Poisson psi polynomials: Computation and analysis

David H. Bailey
Lawrence Berkeley National Laboratory
dhbailey@lbl.gov

September 4, 2025

Abstract

Earlier studies have explored the intriguing phenomenon of algebraic numbers arising from a simple two-dimensional instance of the Poisson potential function of mathematical physics:

$$\phi_2(x, y) = \frac{1}{\pi^2} \sum_{m,n \text{ odd}} \frac{\cos(m\pi x)\cos(n\pi y)}{m^2 + n^2}.$$

In this paper, we address the closely related function (even indices instead of odd, excluding $(0, 0)$):

$$\psi_2(x, y) = \frac{1}{\pi^2} \sum_{m,n \text{ even}'} \frac{\cos(m\pi x)\cos(n\pi y)}{m^2 + n^2}.$$

As with $\phi_2(x, y)$, it is known from an earlier study that when x and y are *rational numbers*, then $\psi_2(x, y) = 1/\pi \cdot \log(\beta(x, y))$, where $\beta(x, y)$ is an *algebraic number* of some degree m.

In this paper we present formulas and techniques for rapid numerical computation of $\psi_2(x, y)$, corrected from an earlier study, together with an initial catalogue of the minimal polynomials satisfied by $\alpha = \exp(8\pi s\psi_2(x, y))$. These computations, which are much more challenging than with $\phi_2(x, y)$, cover the cases $(x, y) = (p/s, q/s)$, where $1 \le p \le q < s/2$ and $\gcd(p, q, s) = 1$, for $10 \le s \le 25$ and also for $s = 26, 28, 30, 32, 34, 36, 40$, a total of 1,017 cases. With this catalogue of computational results in hand, we note several intriguing regularities, including (tentatively): (a) a variant of Kimberley's formula that gives the degrees of the minimal polynomials; and (b) the fact that for a given s, all the cases $(x, y) = (p/s, p/s)$, with $1 \le p < s/2$ and $\gcd(p, s) = 1$, share the same minimal polynomial. These polynomials typically do *not* exhibit the palindromic property observed for $\phi_2(p/s, q/s)$ when s is even.

1 Earlier work on Poisson polynomials

Lattice sums related to the Poisson potential function naturally arise in studies of gravitational and electrostatic potentials, and have been studied in the mathematical physics community for many years [1, 9, 13, 14, 18]. Lord Rayleigh, in his 1892 paper, mentions Lorenz as the inventor of the concept [16]. More recently, researchers have identified applications in practical image processing [5]. These developments have underscored the need to better understand the underlying mathematical theory.

In two earlier papers [5, 6], Jonathan Borwein (deceased 2016), Richard Crandall (deceased 2012) and I. J. Zucker analyzed a simple two-dimensional instance of the Poisson potential function:

$$\phi_2(x, y) = \frac{1}{\pi^2} \sum_{m,n \text{ odd}} \frac{\cos(m\pi x)\cos(n\pi y)}{m^2 + n^2}. \tag{1}$$

These researchers empirically discovered and then proved the intriguing fact that when x and y are rational numbers, then

$$\phi_2(x, y) = \frac{1}{\pi} \log(\beta(x, y)), \tag{2}$$

where $\beta(x,y)$ is an *algebraic number*, namely the root of a degree-m minimal polynomial with integer coefficients, for some m.

This result can be explored computationally as follows: Given rationals x, y and an integer m, compute $\alpha = \exp(8\pi\phi_2(x,y))$ to high precision, generate the $(m+1)$-long vector $(1, \alpha, \alpha^2, \cdots, \alpha^m)$, and then apply an integer relation algorithm to discover the coefficients of the polynomial of degree m, if it exists, satisfied by α. It is not practical to numerically evaluate $\phi_2(x,y)$ by the defining formula (1), but Borwein and Crandall discovered rapidly computable formulas for $\phi_2(x,y)$ in terms of theta functions [5].

Based on some initial computational results, Jason Kimberley of the University of Newcastle, Australia observed that the degree $m(s)$ of the minimal polynomial associated with the special case $(x,y) = (1/s, 1/s)$ appears to be given by the following number-theoretic rule [6]: Set $m(2) = 1/2$. Otherwise for primes p congruent to 1 modulo 4, set $m(p) = (p-1)^2/4$, and for primes p congruent to 3 modulo 4, set $m(p) = (p^2-1)/4$. Then for any other positive integer s whose prime factorization is $s = p_1^{e_1} p_2^{e_2} \cdots p_r^{e_r}$,

$$m(s) = 4^{r-1} \prod_{i=1}^{r} p_i^{2(e_i-1)}\, m(p_i). \tag{3}$$

Subsequent computations confirmed that Kimberley's formula holds for $(x,y) = (1/s, 1/s)$ for all integers s up to 40, and also for most even integers up to 64. By doing Google searches on the coefficients of the resulting polynomials, the authors found a connection to a 2010 paper by Savin and Quarfoot [17]. These investigations ultimately led to a proof, given in [6], that Kimberley's formula (3) is valid in the special case $(x,y) = (1/s, 1/s)$, and, when s is even, the minimal polynomials for $(x,y) = (1/s, 1/s)$ are palindromic (i.e., coefficient $a_k = a_{m-k}$, where m is the degree).

In [3, 4] these computations were extended to the much larger set of mixed arguments, namely $(x,y) = (p/s, q/s)$, where $1 \le p \le q < s/2$ and $\gcd(p,q,s) = 1$, for $10 \le s \le 36$ and also for $s = 38, 40, 42$ and $s = 50$, a total of 2,206 cases. With this extensive catalogue of computational results in hand, we were able to note (tentatively): (a) a variant of Kimberley's formula that gives the degrees; (b) the fact that for a given s, all the cases $(x,y) = (p/s, p/s)$, with $1 \le p < s/2$ and $\gcd(p,s) = 1$, share the same minimal polynomial; and (c) the fact that whenever s is even, the minimal polynomials are palindromic.

2 The Poisson psi function

The 2013 study [5] also mentioned the closely related function

$$\psi_2(x,y) = \frac{1}{\pi^2} \sum_{m,n \text{ even}'} \frac{\cos(\pi m x)\cos(\pi n y)}{m^2+n^2}, \tag{4}$$

which differs from $\phi_2(x,y)$ by replacing *odd* indices with *even*, excluding $(m,n) = (0,0)$. The $\psi_2(x,y)$ function is the natural potential for a classical "jellium" crystal, namely a structure with a positive charge at every integer lattice point, in a bath (a jelly) of uniform negative charge density [11]. As with $\phi_2(x,y)$, the authors of [5] were able to show that when x and y are rational, then $\psi_2(x,y) = 1/\pi \cdot \log(\beta(x,y))$ for some algebraic $\beta(x,y)$.

It is not possible to numerically compute $\psi_2(x,y)$ by formula (4), since millions of terms are required to obtain even a few correct digits. Thus a key breakthrough in this research was the discovery, due to Borwein and Crandall, of formulas permitting fast computation of both $\phi_2(x,y)$ and $\psi_2(x,y)$ [5, Thm. 9] (the formulas given for $\psi_2(x,y)$ in [5] are flawed, thus preventing computer exploration; they are corrected below). These formulas, in turn, are based on the following two formulas (the second was incorrectly presented in earlier literature, but was corrected in [5]):

$$\phi_2(x,y) = \frac{1}{4\pi} \log|A(z,q)|, \tag{5}$$

$$\psi_2(x,y) = \frac{x^2}{2} + \frac{1}{4\pi} \log\left(\frac{\Gamma(1/4)}{\sqrt{8\pi}\,\Gamma(3/4)}\right) - \frac{1}{2\pi} \log\left|\theta_1(2z, e^{-\pi})\right|, \tag{6}$$

where $q = e^{-\pi}$, $z = \pi/2 \cdot (y + ix)$, $A(z,q) = (\theta_2^2(z,q)\theta_4^2(z,q))/(\theta_1^2(z,q)\theta_3^2(z,q))$, and the theta functions are defined below. Based on these formulas, the authors of [5] derived the following equivalents, which we present here in corrected form and with less ambiguous notation:

$$\phi_2(x,y) = \frac{1}{2\pi} \log \left| \frac{\theta_2(z,q)\theta_4(z,q)}{\theta_1(z,q)\theta_3(z,q)} \right| = \frac{1}{4\pi} \log \left| \frac{1 - \lambda(z,q)/\sqrt{2}}{1 - 1/(\lambda(z,q)\sqrt{2})} \right|, \tag{7}$$

$$\psi_2(x,y) = -\frac{1}{4\pi} \log \left| 2\,\mu(2z,q) \left(\sqrt{2}\,\lambda(2z,q) - 1 \right) \right|, \tag{8}$$

where $\mathrm{Im}(z)$ denotes imaginary part, and

$$\lambda(z,q) = \frac{\theta_4^2(z,q)}{\theta_3^2(z,q)} = \prod_{n=1}^{\infty} \frac{(1 - 2\cos(2z)q^{2k-1} + q^{4k-2})^2}{(1 + 2\cos(2z)q^{2k-1} + q^{4k-2})^2}, \tag{9}$$

$$\mu(z,q) = \exp\left(-2\,\mathrm{Im}^2(z)/\pi\right) \frac{\theta_3^2(z,q)}{\theta_3^2\,(0,q)} = \exp\left(-2\,\mathrm{Im}^2(z)/\pi\right) \prod_{n=1}^{\infty} \frac{(1 + 2\cos(2z)q^{2k-1} + q^{4k-2})^2}{(1 + q^{2k-1})^4}. \tag{10}$$

