is precisely the set of coefficients of a polynomial satisfied by $\alpha$. When a relation is not found, the resulting bound means that $\alpha$ cannot possibly be the root of a polynomial of degree $n$, with coefficients of size less than the established bound. Even negative results of this sort are often of interest.

One of us has performed several computations of this type [Bailey and Ferguson 1989], establishing, for example, that if Euler's constant $\gamma$ satisfies an integer polynomial of degree 50 or less, the Euclidean norm of the coefficients must exceed $7 \times 10^{17}$. Computations of this sort have also been applied to study a certain conjecture regarding the Riemann zeta function. It is well known [Borwein and Borwein 1987] that $\zeta(2)$, $\zeta(3)$ and $\zeta(4)$ equal, respectively,

$$3\sum_{k=1}^{\infty} \frac{1}{k^2\binom{2k}{k}}, \quad \frac{5}{2}\sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k^3\binom{2k}{k}}, \quad \frac{36}{17}\sum_{k=1}^{\infty} \frac{1}{k^4\binom{2k}{k}}.$$

This has led some to suggest that

$$\zeta(5) = Z_5 \sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k^5\binom{2k}{k}}$$

for $Z_5$ a simple rational or algebraic number. Unfortunately, integer relation calculations [Bailey] have established that if $Z_5$ satisfies a polynomial of degree 25 or less, the Euclidean norm of the coefficients must exceed $2 \times 10^{37}$.

The present application of Euler sum constants is well suited to analysis with integer relation algorithms. We will present but one example of these computations. Consider

$$s_a(2,3) = 0.156166933381176915881035909687\ldots$$

(see Table 1 for the definition). Based on experience with other constants, we conjectured that this constant satisfies a relation involving homogeneous combinations of $\zeta(2)$, $\zeta(3)$, $\zeta(4)$, $\zeta(5)$, $\ln(2)$, $\mathrm{Li}_4(\frac{1}{2})$ and $\mathrm{Li}_5(\frac{1}{2})$, where $\mathrm{Li}_n(x) = \sum_{k=1}^{\infty} x^k k^{-n}$ denotes the polylogarithm function. The numerical values of these constants, to 30 decimal digits, are:

$$\begin{aligned}
\zeta(2) &= 1.644934066848226436472415166646\ldots\\
\zeta(3) &= 1.202056903159594285399738161511\ldots\\
\zeta(4) &= 1.082323233711138191516003696541\ldots\\
\zeta(5) &= 1.036927755143369926331365486457\ldots\\
\ln(2) &= 0.693147180559945309417232121458\ldots\\
\mathrm{Li}_4(\tfrac{1}{2}) &= 0.517479061673899386330758161898\ldots\\
\mathrm{Li}_5(\tfrac{1}{2}) &= 0.508400579242268707459108849258\ldots
\end{aligned}$$

The terms involving these constants with degree five (see Section 7 for the definition of this term) are the following: $\mathrm{Li}_5(\frac{1}{2})$, $\mathrm{Li}_4(\frac{1}{2})\ln(2)$, $\ln^5(2)$, $\zeta(5)$, $\zeta(4)\ln(2)$, $\zeta(3)\ln^2(2)$, $\zeta(2)\ln^3(2)$, $\zeta(2)\zeta(3)$. When $s_a(2,3)$ is augmented with this set of terms, all computed to 135 decimal digits accuracy, and the resulting vector of length 9 is input to the PSLQ algorithm, the relation $(480, -1920, 0, 16, 255, 660, -840, -160, 360)$ is detected at iteration 390. Solving this relation for $s_a(2,3)$, we obtain the formula

$$\begin{aligned}
s_a(2,3) &= 4\,\mathrm{Li}_5(\tfrac{1}{2}) - \tfrac{1}{30}\ln^5(2) - \tfrac{17}{32}\zeta(5) - \tfrac{11}{8}\zeta(4)\ln(2)\\
&\quad + \tfrac{7}{4}\zeta(3)\ln^2(2) + \tfrac{1}{3}\zeta(2)\ln^3(2) - \tfrac{3}{4}\zeta(2)\zeta(3)\\
&= 4\,\mathrm{Li}_5(\tfrac{1}{2}) - \tfrac{1}{30}\ln^5(2) - \tfrac{17}{32}\zeta(5) + \tfrac{11}{720}\pi^4\ln(2)\\
&\quad + \tfrac{7}{4}\zeta(3)\ln^2(2) + \tfrac{1}{18}\pi^2\ln^3(2) - \tfrac{3}{24}\pi^2\zeta(3)
\end{aligned}$$

