Exploring the eigen-system of the tetration-matrices

Edit-note:
1) Update in tabledata!
2) Error concerning Q statements ~~striked out~~

Why to analyze the eigensystem of Bs?

If we want to perform the analysis of tetration based on eigenanalysis, then we rely on finite dimensional matrices which were computed by numerically very difficult procedurs in the eigensystem-solver. Although we might assume, that for the convergent cases our found matrix is a good approximate, we cannot quantify this approximation, and thus not the error. Also for extending the range of admissible baseparameter s it seems needed to have such an analytical description, from where then analytic continuation can be derived and be well founded. So it seems essential for serious progress to decode the structure of the eigenmatrices and of the eigenvalues. Here I show a short step into this matter.

I'll analyze the eigensystem for one case $t=1.7$, baseparameter $s=t^{1/t}$, to make things simple.

$t= 1.7$
$s = t^{1/t} = 1.36633813256$
$log(t) = 0.530628251062$

From my previous articles I'll take the convention:

$B = matrix(dim,dim,r,c,c^r/r!)$ // *r,c denoting row,column-index, zero-based*
$B_s = {}^dV(log(s)) * B$ // *where* ${}^dV(x)$ *is a diagonal vector of consecutive powers of x*
// $V(x) = column(1,x,x^2,x^3,...)$ *and* ${}^dV(x) = diag(V(x))$

and $\boldsymbol{B_s}$ performs the integer-tetration-iteration

$V(x)\sim * B_s = V(s^x) \sim$
$V(1)\sim * B_s = V(s) \sim$
$V(1)\sim * B_s^y = V(s\text{^^}y)$

I also use here the following notation for $\boldsymbol{B_s}$' eigen-decomposition for a fixed parameter s:

$B_s = W * E * W^{-1}$
$W^{-1} * B_s = E * W^{-1}$

where $\boldsymbol{E}$ is the diagonal-matrix containing the eigenvalues. Note that W and E depend on the parameter s here.

Settings according to my hypotheses:

e_k are the k'th eigenvalues of B_s, which I assume are $log(t)^k$
Let W be the eigenvector-matrix of B_s
W^{-1} contains the powers of t, so I extract them from the coefficients. Documented as t^0, t^1,...

According to the last (very trustworthy) observation, that $\boldsymbol{W^{-1}}$ contains the consecutive powers of t, I separated these components, calling the remaining matrix $\boldsymbol{Q}$ (or also *"kernel"* of $\boldsymbol{W}$):

$W^{-1} * {}^dV(t)^{-1} = Q$

so that the previous

$W^{-1} * B_s = E * W^{-1}$

becomes

$Q * {}^dV(t) * B_s = E * Q * {}^dV(t)$

~~where Q is now independent of s (or t).~~

Note that B_s already contains the parameter t in the following form:

$B_s = {}^dV(log(s))*B = {}^dV(log(t^{1/t})) * B = {}^dV(log(t)/t)) * B$

and

${}^dV(t) * B_s = {}^dV(t) * {}^dV(log(t)/t)) * B = {}^dV(log(t)) * B$

so that this matrix-equation can be reduced to one of the following forms, where I do not know currently, which is the most useful one

$$Q * {}^dV(\log(t)) * B = E * Q * {}^dV(t)$$
$${}^dV(\log(t)) * B * {}^dV(t)^{-1} = Q^{-1} * E * Q$$
$$E^{-1} * Q * E * B = Q * {}^dV(t)$$
$$E\, B\, {}^dV(t)^{-1} = Q^{-1} * E * Q$$

A problem for these formula exist in that in the limit of infinite dimension the sequence of eigenvalues approaches zero according to my hypothesis and thus ***E*** should not be invertible in a first view.