The theta functions can be computed using the following rapidly convergent formulas from [8, pg. 52]:

$$\begin{aligned}
\theta_1(z,q) &= 2\sum_{k=1}^{\infty} (-1)^{k-1} q^{(2k-1)^2/4} \sin((2k-1)z), \\
\theta_2(z,q) &= 2\sum_{k=1}^{\infty} q^{(2k-1)^2/4} \cos((2k-1)z), \\
\theta_3(z,q) &= 1 + 2\sum_{k=1}^{\infty} q^{k^2} \cos(2kz), \\
\theta_4(z,q) &= 1 + 2\sum_{k=1}^{\infty} (-1)^k q^{k^2} \cos(2kz).
\end{aligned} \tag{11}$$

The present author has implemented three variations of these formulas, using both *Mathematica* and a high-precision software package: (a) formulas (5) and (6); (b) formulas (7) through (11) with the first parts of (9) and (10); and (c) formulas (7) through (11) with the second parts of (9) and (10). All three agree on test problems, and run times are within a factor of two. Option (b) was employed below.

The study [5] included several explicit evaluations for $\psi_2(x,y)$, but one of these was in error. Here is a corrected collection, with two additional results due to the present author:

$$\begin{aligned}
\psi_2(1/3,1/3) &= \frac{1}{24\pi} \log\left(\frac{2\sqrt{3} - 3}{9} \right), \\
\psi_2(1/4,1/4) &= \frac{1}{16\pi} \log(1/2), \\
\psi_2(1/5,1/5) &= \frac{1}{40\pi} \log\left(-\frac{709}{2} + \frac{319}{2}\sqrt{5} + 3\sqrt{28090 - 12562\sqrt{5}} \right), \\
\psi_2(1/6,1/6) &= \frac{1}{12\pi} \log(2 + \sqrt{3}), \\
\psi_2(1/8,1/8) &= \frac{1}{64\pi} \log\left(\frac{1737169}{2} + 614182\sqrt{2} + 4\sqrt{2\left(47152440367 + 33341810333\sqrt{2}\right)} \right), \\
\psi_2(1/10,1/10) &= \frac{1}{80\pi} \log\left[\frac{283373459287}{2} + \frac{126728463597\sqrt{5}}{2} \right. \\
&\quad \left. + 12\sqrt{2\left(139410620535209513705 + 62346324860431317259\sqrt{5}\right)} \right].
\end{aligned} \tag{12}$$

In this paper, we describe the computation of minimal polynomials for $\alpha = \exp(8\pi s \psi_2(x, y))$ (which, as can be seen above, is a natural form for this study), for all $(x, y) = (p/s, q/s)$, where $1 \le p \le q < s/2$, for $10 \le s \le 25$, and also for $s = 26, 28, 30, 32, 34, 36$ and 40, a total of 1,017 cases. These computations and analyses are significantly more challenging than with $\phi_2(x, y)$, requiring much higher numeric precision (up to 150,000 digits) and much longer run times (typically 100X or more, compared to equivalent $\phi_2(x, y)$ cases). See Table 3 below for some statistics.

Some high-level details of the algorithms and techniques employed in this study are given in Sections 3 through 6 below, parts of which are adapted and condensed from an earlier $\phi_2(x, y)$ study [4].

3 Finding minimal polynomials using integer relation algorithms

Given an n-long input vector $v = (v_i, 1 \le i \le n)$ of real numbers, typically given as high-precision floating-point values, an integer relation algorithm attempts to find integers (a_i), not all zero, such that

$$a_1 v_1 + a_2 v_2 + \cdots + a_n v_n = 0, \tag{13}$$

to within the tolerance of the numeric precision being used.

If one suspects that a high-precision floating-point value α is an algebraic number of degree m, one may compute the $(m+1)$-long vector $X = (1, \alpha, \alpha^2, \cdots, \alpha^m)$ to high precision and then apply an integer relation algorithm. If an integer relation (a_i) is found for X, then the resulting vector of integers may be the coefficients of an integer polynomial of degree m satisfied by α, subject to further verification.

As an illustration, suppose one suspects that the real constant α, whose numerical value to 40 digits is 2.119591269829175131329848334934687110628 0..., is an algebraic number of degree eight. After computing the vector $(1, \alpha, \alpha^2, \cdots, \alpha^8)$, say to 100-digit precision, and applying the multipair PSLQ integer relation algorithm (see next section), the relation $(1, -216, 860, -744, 454, -744, 860, -216, 1)$ is produced, so that α appears to satisfy the polynomial $1 - 216\alpha + 860\alpha^2 - 744\alpha^3 + 454\alpha^4 - 744\alpha^5 + 860\alpha^6 - 216\alpha^7 + \alpha^8 = 0$. *Maple* or *Mathematica* may then be used to verify that the resulting polynomial is irreducible; alternatively, one may attempt to recover an integer relation with the degree reduced by one, and verify that no numerically significant relation is produced with this smaller degree.

4 The three-level multipair PSLQ algorithm

The multipair PSLQ algorithm [7] is an efficient and moderately parallelizable variant of PSLQ, a widely used integer relation algorithm. Variants of the LLL algorithm are also used [10]. For convenience, full statements of the PSLQ and multipair PSLQ algorithms are presented below in Section 9.

In brief, given an n-long input vector v, the multipair PSLQ algorithm generates a sequence of invertible $n \times n$ integer matrices A_k, their inverses B_k and real $n \times (n-1)$ matrices H_k, so that the reduced vector $w = B_k \cdot v$ has steadily smaller entries, until one entry of w is smaller than the specified epsilon (with the relation given in the corresponding row of B_k), or else precision is exhausted.

Integer relation detection by any algorithm requires very high numeric precision. It can be seen from a combinatorial argument that one must employ at least $n \cdot \max_i \log_{10} |a_i|$ digits, or else the true relation will be lost in a sea of numerical artifacts. Multipair PSLQ can typically detect a relation when the numeric precision is only a few percent higher than this minimum bound [7].

The computations in this study employed a three-level variable precision implementation of the multipair PSLQ algorithm [7]: (a) double precision (15 digits); (b) medium precision (typically 500 to 10,000 digits); and (c) full precision (typically 10,000 to 150,000 digits). With this scheme, almost all iterations of the multipair PSLQ algorithm are performed very rapidly using ordinary double precision (DP) floating-point arithmetic. When an entry of the DP w vector is smaller than 10^{-14}, or when an entry of the DP A or B array exceeds 10^{13}, the medium precision arrays are updated from the DP arrays using matrix multiplication via the formulas

$$w := \hat{B} \cdot w, \; B := \hat{B} \cdot B, \; A := \hat{A} \cdot A, \; H := \hat{A} \cdot H, \tag{14}$$

where the hat notation indicates DP arrays. When an entry of the medium precision w vector is smaller than the medium precision epsilon, or when an entry of the medium precision A or B array nearly exceeds the largest integer value exactly representable in medium precision, then the full precision arrays are updated from the medium precision arrays using similar formulas. On large problems the three-level scheme is typically 100X faster than a straightforward implementation using only full precision.

Considerable care must be taken in this implementation to correctly detect when precision has been exhausted at each level, to reliably process the handoff to higher or lower level of precision, and to recover from a situation where an iteration must be abandoned due to precision overflow. Also, advanced multi-precision arithmetic techniques, such as fast Fourier transform (FFT)-based multiplication, are required to obtain optimal performance on large problems. For full details see [7, 3].

5 Numerical reliability

While these computations do not constitute formal mathematical proofs, with some care these results can be very reliable. Figure 1 illustrates the process of finding a relation using the scheme described above. In particular, the graph shows the base-10 logarithm of the minimum absolute value of the w vector (vertical axis), plotted against the iteration number (horizontal axis), in the multipair PSLQ computer run that produced the 36-degree minimal polynomial corresponding to the case $(x, y) = (1/13, 1/13)$.

Note that as the algorithm proceeds, the minimum absolute value of the w vector steadily decreases, from approximately 10^{-753} to approximately 10^{-3874}, but at iteration 17,021 abruptly drops to approximately 10^{-6000}, a drop of 2126 orders of magnitude. Note that since the run employed 6000-digit precision, the value 10^{-6000} is effectively zero, so the algorithm terminates here. In other words, the polynomial relation found by the computer run holds to roughly 2126 digits beyond the precision level required to discover it. This dynamic range at the iteration of detection can thus be considered a "confidence level" of the result's numerical reliability. In the computer runs for the present study, all results obtained by multipair PSLQ runs exhibited a dynamic range of at least several hundred digits, and in most cases several thousand digits.

