Pictures for 5 different interpolation-methods for Tetration

After a couple of proposals have been popped up in the previous years I needed for myself a more immediate impression of their different properties and characteristics. When one looks at wikipedia he/she can even find some serious suggestion of a *linear interpolation*, so that was triggering me to compile a comparision of different variants. I cannot yet implement all of those methods - but the overview here might still be interesting and an eye-opener for someone mor involved than me.

I am most interested in the behave in the complex plane as the most general application of tetration, so we should look at complex values to be tetrated; also to stress the methods much, we should use a base outside the conventional Euler-interval b of $1..e^{1/e}$; I chose $b=4$ here which is only real - but to use a complex base too would unfortunately have overcomplexified my computations... The initial values, for which I generated the trajectories were however imaginary: $z_1(0)=0.1\ \hat{\imath}$, $z_2(0) = 0.05\ \hat{\imath}$ and $z_3(0) = 0.01\ \hat{\imath}$.

Usually I think in the paradigm of the <u>*regular tetration*</u> and also assume/hope, it shall give meaningful solutions even when this involves powerseries with complex coefficients. Because it seems, that we can compute nearly arbitrary accurate values even if we use the powerseries for its Schröder-functions, I use it as first reference in the following.

Regular tetration

That base $b=4$ provides no real fixpoint but only complex ones (and they all are repelling). So the regular tetration has to employ complex powerseries; we have to use the powerseries for $exp(z+ln(t)) - ln(t)$ whose constant term vanishes by this shift of coordinates and thus can be "regularly" fractionally iterated.

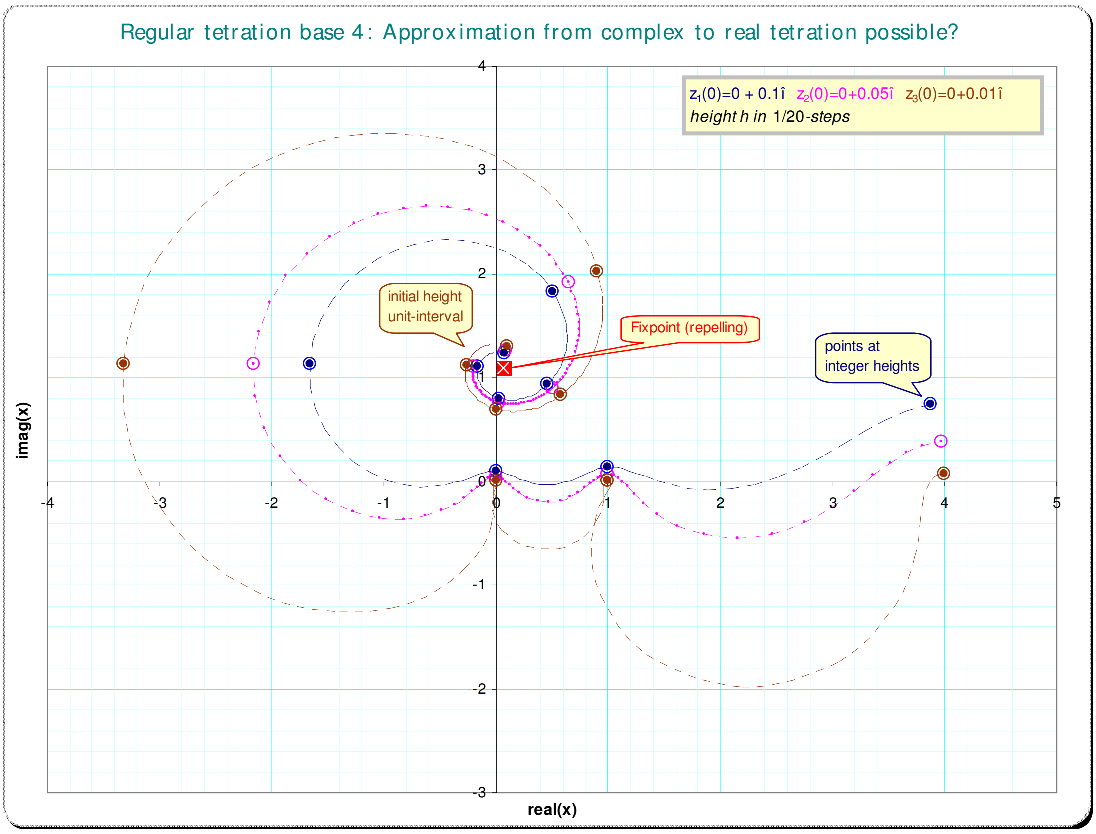
But besides the general inconveniences which occur with powerseries with complex coefficients we have a very uneasy problem of convergence: the real axis is off the radius of convergence and so no immediately meaningful values can be computed, and for iterates from , say, $z_0=1$ (or in general, from powertowers on b) in integer steps we cannot approach the fixpoint due to the singularity of $log(0)$.

When I constructed the powerseries for regular tetration, using the complex fixpoint near $0.06 + 1.091\ \hat{\imath}$ it was obvious, that the series would converge miserably if at all for real initial values. But those values are naturally of special interest. So I computed approximations for a start. I set $z_1(0) = 0 + 0.1*I$ where the notation "(0)" means, it has iteration height $h=0$. This values is thus "the norm" for the notation of the subsequent iterates.

