Tetration

Mathematical Miniatures

Gottfried Helms - Univ Kassel 12 - 2007

Tetra-series

A systematic inconsistency in summing powertowers of increasing heights by serial or matrix-computation.

(update 7, 12.2.2008)

Abstract: In this article I describe the problem of the observed discrepancy, when the infinite alternating series of powertowers of increasing heights ("Tetra-series") are computed according to the matrix-method (using diagonalization for fractional iteration) compared to the evaluation of the partial sums of the powertowers using a summation-method, like for instance, Pari/GP-sumalt-function.

The discrepancy cannot be explained yet, and a set of approaches to understand the source of the discrepancy is presented. At least, there are some properties of these discrepancies found, which may direct into a direction for further research.

Contents:

1. Intro

Assume the following two (and mutual inverse) tetrational-functions:

(1.1.1.) $u_b(x) = b^x - 1$ $\qquad l_b(x) = \log(1+x)/\log(b)$

and their iterations as "*tower*"-functions with "*height*"-parameter h which I call "U-tetration" here:

(1.1.2.) $$ut_b(x,h) = \begin{cases} x & \text{if } h=0 \\ u_b(x)=b^x-1 & \text{if } h=1 \\ u_b(ut_b(x,h-1)) & \text{if } h>1 \end{cases} \qquad lt_b(x,h) = \begin{cases} x & \text{if } h=0 \\ l_b(x) & \text{if } h=1 \\ l_b(lt_b(x,h-1)) & \text{if } h>1 \end{cases}$$

> Originally I considered the more common "T-tetration" (I adapt the notation here)
>
> (1.1.3.) $t_b(x) = b^x$ $\qquad tl_b(x) = \log(x)/\log(b)$
> $tt_b(x,h) = tt_b(t_b(x),h-1)=\{b,x\}$^^$h$ $\qquad tlt_b(x,h) = tlt_b(tl_b(x),h-1)$
>
> but after I could show, that $tt_b()$ and $ut_b()$ are convertible by shift and rescale of x I discuss the U-tetration here, since its matrix-operator is triangular and thus better accessible (for $b=e=\exp(1)$ we can even use rational arithmetic).

In this article I compare ***two methods*** to compute the two infinite alternating sums

(1.1.4.) $$AU_b(x) = x - (b^x-1) + (b^{b^x-1}-1) - \dots = \sum_{h=0}^{inf} (-1)^h * ut_b(x,h)$$

(1.1.5.) $$AL_b(x) = x - l_b(x) + l_b(l_b(x)) - \dots = \sum_{h=0}^{inf} (-1)^h * lt_b(x,h)$$

which show a surprising and striking discrepancy for $AL_b(x)$, for which I don't have an explanation so far. I'll compare the "serial-computation" by Euler-summation using partial sums of the series of towers, and the "matrix-computation" for the summation via my matrix-method.

The following text is an approach to shed some light on the characteristic of this inconsistency, trying to find a way to explain, and possibly cure it. The inconsistency is "small", and also shows some systematic, so that there is a bit of hope to find some description of its properties and for its reason.

For descriptions of the involved matrix-operations see appendices ***3.1*** to ***3.3***. Note, that all matrices are assumed to have infinite size, and for practical purposes I use truncations of $size = 96 \times 96$.

2. The problem

2.1. The conjecture and the inconsistency

Recently I conjectured, based on consideration of the ***matrix-method*** for these computations, that

(2.1.1.) $AU_b(x) + AL_b(x) - x = 0$

for all b and x and computed example-values using the matrix-method.

However, one correspondent in a usenet-discussion-group pointed out, that my computations[1] did not agree with his serial computations, which were based on evaluation the ***partial sums*** and application of Cesaro-summation. Then I confirmed this finding with own computations, where I used Euler-summation based on the partial sums. I call the latter here the ***"serial"-method***.

So the "serial method" is to compute the individual u- or l-towers and their (alternating) sum using Euler-summation, the "matrix-method" is to compute that sums using matrices of coefficients of the involved powerseries $\boldsymbol{M_u}$ and $\boldsymbol{M_l}$. Note, that there is no obvious discrepancy of the two methods for computation of AU; only for AL, and this discrepancy is small – and comes out to be systematic, thus correctable.

Matrix-method

Here the matrix $\boldsymbol{M_u}$ represents the infinite alternating sum of powers of Stirling-matrices 2'nd kind, rowscaled by the logarithm of b, $\boldsymbol{S2_b}$, and the matrix $\boldsymbol{M_l}$ represents the infinite alternating sum of powers of Stirling-matrices 1'st kind, columnscaled by the reciprocal of the logarithm of b, $\boldsymbol{S1_b}$.

(2.1.2.) $S2_b = {}^dV(log(b))*S2$ $\qquad$ $S1_b = S1 * {}^dV(1/log(b))$

which provide the coefficients for the powerseries in x for $b^x\text{-}1$ and $log_b(1+x)$ respectively, and also are their mutual reciprocals ($\boldsymbol{S1_b * S2_b = I}$). Using their powers one can compute the according iterated function $ut_b(x,h)$ and $lt_b(x,h)$ correctly: since the matrices $\boldsymbol{S1}$, $\boldsymbol{S1_b}$, $\boldsymbol{S2}$, $\boldsymbol{S2_b}$ are triangular, the matrix-entries of their powers can be computed by finite sums (which occur in the required vector-products) and provide exact terms[2] for any finite integer power.

Then by linear combination of that matrix-powers I assume (and define):

(2.1.3.) $M_u = (I - S2_b + S2_b^2 - S2_b^3 + ... - ...)$
(2.1.4.) $M_l = (I - S1_b + S1_b^2 - S1_b^3 + ... - ...)$

and $\boldsymbol{M_u}$ and $\boldsymbol{M_l}$ can be computed in two ways, which give the same result

* using Euler-summation of partial sums for each matrix-entry separately (The progressions in the sequence of related entries is asymptotically geometric).

* I can also show, that these computations via infinite sums (evaluated for each matrix-entry separately) and the shortcut-formula for geometric series

(2.1.5.) $M_u = (I + S2_b)^{-1}$
(2.1.6.) $M_l = (I + S1_b)^{-1}$

agree to get the same final $\boldsymbol{M}$-matrices. (see Appendix ***3.2*** and ***3.3***)

---

[1] which actually used the T- instead of U-tetration, but that doesn't matter here

[2] as far as logarithms and their powers are assumed to be "exact". To prevent any possibility of error, we may use $b= exp(1)$, thus the $log(b)=1$ and we have exact rational entries in all involved matrices

So let us assume here, that the matrix-method for computation of $u_b(x)$ and $l_b(x)$ resp $ut_b(x,h)$ and $lt_b(x,h)$ are ***correct for a single tower*** of arbitrary height h. (I verified this also in some examples).

This is not so with the infinite alternating sums.

More precisely, it seems, that the results are consistent (with reasonable approximation in spite of the finite truncation of series) for the alternating sum $AU_b(x)$, but not for its "inverse", $AL_b(x)$.

From matrix-algebra, $\mathbf{M_u} + \mathbf{M_l} = \mathbf{I}$, and thus the expected sum of the two series is

(2.1.7.) $AU_b(x) + AL_b(x) - x = 0$ *(matrix-computation)*

This result appears indeed, as expected, if evaluated using the matrix-method only.