Additionally, in every case studied below, the set of coefficients found for a $\psi_2(x, y)$ polynomial has shape akin to an asymmetric parabola, with small coefficients at the start (often ± 1), a maximum size near the middle and small again at the end. Table 1 (shown in a very small font), presents the degree-36 minimal polynomial found by the author's program for the case $(x, y) = (1/13, 1/13)$, and Table 2 presents the degree-32 minimal polynomial found by the author's program for the case $(x, y) = (1/24, 9/24)$. Each is typical of Poisson $\psi_2(x, y)$ polynomials, in that the initial coefficient is ± 1, then coefficients ascend to a maximum size (here roughly 10^{106} and 10^{130}, respectively), and then descend back down to ± 1.

This asymmetric parabolic pattern, from tiny to huge to tiny, is strong numerical evidence that the polynomial produced by the computer program is the true minimal polynomial associated in these cases, and that all hardware, software and application code performed flawlessly, since otherwise it is *exceedingly unlikely* that the final set of coefficients would have this distinctive and highly improbable pattern. By contrast, in cases where the program fails to find a numerically significant relation, say due to a coding bug, insufficient degree or insufficient precision, the resulting erroneous integer coefficients typically are all roughly the same size, within one or two orders of magnitude, as if generated by a uniform pseudorandom generator. Visually speaking, an erroneous set of coefficients appears as a rectangle rather than an asymmetric parabola.

6 High-level computational algorithm

As mentioned above, a key breakthrough in the study of these Poisson polynomials was the discovery, by Borwein and Crandall, that both $\phi_2(x, y)$ and $\psi_2(x, y)$ can be numerically computed very rapidly using theta functions from the theory of elliptic functions. Unfortunately, the formulas given in [5] for $\psi_2(x, y)$ are flawed, but were presented above (Section 2) in corrected form. In particular, here is the high-level algorithm employed in this study to discover the $\psi_2(x, y)$ polynomials:

$+1\ \alpha^{0}$
$-102008900\ \alpha^{1}$
$+3386359201083610\ \alpha^{2}$
$-45767430603522450027036\ \alpha^{3}$
$+235847871430876886823255114847\ \alpha^{4}$
$-401808595154612767463343530401906914\ \alpha^{5}$
$+322639319964424434060996969082765345466492\ \alpha^{6}$
$-128935196503678705655858436162015626186093449926\ \alpha^{7}$
$+254366151720695039825209947252397855665352247599405 43\ \alpha^{8}$
$-183563571956175981819119019501016765572724369008916067 3300\ \alpha^{9}$
$+129524292842384780491187097906105207534455460055928556272450511\ \alpha^{10}$
$-50643964076651548134188406197745972399247560020455770366680769187 94\ \alpha^{11}$
$+119351474020211942432679618099379351018638670800160392135616726563072711\ \alpha^{12}$
$-224470825349640047710437542854033751040897002752617962303021689881813 7766158\ \alpha^{13}$
$-870786004906133783096226981075279808255547512718656267021486479206827455704 59492\ \alpha^{14}$
$-140638908054506028567444768392928229345022043593595093174631567731349477750975215 3540\ \alpha^{15}$
$-113857580255448942567775801877241706235482318946855691158239995413812959650343613903 10766\ \alpha^{16}$
$-443471139455584677707404662176778135920558558131518180839262426731741040412514105502891 08570\ \alpha^{17}$
$-804291850259893423262606712785456930847650482968385178781804256591775481334031350376286193 91495\ \alpha^{18}$
$-776092482607699435134104584812076912438393426275378796708993445861063506621970494433321376706 14362\ \alpha^{19}$
$-984869729602869350585974274751587527257977783898002807698675464321093584309232417881441351175739 40304\ \alpha^{20}$
$-196916509108561280726348059526173479986764399472822864535007515131473701389796432459908585329453722 05916\ \alpha^{21}$
$+930991878738589274596983200570581437557191635209585348848648176548365191235087708730688914482798705 0001\ \alpha^{22}$
$-106664560108285377057792262093233638788085055339033980271645030858310578824875595368385789115973010767 55566\ \alpha^{23}$
$+354177293967643061141762989295284373185179453110271554764233110846188885153863684384031589356947520154 80023\ \alpha^{24}$
$+116069960465502784558571674771638887485284225126865778906578570502032851030687844459268172813358665411 435964\ \alpha^{25}$
$+881324833527136672038788993863877131688996093514776742873729696518474888307395906386235721770213681480 34789\ \alpha^{26}$
$+166707380030691034516888097700879893892458301449119121624216476871337035450251555949126852167017096871 22038\ \alpha^{27}$
$-704710382216061226800533124363494579736487582187308414415772397548018470271791769799111032904540109 122\ \alpha^{28}$
$+190483525794080343148637055868161392775445889322696089097109677297582954919038070639324562064597 54\ \alpha^{29}$
$+685801375051471321819350009602526290988426139567571256059200121624340254444320223689387767 89\ \alpha^{30}$
$-242852122198134366320150437428979278784648291536502453876854273726532427941829453160 36\ \alpha^{31}$
$-150937181998961371940762728692330965504227666747719379967841633525298631117\ \alpha^{32}$
$-967710126182231085867777501101834999529007048488756535637943516\ \alpha^{33}$
$-287621740841771257846160456053044691979984738239 97\ \alpha^{34}$
$+346262896970171679004695509 86\ \alpha^{35}$
$+1\ \alpha^{36}$

Table 1: Degree-36 minimal polynomial found for the case $(x, y) = (1/13, 1/13)$

$-1\ \alpha^{0}$
$+1517633683648\ \alpha^{1}$
$-10703210741996158640146 08\ \alpha^{2}$
$+160770009694588411495737213776184 96\ \alpha^{3}$
$+532539618188391769214800464078279737718357 84\ \alpha^{4}$
$+102154152904859512445035585931300928840268254701665472\ \alpha^{5}$
$-355134723946655103535462538871183148710150223902144506789662 4\ \alpha^{6}$
$-554863910246571490047397481563908101743994016452066819866994995540416\ \alpha^{7}$
$-456068625135300880404526075979695045564225540721629846219568094179542616399 64\ \alpha^{8}$
$+670291922167927614981502689703928928861430088377593305299059116201930373017645536 64\ \alpha^{9}$
$-595905195905694603588462192244385992695205271977913624546982493714892215344375837627560 3472\ \alpha^{10}$
$+100048740278267488042048256861542700373055776478792172039345529702151821895599229718834514266671936\ \alpha^{11}$
$-350177363341451941455035658205833596702341009743576501156651113523159226127638586494609376001647233636680\ \alpha^{12}$
$+881472974554592249157201775536177591524823556687278499896398088470381359988152667316050413466631613918078 8672\ \alpha^{13}$
$-118549312366609619628564149980792313147852701945571355416016878859311109215178021209826764350986045630608718 881200\ \alpha^{14}$
$+124189769244803224297213171920582098201040916356205758243909300377539493421576241081349518568282185753330021457 6274240\ \alpha^{15}$
$-102994186566750030397131059587509168059366442548514178075101356561299172236636902622684889746481900558250225236756660 70086\ \alpha^{16}$
$+557138661401273802639442495321129962275843542497283947791504515464982377779752850427304417976509822775406063132372235007 51424\ \alpha^{17}$
$-417497986661969369510207854855330800313977605520628549346802809031577998952685439410881788148567492643507241931068561055123 39888\ \alpha^{18}$
$+963282390895422267128761035512142337853138792781497895980862128613688587833090282104829941574833655312753128953409758495785810 7072\ \alpha^{19}$
$-115512897579136122764505330906149262442977461093760752697506071215614526849714890972942434256998202785106529650909857007718301 67880\ \alpha^{20}$
$+758104580411550662205525687559831846613382993226726682193143553620135609621445168198153728648295341027034593935160964005138832 5440\ \alpha^{21}$
$-137118273753493683413564299341385746258100935434045664160804887645808449966868455614839616685968012144385298186325906315910901 92\ \alpha^{22}$
$+894373369131507040177219834867371849482546622186302855430384823931068225279035743469724018466314642980635123061495299061420 8\ \alpha^{23}$
$-231168405849276056952084673193189393782715211765239995759443105968570598164831286457772152433253126803768511861583979292\ \alpha^{24}$
$+729849703471389403379821738469790959323260834357348903968706776872843424809342619623342808385731802560339122707283 2\ \alpha^{25}$
$-313481814534670667696345811744235813685237430447201318642419675490729824838809660008139085193458975759235 04\ \alpha^{26}$
$-125983281723249988735636152804856965060441480145862833730383442926970841992531803334073340933801024\ \alpha^{27}$
$-108397961801245459860567796763706983264832823455262849002828973202977187842059921304757880\ \alpha^{28}$
$+319994525123437142325919236048329182301605258362894026779337231829219977439552\ \alpha^{29}$
$-697418281407673477776633103742501391603178855541498424588651024144\ \alpha^{30}$
$+155515927335146581596452493989574 4\ \alpha^{31}$
$-1\ \alpha^{32}$