(recall that $\zeta(2n) = (2\pi)^{2n}|B_{2n}|/(2(2n)!)$ for $n$ integer).

When the relation is detected, the minimum and maximum $y$ vector entries are $1.60 \times 10^{-134}$ and $5.98 \times 10^{-29}$, whose ratio is of the order of $10^{-105}$. Thus the confidence level of this detection is very high.

Although we were using 135-digit input values and 150-digit working precision when we first detected this relation, the fact that the maximum $y$-vector entry is only $10^{-29}$ at detection implies that such high levels of numeric precision are not required in this case. Indeed, the relation can be successfully detected by the PSLQ algorithm using only 50-digit input values and 50-digit working precision.

## 6. EXPERIMENTAL RESULTS

Many special cases of the proven results listed in Table 2 were first obtained using the experimental method presented in Sections 2-4. In addition, we have obtained a number of experimental identities for which formal proofs have not yet been found. Table 3 lists some of them.

It should be emphasized that the results in Table 3 are not established in any rigorous mathematical sense by these calculations. However, in each case the "confidence level" (see Section 3) of these detections is better than $10^{-50}$, and in most cases is in the neighborhood of $10^{-100}$. Table 3, together with the results in [Borwein et al. 1994], gives all values of $s_h(m,n)$ for $m+n \le 7$ and $m+n = 9$, and all values of the alternating sums for $m+n \le 5$. Some of these identities can be proved by ad hoc methods, based on [Lewin 1981]; they are indicated with an asterisk.

In many other cases we were not able to obtain a formula for the Euler sum explicitly in terms of values of the Riemann zeta, logarithm and polylogarithm functions, but we were able to obtain relations involving two or more Euler sums of the same degree (where by "degree" we mean the sum of the indices $m$ and $n$ of the constant). Some of

$$
\begin{aligned}
{}^*s_h(3,2) &= \tfrac{15}{2}\,\zeta(5) + \zeta(2)\,\zeta(3)\\
{}^*s_h(3,3) &= -\tfrac{33}{16}\,\zeta(6) + 2\,\zeta^2(3)\\
s_h(3,4) &= \tfrac{119}{16}\,\zeta(7) - \tfrac{33}{4}\,\zeta(3)\,\zeta(4) + 2\,\zeta(2)\,\zeta(5)\\
s_h(3,6) &= \tfrac{197}{24}\,\zeta(9) - \tfrac{33}{4}\,\zeta(4)\,\zeta(5) - \tfrac{37}{8}\,\zeta(3)\,\zeta(6) + \zeta^3(3) + 3\,\zeta(2)\,\zeta(7)\\
s_h(4,2) &= \tfrac{859}{24}\,\zeta(6) + 3\,\zeta^2(3)\\
s_h(4,3) &= -\tfrac{109}{8}\,\zeta(7) + \tfrac{37}{2}\,\zeta(3)\,\zeta(4) - 5\,\zeta(2)\,\zeta(5)\\
s_h(4,5) &= -\tfrac{29}{2}\,\zeta(9) + \tfrac{37}{2}\,\zeta(4)\,\zeta(5) + \tfrac{33}{4}\,\zeta(3)\,\zeta(6) - \tfrac{8}{3}\,\zeta^3(3) - 7\,\zeta(2)\,\zeta(7)\\
s_h(5,2) &= \tfrac{1855}{16}\,\zeta(7) + 33\,\zeta(3)\,\zeta(4) + \tfrac{57}{2}\,\zeta(2)\,\zeta(5)\\
s_h(5,4) &= \tfrac{890}{9}\,\zeta(9) + 66\,\zeta(4)\,\zeta(5) - \tfrac{4295}{24}\,\zeta(3)\,\zeta(6) - 5\,\zeta^3(3) + \tfrac{265}{8}\,\zeta(2)\,\zeta(7)\\
s_h(6,3) &= -\tfrac{3073}{12}\,\zeta(9) - 243\,\zeta(4)\,\zeta(5) + \tfrac{2097}{4}\,\zeta(3)\,\zeta(6) + \tfrac{67}{3}\,\zeta^3(3) - \tfrac{651}{8}\,\zeta(2)\,\zeta(7)\\
s_h(7,2) &= \tfrac{134701}{36}\,\zeta(9) + \tfrac{15697}{8}\,\zeta(4)\,\zeta(5) + \tfrac{29555}{24}\,\zeta(3)\,\zeta(6) + 56\,\zeta^3(3) + \tfrac{3287}{4}\,\zeta(2)\,\zeta(7)
\end{aligned}
$$