Also apparently

(2.1.8.) $AU_b(x)$ *(serial-computation)* $= AU_b(x)$ *(matrix-computation)*

seem to occur with arbitrarily well approximation.

So the said inconsistency seems to be based in some special property of the $AL()$-sum only, since:

(2.1.9.) $AL_b(x)$ *(by serial-computation)* $\neq AL_b(x)$ *(by matrix-computation)*

and hence the discrepancy:

(2.1.10.) $AU_b(x) + AL_b(x) - x \neq 0$ *(serial-computation)*

The hypothese and the problem may be expressed in a more concise way (neglecting here, that this notation would change the order of summation):

Definition:

(2.1.11.)
$$\begin{aligned} AL_b(x) + AU_b(x) - x &= \ldots - \ldots + lt_b(x,2) - lt_b(x,1) + x \\ &\quad + x - ut_b(x,1) + ut_b(x,2) - \ldots + \ldots \\ &\quad - x \\ &= \ldots - \ldots + ut_b(x,-2) - ut_b(x,-1) + ut_b(x,0) - ut_b(x,1) + ut_b(x,2) - \ldots + \ldots \end{aligned}$$

(2.1.12.)
$$AL_b(x) + AU_b(x) - x = \sum_{h=-inf}^{inf} ut_b(x,h) = d_b(x)$$

Hypothese, which fits the computations by the matrix-method:

(2.1.13.) $d_b(x) = 0$

Observation by serial summation

(2.1.14.) $d_b(x) \neq 0$

where also the discrepancy occurs only due to the infinite alternating sum $AL_b(x)$ *(serial-computation)*.

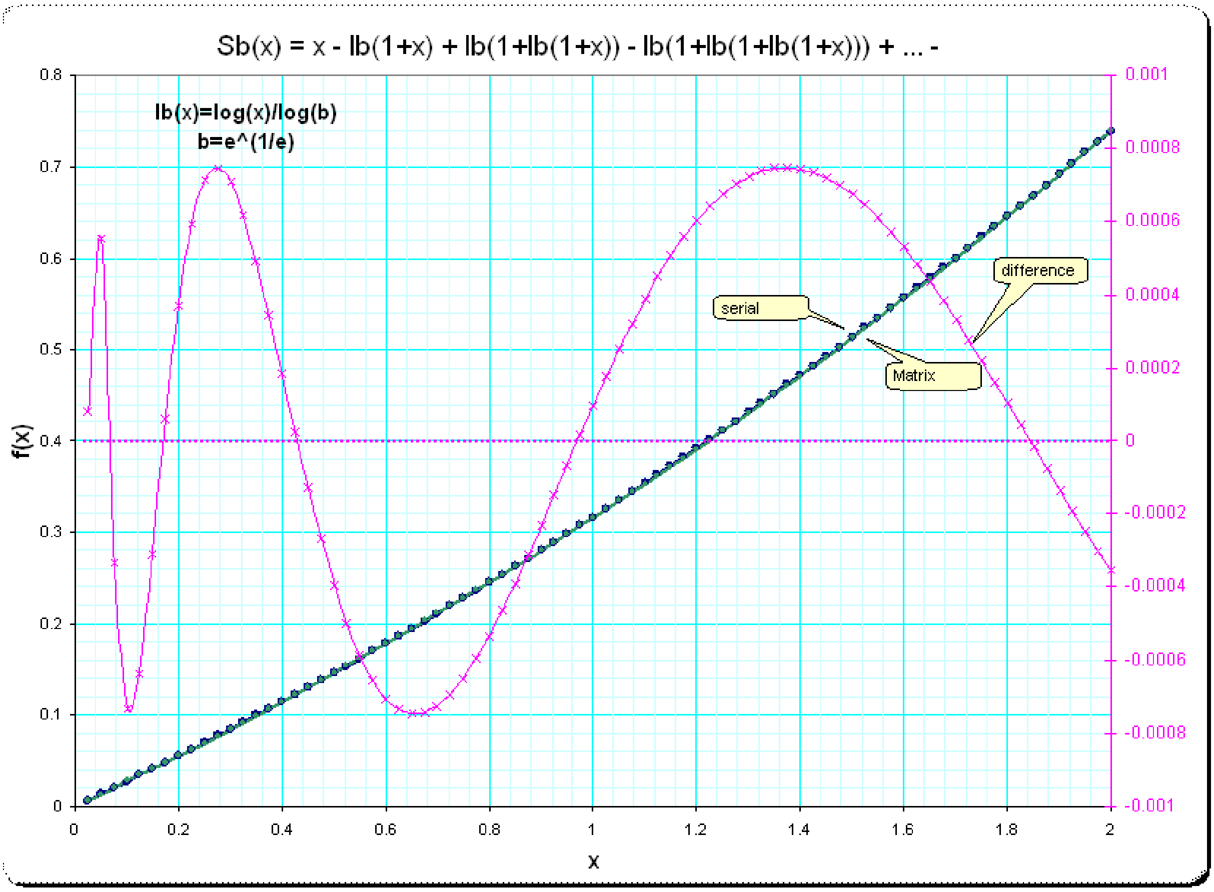
2.2. Some results

First I found, that the inconsictency/deviation depends on x, different for each base b.

Second, the deviation seem to be small and also periodical with some sinusoidal effect.

Observation 1

An example, using $b=e^{1/e}$, and some x between $0<x<2$. The curves for serial and for matrix-computation seem to match within the graphical resolution of the plot, but there is a curve of small differences, which is somehow sinusoidal, having increasing wavelength. The scale for the difference-curve is at the right side of the plot:



From another plot one may assume, that the apparently diminuished value of the first (plotted) local maximum at $x\sim 0.05$ is due to missing density of x-steps in that region – there seem to be a true local maximum of expected value neighboured to this point, instead.

The observation, that the ***differences are not arbitrary***, gives some hope, that the matrix-method is not completely useless for this series, but needs only a certain correction, which captures this sinusoidal effect.

Observation 2

Next idea was then, to check, whether the deviances occur periodically when x was replaced by $l_b(x)$, $l_b(l_b(x))$ and so on for $lt_b(x,h)$ with increasing h. (Since for the serial version I can only sum using integer iterates h of an initial value, the problem is restricted to that cases)

Here I got the first interesting result: taking one x_0, for the first few $x_h = lt_b(x_0,h)$ the *same* deviances occur peridocially with *alternating* signs. That means, that for the partial series

$$AL_b(x) = lt_b(x,0) - lt_b(x,1) + lt_b(x,2) - lt_b(x,3) + \dots - \dots$$
$$AL_b(lt_b(x,1)) = lt_b(x,1) - lt_b(x,2) + lt_b(x,3) + \dots - \dots$$
$$AL_b(lt_b(x,2)) = lt_b(x,2) - lt_b(x,3) + \dots - \dots$$

the differences

$$AL_b(lt_b(x,h)) \text{ (serial computation) } - AL_b(lt_b(x,h)) \text{ (matrix-computation) } = diff_b(x)$$

were the same for each h (except for their sign).

In the following table I document this (including the same for $AU_b(x_h)$, which seem to agree with both methods for the checked h and the used x.

To have converging series in both ways I used the base $b=1.3$ here.