Table 2: Degree-32 minimal polynomial found for the case $(x, y) = (1/24, 9/24)$

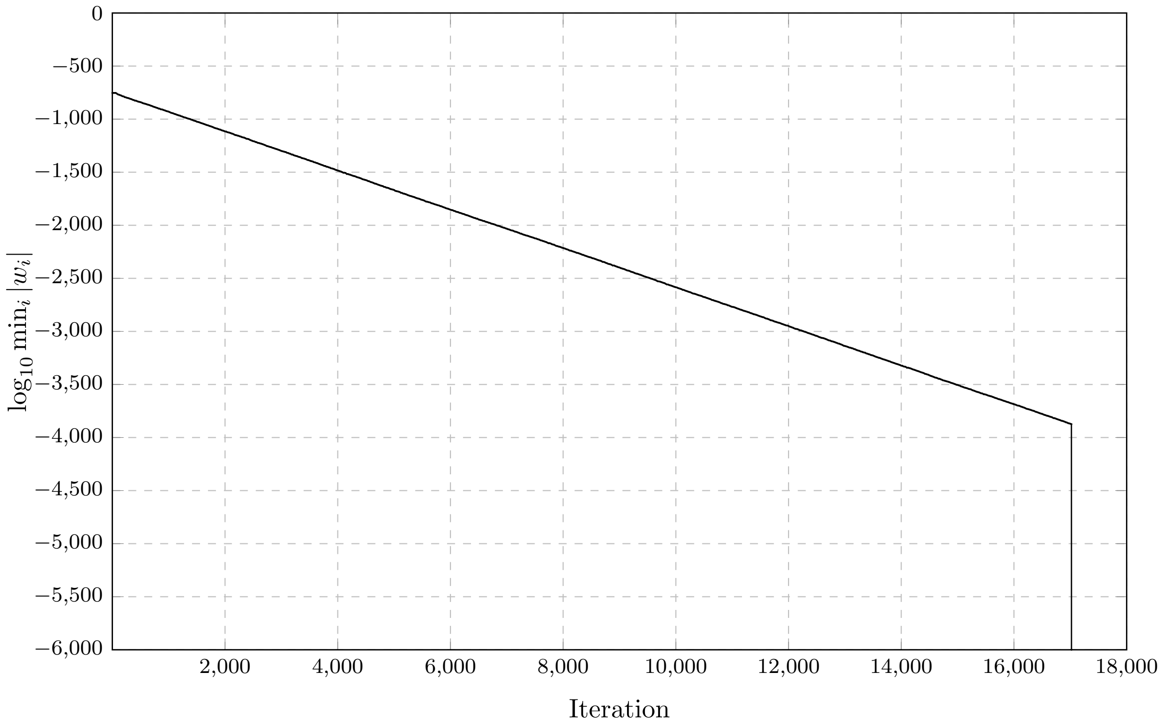

Figure 1: Plot of $\log_{10} \min_i |w_i|$ in the multipair PSLQ computer run for the case $(x, y) = (1/13, 1/13)$, showing the detection of the relation at iteration 17,021.

1. Given rationals $(x, y) = (p/s, q/s)$, typically satisfying $1 \le p \le q < s/2$ for $s \le 40$, with $\gcd(p, q, s) = 1$, select a conjectured minimal polynomial degree m (say from Kimberley's rule), a medium precision level P_1 digits, a full precision level P_2 digits and other parameters for the run.

2. Calculate $\psi_2(x, y)$ to P_2-digit precision using the formula (from Section 2 above)

$$\psi_2(x, y) = -\frac{1}{4\pi} \log \left| 2\, \mu(2z, q) \left(\sqrt{2}\, \lambda(2z, q) - 1 \right) \right|,$$

where $q = e^{-\pi}$, $z = \pi/2 \cdot (y + ix)$, $\mathrm{Im}(z)$ denotes imaginary part, and

$$\mu(z, q) = \exp\left(-2\,\mathrm{Im}^2(z)/\pi\right) \frac{\theta_3^2(z, q)}{\theta_3^2\,(0, q)}, \quad \lambda(z, q) = \frac{\theta_4^2(z, q)}{\theta_3^2(z, q)}. \tag{15}$$

Compute $\theta_3(z, q)$ and $\theta_4(z, q)$ using the following rapidly convergent formulas from [8, pg. 52]:

$$\begin{aligned} \theta_3(z, q) &= 1 + 2 \sum_{k=1}^{\infty} q^{k^2} \cos(2kz), \\ \theta_4(z, q) &= 1 + 2 \sum_{k=1}^{\infty} (-1)^k q^{k^2} \cos(2kz). \end{aligned} \tag{16}$$

3. Calculate $\alpha = \exp(8\pi s \psi_2(x, y))$ to P_2-digit precision and generate the $(m + 1)$-long vector $X = (1, \alpha, \alpha^2, \cdots, \alpha^m)$.

4. Apply a three–level multipair PSLQ algorithm to find a numerically significant integer relation for X, if one exists (see Sections 4, 5 and 9).

5. If a numerically significant relation is not found, try again with a larger degree m and/or higher precision P_2. If a tentative relation is found, employ *Mathematica* or *Maple* to ensure that the resulting polynomial is irreducible. If it is not, rerun the problem with reduced degree m, until a degree m is found that produces a numerically significant relation passing the irreducibility test.

Previous computations in the $\phi_2(x,y)$ study, as catalogued in [4], found that whenever s is even, the corresponding minimal polynomial is always palindromic, i.e., coefficient $a_k = a_{m-k}$, where m is the degree. In such cases, one can apply the fact that if α satisfies a palindromic polynomial of degree m, then $\alpha + 1/\alpha$ satisfies a polynomial of degree $m/2$ (thus greatly reducing the run time), and the degree-m polynomial satisfied by α can then be easily reconstructed from the degree-$m/2$ polynomial satisfied by $\alpha + 1/\alpha$ [15]. Unfortunately, hardly any of the minimal polynomials found in the present study exhibit the palindromic property, either for odd s or even s, so no computational savings of this type is possible in these runs. However, as it turns out, the costs of the $\psi_2(p/s, q/s)$ runs with even s are typically much less than for odd s of similar size, since the degrees and coefficients are smaller.

7 Results and analysis

It can be seen from formula (4) that $\psi_2(a+x, b+y) = \psi_2(x,y)$, for any integers a, b, so there is no need to consider the cases $(x,y) = (p/s, q/s)$, where either p or q is negative or where either p or q exceeds s. In fact, by symmetry it follows that only cases where $1 \le p \le q < s/2$, with $\gcd(p,q,s) = 1$, need be examined, since otherwise these cases are equivalent to cases with smaller p, q and s.

For this study, 1,017 individual cases were run, using the algorithms and software described in Sections 3 through 6. In particular, these cases are: $(x,y) = (p/s, q/s)$, where $1 \le p \le q < s/2$, $\gcd(p,q,s) = 1$, for $10 \le s \le 25$ and also for $s = 26, 28, 30, 32, 34, 36$ and 40. Some of these runs required up to 150,000-digit arithmetic. The minimal polynomials produced by these runs have coefficients as large as 10^{920}. Run times are typically 50–125X higher than the times for the corresponding $\phi_2(x,y)$ cases. The output files, with the full recovered polynomials, are quite large but are available from the author. Statistics for a brief selection of these runs are shown in Table 3.

Most of these runs were performed on an Apple Mac Studio computer with an M4 Max processor and 14 cores. The application program implementing the algorithm described above in Section 6 was coded using an arbitrary precision package, written by the present author, with a high-level language interface and FFT-based multiplication, which greatly accelerates very high precision computation [2, 3]. The resulting performance is comparable to that of MPFR [12], but with a high-level programming interface and a much simpler software installation process.

For each of these cases, the computer run exhibited a drop of at least several hundred orders of magnitude at detection and, in most cases, to several thousand orders of magnitude. Thus the polynomials produced by these calculations hold to hundreds and, in most cases, to thousands of digits beyond the precision required to discover them. Note, for example, in the last row of Table 3, that the full precision level was 140,000 digits and the detection level was 2.00×10^{-118003}. This means that the recovered minimal polynomial relation holds to roughly 22,000 digits beyond the level of precision required to discover it. *Wolfram Mathematica* 14.3 confirmed that each of these polynomials is irreducible.