$$
\begin{aligned}
{}^*s_a(2,2) &= 6\,\mathrm{Li}_4(\tfrac{1}{2}) + \tfrac{1}{4}\ln^4(2) - \tfrac{29}{8}\,\zeta(4) + \tfrac{3}{2}\,\zeta(2)\ln^2(2)\\
{}^*s_a(2,3) &= 4\,\mathrm{Li}_5(\tfrac{1}{2}) - \tfrac{1}{30}\ln^5(2) - \tfrac{17}{32}\,\zeta(5) - \tfrac{11}{8}\,\zeta(4)\ln(2) + \tfrac{7}{4}\,\zeta(3)\ln^2(2) + \tfrac{1}{3}\,\zeta(2)\ln^3(2) - \tfrac{3}{4}\,\zeta(2)\,\zeta(3)\\
{}^*s_a(3,2) &= -24\,\mathrm{Li}_5(\tfrac{1}{2}) + 6\ln(2)\,\mathrm{Li}_4(\tfrac{1}{2}) + \tfrac{9}{20}\ln^5(2) + \tfrac{659}{32}\,\zeta(5) - \tfrac{285}{16}\,\zeta(4)\ln(2) + \tfrac{5}{2}\,\zeta(2)\ln^3(2) + \tfrac{1}{2}\,\zeta(2)\,\zeta(3)
\end{aligned}
$$

$$
\begin{aligned}
{}^*a_h(2,2) &= -2\,\mathrm{Li}_4(\tfrac{1}{2}) - \tfrac{1}{12}\ln^4(2) + \tfrac{99}{48}\,\zeta(4) - \tfrac{7}{4}\,\zeta(3)\ln(2) + \tfrac{1}{2}\,\zeta(2)\ln^2(2)\\
{}^*a_h(2,3) &= -4\,\mathrm{Li}_5(\tfrac{1}{2}) - 4\ln(2)\,\mathrm{Li}_4(\tfrac{1}{2}) - \tfrac{2}{15}\ln^5(2) + \tfrac{107}{32}\,\zeta(5) - \tfrac{7}{4}\,\zeta(3)\ln^2(2) + \tfrac{2}{3}\,\zeta(2)\ln^3(2) + \tfrac{3}{8}\,\zeta(2)\,\zeta(3)\\
{}^*a_h(3,2) &= 6\,\mathrm{Li}_5(\tfrac{1}{2}) + 6\ln(2)\,\mathrm{Li}_4(\tfrac{1}{2}) + \tfrac{1}{5}\ln^5(2) - \tfrac{33}{8}\,\zeta(5) + \tfrac{21}{8}\,\zeta(3)\ln^2(2) - \zeta(2)\ln^3(2) - \tfrac{15}{16}\,\zeta(2)\,\zeta(3)
\end{aligned}
$$