Then to get an improved general impression I observe also the trajectories beginning at $z_2(0) = 0 + 0.05\ \hat{\imath}$ and $z_3(0) = 0 + 0.01\ \hat{\imath}$. From those I computed the values for fractional heights from $h=0$ to $h=1$, which means $z_1(1)= 4^{z_1(0)} \sim 0.99 + 0.138\ \hat{\imath}$, $z_2(1) = 4^{z_2(0)} \sim 0.997 + 0.069\ \hat{\imath}$ and $z_3(1) = 4^{z_2(0)} \sim 0.9999+0.0138\ \hat{\imath}$.

Because the powerseries for the schröderfunction do not converge for those values I shifted that initial values towards the fixpoint by integer-height-tetrating with height $h=-6$. For the interval $z_k(-6)..z_k(-5)$ in height steps of $1/20$ the powerseries for the schröder-function seem to converge nicely (at least in the first 64 terms) and give the values for the according fractional heights. That list of 20 values for fractional heights are then reshifted using inter-heights to cover the wider interval $z_k(-5)..z_k(2)$

This is shown in the first plot.



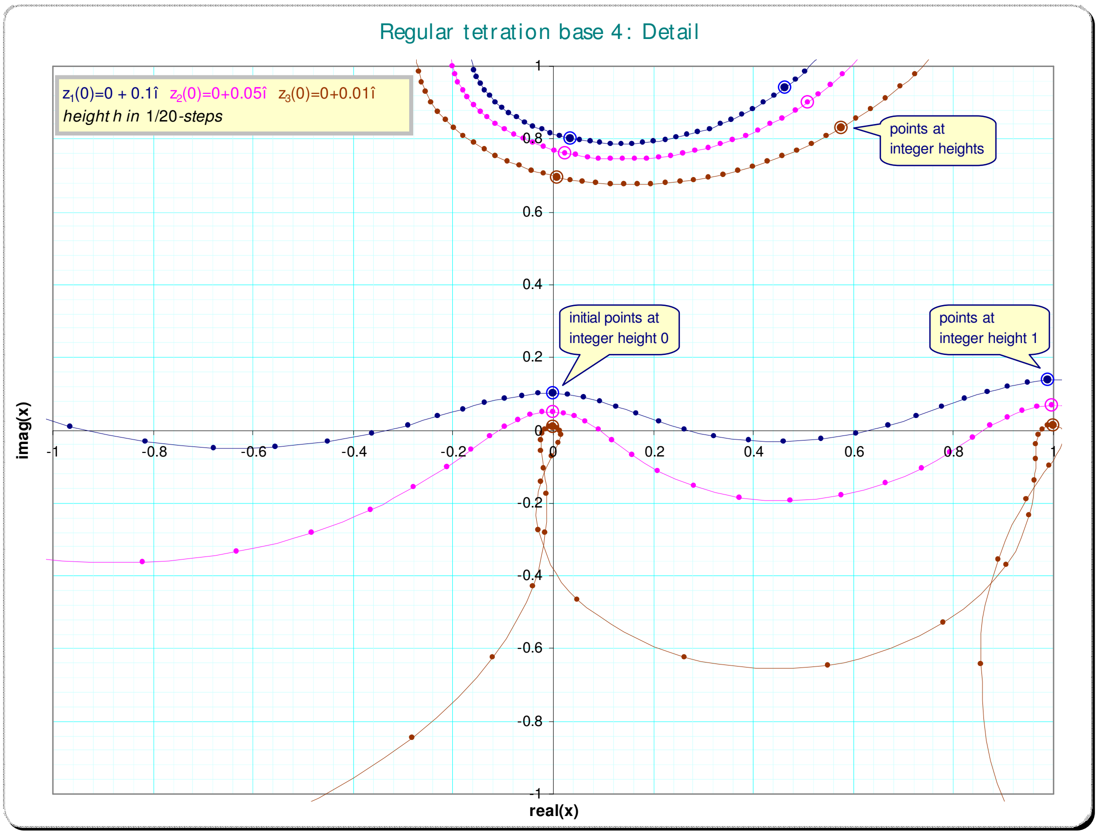
The blue line is the trajectory of $z_0(0) = 0.1\ \hat{\imath}$, the magenta line that of $z_1(0) = 0.05\ \hat{\imath}$ and the brown line that of $z_2(0) = 0.01\ \hat{\imath}$

The big points are the values of the integer iteration, which means, that value for which no interpolation is needed. We see near the fixpoint the begin of the spirals – this is the interval of heights $h=-6...-5$ relative to the three $z_k(0)$ for which the fractional iterates are computed using the <u>*regular tetration*</u>-method and the schröder-functions.

What is surprising is that the innocent looking spirals begin to wobble in the very first height-interval $h=0..1$ and even more in that of $h=1..2$. Also the crossings of the real axis to negative imaginary numbers looks very suspicious to me.

But well, here are two views into details:

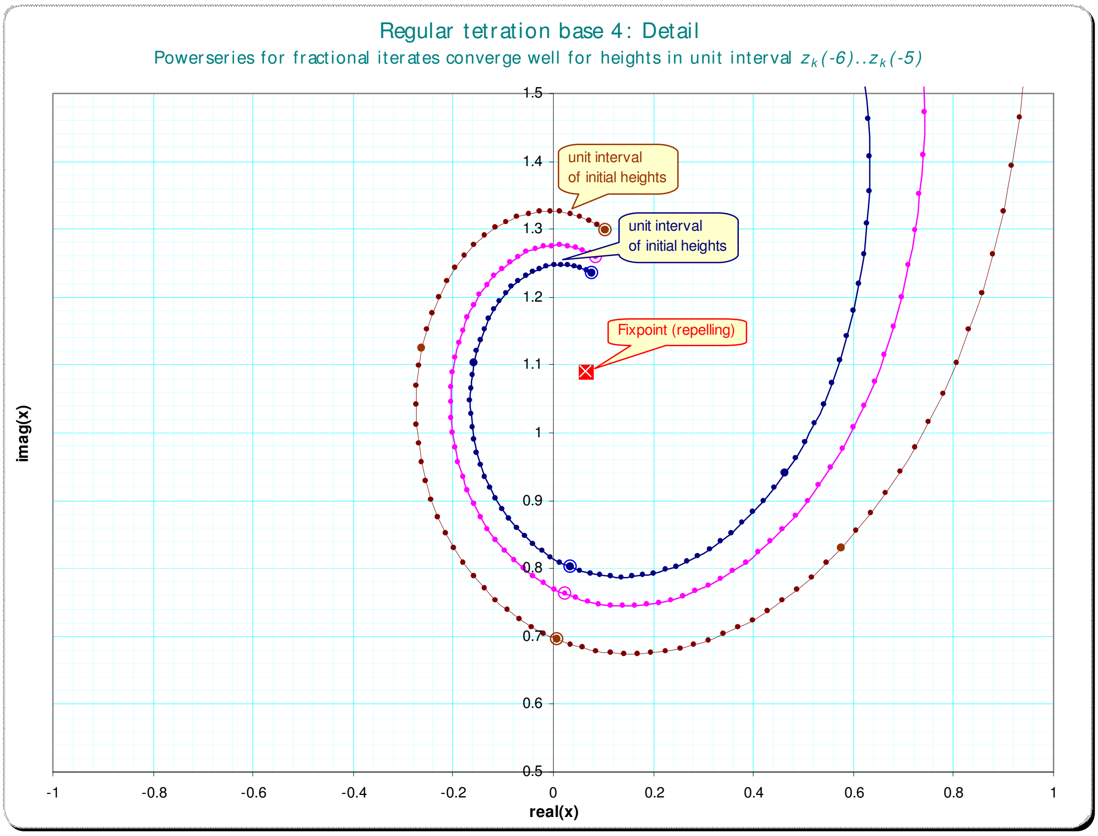
Detail in the height-interval *h=0..1*



which shows the wobbling, and even self-crossing of the $z_2(0)$-trajectory.

That special shape suggests, that it would be completely meaningless to use regular tetration with base *4* as some approximation to the real tetration of this base.

In the following we see the much more pleasing region in the near of the fixpoint. The marked unit-interval was used for the fractional tetration and from here the other height-intervals were computed using the functional equation $z_k(h+1) = 4^{z_k(h)}$ The computations were done in Pari/GP with decimal precision to 200 digits, so rounding errors should not be visible or significant for this interval of computations.



The log-polar /spiral-interpolation

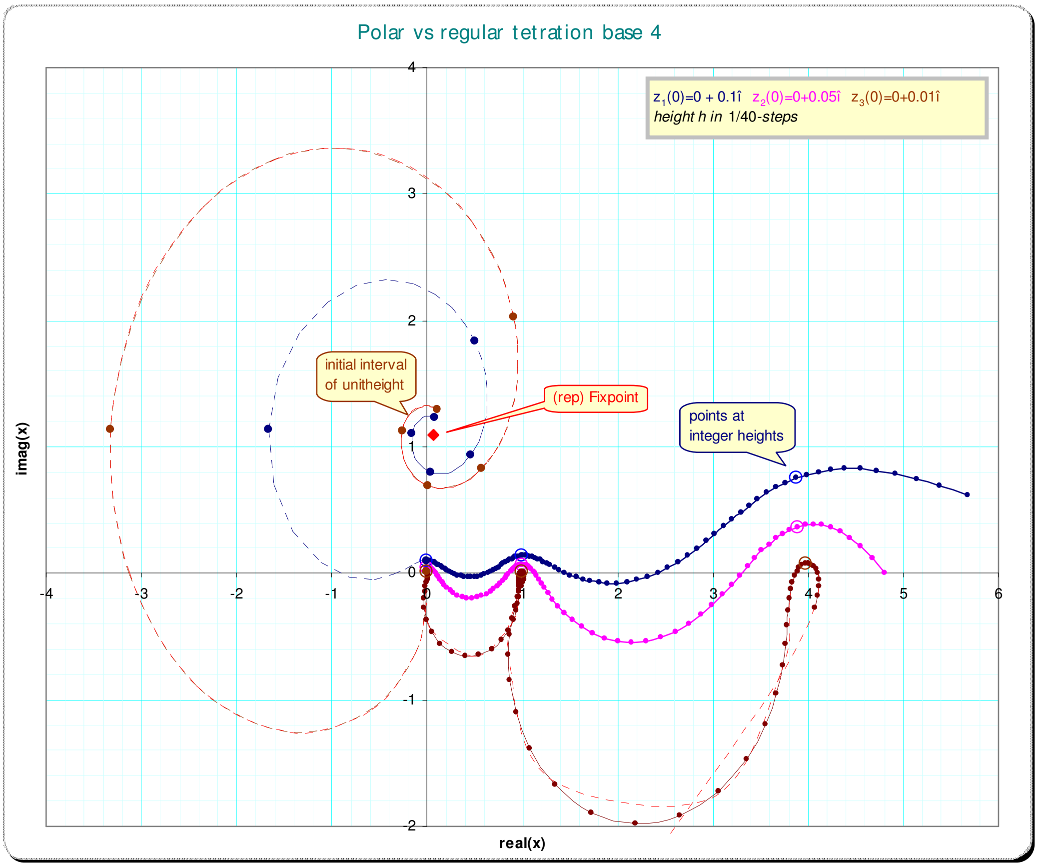
This idea follows from the observation, that the regular tetration near the fixpoint forms a smooth spiral – but that then there occurs an uneasy/unreasonable wobbling from $h=0$ towards $h=1$ and higher. It seems "unreasonable" because one should expect that the imaginary part of the iterates should always be positive or in other words, the trajectory should not cross the real axis. (I assume this as "reasonable" because we have symmetry around the real-axis and the picture of the lower half-plane should just mirror the picture and the trajectories of the conjugate initial points should not interfere).