The principal experimental findings of this study are the following:

1. A generalized Kimberley rule. Given $(x,y) = (p/s, q/s)$, with $1 \le p \le q < s/2$ and $\gcd(p,q,s) = 1$, let $\phi_2(x,y)$ be defined as in (1), with $\alpha = \exp(-8\pi s \psi_2(x,y))$. Then the degree of the minimal polynomial of α is given by this rule:

1. If s is even or odd, and $p = q$, then the degree is given by Kimberley's rule (3).
2. Otherwise if s is odd, then the degree is given by Kimberley's rule, except for a few cases where the degree is half Kimberley's rule.

s	p	q	m	P_1	P_2	Detection level	Largest coefficient	CPU time
10	1	4	8	200	1000	7.25e -50	1.13e 45	0.04
11	2	2	30	600	3000	3.01e -2301	2.83e 76	5.45
12	4	5	16	400	2000	9.04e -555	6.13e 71	0.35
13	2	2	36	800	5000	1.51e -3874	4.93e 256	17.67
14	2	3	24	600	4000	5.13e -1392	1.16e 107	2.94
15	2	2	32	1000	6000	2.46e -3640	2.78e 113	14.90
16	2	3	32	800	4000	2.10e -2854	6.58e 88	8.94
17	3	3	64	2500	20000	2.20e -15969	3.91e 286	911.91
18	2	5	36	800	5000	3.62e -4281	6.99e 237	18.93
19	3	3	90	4000	40000	3.70e -35399	1.36e 392	8426.15
20	3	4	32	1000	5000	2.94e -3757	1.93e 283	11.23
21	4	4	96	5000	50000	1.35e -44538	3.36e 462	24148.22
22	1	6	60	2000	18000	4.46e -13901	6.88e 230	863.87
23	6	6	132	10000	100000	2.44e -92118	4.59e 695	293444.22
24	2	5	64	3000	22000	3.68e -17273	4.82e 504	1572.62
25	4	4	100	7500	75000	3.78e -57355	1.12e 572	52476.92
26	4	5	72	4000	35000	1.24e -23557	2.26e 711	4117.15
28	1	8	96	5000	50000	1.31e -46099	2.21e 666	24986.04
30	2	9	64	4000	30000	5.50e -22683	8.55e 477	2669.25
32	2	9	128	10000	100000	2.78e -92006	6.44e 716	266489.97
34	4	15	128	11000	110000	1.34e -98262	4.33e 765	335228.14
36	4	13	144	15000	150000	5.33e-132837	3.36e 920	1014451.90
40	1	4	128	16000	140000	2.00e-118003	1.20e 919	414242.84

Table 3: Run statistics for a brief sample of runs from the catalogue. Columns:

s, p, q: Identify the case $(x, y) = (p/s, q/s)$.

m: Degree of the discovered minimal polynomial.

P_1 and P_2: Medium and full precision levels employed for the run in decimal digits.

Detection level: Size of $\max_i |w_i|$ at detection; in each case, $\min_i |w_i| \approx 10^{-P_2}$.

Largest coefficient: Approx. size of the largest coefficient in the resulting minimal polynomial.

CPU time: Total run time in processor core seconds (shown for uniformity to two decimal places, but not repeatable beyond about three significant digits).

3. If s is even, and both p and q are odd, then the degree is given by Kimberley's rule, except for a few cases where the degree is half Kimberley's rule.

4. If s is even, with one of p or q even and the other odd, then the degree is given by *twice* Kimberley's rule, except for a few cases where the degree is equal to Kimberley's rule.

2. Sharing of minimal polynomials. One particularly intriguing feature of the catalogue of results (a feature also of the $\phi_2(x, y)$ results) is that for a given integer s, many of the minimal polynomials corresponding to various $(x, y) = (p/s, q/s)$ cases are identical, even though the α numerical values are distinct. Tables 4 through 7 below present a complete summary of these data extracted from the computer runs: for a given s, each row lists (p, q) cases, corresponding to $(x, y) = (p/s, q/s)$, whose minimal polynomials are identical.

In examining these data, one striking regularity is observed: For a given s, all the cases $(x, y) = (p/s, p/s)$, where $1 \leq p < s/2$ and $\gcd(p, s) = 1$, share the same minimal polynomial. For example, for $s = 36$, the minimal polynomials for the cases $(1, 1), (5, 5), (7, 7), (11, 11), (13, 13), (17, 17)$ are all identical. Note that this represents a complete set of $(p/36, p/36)$ with $1 \leq p < 18$ and $\gcd(p, 36) = 1$.

This sharing feature of $\phi_2(x, y)$, like that of $\psi_2(x, y)$, has not been observed before in studies of Poisson polynomials, and the simplicity of this assertion suggests that it might well be amenable to further theoretical analysis. Doubtless other regularities exist in this large set of data, as yet unrecognized. The reader is invited to search these tables for additional interesting regularities.

It should be emphasized again, however, that these findings are experimental and tentative; the present author has not been able to find formal proofs. But the relative simplicity of these assertions suggests that they may well be amenable to proof or disproof.

8 Conclusions and future research

While these computational results and observations are a useful start, it is clear that a fuller understanding of the structure and behavior of Poisson polynomials will require additional effort. In particular, recall that the catalogued computations merely cover the cases $(x, y) = (p/s, q/s)$, where $1 \leq p, q < s/2$, with $\gcd(p, q, s) = 1$, for $10 \leq s \leq 25$ and also for $s = 26, 28, 30, 32, 34, 36$ and 40. To obtain further confidence in the three assertions mentioned in the previous section, these limits should be increased, which will require substantial additional computation. In addition, several questions still remain, such as what regularity is exhibited by the exceptional cases, and, even more intriguingly, why certain sets of cases share the same minimal polynomial, as noted in Tables 4 through 7.

Note also that all of the research results and analyses to date are for the simple two-dimensional cases, namely $\phi_2(x, y)$ and $\psi_2(x, y)$. The study [5] included brief mention of three and higher dimensions, but at present this research is hampered by the lack of rapid and universally applicable computational algorithms for higher dimensions, analogous to those listed above for $\phi_2(x, y)$ and $\psi_2(x, y)$ in Section 2. Clearly one important next step in this research is to re-examine earlier studies and elsewhere for hints to computational techniques and theoretical results applicable to higher dimensions.

The arbitrary precision package employed in this study [2] is thread-safe, and the multi-pair PSLQ algorithm exhibits moderate parallelism for large problems. Speedups of 12X on a 16-core system have been achieved. But the question remains whether some other algorithm is more effective for these very large problems and precision levels, and further is amenable to highly parallel processing. Note that simply converting a straightforward full-precision implementation of an algorithm such as multipair PSLQ for parallel processing, which may achieve large parallel speedups, is not helpful, since performance timings in parallel computing must be compared to the most efficient practical serial algorithm (which in this case is an algorithm, such as the three-level multipair PSLQ algorithm, that performs nearly all iterations in double precision); otherwise parallel speedups are illusory.

References

[1] J. Ablinger, J. Blumlein and C. Schneider, "Harmonic sums and polylogarithms generated by cyclotomic polynomials," *Journal of Mathematical Physics*, vol. 52 (2011), 102301.

[2] D. H. Bailey, "MPFUN2020: A new thread-safe arbitrary precision package," manuscript, 7 Apr 2024, `https://www.davidhbailey.com/dhbpapers/mpfun2020.pdf`.

[3] D. H. Bailey, "Finding large Poisson polynomials using four-level variable precision," *Correctness 2021: 5th International Workshop on Software Correctness for HPC Applications*, 10 Nov 2021, `https://www.davidhbailey.com/dhbpapers/correct-2021.pdf`.

[4] D. H. Bailey, "Large Poisson polynomials: Computation, results and analysis," manuscript, 8 Jul 2023, available at `https://www.davidhbailey.com/dhbpapers/poisson-2023.pdf`.

[5] D. H. Bailey, J. M. Borwein, R. E. Crandall and I. J. Zucker, "Lattice sums arising from the Poisson equation," *Journal of Physics A: Mathematical and Theoretical*, vol. 46 (2013), 115201.

[6] D. H. Bailey, J. M. Borwein, J. Kimberley and W. Ladd, "Computer discovery and analysis of large Poisson polynomials," *Experimental Mathematics*, vol. 26 (27 Aug 2016), 349–363.

[7] D. H. Bailey and D. J. Broadhurst, "Parallel integer relation detection: Techniques and applications," *Mathematics of Computation*, vol. 70, 236 (2000), 1719–1736.

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

[9] J. M. Borwein, M. L. Glasser, R. C. McPhedran, J. G. Wan and I. J. Zucker, *Lattice Sums Then and Now*, Cambridge University Press, Cambridge, U.K., 2013.

[10] J. Chen, Y. Feng and W. Wu, "Two variants of HJLS-PSLQ with applications," *Proceedings of SNC'14*, 88–96, Shanghai, China, 2014.

[11] R. Crandall, "The Poisson equation and 'natural' Madelung constants," manuscript, 2012, available at `https://www.davidhbailey.com/dhbpapers/REC-PoissonAnalytic.pdf`.

[12] L. Fousse, G. Hanrot, V. Lefevre, P. Pelissier and P. Zimmermann, "MPFR: A multiple-precision binary floating-point library with correct rounding," *ACM Transactions on Mathematical Software*, vol. 33 (2 Jun 2007), 13–es; code available at `https://www.mpfr.org`.