$$
\begin{aligned}
{}^*a_a(2,2) &= -4\,\mathrm{Li}_4(\tfrac{1}{2}) - \tfrac{1}{6}\ln^4(2) + \tfrac{37}{16}\,\zeta(4) + \tfrac{7}{4}\,\zeta(3)\ln(2) - 2\,\zeta(2)\ln^2(2)\\
{}^*a_a(2,3) &= 4\ln(2)\,\mathrm{Li}_4(\tfrac{1}{2}) + \tfrac{1}{6}\ln^5(2) - \tfrac{79}{32}\,\zeta(5) + \tfrac{11}{8}\,\zeta(4)\ln(2) - 1\,\zeta(2)\ln^3(2) + \tfrac{3}{8}\,\zeta(2)\,\zeta(3)\\
{}^*a_a(3,2) &= 30\,\mathrm{Li}_5(\tfrac{1}{2}) - \tfrac{1}{4}\ln^5(2) - \tfrac{1813}{64}\,\zeta(5) + \tfrac{285}{16}\,\zeta(4)\ln(2) + \tfrac{21}{8}\,\zeta(3)\ln^2(2) - \tfrac{7}{2}\,\zeta(2)\ln^3(2) + \tfrac{3}{4}\,\zeta(2)\,\zeta(3)
\end{aligned}
$$

**TABLE 3.** Experimentally detected identities. Those marked with an asterisk have been formally proved.

these relations are shown in Table 4. This is not a complete list; we have obtained numerous other relations of this type. The confidence level of each of these relations is better than $10^{-25}$. These relations are nonredundant, in the sense that for each one no relation involving fewer constants can be detected (we check this by repeating the runs with each constant removed in turn).

In still other cases we were not successful in finding relations, but we were able to obtain bound results from the PSLQ program that exclude a large class of potential relations among the list of candidate terms. These results do not conclusively prove that there is no such relation, only that if one exists, the Euclidean norm of its coefficients must be larger than a certain bound. Some of these "negative" results are listed in Table 5.

One interesting byproduct of the bound results in Table 5 is that there are no modest-sized integer relations among homogeneous products of $\zeta(k)$ with degree 12 or less (see Section 7), except of course the well-known relations when all $k$ are even integers.

The bound result for $a_h(1,5)$ in Table 5 confirms the observation in [Borwein et al. 1994] that $a_h(1,n)$, which equals $\alpha_h(1,n)$, does not appear to possess an explicit evaluation when $n$ is odd and greater than three. The bound results for $\sigma_h(2,6)$ and $\sigma_h(2,8)$ confirm the observation in [Borwein et al. 1994] that $\sigma_h(2,n)$ does not appear to possess

| |
|---|
| $84549 s_h(1,7) + 211468 s_h(2,6) + 148902 s_h(3,5) - 13360 s_h(4,4) - 1978 s_h(5,3)$ |
| $-127\,\zeta(8) + 336\,\zeta(3)\,\zeta(5) - 120\,\zeta(2)\,\zeta^2(3) - 24 s_h(2,6) - 96 s_h(3,5)$ |
| $-2718587 s_h(1,8) - 164525664 s_h(2,7) - 178042944 s_h(3,6) - 88947862 s_h(4,5) + 3863940 s_h(5,4) + 672100 s_h(6,3)$ |
| $-5138 s_h(1,8) - 566656 s_h(2,7) - 624016 s_h(3,6) - 316988 s_h(4,5) + 6480 s_h(5,4) + 33605\,\zeta(3)\,\zeta(6)$ |
| $-14269408 s_h(1,9) + 2578470 s_h(2,8) + 2815376 s_h(3,7) + 5814550 s_h(4,6) + 6238884 s_h(5,5) + 3938912 s_h(6,4) + 1122784 s_h(7,3) - 1860 s_h(8,2) + 63164285\,\zeta(10)$ |
| $321\,\zeta(10) - 440\,\zeta^2(5) - 720\,\zeta(3)\,\zeta(7) - 80\,\zeta^2(3)\,\zeta(4) + 560\,\zeta(2)\,\zeta(3)\,\zeta(5) - 40 s_h(2,8) + 160 s_h(3,7)$ |
| $-1691755503 s_h(1,10) - 3172589688 s_h(2,9) + 837511504 s_h(3,8) - 7302717576 s_h(4,7) - 13958660016 s_h(5,6) - 12910466064 s_h(6,5) - 7099332912 s_h(7,4) - 1773212688 s_h(8,3) + 658360 s_h(9,2) + 53491434679\,\zeta(11) - 21868248971\,\zeta(2)\,\zeta(9)$ |
| $-589\,\zeta(11) + 322\,\zeta(5)\,\zeta(6) + 756\,\zeta(4)\,\zeta(7) + 254\,\zeta(3)\,\zeta(8) - 336\,\zeta^2(3)\,\zeta(5) - 368\,\zeta(2)\,\zeta(9) + 80\,\zeta(2)\,\zeta^3(3) - 16 s_h(3,8) - 48 s_h(4,7)$ |
| $70663\,\zeta(12) - 165840\,\zeta(5)\,\zeta(7) - 121616\,\zeta(3)\,\zeta(9) - 33168\,\zeta^2(3)\,\zeta(6) + 5528\,\zeta^4(3) + 49752\,\zeta(2)\,\zeta^2(5) + 99504\,\zeta(2)\,\zeta(3)\,\zeta(7) - 16584 s_h(2,10) + 22112 s_h(3,9)$ |
| $1152 s_a(2,4) + 640 s_a(3,3) - 7680 \ln(2)\,\mathrm{Li}_5(1/2) + 64 \ln^6(2) - 1881\,\zeta(6) + 7440\,\zeta(5)\ln(2) - 1680\,\zeta(4)\ln^2(2) - 1120\,\zeta(3)\ln^3(2) + 864\,\zeta(3)\,\zeta(3) - 640\,\zeta(2)\ln^4(2) - 432\,\zeta(2)\,\zeta(3)\ln(2)$ |