Well, that crossing of the real axis may be due to some fine distortions in the numerical computation of the parts of the spiral near the fixpoint.

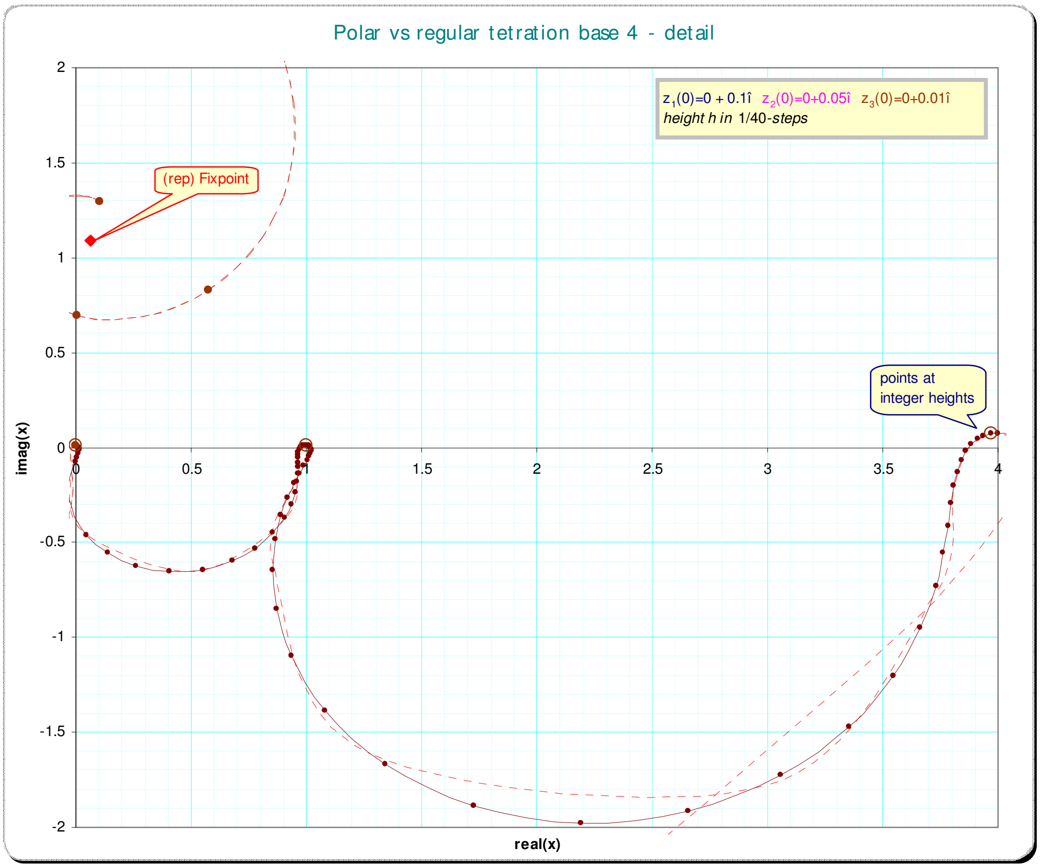
So I try a polar/spiral interpolation, where the coordinates at the integer heights $z(-6)$ and $z(-5)$ are written in log-polar-form with the fixpoint as origin. For a complex number $z = x+y\,\hat{\imath} = exp(\lambda + \varphi\,\hat{\imath})$ we have $[\lambda ,\varphi]$ where $\lambda = ln(abs(z))$ and $\varphi = arg(z)$.

Denoting the fixpoint as t I compute then $[\lambda_{-6} , \varphi_{-6}] = logpolar(z(h-6) - t)$ and $[\lambda_{-5} , \varphi_{-5}] = logpolar(z(h-5) - t)$, then interpolate between $[\lambda_{-6} , \varphi_{-6}]$ and $[\lambda_{-5} , \varphi_{-5}]$ linearly and rewrite this again as complex number. This provides a very close approximation to the regular tetration!

In the following plot the lines are nearly overlaid. To see some difference at all I used the more difficult initial value $z_2(0) = 0.01\,\hat{\imath}$ which provides more wobble in the $h=0..2$ – interval: only there we see that fine differences to the regular tetration! In the intervals $h<0$ the red line of the interpolation-trajectory and the brown line of the z_2 - trajectory are visually not discernable.



Here is a bit more detail which focuses the small differences in the region of *h=0 to 2* at the positive real-axis:



That suggests, that the linear interpolation using the logpolar-representation is a very good poor-mans-solution for some intervals.