[13] S. Kanemitsu, Y. Tanigawa, H. Tsukada and M. Yoshimoto, "Crystal symmetry viewed as zeta symmetry," p. 91–130 in T. Aoki, S. Kanemitsu, M. Nakahara and Y. Ohno, eds., *Zeta Functions, Topology and Quantum Physics*, Springer, New York, 2005 (rep. 2010).

[14] S. Kanemitsu and H. Tsukada, "Crystal symmetry viewed as zeta symmetry II," 275–292 in K. Alladi, J. R. Klauder and C. R. Rao, eds., *The Legacy of Alladi Ramakrishnan in the Mathematical Sciences*, Springer, New York, 2010.

[15] N. Mani, "Palindrome degree reduction," manuscript, 19 Dec 2015, `https://www.davidhbailey.com/dhbpapers/mani-palindrome.pdf`.

[16] Lord Rayleigh, "On the influence of obstacles arranged in rectangular order upon the properties of a medium," *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, vol. 34 (1892), pg. 481–502, available at `https://www.tandfonline.com/doi/abs/10.1080/14786449208620364`.

[17] G. Savin and D. Quarfoot, "On attaching coordinates of Gaussian prime torsion points of $y^2 = x^3 + x$ to $Q(i)$," March 2010, `https://www.math.utah.edu/~savin/EllipticCurvesPaper.pdf`.

[18] I. J. Zucker, "The exact evaluation of some new lattice sums," *Symmetry*, vol. 9 (2017), pg. 314.

9 Appendix: PSLQ and multi-pair PSLQ

Given an input vector $x = (x_j, 1 \le j \le n)$ of real numbers, typically given as high-precision floating-point values, the PSLQ and multipair PSLQ integer relation algorithms attempt to find a nontrivial vector of integers (a_j), if one exists, such that

$$a_1 x_1 + a_2 x_2 + \cdots + a_n x_n = 0, \tag{17}$$

to within the numeric precision being employed. The name "PSLQ" derives from its usage of a partial sum of squares vector and an LQ (lower-diagonal-orthogonal) matrix factorization.

The multipair PSLQ algorithm attempts to perform multiple iterations of the standard PSLQ algorithm in a single iteration. It is moderately parallelizable and has the added benefit of running faster, even on a single processor, and of being even more efficient with precision: in most cases it can detect a relation when the numeric precision is only a few percent higher than a minimum bound [7]. More complete details on these algorithms, including details on multilevel precision implementations, are given in [7] and [3].

9.1 The standard PSLQ algorithm

Let x be the n-long input real vector, let nint denote the nearest integer function (for exact half-integer values, define nint to be the integer with greater absolute value) and select $\gamma \ge \sqrt{4/3}$ (we typically select $\gamma = \sqrt{4/3}$, since this is the most efficient with precision).

Initialize:

1. For $j := 1$ to n: for $i := 1$ to n: if $i = j$ then set $A_{ij} := 1$ and $B_{ij} := 1$ else set $A_{ij} := 0$ and $B_{ij} := 0$; endfor; endfor.
2. For $k := 1$ to n: set $s_k := \sqrt{\sum_{j=k}^n x_j^2}$; endfor. Set $t = 1/s_1$. For $k := 1$ to n: set $y_k := t x_k$; $s_k := t s_k$; endfor.
3. Initial H: For $j := 1$ to $n-1$: for $i := 1$ to $j-1$: set $H_{ij} := 0$; endfor; set $H_{jj} := s_{j+1}/s_j$; for $i := j+1$ to n: set $H_{ij} := -y_i y_j/(s_j s_{j+1})$; endfor; endfor.
4. Reduce H: For $i := 2$ to n: for $j := i-1$ to 1 step -1: set $t := \mathrm{nint}(H_{ij}/H_{jj})$; and $y_j := y_j + t y_i$; for $k := 1$ to j: set $H_{ik} := H_{ik} - t H_{jk}$; endfor; for $k := 1$ to n: set $A_{ik} := A_{ik} - t A_{jk}$ and $B_{kj} := B_{kj} + t B_{ki}$; endfor; endfor; endfor.

Iteration: Repeat the following steps until precision has been exhausted or a relation has been detected.

1. Select m such that $\gamma^i |H_{ii}|$ is maximal when $i = m$.
2. Exchange the entries of y indexed m and $m+1$, the corresponding rows of A and H, and the corresponding columns of B.
3. Remove corner on H diagonal: If $m \le n-2$ then set $t_0 := \sqrt{H_{mm}^2 + H_{m,m+1}^2}$, $t_1 := H_{mm}/t_0$ and $t_2 := H_{m,m+1}/t_0$; for $i := m$ to n: set $t_3 := H_{im}$, $t_4 := H_{i,m+1}$, $H_{im} := t_1 t_3 + t_2 t_4$ and $H_{i,m+1} := -t_2 t_3 + t_1 t_4$; endfor; endif.
4. Reduce H: For $i := m+1$ to n: for $j := \min(i-1, m+1)$ to 1 step -1: set $t := \mathrm{nint}(H_{ij}/H_{jj})$ and $y_j := y_j + t y_i$; for $k := 1$ to j: set $H_{ik} := H_{ik} - t H_{jk}$; endfor; for $k := 1$ to n: set $A_{ik} := A_{ik} - t A_{jk}$ and $B_{kj} := B_{kj} + t B_{ki}$; endfor; endfor; endfor.
5. Norm bound: Compute $M := 1/\max_j |H_{jj}|$. Then there can exist no relation vector whose Euclidean norm is less than M.

6. Termination test: If the largest entry of A or B 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, and the dynamic range between that smallest entry and the largest entry of y is sufficiently large (say at least 30 orders of magnitude), then a relation may have been detected and is given in the corresponding row of B.

9.2 The multipair PSLQ algorithm

Let x be the n-long input real vector, let nint denote the nearest integer function as before and select $\gamma \geq \sqrt{4/3}$ (we typically select $\gamma = \sqrt{4/3}$, since this is the most efficient with precision) and $\beta = 0.4$.

Initialize:

1. For $j := 1$ to n: for $i := 1$ to n: if $i = j$ then set $A_{ij} := 1$ and $B_{ij} := 1$ else set $A_{ij} := 0$ and $B_{ij} := 0$; endfor; endfor.

2. For $k := 1$ to n: set $s_k := \sqrt{\sum_{j=k}^{n} x_j^2}$; endfor; set $t = 1/s_1$; for $k := 1$ to n: set $y_k := tx_k$; $s_k := ts_k$; endfor.

3. Initial H: For $j := 1$ to $n - 1$: for $i := 1$ to $j - 1$: set $H_{ij} := 0$; endfor; set $H_{jj} := s_{j+1}/s_j$; for $i := j + 1$ to n: set $H_{ij} := -y_i y_j/(s_j s_{j+1})$; endfor; endfor.

Iteration: Repeat the following steps until precision has been exhausted or a relation has been detected.

1. Sort the entries of the $(n - 1)$-long vector $\{\gamma^i |H_{ii}|\}$ in decreasing order, producing the sort indices.

2. Beginning at the sort index m_1 corresponding to the largest $\gamma^i |H_{ii}|$, select pairs of indices $(m_i, m_i + 1)$, where m_i is the sort index. If at any step either m_i or $m_i + 1$ has already been selected or is outside the array bound, pass to the next index in the list. Continue until either βn pairs have been selected, or the list is exhausted. Let p denote the number of pairs actually selected in this manner.

3. For $i := 1$ to p, exchange the entries of y indexed m_i and $m_i + 1$, and the corresponding rows of A, B and H; endfor.

4. Remove corners on H diagonal: For $i := 1$ to p: if $m_i \leq n - 2$ then set $t_0 := \sqrt{H^2_{m_i,m_i} + H^2_{m_i,m_i+1}}$, $t_1 := H_{m_i,m_i}/t_0$ and $t_2 := H_{m_i,m_i+1}/t_0$; for $i := m_i$ to n: set $t_3 := H_{i,m_i}$; $t_4 := H_{i,m_i+1}$; $H_{i,m_i} := t_1 t_3 + t_2 t_4$; and $H_{i,m_i+1} := -t_2 t_3 + t_1 t_4$; endfor; endif; endfor.

5. Reduce H: For $i := 2$ to n: for $j := 1$ to $n - i + 1$: set $l := i + j - 1$; for $k := j + 1$ to $l - 1$: set $H_{lj} := H_{lj} - T_{lk} H_{kj}$; endfor; set $T_{lj} := \mathrm{nint}(H_{lj}/H_{jj})$ and $H_{lj} := H_{lj} - T_{lj} H_{jj}$; endfor; endfor.