**TABLE 4.** Expressions that have been experimentally found to vanish.

an explicit evaluation for $n$ even and greater than four.

The numerical values of the various Euler sum constants, which were used to obtain the results listed in Tables 3–5, were computed as described in Sections 2 and 3. The explicit sum in formula (2.2) was computed using Algorithm 1 (or its equivalent for alternating sums), on the Intel Paragon parallel computer system, with $P = 128$, $J = 2^{13}$, $I = 100$. The symbolic operations indicated in (2.2) were performed in Maple. The final numerical values were checked by comparing them with the values obtained from Algorithm 1 (or equivalent) with $I = 99$ instead of 100.

## 7. CONJECTURES

It is not known whether closed-form evaluations of the type listed in Table 3 exist for all of the various classes of Euler sums studied in this paper. It is possible that such formulas always exist and could be uncovered by the techniques described in this paper, if one could only deduce the form of the missing terms. We present this as an open question for further research.

One principle we have observed in this work is that in every case where we have obtained a relation, this relation has always involved homogeneous terms, in the sense that the degree $m + n$ of each term involved in the relation is the same. (For these purposes, the degree of $\zeta(k)$ is taken to be $k$, as is the degree of $\mathrm{Li}_k(\frac{1}{2})$, while that of $\ln(2)$ is taken as one.)

Although we believe this principle may hold in general, we have no idea how to prove it. We therefore present it as a conjecture. However, since it is important to limit the number of constants input to the PSLQ algorithm in order to enhance the possibility of detecting a relation, we have often used this principle in selecting the candidate constants.