Note, that for the ***tetration on the real domain only*** this reduces to linear interpolation of the logs of one unit-height-interval:

$$\lg_4(x(h)) = \lg_4(x(0)) + h \cdot \left(\lg_4(x(1)) - \lg_4(x(0))\right) \qquad \textit{for} \quad x\,\textit{real}, \;\; h = 0..1$$

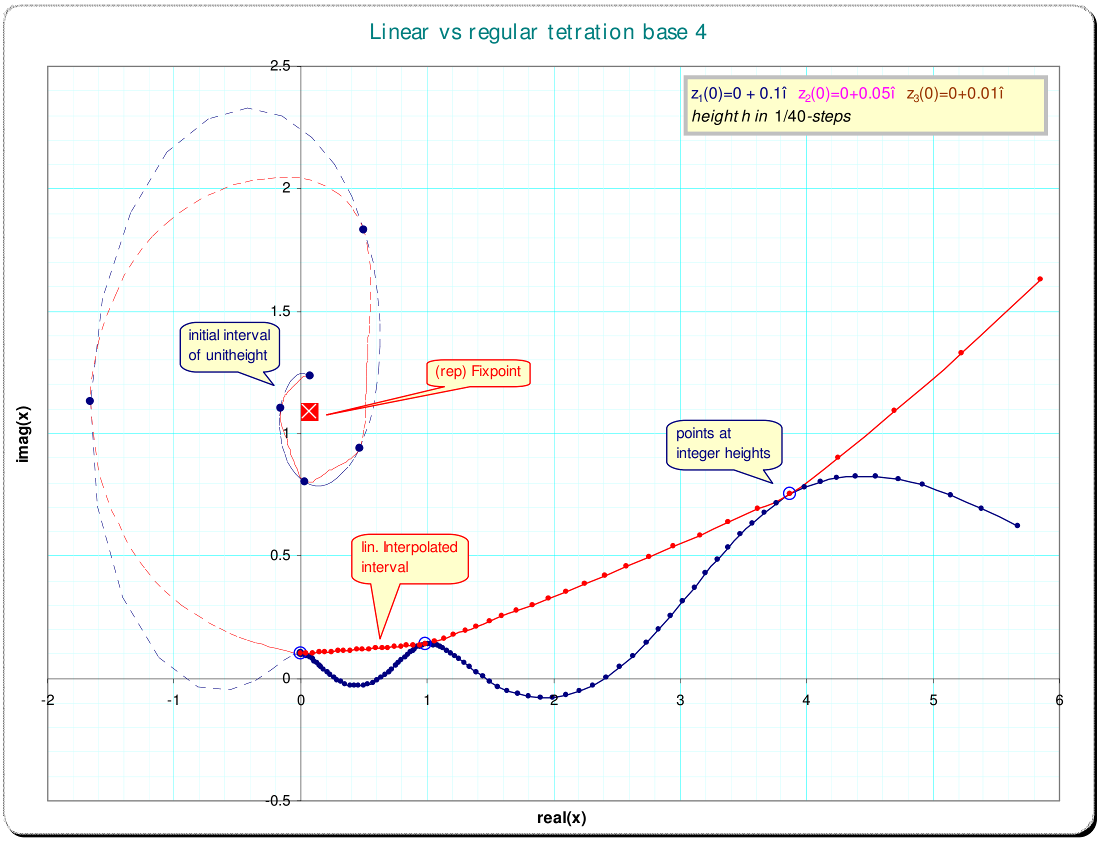
This linear interpolation of the logarithms should not be confused with the linear interpolation of the values themselves, as introduced below.

Linear interpolation

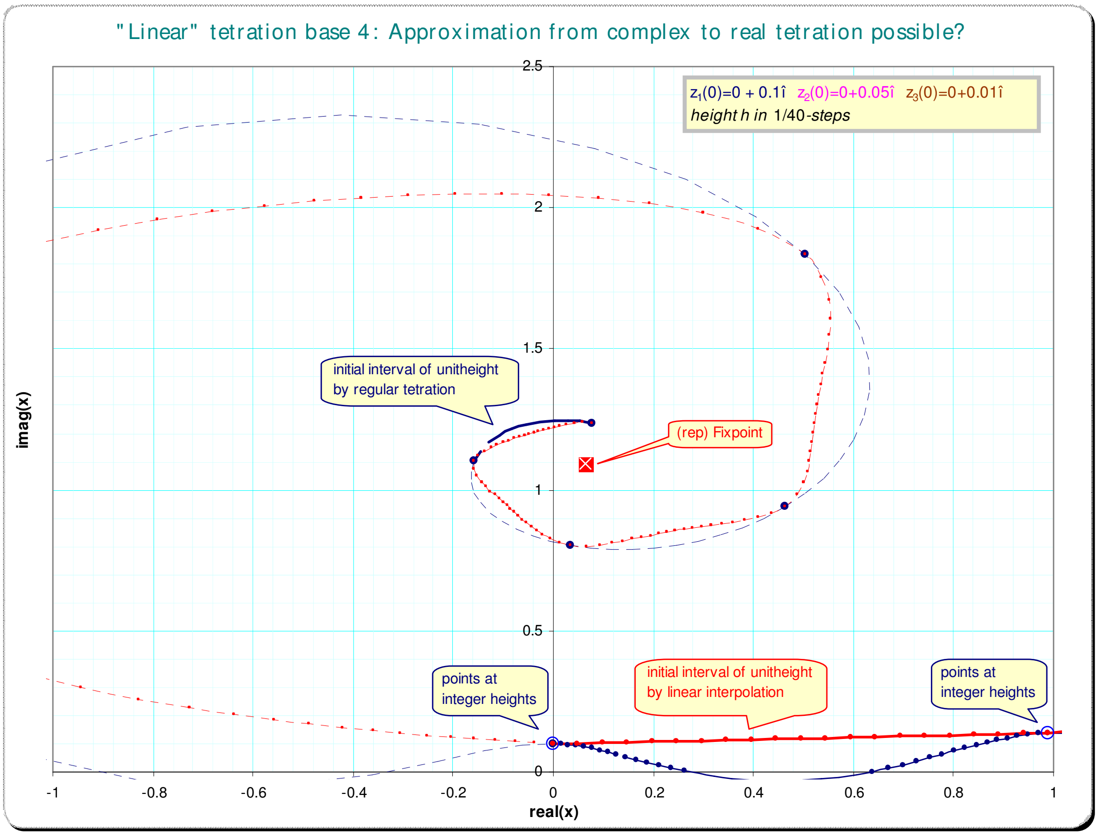
Perhaps the most naive model is the linear interpolation in one unit-interval and the extension of this to the other unit-intervals applying the functional equation.