Table 2.2.1

b=1.3	*using matrix* $\mathbf{M_l}$; *serial:* $AL_b(lt_b(x,h))$			*using matrix* $\mathbf{M_u}$; *serial:* $AU_b(lt_b(x,h))$		
x=0.1	**Matrix**	**Serial**	**diff**	**Matrix**	**Serial**	**diff**
h=0	0.0210405	0.0261459	-0.00510539	0.0789595	0.0789595	1.57901E-188
h=1	0.0789595	0.0738541	0.00510539	0.284315	0.284315	9.88668E-135
h=2	0.284315	0.289420	-0.00510539	0.896827	0.896827	1.58715E-85
h=3	0.896827	0.891721	0.00510539	2.07556	2.07556	6.35219E-47
h=4	2.07556	2.08067	-0.00510539	3.18189	3.18189	6.57891E-23
h=5	3.18189	3.17678	0.00510539	3.80753	3.80753	1.07150E-10
h=6	*3.80752*	3.81263	*-0.00511031*	*4.11321*	4.11321	*0.00000491495*
h=7	*4.11318*	4.10810	*0.00507446*	*4.22781*	4.22778	*0.0000310898*
x=0.2						
h=0	0.0426027	0.0304611	0.0121416	0.157397	0.157397	1.26643E-159
h=1	0.157397	0.169539	-0.0121416	0.537520	0.537520	1.13926E-107
h=2	0.537520	0.525379	0.0121416	1.47355	1.47355	2.76915E-63
h=3	1.47355	1.48570	-0.0121416	2.72785	2.72785	2.20361E-32
h=4	2.72785	2.71570	0.0121416	3.55703	3.55703	2.69669E-15
h=5	*3.55703*	3.56918	*-0.0121419*	*4.01184*	4.01184	*0.000000360360*
h=6	*4.01142*	3.99969	*0.0117312*	*4.17618*	4.17577	*0.000410345*
h=7	*4.17670*	4.18791	*-0.0112110*	*4.27664*	4.27757	*-0.000930386*

The red marked entries are not reliable due to difficult conditions for Euler-summation in that cases. The other differences in the Mu-column seem to be numerical errors due to the finite truncation of the matrices and the increasing heights, which need high orders of Euler-summation. See Appendix 3.3

In the following table I compare the theoretical identity, based on matrix-algebra-properties. The sum of the two alternating series $AL_b(x) + AU_b(x)$ should add up to x or more general

Recall the hypothese from matrix-identities:

$$AL_b(lt_b(x,h)) + AU_b(lt_b(x,h)) - lt_b(x,h) = d_b(x) \qquad \text{(2.2.1.)}$$
$$d_b(x) = 0 \quad \text{by hypothesis}$$

The matrix-computation fits well for several heights, and for the serial computation the deviance is nicely constant over all heights aside of the changing sign:

x_0=0.1	using M_l + M_u AL + AU − $lt_b(x_0,h)$	serial AL + AU −lt_b(x0,h)
h=0	-7.01691E-203	0.00510539
h=1	-1.02064E-202	-0.00510539
h=2	-2.04128E-202	0.00510539
h=3	6.72398E-199	-0.00510539
h=4	-1.49034E-174	0.00510539
h=5	-8.80295E-163	-0.00510539
h=6	*-0.000000119785*	0.00510539
h=7	*0.00000352142*	-0.00510539

However, this result is not really surprising, since from the construction of the formula

$$AL_b(x) + AU_b(x) - x = \dots - \dots + lt_b(x,2) - lt_b(x,1) + lt_b(x,0)$$
$$+ ut_b(x,0) - ut_b(x,1) + ut_b(x,2) - \dots + \dots$$
$$- x$$
$$= \sum_{h=-inf}^{inf} ut_b(x,h) = d_b(x)$$

the replacement of x by $ut_b(x,h)$ (or $lt_b(x,h)$) means simply to move the center-point h some indexes to the left or to the right, and the result should always be the same.

Observation 3

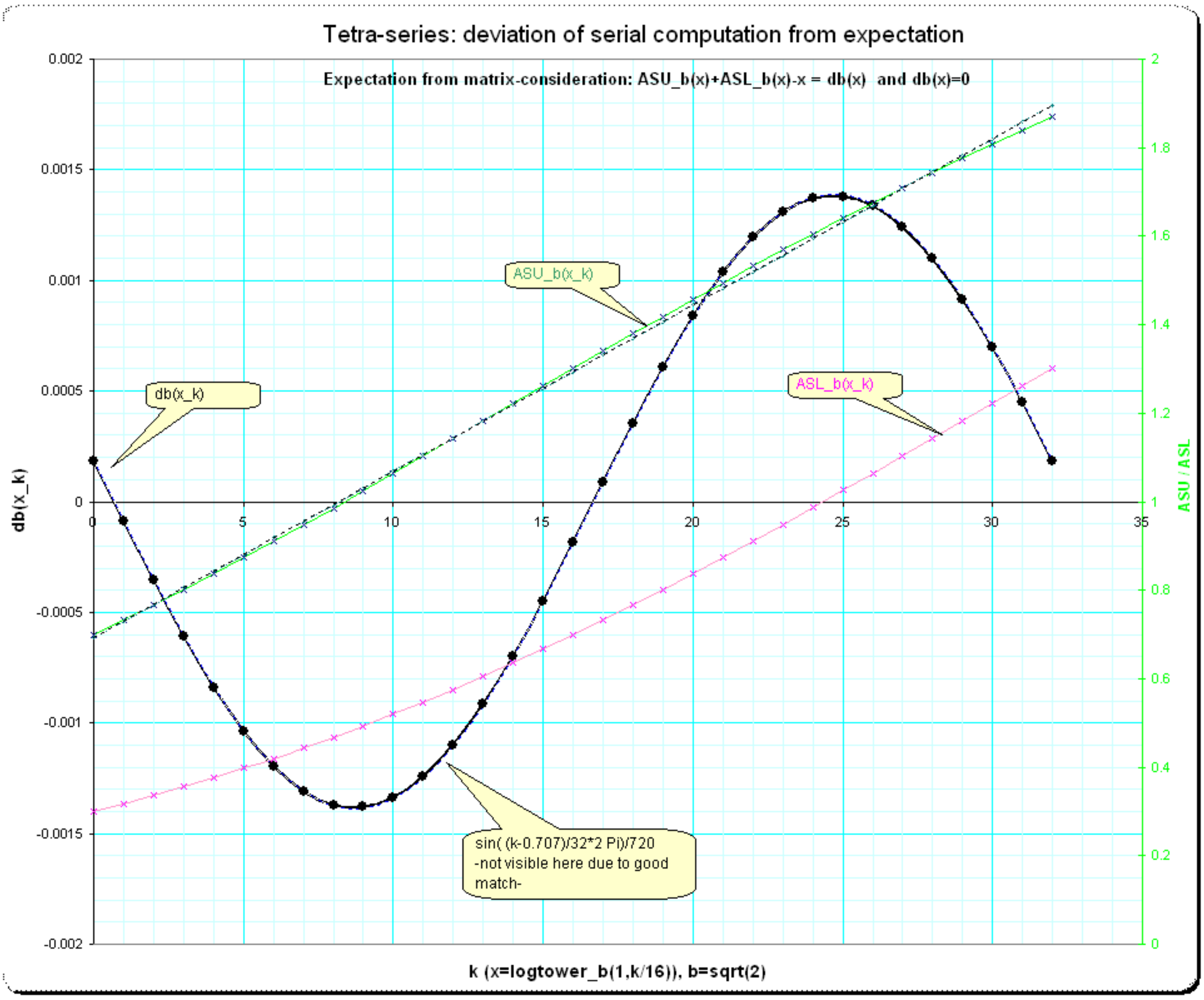
The sinusoidal form of the $d_b(x)$ – curve suggests to check, whether the increasing wavelength could be controlled by substituting some function of x for x instead. This is indeed the case.