| Set of constants that do not appear to be linked by a relation | Bound |
|---|---|
| $s_h(2,6)$, $\zeta(8)$, $\zeta(3)\,\zeta(5)$, $\zeta(2)\,\zeta^2(3)$ | $7.06 \times 10^{41}$ |
| $s_h(3,5)$, $\zeta(8)$, $\zeta(3)\,\zeta(5)$, $\zeta(2)\,\zeta^2(3)$ | $4.31 \times 10^{41}$ |
| $s_h(4,4)$, $\zeta(8)$, $\zeta(3)\,\zeta(5)$, $\zeta(2)\,\zeta^2(3)$ | $1.63 \times 10^{40}$ |
| $s_h(2,8)$, $\zeta(10)$, $\zeta^2(5)$, $\zeta(3)\,\zeta(7)$, $\zeta^2(3)\,\zeta(4)$, $\zeta(2)\,\zeta(3)\,\zeta(5)$ | $1.28 \times 10^{26}$ |
| $s_h(3,7)$, $\zeta(10)$, $\zeta^2(5)$, $\zeta(3)\,\zeta(7)$, $\zeta^2(3)\,\zeta(4)$, $\zeta(2)\,\zeta(3)\,\zeta(5)$ | $3.03 \times 10^{26}$ |
| $s_h(4,6)$, $\zeta(10)$, $\zeta^2(5)$, $\zeta(3)\,\zeta(7)$, $\zeta^2(3)\,\zeta(4)$, $\zeta(2)\,\zeta(3)\,\zeta(5)$ | $3.33 \times 10^{24}$ |
| $s_h(3,8)$, $\zeta(11)$, $\zeta(5)\,\zeta(6)$, $\zeta(4)\,\zeta(7)$, $\zeta(3)\,\zeta(8)$, $\zeta^2(3)\,\zeta(5)$, $\zeta(2)\,\zeta(9)$, $\zeta(2)\,\zeta^3(3)$ | $2.01 \times 10^{17}$ |
| $s_h(4,7)$, $\zeta(11)$, $\zeta(5)\,\zeta(6)$, $\zeta(4)\,\zeta(7)$, $\zeta(3)\,\zeta(8)$, $\zeta^2(3)\,\zeta(5)$, $\zeta(2)\,\zeta(9)$, $\zeta(2)\,\zeta^3(3)$ | $1.89 \times 10^{17}$ |
| $s_h(2,10)$, $\zeta(12)$, $\zeta(5)\,\zeta(7)$, $\zeta(3)\,\zeta(9)$, $\zeta(3)\,\zeta(4)\,\zeta(5)$, $\zeta^2(3)\,\zeta(6)$, $\zeta^4(3)$, $\zeta(2)\,\zeta^2(5)$, $\zeta(2)\,\zeta(3)\,\zeta(7)$ | $1.43 \times 10^{15}$ |
| $s_h(3,9)$, $\zeta(12)$, $\zeta(5)\,\zeta(7)$, $\zeta(3)\,\zeta(9)$, $\zeta(3)\,\zeta(4)\,\zeta(5)$, $\zeta^2(3)\,\zeta(6)$, $\zeta^4(3)$, $\zeta(2)\,\zeta^2(5)$, $\zeta(2)\,\zeta(3)\,\zeta(7)$ | $8.21 \times 10^{14}$ |
| $s_h(4,8)$, $\zeta(12)$, $\zeta(5)\,\zeta(7)$, $\zeta(3)\,\zeta(9)$, $\zeta(3)\,\zeta(4)\,\zeta(5)$, $\zeta^2(3)\,\zeta(6)$, $\zeta^4(3)$, $\zeta(2)\,\zeta^2(5)$, $\zeta(2)\,\zeta(3)\,\zeta(7)$ | $1.06 \times 10^{15}$ |
| $s_h(1,9), s_h(2,8), s_h(3,7), s_h(4,6), s_h(5,5), s_h(6,4), s_h(7,3), s_h(8,2)$ | $2.31 \times 10^{16}$ |
| $s_h(1,10), s_h(2,9), s_h(3,8), s_h(4,7), s_h(5,6), s_h(6,5), s_h(7,4), s_h(8,3), s_h(9,2)$ | $1.05 \times 10^{15}$ |
| $s_h(1,10), s_h(2,9), s_h(3,8), s_h(4,7), s_h(5,6), s_h(6,5), s_h(7,4), s_h(8,3), s_h(9,2)$, $\zeta(11)$ | $6.54 \times 10^{13}$ |
| $s_h(1,11), s_h(2,10), s_h(3,9), s_h(4,8), s_h(5,7), s_h(6,6), s_h(7,5), s_h(8,4), s_h(9,3), s_h(10,2)$ | $6.77 \times 10^{13}$ |
| $s_h(1,11), s_h(2,10), s_h(3,9), s_h(4,8), s_h(5,7), s_h(6,6), s_h(7,5), s_h(8,4), s_h(9,3), s_h(10,2)$, $\zeta(12)$ | $2.67 \times 10^{11}$ |
| $a_h(1,5)$ plus $A_6$ (see caption) | $7.29 \times 10^{10}$ |
| $s_a(2,4)$ plus $A_6$ | $6.08 \times 10^{10}$ |
| $s_a(3,3)$ plus $A_6$ | $5.95 \times 10^{10}$ |
| $s_a(2,5)$ plus $A_7$ | $2.63 \times 10^{6}$ |
| $s_a(3,4)$ plus $A_7$ | $4.73 \times 10^{6}$ |
| $s_a(2,5), s_a(3,4), s_a(4,3), s_a(5,2)$ plus $A_7$ | $3.16 \times 10^{5}$ |
| $\sigma_h(2,6)$, $\zeta(8)$, $\zeta(3)\,\zeta(5)$, $\zeta(2)\,\zeta^2(3)$ | $6.81 \times 10^{41}$ |
| $\sigma_h(3,5)$, $\zeta(8)$, $\zeta(3)\,\zeta(5)$, $\zeta(2)\,\zeta^2(3)$ | $6.26 \times 10^{41}$ |
| $\sigma_h(2,8)$, $\zeta(10)$, $\zeta^2(5)$, $\zeta(3)\,\zeta(7)$, $\zeta^2(3)\,\zeta(4)$, $\zeta(2)\,\zeta(3)\,\zeta(5)$ | $3.92 \times 10^{26}$ |
| $\sigma_h(3,7)$, $\zeta(10)$, $\zeta^2(5)$, $\zeta(3)\,\zeta(7)$, $\zeta^2(3)\,\zeta(4)$, $\zeta(2)\,\zeta(3)\,\zeta(5)$ | $2.78 \times 10^{24}$ |