The following observation concerning independence of Q from base-parameter s was wrong :

A very interesting, general, observation is, that the kernel ***Q*** of the eigenvector-matrix $\boldsymbol{W^{-1}}$ ~~seems to be constant for any parameter *t* (and the subsequent computed base-parameter *s*), so the same ***Q***-matrix is then valid for all parameters *s* in tetration (given the usual bounds for *s*, resp *t*.)~~

Example: (Q still dependent on parameter t)

$$Q = W^{-1} * {}^dV(t)^{-1}$$

VecNr	*Update: the previously documented table was quite imprecise*						
0	1	1	1	1	1	1	1
1	0	1	2	3	4	5	6
2	0	1.0	1.11460	0.34382	-1.31234	-3.85390	-7.28085
3	0	1.0	0.33009	-1.27044	-3.06237	-4.30643	-4.26337
4	0	1.0	-0.36595	-2.08416	-2.73524	-1.49410	1.87003
5	0	1.0	-0.98470	-2.28881	-1.34707	1.77055	5.82374
6	0	1.0	-1.53611	-2.03517	0.42514	4.12661	6.36486
7	0	1.0	-2.02904	-1.44101	2.14616	5.08344	4.10682
8	0	1.0	-2.47160	-0.59679	3.55094	4.64078	0.28767
9	0	1.0	-2.87293	0.43478	4.49277	3.03516	-3.86546
10	0	1.0	-3.24682	1.63276	4.88987	0.54444	-7.40559
11	0	1.0	-3.60967	3.02164	4.64016	-2.59963	-9.55913

grey shaded cells are different in 5'th place using dim=72

So this is the matrix, for which I try to decode the compositions of rows.

An explicit formal description for the first two rows of ***Q***, associated to the eigenvalues *1* and *log(t)*, such that

$$W^{1}[0,] * Bs = \quad 1 \quad * W^{1}[0,] \quad or \qquad Q[0,]* {}^{d}V(t) * Bs = \quad 1 \quad * Q[0,1]* {}^{d}V(t)$$
$$W^{1}[1,] * Bs = log(t) * W^{1}[1,] \quad or \qquad Q[1,]* {}^{d}V(t) * Bs = log(t) * Q[0,1]* {}^{d}V(t)$$

can easily be derived. To show the idea which I am following here to get hints for the structure of the following rows in ***Q***.

Denote the guessed version for ***Q*** as ***Q^*** and the version, which occurs by acutally computing the rhs as ***R***, where ***R*** should equal ***Q^***, if the proposition is correct.

Example: (the columns [e0,e1] , [1,1], [t^0,t^1] are meant as diagonal matrices)

first two rows: $Q * {}^{d}V(t) *Bs = E * R * {}^{d}V(t)$	$\begin{bmatrix} 1.000 & 1.000 & 1.000 & 1.000 & 1.000 & 1.000 \\ 0 & 0.3121 & 0.6243 & 0.9364 & 1.249 & 1.561 \\ 0 & 0.04871 & 0.1949 & 0.4384 & 0.7794 & 1.218 \\ 0 & 0.005068 & 0.04055 & 0.1368 & 0.3244 & 0.6336 \\ 0 & 0.0003955 & 0.006328 & 0.03204 & 0.1012 & 0.2472 \\ 0 & 0.00002469 & 0.0007901 & 0.006000 & 0.02528 & 0.07716 \end{bmatrix}$
$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ . & 1 & 2 & 3 & 4 & 5 \end{bmatrix} * \begin{bmatrix} t^0 \\ t^1 \end{bmatrix}$	$\begin{bmatrix} e0 \\ e1 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ . & 1 & 2 & 3 & 4 & 5 \end{bmatrix} * \begin{bmatrix} t^0 \\ t^1 \end{bmatrix}$

The deviations from integers in the result ***R*** were so small, that I just approximated these values to integers for display.

In the above example we see, that the input-vectors are identical to the output-vectors except the scaling by $e_0 = 1$, $e_1 = log(t)$, in the first an second row, where e_1 is assumed to be the second eigenvalue

$$e_1 = eigenvalue\_1 = log(t) \text{ by hypothesis}$$

So this is obviously a solution for the first two rows for the left eigenmatrix (=inverse of the right eigenmatrix ***W***) resp. its kernel ***Q***.

Extrapolating from the first two correct rows, the next proposal to explore this further was, that possibly the Pascal-/Binomial-matrix ***P*** is in any way part of ***Q***

It is obvious from the numerical display of *Q*, that this is not the case, but possibly one gets hints for how to search further. See the interesting result of the computation.

The very good approximation to the integer entries of the matrix ***VZ*** is so much convincing, that ***P*** is in fact not part of ***Q***, what is the bad news.