If I define the fractional iterates of $lt_b(1,h)$, for instance h_k=0/16, 1/16,...,32/16 and use

$$x_k = lt_b(1,h_k)$$
$$AL_b(x_k) + AU_b(x_k) - x_k = d_b(x_k)$$

then I get a perfect sinus-curve for $d_b(x_k)$ if for the x-axis k is used instead of x_k..:

Here I used $b=sqrt(2)$ to have good convergence for both series AL and AU: (in the plot, which were made in connection with another article, I used notation „*ASU*“ for „*AU*“ and „*ASL*“ for „*AL*“)

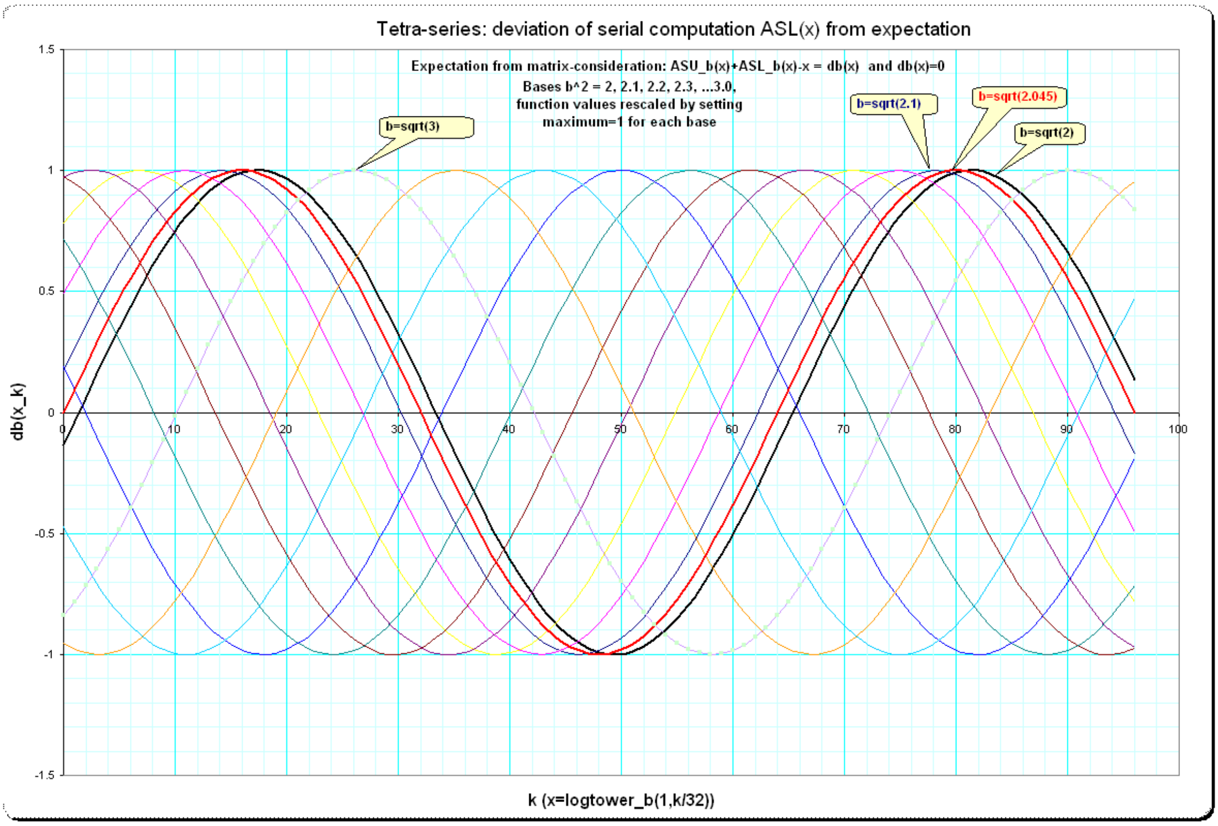


This result is encouraging. Another plot, where I also inserted different bases b, $2 \le b^2 \le 3$, shows the same sinus-form for all that bases.

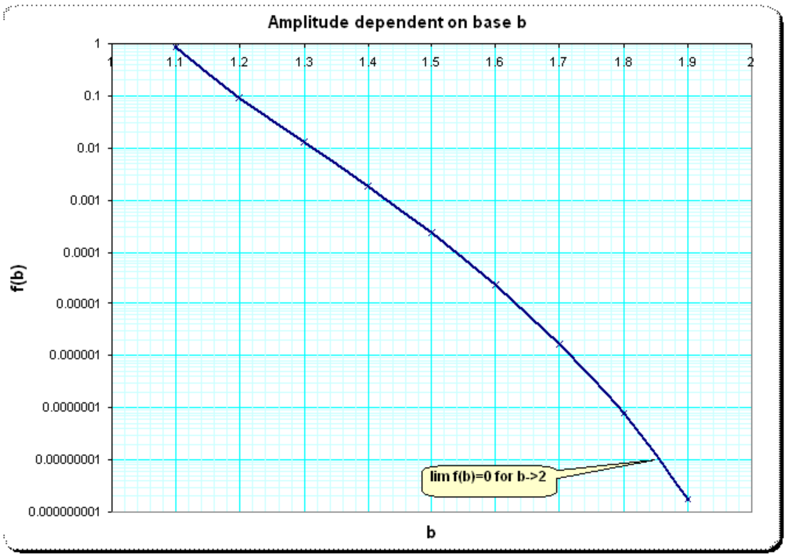
Note, that the same index k leads to different fractional iterates for each base, which define then the x_k-values for the individual bases.

Also note, that the amplitude-height of the curves was normalized by dividing by their empirical maximum so that the maximum of each curve is 1.

The plot suggests, that each base provides a certain shift of the phase; a base in the near of $b=1.43$ has zero-phase-shift (red curve).

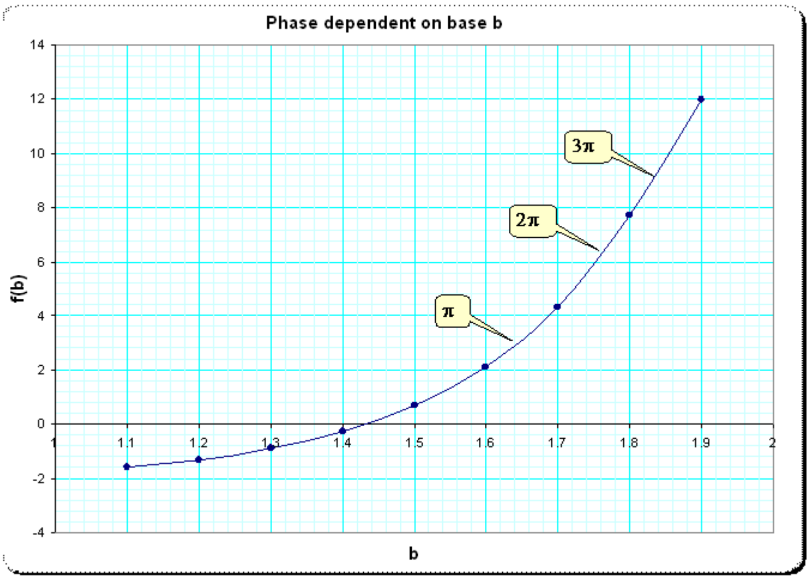


Here is a plot of the diminuishing amplitude, when $b\text{-}> 2$



At $b=2$ we have zero-amplitude, since for all x_k, which are the fractional iterates of $lt_2(1, h/32)$, the result is the same for all such h_k.