I used the most suggestive unit-interval *h=0..1* for linear interpolation, such that $z_0(h) =_{linear} z_0(0) + h \cdot (z_0(1) - z_0(0))$ (for the graphics in steps of *1/20*) .

Then the interpolations for the other intervals were computed using the functional equation $z(h+1) = 4^{z(h)}$. This gives the *red curve* in the graph. Obviously the results are incompatible with the *blue curve* of the regular tetration.



A nice effect of this is that we have now no crossing of the real axis, and still the extrapolations into the other unit-intervals near the fixpoint are roughly spirals. However, the form of the spiral does not really become smoother when the fixpoint is approached. Here is a bit more detail:



We see, that the form of the spiral is so distorted, that we even get changes between convex and concave segments of the partial curves.

I think, this is not acceptable, so linear interpolation is surely no serious candidate for the tetration with continuous heights.

The polynomial interpolation (by eigen-decomposition of the truncated Carlemanmatrix)

But what, if we take some polynomial of higher order? This leads to the *diagonalization of a carleman-matrix*, truncated to the desired order. The mathematical model might be much more appropriate than that of linear interpolation (which can be seen as an polynomial of order 1) but is still imprecise because the truncation of the carleman-matrix introduces numerical and systematic errors in an intractable way into the computation of the eigen-system.

Just ignoring this, in the first view it is a straightforward method for the tetration with continuous heights. Also it gives very reasonable results in certain intervals! This is simply the eigensystem-decomposition of the truncated Carlemanmatrix for the exponentiation as we would compute it with any common mathematical software program.

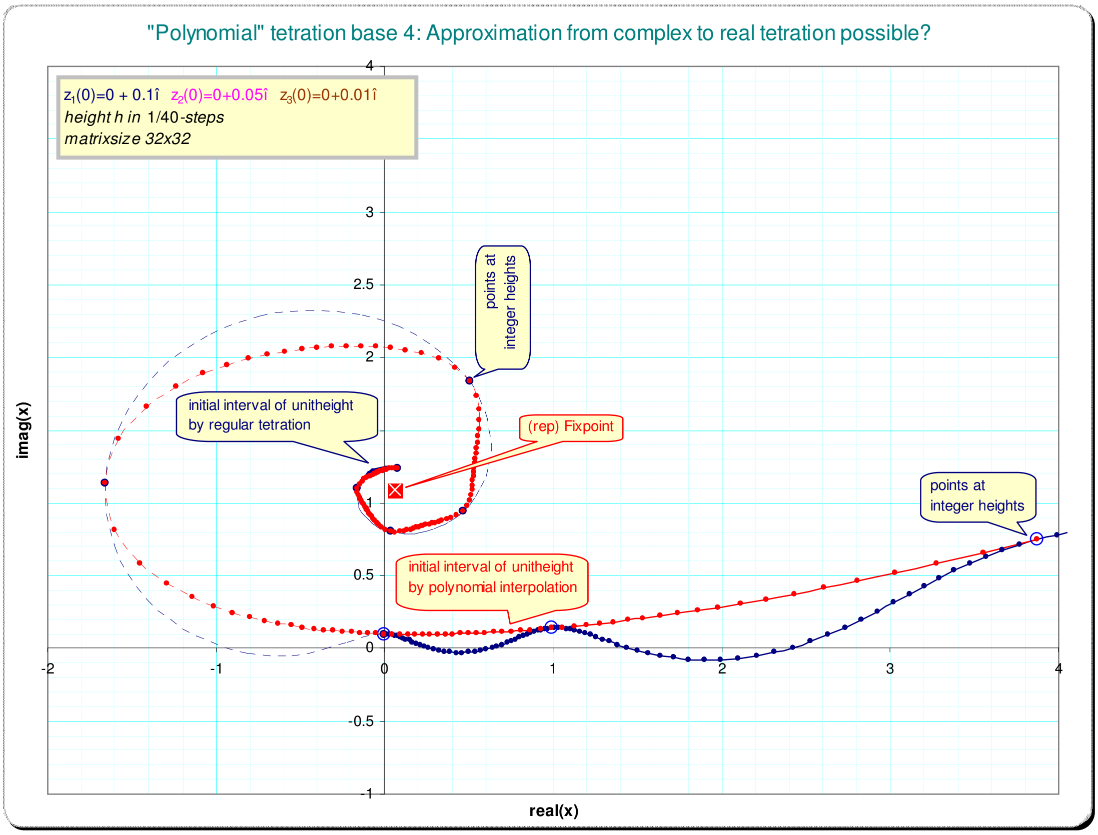
Assume some size *n=16x16* or *n=32x32* or as high as possible and very(!) high floatingpoint precision (we have to diagonalize a Vandermonde matrix!) and compute the fractional powers of the Carlemanmatrix/the matrix-operator by diagonalization and fractional powers of its eigenvalues. Then this matrix-operator in its fractional power give the coefficients for a power series for the tetration of fractional height in the sense which I tend to call *"polynomial interpolation"* because of the underlying idea of polynomials of finite orders, which are introduced by the characteristic polynomial of the finitely-truncated Carlemanmatrix.

