# Exp(Z) Fractal Algorithm

Jens Köplinger*

(Dated: 18 March 2008)

This is a quick detail dump about the algorithm and computation challenges for calculating the exponential function fractal in W space.

## I. INTRODUCTION

In $\mathbb{W}$, one can define $(w)^n := (w) \times (w) \times \ldots \times (w)$, where $(w)$ is a factor $n$ times under $\times$. Similarly, there is $(-w)^n := (-w) \circ (-w) \circ \ldots \circ (-w)$, where $(-w)$ is a factor $n$ times under $\circ$. In general, a set $Y^{(n)}$ is possible for any $Y \in \mathbb{W}$, to contain all products of $n$ factors $Y$, under $\times$ or $\circ$ multiplication:

$$Y^{(n)} := \{Y * Y * \ldots * Y,\ n \text{ times, where each occurrence of } * \text{ can be } \times \text{ or } \circ\}. \quad (1)$$

This offers the interesting possibility to define the exponential function through its Taylor polynomial, to receive a solution set that contains an infinite number of points:

$$\mathrm{Exp}\,(Y) := \left\{ \sum_{n=0}^{\infty} \frac{y_n}{n!}\ ,\ \text{where } y_n \in Y^{(n)} \right\}. \quad (2)$$

The notation $\left\{\sum \frac{y_n}{n!}\right\}$ indicates summation over all possible members $y_n$ of $Y^{(n)}$, with one $y_n$ from each $Y^{(n)}$.

As the sum for $\mathrm{Exp}(Y)$ is expected to be convergent, the infinite solution set for $n \to \infty$, plotted in the two dimensional plane, should exhibit qualities of a fractal[1].

## II. ALGEBRA

For every two points $A$ and $B$ with coefficients $A \equiv (a, b)$ and $B \equiv (c, d)$ in the two dimensional $\{1, w\}$ plane, we have the two multiplications:

$$A \times B = (ac - bd, ad + bc + bd)\,, \quad (3)$$
$$A \circ A = (ac - bd, ad + bc - bd)\,. \quad (4)$$

The two products $\times$ and $\circ$ are only different by the term $\pm bd$ in the $w$-part of the product.

## III. ALGORITHM

We want to compute the complete solution set of the $\mathrm{Exp}\,(Y)$ fractal (relation 2) and plot it in the two dimensional plane. The size of the plot (bounds, pixel resolution) is given. The following needs to be done:

1. Calculate the $Y^{(n)}$ pyramid until the contributions of the smallest terms (i.e., deepest iteration, highest $n$) cannot result in a change of the plot anymore (i.e., any deeper contributions are smaller than one pixel).
2. During (1.), prune the $Y^{(n)}$ pyramid off branches that cannot result in a change to the plot anymore, either.
3. Sum over all possible members of the $\mathrm{Exp}\,(Y)$ solution set, and set a pixel for each result.
4. During (3.), abort calculation on points that are outside the plot area, and can be shown to never contain a summation that would end up within the plot bounds.

---

*URL: http://www.jenskoeplinger.com/P

## A. Fractal calculation

In order to calculate the $\mathrm{Exp}(Y)$ fractal, the following input parameters are provided:

- An arbitrary point $Y = (x, y)$.
- Horizontal bounds $x_{min}$ and $x_{max}$, where $x_{max} > x_{min}$.
- Vertical bounds $y_{min}$ and $y_{max}$, where $y_{max} > y_{min}$.
- Resolution $R_x$ which is the number of pixels in $x$ direction; $R_x > 0$.

The following supporting parameters are derived:

- The resolution $R_y$ is the number of pixels in the $y$ direction is: $R_y = R_x\,(y_{max} - y_{min}) / (x_{max} - x_{min})$ and $R_y$ integer, and $R_y > 0$.
- The width of a single pixel $d_{pix}$ where $d_{pix} := (x_{max} - x_{min}) / R_x$.

## B. Calculating the $Y^{(n)}$ pyramid

From relation (1) we have:

$$
\begin{aligned}
n = 0:\ Y^{(0)} &= \{1\} && (5)\\
n = 1:\ Y^{(1)} &= \{Y\} && (6)\\
n = 2:\ Y^{(2)} &= \{Y \times Y, Y \circ Y\} && (7)\\
n = 3:\ Y^{(3)} &= \{(Y \times Y) \times Y, (Y \times Y) \circ Y, (Y \circ Y) \times Y, (Y \circ Y) \circ Y\} && (8)\\
&= \left\{Y^{(2)} \times Y, Y^{(2)} \circ Y\right\} && (9)\\
n = 4:\ Y^{(4)} &= \left\{Y^{(3)} \times Y, Y^{(3)} \circ Y\right\} && (10)
\end{aligned}
$$

and so on.

## C. Pruning the $Y^{(n)}$ pyramid

When calculating $Y^{(n)} = \left\{Y^{(n-1)} \times Y, Y^{(n-1)} \circ Y\right\}$ for a given $n$, we want to discard one (or both) members of $Y^{(n)}$ if they will not contribute anymore to the final plot (i.e., the total contributions of all subsequent summatrions would be smaller than one pixel; the resulting precision of the plot will thus be $\pm 1$ pixel.