Here is a plot for the ***phase-shift*** of a slightly extended set of different bases b:



We see, that the zero-phase-shift is near the base $b=1.43$; but no more accurate computation was done yet.

Conclusion based on the above observations:

The values of $d_b(x)$, which deviate from the expectation of equalling zero seem to match a modified sine-function of an index k, when the x-values are taken from consecutive fractional iterates of a starting value x_0 (here starting-value $x_0=1$ was assumed)

Having the phase-shift and the scaling of the amplitude, we could thus express the $d_b(x)$-function by a sine-function and thus capture the deviance of $d_b(x)$ as a function of b and x.

This would then allow to correctly compute the values of the tetra-series; however it does not explain the reason, why the matrix-formula using the ***M***-matrices is not correct.

2.3. Quality of approximation of the powerseries based on M_l in the $x = x_{inf}$ - case

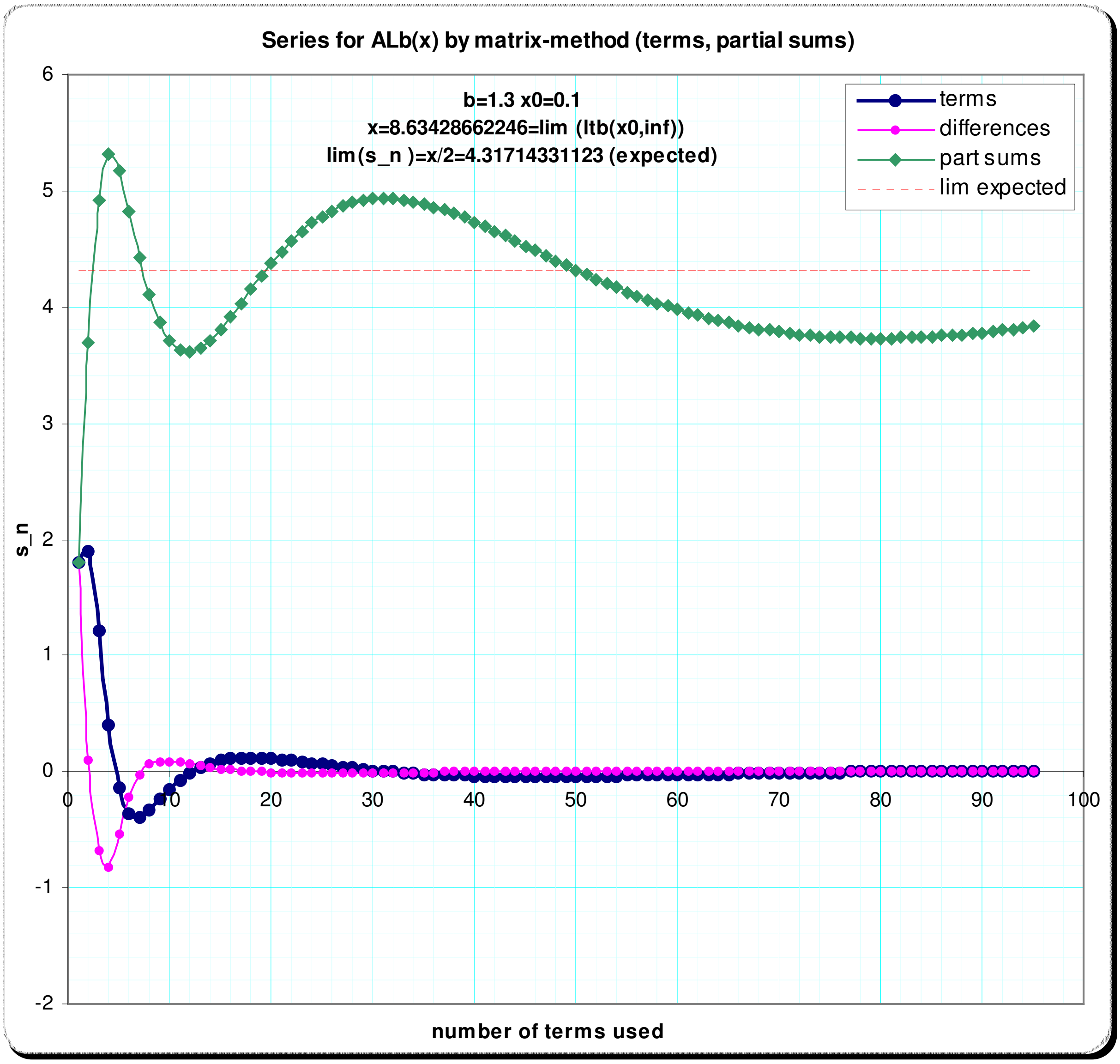
A systematic view into the matrices M_u and M_l might give an idea, where the reason for the discrepancy between matrix- and serial computation lies. Perhaps it is an approximation-problem or an effect of summing divergent series inplicitely with wrong results.

The entries of the second column in M_l are the coefficients of a powerseries in x for $AL_b(x)$.

If x equals the limit $x_{inf} = \lim_{h->inf} lt_b(x_0,h)$ (where this converges), x_{inf} is a fixed point, and the alternating series consists only of the values of x_{inf} with alternating signs. Thus the Cesaro- or Eulersummation must give the value

(2.3.1.) $$AL_b(x_{inf}) = x_{inf}/2$$

However, the terms of M_l, multiplied by the powers of x_{inf} do not converge at early stages, and the series is not well Euler-summable (if at all), so a comparision between the serial- and the matrix-method cannot be done here. Below is a plot, which shows the bad or non-convergence of the terms and the sequence of partial sums for matrix-dimension of $N=96$.



A symbolic description of M_l and M_u is relatively simple, since they are, using $u = log(b)$ for a base b

$$M_u = ({}^dV(u)*S2 + I)^{-1} \qquad\qquad M_l = (S1 * {}^dV(1/u) + I)^{-1}$$

We need only the second column; expressed symbolically this gives for the first few rows the following polynomials in u:

M_u	**M_l**
0	0
$\frac{1}{1!\,(1+u)}$	$\frac{u}{1!\,(1+u)}$
$\frac{-u^2}{2!\,(1+u)(1+u^2)}$	$\frac{u^2}{2!\,(1+u)(1+u^2)}$
$\frac{2*u^5-u^3}{3!\,(1+u)(1+u^2)(1+u^3)}$	$\frac{-2*u^5+u^3}{3!\,(1+u)(1+u^2)(1+u^3)}$
$\frac{-6*u^9+5*u^7+6*u^6-u^4}{4!\,(1+u)(1+u^2)(1+u^3)(1+u^4)}$	$\frac{6*u^9-5*u^7-6*u^6+u^4}{4!\,(1+u)(1+u^2)(1+u^3)(1+u^4)}$
$\frac{24*u^{14}-26*u^{12}-46*u^{11}-36*u^{10}+9*u^9+24*u^8+14*u^7-u^5}{5!\quad(1+u)(1+u^2)(1+u^3)(1+u^4)(1+u^5)}$	$\frac{-24*u^{14}+26*u^{12}+46*u^{11}+36*u^{10}-9*u^9-24*u^8-14*u^7+u^5}{5!\quad(1+u)(1+u^2)(1+u^3)(1+u^4)(1+u^5)}$

where each row has to be multiplied with a consecutive power of x, beginning at x^0

Example:

$$AU_b(x) = 1/(1+u)\ x/1!\ -\ u^2/((1+u)(1+u^2))\ x^2/2!\ +\ (2\ u^5 - u^3)/((1+u)(1+u^2)(1+u^3)\ x^3/3!\ + ...$$

One can immediately see, that these coefficents cancel, if M_u and M_l are added, except the second row, wich gives then $(1+u)/(1+u)/1! = 1$. So the only coefficient, which remains, is 1 at the first power of x, so the matrix-computation gives

$$AU_b(x) + AL_b(x) = x$$

also by this symbolic representation – with no obvious error.

