

# INTERACTIVE VISUALIZATION IN COMPLEX ANALYSIS

**Matthias KAWSKI**

Department of Mathematics  
Arizona State University  
Tempe, Arizona 85287, U.S.A.  
e-mail: kowski@asu.edu

## ABSTRACT

Analytic functions of a complex variable exhibit some of the most striking beauty found anywhere -- but in the ages of black-on-white printed textbooks, this facet has been largely inaccessible to all except a few. Needham's recent text "Visual Complex Analysis" clearly demonstrates the power of a visual language as an organizing principle and as a useful tool to develop promising strategies for analytic arguments.

But modern technology allows one to go much further: We discuss selected implementations in computer algebra systems and especially in JAVA applets whose ultimate interactivity transforms every learner into an experimenter and researcher! Selected examples include zooming into essential singularities, mappings of the complex plane, winding numbers, and convergence of Laurent series.

We report how such tantalizing imagery transformed our own class, where amazing beauty led to inquiry and an urgent sense of "I want to know how/why that works". We contrast CAS-worksheets with model JAVA applets: On one side the user may modify and change everything -- but the algebraic-symbolic language of CAS worksheets usually requires a nontrivial "manual". On the other side, well-designed JAVA applets ideally require no instructions at all. Moreover, by using the "mouse" for input, and a graphic language for output, they take advantage of tactile, kinaesthetic and visual pathways that arguably have been much underutilized in mathematics teaching in recent centuries.

# 1. Introduction

The study of analytic functions of a complex variable is a center-piece of classical mathematics. Some of its distinguishing characteristics are its beauty and symmetry, which must have led many a researcher to go into complex analysis simply for aesthetic reasons. On the other hand, a large proportion of students in traditional introductory complex analysis classes never reach this level where they truly enjoy this beauty, but instead get stuck in a morass of algebraic-symbolic manipulations.

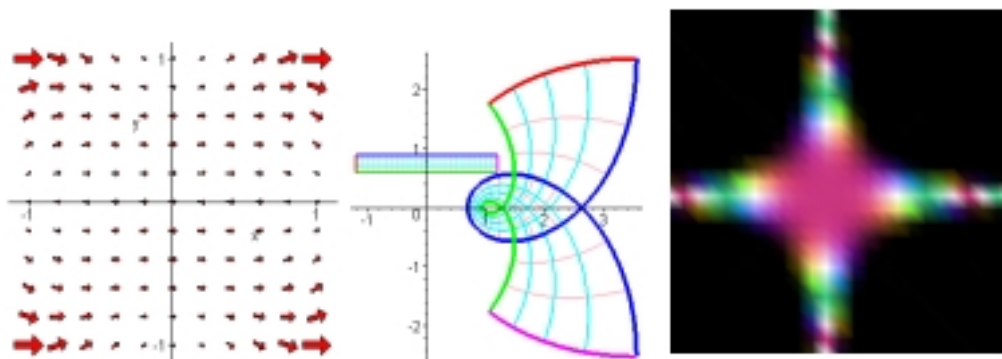
In the past decade we have seen calculus reform make much mileage from the “Rule of Three”: Students achieve a deeper level of understanding when offered (and forced) to address concepts from multiple perspectives, commonly algebraic/symbolic, graphical, and numerical. Complex analysis faces the difficulty that a simple dimension count precludes the naïve implementation of a graphical approach as each of domain and range requires two (real) dimensions. Thus it is no surprise that for well over a century complex analysis was almost exclusively approached from the symbolic/algebraic perspective. However, the last two decades have seen a proliferation of graphical perspectives of complex analysis. The mesmerizing images of fractals and Julia sets, especially the “Mandelbrot set” may well be considered the starting point of this revolution. While Julia sets were known for several decades before that, it was only the dramatic increase of computational power (combined with mathematical ingenuity) that allowed one to “calculate” these beautiful images that result from (complex) function iteration. More recently the spectacular book “Visual Complex Analysis” by T. Needham [13] has much further propelled the move to also include graphical aspects into the complex analysis courses. These days queries of standard search engines yield an abundance of articles, applets and various course materials on the World Wide Web that implement graphical approaches to Complex Analysis. An excellent starting point is the page “Websites related to Visual Complex Analysis” [18].