**TABLE 5.** Relation exclusion bounds. The right-hand column gives the minimum Euclidean norm of any possible integer relation involving the listed constants. The abbreviations used are: $A_6 = \{\mathrm{Li}_6(\frac{1}{2}), \ln(2)\,\mathrm{Li}_5(\frac{1}{2}), \ln^2(2)\,\mathrm{Li}_4(\frac{1}{2}), \ln^6(2), \zeta(6), \zeta(5)\ln(2), \zeta(4)\ln^2(2), \zeta(3)\ln^3(2), \zeta^2(3), \zeta(2)\ln^4(2), \zeta(2)\,\zeta(3)\ln(2)\}$; $A_7 = \{\mathrm{Li}_7(\frac{1}{2}), \ln(2)\,\mathrm{Li}_6(\frac{1}{2}), \ln^2(2)\,\mathrm{Li}_5(\frac{1}{2}), \ln^3(2)\,\mathrm{Li}_4(\frac{1}{2}), \ln^7(2), \zeta(7), \zeta(6)\ln(2), \zeta(5)\ln^2(2), \zeta(4)\ln^3(2), \zeta(3)\,\mathrm{Li}_4(\frac{1}{2}), \zeta(3)\ln^4(2), \zeta(3)\,\zeta(4), \zeta^2(3)\ln(2), \zeta(2)\,\mathrm{Li}_5(\frac{1}{2}), \zeta(2)\ln^5(2), \zeta(2)\,\zeta(5), \zeta(2)\,\zeta(3)\ln^2(2)\}$.

The modular properties of the sums $\sigma_h(m,n)$ are being investigated by D. Zagier (private communication). His work provides an alternate, abstract proof that $\sigma_h(m,n)$ evaluates in terms of zeta functions if $m+n$ is odd. This corresponds to our results and still leaves the case $m+n$ even as an open problem.

## ACKNOWLEDGMENTS

The authors wish to thank David Borwein, Peter Borwein, Helaman R. P. Ferguson and Paul O. Frederickson for valuable discussions during the course of this work.

## REFERENCES

[Abramowitz and Stegun 1972] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*, Dover, New York, 1972.

[Atkinson 1989] K. E. Atkinson, *An Introduction to Numerical Analysis*, Wiley, New York, 1989.