The polynomials in u of each row r $(r=0..inf)$ has the highest power of u with of $(-1)^r/r*1/u$, (the row $r=0$ must be omitted) so asymptotically for large base b and thus large u ($=log(b)$) we should have

$$AU_b(x) \sim\ log(1+x) / u = log_b(1+x) \qquad \textit{for large } b$$

(but I didn't check this further)

The coefficients of u of the numerators in M_u only, as matrix show this image:

$$\begin{bmatrix} . & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\ 1 & . & . & . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & -1 & . & . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & -1 & . & 2 & . & . & . & . & . & . & . & . & . \\ . & . & . & . & -1 & . & 6 & 5 & . & -6 & . & . & . & . & . \\ . & . & . & . & . & -1 & . & 14 & 24 & 9 & -36 & -46 & -26 & . & 24 \end{bmatrix}$$

co(Mu)

where each column c is associated with the c'th power of u (c beginning at zero)

Using $u=1$ ($b=exp(1)$) we get the same coefficients as in chap **2.4** below(see "rowsums" in **2.4.2**)

$$AU_e(x) = 1/2\ x\ -\ 1/8\ x^2\ +\ 1/48\ x^3 + 1/96\ x^4 \ ...$$

2.4. A version, where $AL_b(x)$ is expressed by a series using values of the Dirichlet-eta-function

The observed discrepancy between the two method reminds me of a similar effect, if the zeta-/eta-function is naively summed, again by (insufficient) consideration of matrix-identities.

There may be another way of obtaining the row-sums of the list-matrix of second columns of the powers of $S1_b$.

Note: to have exact arithmetic, I use $b=e=exp(1)$ as base for the matrices in the following examples.

(2.4.1.) *Example:*

The 2'columns of powers of $S1_e$ — *alternate summed rows*

						alternate summed rows
0	0	0	0	0	0	0
1	1	1	1	1	1	1/2
0	-1/2	-1	-3/2	-2	-5/2	1/8
0	1/3	7/6	5/2	13/3	20/3	-1/48
0	-1/4	-35/24	-35/8	-39/4	-55/3	-1/96
0	1/5	19/10	947/120	337/15	617/12 ...	19/1920

If we construct polynomials based on the progressions in each row and then consider, how to compute the row-sums, then we can use the polynomials for each subsequent $x=1,2,3,...$ and sum then with alternating sign.

(2.4.2.) *Example*

The 2'columns of powers of $S1_e$						polynomials in x for computing entries	rowsums
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1/2
0	-1/2	-1	-3/2	-2	-5/2	-1/2*x+1/2	1/8
0	1/3	7/6	5/2	13/3	20/3	1/4*x^2-5/12*x+1/6	-1/48
0	-1/4	-35/24	-35/8	-39/4	-55/3	-1/8*x^3+13/48*x^2-3/16*x+1/24	-1/96
0	1/5	19/10	947/120	337/15	617/12 ...	1/16*x^4-23/144*x^3+7/48*x^2-41/720*x+1/120	19/1920

The evaluation of the first and second row is trivial.

For the third row (where the polynomial is $1/2 - 1/2x$), if we insert $1,2,3,4,...$ subsequently for x, we get for the infinite alternating sum

(2.4.3.)
$$\begin{aligned}
&(\ 1/2*1 \ - 1/2*1)\\
-&(\ 1/2*1 \ - 1/2*2)\\
+&(\ 1/2*1 \ - 1/2*3)\\
-&(\ 1/2*1 \ - 1/2*4)\\
&....\\
&==============\\
&1/2*\eta(0) \ - 1/2\ \eta(-1) \quad = 1/2*1/2 - 1/2*1/4 = 1/8
\end{aligned}$$

which matches the row-sum, computed the other ways, perfectly.

Let's see the polynomial for the fourth row, where I insert the $\eta()$-expressions at once:

(2.4.4.)
$$\begin{aligned}
&1/6*\eta(0) - 5/12*\eta(-1) + 1/4*\eta(-2)\\
&= 1/6*1/2 \ - 5/12*1/4 \ + 1/4*0\\
&= 4/48 - 5/48\\
&= -\ 1/48
\end{aligned}$$

This again fits perfectly the otherwise computed sums. (I proceeded here to check this for a few more rows). The heuristic suggests, that this may be a valid procedure.

The next step is, to display these coefficients (as computed for the *eta()*-s), collected by equal powers of *x* as a matrix ***C*** itself, where the coefficients of each polynomial are expressed as entries of one row.

The matrix ***C*** has then a special property by its construction. Postmultiplied by a powerseries, it gives interesting sequences of numbers: Bell-numbers, factorials, in general the *e.g.f.*s for the iterated exponentials and logarithms (see *OEIS A000258, A003713*)

(2.4.5.) *Examples*

$C * V(x) = {}^dF^{-1} * (interesting\ sequence)$

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 & 3 \\ 4 & 1 & 0 & 1 & 4 & 9 \\ -8 & -1 & 0 & 1 & 8 & 27 \\ 16 & 1 & 0 & 1 & 16 & 81 \\ -32 & -1 & 0 & 1 & 32 & 243 \end{bmatrix}$$

$$\begin{bmatrix} 0 & . & . & . & . & . \\ 1 & 0 & . & . & . & . \\ 1/2 & -1/2 & 0 & . & . & . \\ 1/6 & -5/12 & 1/4 & 0 & . & . \\ 1/24 & -3/16 & 13/48 & -1/8 & 0 & . \\ 1/120 & -41/720 & 7/48 & -23/144 & 1/16 & 0 \end{bmatrix} \qquad \begin{bmatrix} 1/0! \\ 1/1! \\ 1/2! \\ 1/3! \\ 1/4! \\ 1/5! \end{bmatrix} * \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 3 & 2 & 1 & 0 & -1 & -2 \\ 12 & 5 & 1 & 0 & 2 & 7 \\ 60 & 15 & 1 & 0 & -6 & -35 \\ 358 & 52 & 1 & 0 & 24 & 228 \end{bmatrix}$$

Obviously, this is due to its property to define the *2*'nd column of the *h*'th power of $\boldsymbol{S1_e}$, and in the column for *V(2) = [1,2,4,8,...]* we find for the result in that column just the coefficients *[1,-1/2,1/3,-1/4,...]* of the powerseries, which defines *log(1+x)* (if the reciprocal factorials, which I factored out here for display, are also used), just the second column of the first power $\boldsymbol{S1_e}^h$, where *h=1* .