6. Update y: For $j := 1$ to $n - 1$: for $i := j + 1$ to n: set $y_j := y_j + T_{ij} y_i$; endfor; endfor.

7. Update A and B: For $k := 1$ to n: for $j := 1$ to $n - 1$: for $i := j + 1$ to n: set $A_{ik} := A_{ik} - T_{ij} A_{jk}$ and $B_{jk} := B_{jk} + T_{ij} B_{ik}$; endfor; endfor; endfor.

8. Norm bound: Compute $M := 1/\max_j |H_{jj}|$. Then there can exist no relation vector whose Euclidean norm is less than M.

9. Termination test: If the largest entry of A or B 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, and the dynamic range between that smallest entry and the largest entry of y is sufficiently large (say at least 30 orders of magnitude), then a relation may have been detected and is given in the corresponding row of B.

$s = 10$	(1, 1), (3, 3)
	(1, 2), (3, 4)
	(1, 4), (2, 3)
	(1, 3)
$s = 11$	(1, 2), (1, 5), (2, 4), (3, 4), (3, 5)
	(1, 1), (2, 2), (3, 3), (4, 4), (5, 5)
	(1, 3), (1, 4), (2, 3), (2, 5), (4, 5)
$s = 12$	(1, 2), (1, 4), (2, 5), (4, 5)
	(1, 5)
	(2, 3), (3, 4)
	(1, 3), (3, 5)
	(1, 1), (5, 5)
$s = 13$	(1, 2), (1, 6), (2, 4), (3, 5), (3, 6), (4, 5)
	(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6)
	(1, 5), (2, 3), (4, 6)
	(1, 3), (1, 4), (2, 5), (2, 6), (3, 4), (5, 6)
$s = 14$	(1, 6), (2, 5), (3, 4)
	(1, 2), (1, 4), (2, 3), (3, 6), (4, 5), (5, 6)
	(1, 3), (1, 5), (3, 5)
	(1, 1), (3, 3), (5, 5)
$s = 15$	(1, 4), (2, 7)
	(1, 2), (1, 7), (2, 4), (4, 7)
	(1, 6), (2, 3), (3, 7), (4, 6)
	(1, 1), (2, 2), (4, 4), (7, 7)
	(3, 5), (5, 6)
	(1, 3), (2, 6), (3, 4), (6, 7)
	(1, 5), (2, 5), (4, 5), (5, 7)
$s = 16$	(1, 1), (3, 3), (5, 5), (7, 7)
	(1, 7), (3, 5)
	(1, 4), (3, 4), (4, 5), (4, 7)
	(1, 2), (1, 6), (2, 3), (2, 5), (2, 7), (3, 6), (5, 6), (6, 7)
	(1, 3), (1, 5), (3, 7), (5, 7)
$s = 17$	(1, 2), (1, 8), (2, 4), (3, 6), (3, 7), (4, 8), (5, 6), (5, 7)
	(1, 5), (1, 7), (2, 3), (2, 7), (3, 4), (4, 6), (5, 8), (6, 8)
	(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7), (8, 8)
	(1, 4), (2, 8), (3, 5), (6, 7)
	(1, 3), (1, 6), (2, 5), (2, 6), (3, 8), (4, 5), (4, 7), (7, 8)
$s = 18$	(1, 3), (1, 5), (1, 7), (3, 5), (3, 7), (5, 7)
	(1, 2), (1, 4), (2, 5), (4, 7), (5, 8), (7, 8)
	(1, 1), (5, 5), (7, 7)
	(1, 8), (2, 7), (4, 5)
	(1, 6), (2, 3), (3, 4), (3, 8), (5, 6), (6, 7)
$s = 19$	(1, 2), (1, 9), (2, 4), (3, 6), (3, 8), (4, 8), (5, 7), (5, 9), (6, 7)
	(1, 7), (1, 8), (2, 3), (2, 5), (3, 5), (4, 6), (4, 9), (6, 9), (7, 8)
	(1, 3), (1, 6), (2, 6), (2, 7), (3, 9), (4, 5), (4, 7), (5, 8), (8, 9)
	(1, 4), (1, 5), (2, 8), (2, 9), (3, 4), (3, 7), (5, 6), (6, 8), (7, 9)
	(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7), (8, 8), (9, 9)

Table 4: Each row gives cases $(p/s, q/s)$ that share the same minimal polynomial; $10 \leq s \leq 19$

$s = 20$	(1, 5), (3, 5), (5, 7), (5, 9)
	(1, 4), (1, 6), (2, 3), (2, 7), (3, 8), (4, 9), (6, 9), (7, 8)
	(1, 2), (1, 8), (2, 9), (3, 4), (3, 6), (4, 7), (6, 7), (8, 9)
	(2, 5), (4, 5), (5, 6), (5, 8)
	(1, 3), (1, 7), (3, 9), (7, 9)
	(1, 9), (3, 7)
	(1, 1), (3, 3), (7, 7), (9, 9)
$s = 21$	(1, 6), (2, 9), (3, 4), (3,10), (5, 9), (6, 8)
	(1, 2), (1,10), (2, 4), (4, 8), (5, 8), (5,10)
	(1, 9), (2, 3), (3, 5), (4, 6), (6,10), (8, 9)
	(1, 8), (2, 5), (4,10)
	(1, 1), (2, 2), (4, 4), (5, 5), (8, 8), (10,10)
	(1, 7), (2, 7), (4, 7), (5, 7), (7, 8), (7,10)
	(1, 3), (2, 6), (3, 8), (4, 9), (5, 6), (9,10)
	(1, 4), (1, 5), (2, 8), (2,10), (4, 5), (8,10)
	(3, 7), (6, 7), (7, 9)
$s = 22$	(1, 2), (1, 6), (2, 7), (3, 4), (3, 6), (4, 9), (5, 8), (5,10), (7, 8), (9,10)
	(1,10), (2, 9), (3, 8), (4, 7), (5, 6)
	(1, 3), (1, 5), (1, 7), (1, 9), (3, 5), (3, 7), (3, 9), (5, 7), (5, 9), (7, 9)
	(1, 4), (1, 8), (2, 3), (2, 5), (3,10), (4, 5), (6, 7), (6, 9), (7,10), (8, 9)
	(1, 1), (3, 3), (5, 5), (7, 7), (9, 9)
$s = 23$	(1, 2), (1,11), (2, 4), (3, 6), (3,10), (4, 8), (5, 9), (5,10), (6,11), (7, 8), (7, 9)
	(1, 7), (1,10), (2, 3), (2, 9), (3, 7), (4, 5), (4, 6), (5,11), (6, 9), (8,10), (8,11)
	(1, 1), (2, 2), (3, 3), (4, 4), (5, 5), (6, 6), (7, 7), (8, 8), (9, 9), (10,10), (11,11)
	(1, 3), (1, 8), (2, 6), (2, 7), (3, 9), (4, 9), (4,11), (5, 6), (5, 8), (7,10), (10,11)
	(1, 4), (1, 6), (2, 8), (2,11), (3, 5), (3,11), (4, 7), (5, 7), (6,10), (8, 9), (9,10)
	(1, 5), (1, 9), (2, 5), (2,10), (3, 4), (3, 8), (4,10), (6, 7), (6, 8), (7,11), (9,11)
$s = 24$	(1, 7), (5,11)
	(1,11), (5, 7)
	(1, 9), (3, 5), (3,11), (7, 9)
	(3, 4), (3, 8), (4, 9), (8, 9)
	(1, 1), (5, 5), (7, 7), (11,11)
	(1, 3), (3, 7), (5, 9), (9,11)
	(1, 4), (1, 8), (4, 5), (4, 7), (4,11), (5, 8), (7, 8), (8,11)
	(1, 6), (5, 6), (6, 7), (6,11)
	(1, 5), (7,11)
	(1, 2), (1,10), (2, 5), (2, 7), (2,11), (5,10), (7,10), (10,11)
	(2, 3), (2, 9), (3,10), (9,10)
$s = 25$	(1, 4), (1, 6), (2, 8), (2,12), (3, 7), (3,12), (4, 9), (6,11), (7, 8), (9,11)
	(1, 9), (2, 3), (2, 7), (3, 8), (4, 6), (4,11), (6, 9), (7,12), (8,12)
	(1, 2), (1,12), (2, 4), (3, 6), (3,11), (4, 8), (6,12), (7, 9), (7,11), (8, 9)
	(1,10), (2, 5), (3, 5), (4,10), (5, 7), (5, 8), (5,12), (6,10), (9,10), (10,11)
	(1, 1), (2, 2), (3, 3), (4, 4), (6, 6), (7, 7), (8, 8), (9, 9), (11,11), (12,12)
	(1, 5), (2,10), (3,10), (4, 5), (5, 6), (5, 9), (5,11), (7,10), (8,10)
	(1, 7), (2,11), (3, 4), (6, 8)
	(1, 3), (1, 8), (2, 6), (2, 9), (3, 9), (4, 7), (4,12), (6, 7), (8,11), (11,12)

Table 5: Each row gives cases $(p/s, q/s)$ that share the same minimal polynomial; $20 \leq s \leq 25$