[Bailey] D. H. Bailey, "Multiprecision translation and execution of Fortran programs," *ACM Transactions on Mathematical Software*, to appear. This software and documentation may be obtained by sending electronic mail to `mp-request@nas.nasa.gov`.

[Bailey and Ferguson 1989] D. H. Bailey and H. R. P. Ferguson, "Numerical results on relations between numerical constants using a new algorithm," *Math. Comp.* **53** (1989), 649–656.

[Berndt 1985] B. C. Berndt, *Ramanujan's Notebook*, Part I, Springer, New York, 1985.

[Borwein and Borwein 1994] D. Borwein and J. M. Borwein, "On an intriguing integral and some series related to $\zeta(4)$," to appear in *Proc. Amer. Math. Soc.*

[Borwein et al. 1994] D. Borwein, J. M. Borwein and R. Girgensohn, "Explicit evaluation of Euler sums," to appear in *Proc. Edinburgh Math. Soc.*

[Borwein and Borwein 1987] J. M. Borwein and P. B. Borwein, *Pi and the AGM*, Wiley, New York, 1987.

[Borwein et al. 1989] J. M. Borwein, P. B. Borwein and K. Dilcher, "Pi, Euler Numbers and Asymptotic Expansions," *Amer. Math. Monthly* **96** (1989), 681–687.

[Char et al. 1991] B. W. Char, K. O. Geddes, G. H. Gonnet, B. L. Leong, M. B. Monagan, S. M. Watt, *Maple V Language Reference Manual*, Springer, New York, 1991.

[De Doelder 1991] P. J. De Doelder, "On some series containing $\Psi(x)-\Psi(y)$ and $(\Psi(x)-\Psi(y))^2$ for certain values of $x$ and $y$," *J. Comp. Appl. Math.* **37** (1991), 125–141.

[Ferguson and Bailey 1991] H. R. P. Ferguson and D. H. Bailey, "A polynomial time, numerically stable integer relation algorithm," RNR Technical Report RNR-91-032, NASA Ames Research Center, MS T045-1, Moffett Field, CA 94035-1000.

[Ferguson and Forcade 1979] H. R. P. Ferguson and R. W. Forcade, "Generalization of the Euclidean algorithm for real numbers to all dimensions higher than two," *Bull. Amer. Math. Soc.* **1** (1979), 912–914.

[Ferguson 1987] H. R. P. Ferguson, "A non-inductive $\mathrm{GL}(n,\mathbb{Z})$ algorithm that constructs linear relations for $n$ $\mathbb{Z}$-linearly dependent real numbers," *J. Algorithms* **8** (1987), 131–145.

[Hastad et al. 1988] J. Hastad, B. Just, J. C. Lagarias and C. P. Schnorr, "Polynomial time algorithms for finding integer relations among real numbers," *SIAM J. Computing* **18** (1988), 859–881.

[Knuth 1973] D. E. Knuth, *The Art of Computer Programming*, vol. 1, Addison-Wesley, Reading, MA, 1973.

[Knuth 1981] D. E. Knuth, *The Art of Computer Programming*, vol. 2, Addison-Wesley, Reading, MA, 1981.

[Lenstra et al. 1982] A. K. Lenstra, H. W. Lenstra and L. Lovasz, "Factoring polynomials with rational coefficients", *Math. Annalen* **261** (1982), 515–534.

[Lewin 1981] L. Lewin, *Polylogarithms and associated functions*, North-Holland, New York, 1981.

[Wolfram 1991] S. Wolfram, *Mathematica: A System for Doing Mathematics by Computer*, 2nd ed., Addison-Wesley, Reading, MA, 1991.

David H. Bailey, NAS Applied Research Branch, NASA Ames Research Center, Moffett Field, CA 94035-1000, USA (dbailey@nas.nasa.gov)

Jonathan M. Borwein, Department of Mathematics and Statistics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (jborwein@cecm.sfu.ca)

Roland Girgensohn, Department of Mathematics and Statistics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada (girgen@cecm.sfu.ca)

Received October 25, 1993; accepted in revised form May 19, 1994