Now, if we don't look at an individual powerseries, but at the infinite alternating sum of all consecutive powerseries, we have the vector of *eta()*-values at the rhs

(2.4.6.) *Example:*

$$\lim\left(\begin{bmatrix} 1 & -1 & +1 & -1 & +1 & -1 \\ 1 & -2 & +3 & -4 & +5 & -6 \\ 1 & -4 & +9 & -16 & +25 & -36 \\ 1 & -8 & +27 & -64 & +125 & -216 \\ 1 & -16 & +81 & -256 & +625 & -1296 \\ 1 & -32 & +243 & -1024 & +3125 & -7776 \end{bmatrix}\right) = \begin{bmatrix} h(0) \\ h(-1) \\ h(-2) \\ h(-3) \\ h(-4) \\ h(-5) \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1/4 \\ 0 \\ -1/8 \\ 0 \\ 1/4 \end{bmatrix}$$

and get, as expected:

(2.4.7.) $C * H = M_l\,[,1]$

$$* \begin{bmatrix} h(0) \\ h(-1) \\ h(-2) \\ h(-3) \\ h(-4) \\ h(-5) \end{bmatrix}$$

$$\begin{bmatrix} 0 & . & . & . & . & . \\ 1 & 0 & . & . & . & . \\ 1/2 & -1/2 & 0 & . & . & . \\ 1/6 & -5/12 & 1/4 & 0 & . & . \\ 1/24 & -3/16 & 13/48 & -1/8 & 0 & . \\ 1/120 & -41/720 & 7/48 & -23/144 & 1/16 & 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1/2 \\ 1/8 \\ -1/48 \\ -1/96 \\ 19/1920 \end{bmatrix}$$

Using column-sums of C

By this construction, we had now, writing

$H = columnvector(\eta(0), \eta(-1), \eta(-2),...)$

the equivalent expression

(2.4.8.) $AL_e(x) = V(x)\sim * C * H$
(2.4.9.) $AL_e(1) = V(1)\sim * C * H$

Thus, if we assume a fixed x, for instance $x=1$, then we might use associativity and rewrite:

(2.4.10.) $AL_e(1) = (V(1)\sim * C) * H = A\sim * H = [a_0, a_1, a_2, a_3, ...]\sim * H$

and we can compute $AL_e(1)$ by a series of $\eta()$-values:

(2.4.11.) $AL_e(1) = a_0\ \eta(0) + a_1\ \eta(-1) + a_2\ \eta(-2) + a_3\ \eta(-3) + ...$

The values for the a_k are not obvious. Here I only recognize

$a_0 = exp(1)-1$

having a known value, and numerically I get for the other coefficients a_k about

(2.4.12.)
$a_0 \sim\ \ 1.71828$
$a_1 \sim -1.17714$
$a_2 \sim\ \ 0.732788$
$a_3 \sim -0.430009$
$a_4 \sim\ \ 0.241939$
$a_5 \sim -0.131792$
$a_6 \sim\ \ 0.0699448$
$a_7 \sim -0.0363238$

Note, the sign change in connection with the fact, that each $\eta(-2k)$ -value is zero, and the non-zero values alternate in sign, so we get effectively:

(2.4.13.) $AL_e(1) =$

1.71828...	* 1/2
- 1.17714...	* 1/4
+ 0	
+ 0.430009	* 1/8
+ 0	
- 0.131792	* 1/4
+ 0	
+ 0.0363238	* 17/16
+ 0	
...	

For $x=1$, $x=log(1+1)$, $x=log(1+log(1+1))$ we get the following a_k:

	x=1	x=log(1+1)	x=log(1+log(1+1))
$a_0 \sim$	1.71828	1.00000	0.693147
$a_1 \sim$	-1.17714	-0.433046	-0.216523
$a_2 \sim$	0.732788	0.175814	0.0644657
$a_3 \sim$	-0.430009	-0.0685215	-0.0186106
$a_4 \sim$	0.241939	0.0259260	0.00525275
$a_5 \sim$	-0.131792	-0.00958558	-0.00145651
$a_6 \sim$	0.0699448	0.00347799	0.000398033
$a_7 \sim$	-0.0363238	-0.00124213	-0.000107444

but which does not help much, since the $h()$-values increase in absolute value and seem to make the powerseries a divergent series in all cases by the dominance of their hypergeometric progression.

Gottfried Helms

(Note: I'll be updating this text as I get new results. Check this url occasionally)

3. Appendix:

3.1. The matrices S1 and S2

I use the matrix-representation beginning with:

(3.1.1.) $u_b(x) \quad = V(x)\sim * S2_b\,[,1]$ // [,1] meaning the second column of $S2_b$ only

(3.1.2.) $l_b(x) \quad = V(x)\sim * S1_b\,[,1]$

where $S1_b$ and $S2_b$ are matrices, which contain essentially the Stirling-numbers of first and second kind.

Let $V(x)$ be a notation for a vector, which contains consecutive powers of its argument, such that

(3.1.3.) $V(x) = column(1, x, x^2, x^3, \ldots)$

$V(x)\sim$ its transpose (row-vector) and ${}^dV(x)$ its diagonal arrangement.

Let $b=e=exp(1)$, then the basic Stirling-matrices are:

(3.1.4.) $S1_e := S1 =$

$$\begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -1/2 & 1 & . & . & . \\ 0 & 1/3 & -1 & 1 & . & . \\ 0 & -1/4 & 11/12 & -3/2 & 1 & . \\ 0 & 1/5 & -5/6 & 7/4 & -2 & 1 \end{bmatrix}$$

(3.1.5.) $S2_e := S2 =$

$$\begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & 1/2 & 1 & . & . & . \\ 0 & 1/6 & 1 & 1 & . & . \\ 0 & 1/24 & 7/12 & 3/2 & 1 & . \\ 0 & 1/120 & 1/4 & 5/4 & 2 & 1 \end{bmatrix}$$

and for a general b

(3.1.6.) $S1_b = S1_e * {}^dV(1/log(b))$

(3.1.7.) $S2_b = {}^dV(log(b)) * S2_e$

Then according to the "Handbook of mathematical functions" (Abramowitz-Stegun [AS-St]) we have the following identities for $b=e=exp(1)$:

(3.1.8.) $V(x)\sim * S2_e = V(exp(x)-1)\sim = V(u_e(x))\sim$

(3.1.9.) $V(x)\sim * S1_e = V(log(1+x))\sim = V(l_e(x))\sim$

and finite integer iterations can be expressed by powers of $S1$ and $S2$:

(3.1.10.) $V(x)\sim * S2_e^{\,h} = V(ut_e(x,h))\sim$

(3.1.11.) $V(x)\sim * S1_e^{\,h} = V(lt_e(x,h))\sim$

For instance, correctly, (where $x=0.6$ and $b=e=exp(1)$)

$V(0.6)\sim * S1_e = V(l_e(0.6))\sim$

$$\begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -1/2 & 1 & . & . & . \\ 0 & 1/3 & -1 & 1 & . & . \\ 0 & -1/4 & 11/12 & -3/2 & 1 & . \\ 0 & 1/5 & -5/6 & 7/4 & -2 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0.6 & 0.6\text{^}2 & 0.6\text{^}3 & 0.6\text{^}4 & 0.6\text{^}5 \end{bmatrix} \quad \begin{bmatrix} 1 & 0.470004 & 0.220903 & 0.103825 & 0.0487983 & 0.0229354 \end{bmatrix}$$

where in the second column of the result is $lt_e(0.6,1)= l_e(0.6) = log(1+0.6) \sim 0.470004$

and the iteration (by second power of ***S1***)