In this article we give a personal account of teaching experiences using home-made implementations in computer algebra system (CAS) worksheets and JAVA applets, and discuss the merits of key features of such visual aids. One focus is on the balance of minimal start-up-costs (ease of use) versus universality (can do everything). In particular, in some cases a single slide or a movie (animation) is appropriate, whereas in others the kinesthetic aspects of direct interactivity via the mouse appear to be essential. Many of the insights, experiences and course materials shared in this article date back to an introductory complex analysis class taught at ASU in the fall of 1999. The class-size was very small, but the student body was very diverse as this was not a required class for any degree program. While many of our implementations into MAPLE and JAVA may no longer be unique as similar efforts are proliferating, we believe that they still have valuable unique aspects worth to be shared.

## 2. Visualizing functions of a complex variable

As suggested in the introduction, one of the first challenges encountered with functions of a complex variable is the difficulty to “visualize” their graphs: Upon identifying the complex line with the real plane, the graphs are simply (real) two-dimensional surfaces in (real) four-dimensional space – too hard for almost all human brains. A classical alternative identifies a complex valued function  $f$  of a complex variable  $z=x+iy$  with the vector field  $(x,y)\rightarrow(\text{Re } f, -\text{Im } f)$ , the so-called Polya vector field [15,16], see figure 1.a for the example  $f(z)=\cos(z^2)$ . This approach

helps especially well to connect contour integration from complex analysis with line integrals from vector calculus. But until the emergence of powerful graphing software in the 1990s, plots of vector fields were just as rarely produced by hand and only a few static sample images were found in textbooks. However, modern courses in vector calculus and differential equations very much rely on plots of vector fields, and thus the Polya vector field is expected to become more important in complex analysis.



**Figure 1** Three different views of the function  $f(z)=\cos(z^2)$

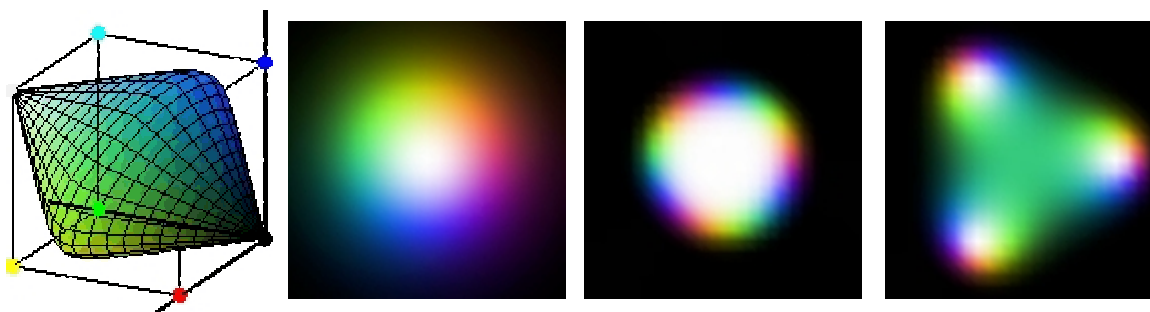
Here we shall concentrate on two other ways of visualizing functions of a complex variable. The first and traditional approach investigates images of specific curves (and regions) under the mapping  $f$ . For example, the complex exponential maps any line segment of length  $2\pi$  on the imaginary axis onto the unit circle. Similarly, it maps any rectangular region with vertices  $r$ ,  $R$ ,  $R+2\pi i$ ,  $r+2\pi i$ , ( $r, R > 0$  real) onto an annulus with radii  $r$  and  $R$  centered at 0. The study of such special cases forms an integral part of the introductory sections of any textbook, and is considered essential for getting an intuitive understanding of the elementary functions.