In my opinion the diagonalization of the ***infinite*** Carleman-matrix for exponentiation should be the best step into the matter and is in principle realized in the "regular tetration"-method. However, for that method we introduce a recentering of the power series for exponentiation towards the fixpoint - and the method is thus fixpoint-dependent and its power series get complex coefficients and might also have very limited or zero radius of convergence. And thus, also the computed values are dependend on the choice of the selected fixpoint. As said above, it is also unsatisfactory because of the observations shown in the first picture.

But as a first step into the matter, truncating the Carleman-matrix to a feasible finite size, we can do a diagonalization, just quick & dirty. As said, this results in a matrix of real coefficients for the power series and thus the fractional iteration of real numbers towards positive infinity remains in the real numbers.

It could be expected (and is much interesting), that this polynomial interpolation is much smoother than the linear interpolation and inherits still the advantage, that the fractional iterates seem to not to cross the real axis, and can such way represent a more reasonable approximation (or even implementation) especially to the tetration over the real numbers.

Here is a plot of that polynomial interpolation using matrix-size *32x32* (where *800* digits float-precision were required for calculation of the eigen-decomposition). The ***red curve*** shows the trajectory of the tetration, implemented by polynomial interpolation, and the ***blue curve*** shows that of the "regular" tetration:



Unfortunately, and still unsatisfying, we have the distortion of the spiral near the fixpoint like in the linear interpolation.

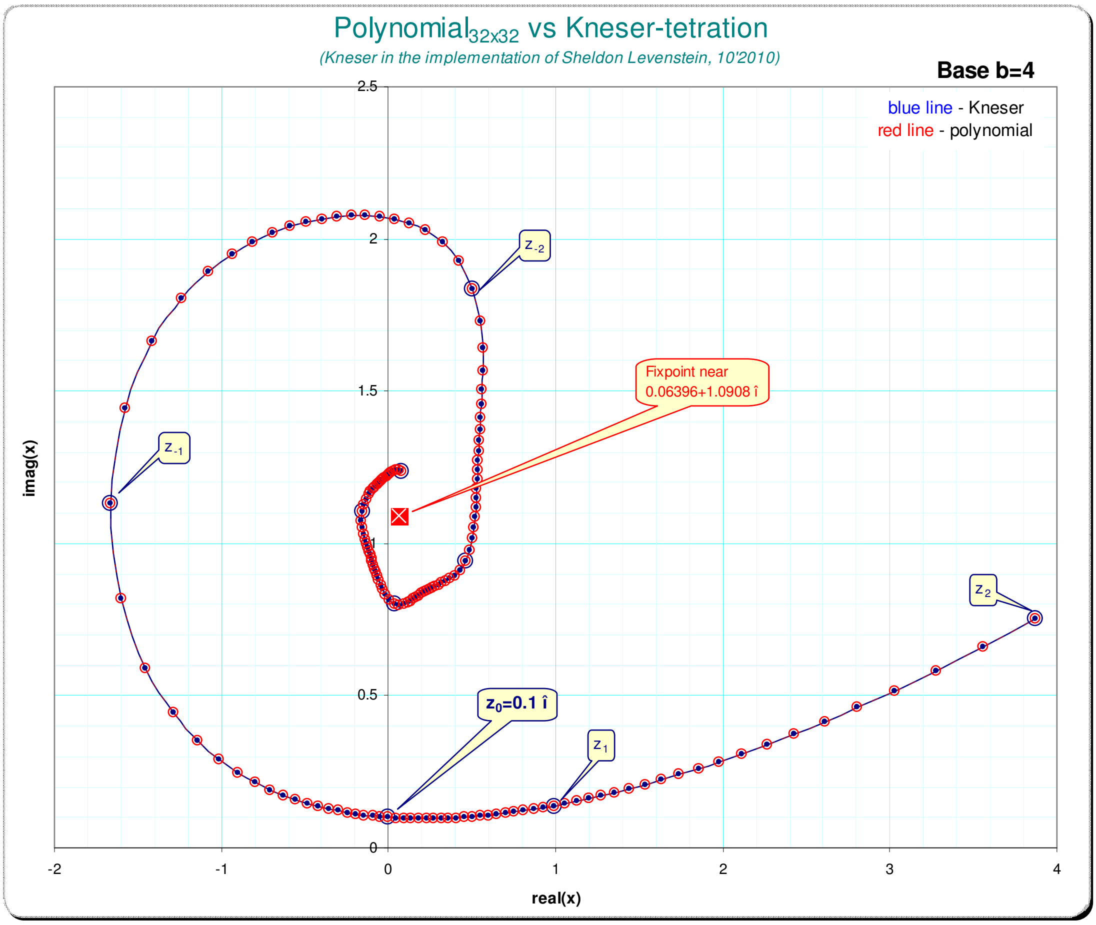
The more serious (because being "systematic") problem of this approach lies in the fact, that the eigenvalues (and thus their fractional powers) of this finitely truncated matrix-operator are heavily dependend on the truncation size, and I did not find any pattern which should allow the extrapolation to the case of the infinite size of the matrix. Only such pattern (and then the mathematical derivation) would allow to control and limit the unavoidable approximation errors.