$V(0.6)\sim * S1_e^2 = V(l_e(l_e(x)))\sim$

S1^2

$$\begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -1 & 1 & . & . & . \\ 0 & 7/6 & -2 & 1 & . & . \\ 0 & -35/24 & 10/3 & -3 & 1 & . \\ 0 & 19/10 & -21/4 & 13/2 & -4 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0.6 & 0.6\text{^}2 & 0.6\text{^}3 & 0.6\text{^}4 & 0.6\text{^}5 \end{bmatrix} \quad \begin{bmatrix} 1 & 0.385265 & 0.148429 & 0.0571845 & 0.0220312 & 0.00848784 \end{bmatrix}$$

where in the second column of the result is $lt_e(0.6,2)= l_e(l_e(0.6)) = \log(1+\log(1+0.6)) \sim 0.385265$

3.2. Powers of S1 and the matrix M_l

The top-left-edge of the first three powers of ***S1*** look like

S1 S1^2 S1^3

$$\begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -1/2 & 1 & . & . & . \\ 0 & 1/3 & -1 & 1 & . & . \\ 0 & -1/4 & 11/12 & -3/2 & 1 & . \\ 0 & 1/5 & -5/6 & 7/4 & -2 & 1 \end{bmatrix} \begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -1 & 1 & . & . & . \\ 0 & 7/6 & -2 & 1 & . & . \\ 0 & -35/24 & 10/3 & -3 & 1 & . \\ 0 & 19/10 & -21/4 & 13/2 & -4 & 1 \end{bmatrix} \begin{bmatrix} 1 & . & . & . & . & . \\ 0 & 1 & . & . & . & . \\ 0 & -3/2 & 1 & . & . & . \\ 0 & 5/2 & -3 & 1 & . & . \\ 0 & -35/8 & 29/4 & -9/2 & 1 & . \\ 0 & 947/120 & -65/4 & 57/4 & -6 & 1 \end{bmatrix}$$

Since only the second columns are interesting here for to obtain the iterated logarithm, I note the sequence of that columns in the following, and show also the row-wise sums, as obtained by any method of summation of alternating divergent series:

(3.2.1.) *The 2'columns of powers of S1*

*The **alternating** sums of the row-entries, computed by **serial** summation of the entries using Euler-summation*

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & -1/2 & -1 & -3/2 & -2 & -5/2 \\ 0 & 1/3 & 7/6 & 5/2 & 13/3 & 20/3 \\ 0 & -1/4 & -35/24 & -35/8 & -39/4 & -55/3 \\ 0 & 1/5 & 19/10 & 947/120 & 337/15 & 617/12 \end{bmatrix} \ldots \qquad \begin{bmatrix} 0 \\ 1/2 \\ 1/8 \\ -1/48 \\ -1/96 \\ 19/1920 \end{bmatrix}$$

3.3. M_l by geometric-series formula

The alternating sum of the 2'nd columns of powers of $S1_e$ is also correctly reproduced by

(3.3.1.) $M_{l,e}=(I+S1_e)^{-1}$

see the relevant result in second column:

(3.3.2.) $M_{l,e}=(I+S1_e)^{-1}$

M

$$\begin{bmatrix} 1/2 & . & . & . & . & . \\ 0 & 1/2 & . & . & . & . \\ 0 & 1/8 & 1/2 & . & . & . \\ 0 & -1/48 & 1/4 & 1/2 & . & . \\ 0 & -1/96 & -1/24 & 3/8 & 1/2 & . \\ 0 & 19/1920 & -5/96 & -1/16 & 1/2 & 1/2 \end{bmatrix}$$

3.4. Approximation to $AU_b(x)$ by M_u

Various dimensions for M_u . diff is the difference between results of matrix-computation and serial computation (using Pari/Gp *sumalt()*-function with prec *400*)

,

dimension	diff $AU_b(x)$ M_u vs. serial	diff $AL_b(x)$ M_l vs. serial
b=1.3, x=1		
32	-4.90491008447 E-33	-0.00990225809450
48	-5.44017005559 E-47	-0.00990225809450
64	-2.90200439567 E-62	-0.00990225809450
80	-8.20471938736 E-81	-0.00990225809450
96	1.77318320360 E-92	-0.00990225809450
b=2.1, x=0.2		
32	-1.40198762190 E-25	2.30682882083 E-14
48	5.04868595534 E-36	2.30682882082 E-14
64	-5.96667339306 E-46	2.30682882082 E-14
80	1.22482311871 E-54	2.30682882082 E-14
96	-6.23842939347 E-64	2.30682882082 E-14

Other examples behave similarly.

3.5. x_k-values for $d_b(x)$ - plots

The x_k values of fractional heights $h_k=-1+k/32$ for base $b=sqrt(2)$ and $a=1$ where

$$x_k = lt_b(a,h_k)$$

For $k=32$, thus $h=0$, this is $x_k = a = 1.$, for $k=33$, thus $h=1/32$, this is $x_{33}=1.025$ and for $k=64$, thus $h=32/32=1$ this is $x_{64}=log(1+x_0)/log(sqrt(2))=2$ The other values, for $k=0..31$ and $k=65..96$ are computed by integer iteration.

k	xk	k	xk
32	1		
33	1.025235216	49	1.486307197
34	1.050907909	50	1.518446661
35	1.077016461	51	1.550933184
36	1.103558812	52	1.583757306
37	1.13053245	53	1.616909148
38	1.157934408	54	1.650378416
39	1.185761257	55	1.684154419
40	1.214009098	56	1.718226083
41	1.242673563	57	1.752581962
42	1.27174981	58	1.787210258
43	1.301232523	59	1.822098835
44	1.331115909	60	1.85723524
45	1.361393704	61	1.892606721
46	1.392059168	62	1.928200243
47	1.423105095	63	1.964002514
48	1.454523815	64	2

This computation of x_k was actually performed by (Pari/Gp-Pseudocode):

```
L = S1 * dV(1/b)
L32 = mpow(L,1/32)  \\ using eigenvalue-decomposition
a = 1
for(k=0,96, xlist[1+k]= V(a)~ * (L32^(k/32-1))[,2] )
```

4. References

[AS-St] Abramowitz, M. and Stegun, I. A. (Eds.). "Stirling Numbers of the First Kind." §24.1.3 in Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, 9th printing. New York: Dover, p. 824, 1972.
Online-copy at *http://www.convertit.com/Go/ConvertIt/Reference/AMS55.ASP*

Initial conjecture in newsgroup news:sci.math of 20.11.2007,
"[Tetration] Alternating series of powertowers of increasing heights/a conjecture"
(Gottfried Helms)
the counterexample of G.A.Edgar
http://groups.google.de/group/sci.math/browse_frm/thread/82ab36e382fccaf2/4e2670c01293ec57?lnk=gst&q=tetration+edgar#4e2670c01293ec57

a subsequent series of discussion of this topic/related observations in the "tetration-forum"
http://math.eretrandre.org/tetrationforum/showthread.php?tid=99
http://math.eretrandre.org/tetrationforum/showthread.php?tid=93