$s = 26$	(1, 4), (1,10), (2, 5), (2, 7), (3, 4), (3,12), (5, 6), (6,11), (7, 8), (8,11), (9,10), (9,12)
	(1, 2), (1, 6), (2, 9), (3, 6), (3, 8), (4, 5), (4,11), (5,10), (7,10), (7,12), (8, 9), (11,12)
	(1, 8), (2, 3), (4, 7), (5,12), (6, 9), (10,11)
	(1, 5), (3,11), (7, 9)
	(1, 1), (3, 3), (5, 5), (7, 7), (9, 9), (11,11)
	(1, 3), (1, 7), (1, 9), (1,11), (3, 5), (3, 7), (3, 9), (5, 7), (5, 9), (5,11), (7,11), (9,11)
	(1,12), (2,11), (3,10), (4, 9), (5, 8), (6, 7)
$s = 28$	(2, 7), (4, 7), (6, 7), (7, 8), (7,10), (7,12)
	(1,13), (3,11), (5, 9)
	(1, 2), (1,12), (2,13), (3, 6), (3, 8), (4, 5), (4, 9), (5,10), (6,11), (8,11), (9,10), (12,13)
	(1, 6), (1, 8), (2, 5), (2, 9), (3, 4), (3,10), (4,11), (5,12), (6,13), (8,13), (9,12), (10,11)
	(1, 5), (1,11), (3, 5), (3,13), (9,11), (9,13)
	(1, 7), (3, 7), (5, 7), (7, 9), (7,11), (7,13)
	(1, 4), (1,10), (2, 3), (2,11), (3,12), (4,13), (5, 6), (5, 8), (6, 9), (8, 9), (10,13), (11,12)
	(1, 1), (3, 3), (5, 5), (9, 9), (11,11), (13,13)
	(1, 3), (1, 9), (3, 9), (5,11), (5,13), (11,13)
$s = 30$	(1,14), (2,13), (4,11), (7, 8)
	(1, 3), (1, 7), (1,13), (3,11), (7, 9), (7,11), (9,13), (11,13)
	(3,10), (5, 6), (5,12), (9,10)
	(1,10), (2, 5), (4, 5), (5, 8), (5,14), (7,10), (10,11), (10,13)
	(1,11), (3, 5), (5, 9), (7,13)
	(1, 4), (2, 7), (8,13), (11,14)
	(1, 2), (1, 8), (2,11), (4, 7), (4,13), (7,14), (8,11), (13,14)
	(1, 6), (2, 3), (3, 8), (4, 9), (6,11), (7,12), (9,14), (12,13)
	(1, 1), (7, 7), (11,11), (13,13)
	(1, 5), (1, 9), (3, 7), (3,13), (5, 7), (5,11), (5,13), (9,11)
	(1,12), (2, 9), (3, 4), (3,14), (6, 7), (6,13), (8, 9), (11,12)
$s = 32$	(1, 6), (1,10), (2, 3), (2, 5), (2,11), (2,13), (3,14), (5,14), (6, 7), (6, 9), (6,15), (7,10), (9,10), (10,15), (11,14), (13,14)
	(1, 5), (1,13), (3, 7), (3,15), (5, 7), (9,11), (9,13), (11,15)
	(1, 1), (3, 3), (5, 5), (7, 7), (9, 9), (11,11), (13,13), (15,15)
	(1, 7), (1, 9), (3, 5), (3,11), (5,13), (7,15), (9,15), (11,13)
	(1, 3), (1,11), (3, 9), (5, 9), (5,15), (7,11), (7,13), (13,15)
	(1, 8), (3, 8), (5, 8), (7, 8), (8, 9), (8,11), (8,13), (8,15)
	(1,15), (3,13), (5,11), (7, 9)
	(1, 2), (1,14), (2, 7), (2, 9), (2,15), (3, 6), (3,10), (5, 6), (5,10), (6,11), (6,13), (7,14), (9,14), (10,11), (10,13), (14,15)
	(1, 4), (1,12), (3, 4), (3,12), (4, 5), (4, 7), (4, 9), (4,11), (4,13), (4,15), (5,12), (7,12), (9,12), (11,12), (12,13), (12,15)

Table 6: Each row gives cases $(p/s, q/s)$ that share the same minimal polynomial; $26 \le s \le 32$

$s=34$	(1, 2), (1, 8), (2,13), (3, 6), (3,10), (4, 9), (4,15), (5, 6), (5,10), (7,12), (7,14), (8,13), (9,16), (11,12), (11,14), (15,16)
	(1, 5), (1, 7), (3,13), (3,15), (5, 9), (7,15), (9,11), (11,13)
	(1, 4), (2, 9), (3,12), (5,14), (6, 7), (8,15), (10,11), (13,16)
	(1, 1), (3, 3), (5, 5), (7, 7), (9, 9), (11,11), (13,13), (15,15)
	(1, 6), (1,14), (2, 5), (2,11), (3, 8), (3,16), (4, 5), (4, 7), (6,15), (7, 8), (9,10), (9,14), (10,13), (11,16), (12,13), (12,15)
	(1,10), (1,12), (2, 3), (2, 7), (3, 4), (4,11), (5, 8), (5,16), (6, 9), (6,13), (7,16), (8,11), (9,12), (10,15), (13,14), (14,15)
	(1,13), (3, 5), (7,11), (9,15)
	(1,16), (2,15), (3,14), (4,13), (5,12), (6,11), (7,10), (8, 9)
	(1, 3), (1, 9), (1,11), (1,15), (3, 7), (3, 9), (3,11), (5, 7), (5,11), (5,13), (5,15), (7, 9), (7,13), (9,13), (11,15), (13,15)
$s=36$	(1, 6), (1,12), (5, 6), (5,12), (6, 7), (6,11), (6,13), (6,17), (7,12), (11,12), (12,13), (12,17)
	(1,15), (3, 5), (3, 7), (3,17), (11,15), (13,15)
	(1, 2), (1,16), (2,17), (4, 7), (4,11), (5, 8), (5,10), (7,14), (8,13), (10,13), (11,14), (16,17)
	(1, 9), (5, 9), (7, 9), (9,11), (9,13), (9,17)
	(1, 4), (1,14), (2, 5), (2,13), (4,17), (5,16), (7, 8), (7,10), (8,11), (10,11), (13,16), (14,17)
	(1, 8), (1,10), (2, 7), (2,11), (4, 5), (4,13), (5,14), (7,16), (8,17), (10,17), (11,16), (13,14)
	(1, 1), (5, 5), (7, 7), (11,11), (13,13), (17,17)
	(2, 3), (2,15), (3, 4), (3, 8), (3,10), (3,14), (3,16), (4,15), (8,15), (10,15), (14,15), (15,16)
	(1, 5), (1, 7), (5,11), (7,13), (11,17), (13,17)
	(1,17), (5,13), (7,11)
	(2, 9), (4, 9), (8, 9), (9,10), (9,14), (9,16)
	(1,11), (1,13), (5, 7), (5,17), (7,17), (11,13)
	(1, 3), (3,11), (3,13), (5,15), (7,15), (15,17)
$s=40$	(1, 1), (3, 3), (7, 7), (9, 9), (11,11), (13,13), (17,17), (19,19)
	(1,15), (3, 5), (5,11), (5,19), (7,15), (9,15), (15,17)
	(2, 5), (2,15), (5, 6), (5,14), (5,18), (14,15), (15,18)
	(1,10), (3,10), (7,10), (9,10), (10,11), (10,13), (10,17), (10,19)
	(1, 7), (1,17), (3,11), (3,19), (7, 9), (9,17), (11,13), (13,19)
	(1, 6), (1,14), (2, 3), (2, 7), (2,13), (2,17), (3,18), (6,11), (6,19), (7,18), (9,14), (11,14), (13,18), (14,19), (17,18)
	(1,11), (3, 7), (9,19), (13,17)
	(1, 5), (3,15), (5, 7), (5, 9), (5,17), (11,15), (13,15), (15,19)
	(1, 8), (1,12), (3, 4), (3,16), (4, 7), (4,13), (4,17), (7,16), (8, 9), (8,11), (8,19), (9,12), (11,12), (12,19), (13,16), (16,17)
	(1, 3), (1,13), (3, 9), (7,19), (11,17), (17,19)
	(1, 9), (3,13), (7,17), (11,19)
	(1, 2), (1,18), (2, 9), (2,11), (2,19), (3, 6), (3,14), (6, 7), (6,13), (6,17), (7,14), (9,18), (11,18), (13,14), (14,17), (18,19)
	(4, 5), (4,15), (5, 8), (5,12), (5,16), (8,15), (12,15), (15,16)
	(1, 4), (1,16), (3, 8), (3,12), (4, 9), (4,11), (4,19), (7, 8), (7,12), (8,13), (8,17), (9,16), (11,16), (12,13), (12,17), (16,19)
	(1,19), (3,17), (7,13), (9,11)

Table 7: Each row gives cases $(p/s, q/s)$ that share the same minimal polynomial; $34 \leq s \leq 40$