The good news is possibly, that the hypothesis concerning the structure of the set of eigenvalues as well as of the powers of *t* in $\boldsymbol{W}^{-1}$ has a very good backing (given I didn't just do some trivial cancelling here, which also may be possible)

Example:

The column E is meant as diagonal-matrix: E=diag (e_0, e_1,...)
containing the assumed structure for eigenvalues $e_k = log(t^k)$
The column [0!,1!,2!...] is meant as diagonalmatrix F
The column [t0,t1,t2,...] is meant as diagonalmatrix ${}^dV(t)$ containing consecutive powers of t

P is the Pascal-matrix, containing the binomial-coefficients

*Here the matrix-product F*P~ was tried, whether it is the eigenvectormatrix Q*

$$Bs = \begin{bmatrix} 1.000 & 1.000 & 1.000 & 1.000 & 1.000 & 1.000 \\ 0 & 0.3121 & 0.6243 & 0.9364 & 1.249 & 1.561 \\ 0 & 0.04871 & 0.1949 & 0.4384 & 0.7794 & 1.218 \\ 0 & 0.005068 & 0.04055 & 0.1368 & 0.3244 & 0.6336 \\ 0 & 0.0003955 & 0.006328 & 0.03204 & 0.1012 & 0.2472 \\ 0 & 0.00002469 & 0.0007901 & 0.006000 & 0.02528 & 0.07716 \end{bmatrix}$$

$$\begin{bmatrix} 0! \\ 1! \\ 2! \\ 3! \\ 4! \\ 5! \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ . & 1 & 2 & 3 & 4 & 5 \\ . & . & 1 & 3 & 6 & 10 \\ . & . & . & 1 & 4 & 10 \\ . & . & . & . & 1 & 5 \\ . & . & . & . & . & 1 \end{bmatrix} * \begin{bmatrix} t\hat{}0 \\ t\hat{}1 \\ t\hat{}2 \\ t\hat{}3 \\ t\hat{}4 \\ t\hat{}5 \end{bmatrix} \qquad \begin{bmatrix} e0 \\ e1 \\ e2 \\ e3 \\ e4 \\ e5 \end{bmatrix} * \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 4 & 9 & 16 & 25 \\ 0 & 1 & 8 & 27 & 64 & 125 \\ 0 & 1 & 16 & 81 & 256 & 625 \\ 0 & 1 & 32 & 243 & 1024 & 3125 \end{bmatrix} * \begin{bmatrix} t\hat{}0 \\ t\hat{}1 \\ t\hat{}2 \\ t\hat{}3 \\ t\hat{}4 \\ t\hat{}5 \end{bmatrix}$$

So the result does not help much from the approach itself. But it has another impact.

The factorial-scaling of ***P*** constitutes in fact the coefficients, which occur with derivatives. So this accidental result gives raise to a formulation about the values of the derivatives with respect to x:

$$\frac{d^{(m)}(V(x)\sim * {}^{diag}V(t) * Bs)}{(dx)^m} \quad \text{at } x=1.$$

For a column c, where the r'th row-entries of ***Bs*** may be denoted as b_r and b_r's structure explicated:

$$\frac{d^{(m)}\sum_{r=0}^{oo} t^r b_r x^r}{(dx)^{(m)}} = \frac{d^{(m)}\sum_{r=0}^{oo} t^r \frac{\frac{log(t)^r}{t^r} c^r}{r!} x^r}{(dx)^{(m)}} = \frac{d^{(m)}\sum_{r=0}^{oo} \frac{log(t^c)^r}{r!} x^r}{(dx)^{(m)}}$$

By rules of derivatives this is for the m'th derivative with respect to x:

$$= m! \sum_{r=0}^{oo} \binom{r}{m} \frac{log(t^c)^r}{r!} x^{r-m}$$

This reduces to

$$= m! \sum_{r=0}^{oo} \frac{r!}{m!(r-m)!} \frac{log(t^c)^r}{r!} x^{r-m}$$

$$= \sum_{r=0}^{oo} \frac{log(t^c)^r}{(r-m)!} x^{r-m}$$

Since the terms with negative factorials in the denominator vanish, we may start with the index $r=m$

$$= \sum_{r=m}^{oo} \frac{log(t^c)^r}{(r-m)!} x^{r-m}$$

We re-index this with $q=r-m$ and extract constant factors:

$$= \sum_{q=0}^{oo} \frac{log(t^c)^{q+m}}{q!} x^q$$

$$= log(t^c)^m \sum_{q=0}^{oo} \frac{log(t^c)^q}{q!} x^q$$

Now the sum is an exponential-series, giving t^{cx} and we position the variable x nearest to t:

$$= log(t^c)^m t^{cx}$$

$$= c^m \, log(t)^m (t^x)^c$$

which is then at $x=1$, using the eigenvalue-notation for entries of $\mathbf{E}$ and the powers of t fo entries of ${}^d\mathbf{V}(t)$:

$$= c^m \, log(t)^m t^c$$

$$= e_m * c^m * t^c$$

This reflects perfectly the numerical result of the result-matrix with m as row-index and c as column-index.