The "Kneser" interpolation

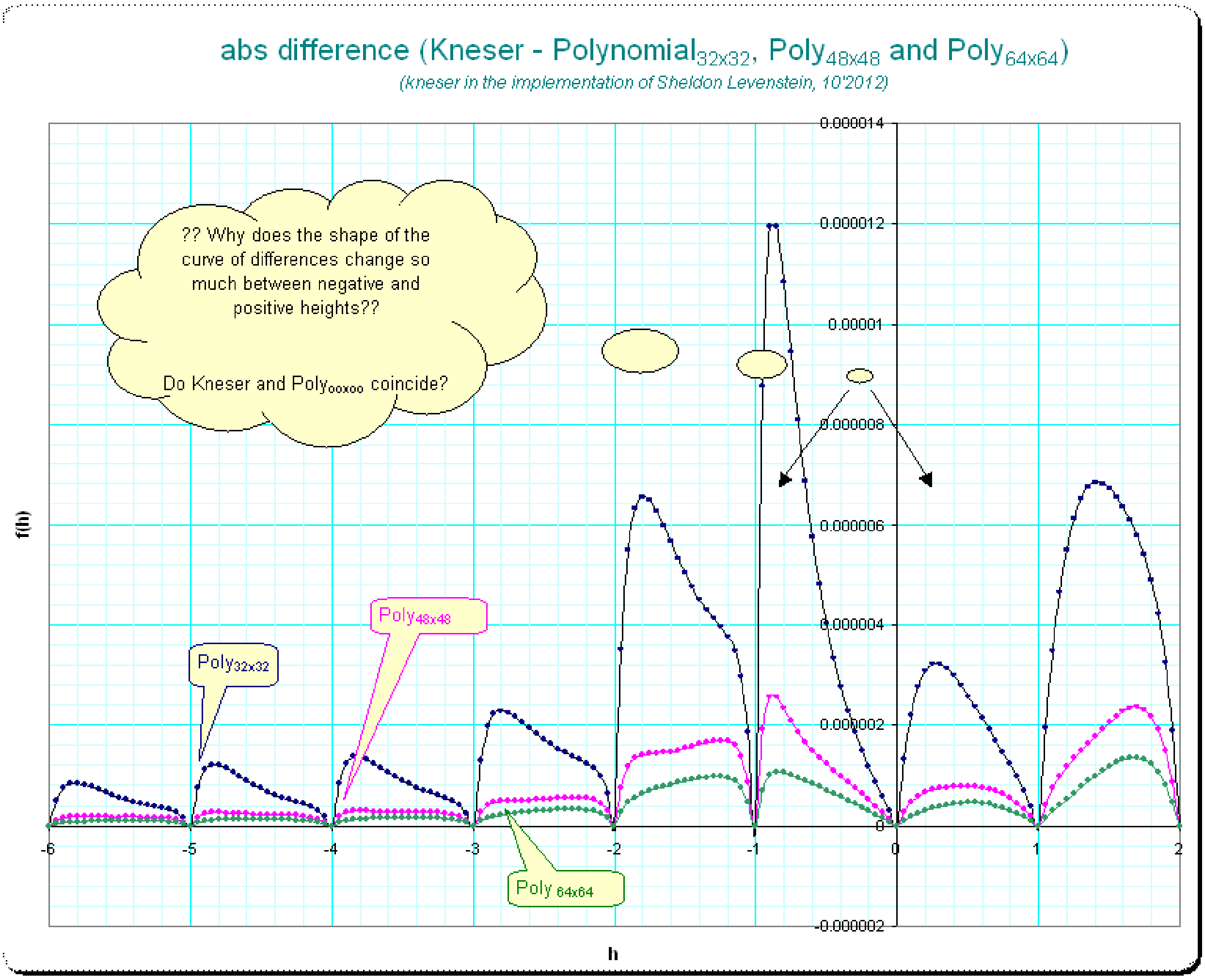
There exists a further proposal for the interpolation towards a continuous tetration which is based on a construction of Hellmuth Kneser in the 40ies of the previous century. I call this here "Kneser-interpolation". The mathematical method starts with the "regular tetration" (as given in the first chapter above) and improves it then by some mappings and iterations to modify the power series, but which I did not yet really understand. In the Tetration-Forum two members ("Mike4"and Sheldon Levenstein) provided code for the software package Pari/GP, which allows to actually compute values for that interpretation of tetration. (Here I used the code of the latter member) .

Surprisingly, the result, which starts with computations along the method of regular tetration and improves then the found power series, becomes finally very similar to that of the polynomial interpolation (as just described before). Differences in the current example were in the order of *1e-7* and are thus not visible in the plot.

Here is the picture, the ***red curve*** is the trajectory of the "Kneser-interpolation" and the ***blue curve*** that of the "polynomial interpolation"-method:



Here is a graph which shows the smallness of differences, in terms of absolute value of the differences of the values computed for each of the two methods. The blue line is drawn based on the computation with *32x32*-matrices the magenta line based on *48x48*- and the green line on *64x64*- matrices:



The different shapes between the negative and positive height unit-intervals might indicate some method-artifact, but which I cannot trace down at the moment.

But much more important: in the previous picture it is much surprising, that even the unwanted, unsatisfactory effect of inconstancy of the convex/concave-shape of the trajectory near the fixpoints which occurs as weakness in the "polynomial-interpolation" matches exactly that trajectory of the Kneser-interpolation. The second picture shows that this improves with size of the polynomial-method matrices.

Concluding question: ***Does the Kneser-method simply define the limit of the polynomial interpolation, where the truncation-size goes to infinity?***

Gottfried Helms, 12.10.2013