In order to do so, execute the following:

1. Calculate all members of the set $\left\{Y^{(n)}\right\}$ for a given $n$.
2. For a given member $A := (x, y) \in Y^{(n)}$ calculate:

$$
\begin{aligned}
|A|_{max} &:= \sqrt{x^2 + y^2 + |xy|} && (11)\\
d_A &:= \exp |A|_{max} - \sum_{m=0}^{n} \frac{\left(|A|_{max}\right)^m}{m!} && (12)
\end{aligned}
$$

3. If $(1.5 * d_A) < d_{pix}$, then mark $A$ as end point in the $Y^{(n)}$ tree: This point will not contribute anymore to the $Y^{(n+1)}$ set when plotted at the specified resolution (pixel size $d_{pix}$). The factor 1.5 in this comparison should be configurable, as it is only an estimate.

### D. Calculate maximum contributions in the pruned $Y^{(n)}$ tree

In order to prepare the pruned $Y^{(n)}$ tree from section III C for plotting, the following helper variables will now be calculated for each $n$: Maximum and minimum contribution in $x$ direction ($\Delta x_{max}^{(n)}$ and $\Delta x_{min}^{(n)}$) and $y$ direction ($\Delta y_{max}^{(n)}$ and $\Delta y_{min}^{(n)}$). This will allow to determine whether or not a certain point will ever be plotted inside the requested plot area, or whether the iteration can be aborted for the current point.

The values will be obtained in the following manner:

1. Begin at the deepest level in the pruned $Y^{(n)}$ tree (i.e., with highest $n$).
2. Loop through all members in $Y^{(n)}$ and get $\Delta x_{min}^{(n)}$, $\Delta x_{max}^{(n)}$, $\Delta y_{min}^{(n)}$, and $\Delta y_{max}^{(n)}$ from the smallest and largest $x/n!$ and $y/n!$ parts. Treat $x$ and $y$ parts separately.
3. Now loop through all members in $Y^{(n-1)}$, and calculate $\Delta x_{min}^{(n-1)}$ by adding the smallest $x/(n-1)!$ value in $Y^{(n-1)}$ to $\Delta x_{min}^{(n)}$; get $\Delta x_{max}^{(n-1)}$ by adding the largest $x/(n-1)!$ in $Y^{(n-1)}$ to $\Delta x_{max}^{(n)}$; and similarily for $\Delta y_{min}^{(n-1)}$ and $\Delta y_{max}^{(n-1)}$.
4. Decrease $n$ by 1; if $n > 0$ then contine at (3.), otherwise abort.

Now we have for each level $n$ the maximum and minimum contributions that could ever arise when executing summation through over all combinations of members of a larger $n$.

### E. Plotting the $\mathrm{Exp}(Z)$ fractal

The plot will, schematically, calculate the set from equation (2) by executing the sum $\left\{\sum \frac{y_n}{n!}\right\}$, i.e. over all possible members $y_n$ of the pruned tree $Y^{(n)}$, with one $y_n$ from each $Y^{(n)}$.

Depending on the desired plot area, a substantial amount of points could lie outside the plot are. It is therefore important that iteration over combinations is aborted as soon as no futher point (i.e. from a higher $n$) may ever lie within the plottable area.

For a given $n$ and point $(x, y)$, the iteration can be aborted if any one of the following five conditions are met:

1. $n$ has reached its highest possible value (i.e. the deepest iteration level). If the point is in the plottable area, set a point.
2. All further iterations will only yield points that are to the left of the plot: $x + \Delta x_{max}^{(n)} < x_{min}$.
3. All further iterations will only yield points to the right: $x + \Delta x_{min}^{(n)} > x_{max}$.
4. All will be below: $y + \Delta y_{max}^{(n)} < y_{min}$.
5. All will be above: $y + \Delta y_{min}^{(n)} > y_{max}$.

## IV. CONCLUSION AND OUTLOOK

The resulting fractal will display the $\mathrm{Exp}(Z)$ solution set, in the preselected plottable area, with resolution $\pm 1$ pixel.

It will probably be valuable and interesting to create a series of screens, to merge the graphs together into a little movie. For example, a path could be parametricized by an integral number $M$ that runs from 1 through $N$. This way, one could e.g. calculate 40 pictures going up on the $w$ axis: $\mathrm{Exp}(f(M))$ where $f(M) := (M/4)\, w$ and $M \in \{1, \ldots, 40\}$. Or calculate the fractals on the unit circle in 50 steps: $f(M) := \cos(100\pi/M) + w \sin(100\pi/M)$. This could then be written e.g. into an animated GIF. When doing the animation, we want to write the iteration parameters into each frame: left-, right-, upper-, lower-bound; current point $(x, y)$.

---

[1] J. Shuster, J. Köplinger, *Elliptic complex numbers with dual multiplciation*. Preprint at `http://www.jenskoeplinger.com/P`.