Well, this was a nice oberservation, but there is not much progress concerning the initial question of the structure of the eigenmatrix-kernel $\mathbf{Q}$. Here it is again:

Example:

$$Q = W^1 * {}^dV(t)^{-1}$$

VecNr	*Update: the previously documented table was quite imprecise*						
0	1	1	1	1	1	1	1
1	0	1	2	3	4	5	6
2	0	1.0	1.11460	0.34382	-1.31234	-3.85390	-7.28085
3	0	1.0	0.33009	-1.27044	-3.06237	-4.30643	-4.26337
4	0	1.0	-0.36595	-2.08416	-2.73524	-1.49410	1.87003
5	0	1.0	-0.98470	-2.28881	-1.34707	1.77055	5.82374
6	0	1.0	-1.53611	-2.03517	0.42514	4.12661	6.36486
7	0	1.0	-2.02904	-1.44101	2.14616	5.08344	4.10682
8	0	1.0	-2.47160	-0.59679	3.55094	4.64078	0.28767
9	0	1.0	-2.87293	0.43478	4.49277	3.03516	-3.86546
10	0	1.0	-3.24682	1.63276	4.88987	0.54444	-7.40559
11	0	1.0	-3.60967	3.02164	4.64016	-2.59963	-9.55913

grey shaded cells are different in 5'th place using dim=72

The problem is still to find an analytical description for the next rows. Note, that the scalings of the rows are arbitrary; I just scaled them so that the leading entry is 1. It seems, they are composed of combinations of more simple row-vectors, for instance it looks suspiciously as if they represent polynomials of order of their row-index. But some obvious compositions didn't work, so this is still an open question for research, and I'm stuck here.

Gottfried Helms, 29.8.2007

Update 2:

With dimension dim=96 and float-precision I found the following improvement:

	1	1	1	1	1	1	1
	0	1	2	3	4	5	6
	0	1.0	1.1146	0.34383	-1.3123	-3.8539	-7.2809
	0	1.0	0.33010	-1.2704	-3.0624	-4.3064	-4.2634
	0	1.0	-0.36595	-2.0842	-2.7352	-1.4941	1.8700
	0	1.0	-0.98470	-2.2888	-1.3471	1.7706	5.8237
	0	1.0	-1.5361	-2.0352	0.42514	4.1266	6.3649
	0	1.0	-2.0290	-1.4411	2.1461	5.0834	4.1071
	0	1.0	-2.4712	-0.59781	3.5497	4.6419	0.29215
	0	1.0	-2.8693	0.42431	4.4863	3.0548	-3.8280
	0	1.0	-3.2294	1.5717	4.8845	0.67969	-7.2623
	0	1.0	-3.5565	2.8037	4.7236	-2.1059	-9.3610

The very interesting aspect is, that if the rows are combined with binomial weights,

test $= P^{-1} * Q$

then we get a very simple pattern:

Example: test

	1.0	1.0	1.0	1.0	1.0	1.0	1.0
	-1.0	0	1.0	2.0	3.0	4.0	5.0
	1.0	0	-1.8854	-4.6562	-8.3123	-12.854	-18.281
	-1.0	0	1.9863	5.6981	11.875	21.255	34.579
	1.0	0	-1.9987	-5.9394	-13.360	-26.392	-47.762
	-1.0	0	1.9999	5.9893	13.829	28.716	55.648
	1.0	0	-2.0	-5.9984	-13.959	-29.588	-59.472
	-1.0	0	2.0	5.9998	13.991	29.879	61.080
	1.0	0	-2.0	-6.0	-13.998	-29.967	-61.690
	-1.0	0	2.0	6.0	14.000	29.992	61.903
	1.0	0	-2.0	-6.0	-14.000	-29.998	-61.972
	-1.0	0	2.0	6.0	14.000	30.000	61.992