Today's computing tools make it very easy to automate such tasks. We distinguish those which require algebraic/symbolic definitions of the curves/regions to be mapped (in computer algebra systems, CAS, such as MAPLE, see e.g. [9] for sample implementations) entered from the keyboard, and JAVA applets [10] and similar programs, e.g. [2,11,17] that allow one to specify the input region by "drawing" it with the mouse. The main advantage of CAS implementations is that the learner must face e.g. explicit parameterizations of the curve/region and face the compositions of functions needed to obtain the image. See figure 1.b for the example  $f(z)=\cos(z^2)$ . The main disadvantage is that it takes quite some effort and time to modify the input. This serves to suggest a more thoughtful, planning approach as opposed to simply playing around. Nonetheless, students in our class spent significant time exploring the mappings using our CAS implementation [9], spending much time trying to understand where intersection points occur, reversal of orientation, "foldovers" that cause the boundary of the image not being a subset of the image of the boundary of the original region etc. A key effort that made this implementation so successful was the attention to detail, like the coloring of opposing edges by red/magenta and green/blue together with the associated internal grids in pastel tones (pink and cyan). This proved to be essential to help track features in more complicated mappings, and to forcefully convey the image of conformality (here preservation of orthogonal angles). On the other hand, JAVA implementations such as [10] and free-standing programs such as [2,11,17] provide much more immediacy and foster a much more playful attitude. They are great tools to get a class excited, but they generally require much more guidance by the instructor to again focus on relevant mathematical questions.

We see their main uses in getting a quick overall feeling, to quickly look up specific cases, and, most importantly, to discover interesting special cases which then warrant further investigation and theoretical follow-up study. A major benefit is that students generally are much more excited to study a question/problem/special case that they asked/posed/discovered themselves rather than a question from the textbook. For a minimalist implementation of the squaring map  $f(z)=z^2$ , and its (multi-valued inverse) see [10]. For professional large programs see the commercial software [11] and freeware [2, 17].

### 3. Graphing functions via colormaps

An exciting alternative approach, which yields immediate global images, uses colour maps. The basic idea is to assign to each point of the range a color, and then color each point  $z$  of the domain by the color of its image  $f(z)$ , see [12] for a detailed and more general description. While colormaps have been used for quite some time, e.g. for visualizing curvature on surfaces [9] via the color functions in MAPLE, the first use of color maps for complex functions on the WWW is attributed to Farris [6]. A common color map uses the polar form of complex numbers, mapping the magnitude to the interval  $[0,1]$  for brightness (e.g. zero is white and infinity is black) and mapping the argument to some “rainbow” (colorwheel). While the standard graphics on computers uses the RGB color model, we prefer a variation of the LAB model (as found in Photoshop) more suitable as the complex plane has natural 2-symmetries (e.g. conjugation), and thus the primary colors should be assumed by  $1, i, -1,$  and  $-i$ . In order to achieve bright colors (but fast calculations) one needs maps from the complex plane (Riemann sphere) into the color cube that “hug” the faces and stay away from the diagonal, compare figure 2.a for our construction.

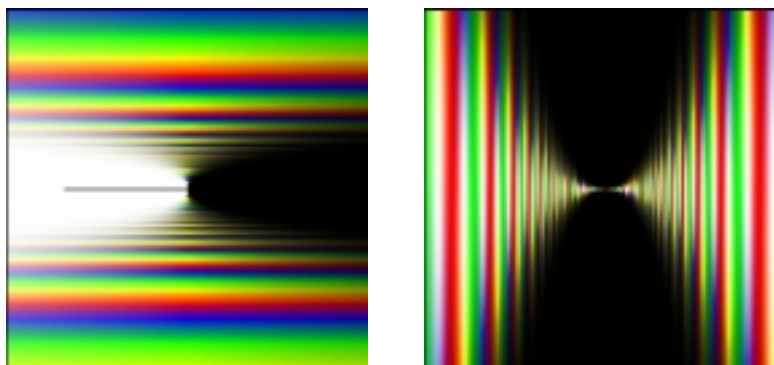


**Figure 2** Mapping the complex plane into the colour cube, views of  $f(z)=z, f(z)=z^3,$  and  $f(z)=z^3-1$

Figure 2.b. shows the resulting image of the identity (which defines the colormap). Figures 2.c illustrates the image of the map  $z \rightarrow z^3$  with beautifully shows that its degree is 3 (traverse the rainbow three times when encircling the origin one). The narrower colored band and the wider whitish region clearly correspond to the growth rate of the real function  $|z| \rightarrow |z|^3$ . The last image figure 2.d shows  $z \rightarrow z^3 - 1$  with its three simple zeros. Implementations of such colormaps may be found in freestanding packages such as [2,11], applets such as [10, 17], and CAS worksheets such as [9].

## 4. Zooming in on essential singularities

One of the more exciting uses of such color maps is to zoom in on essential singularities, such as those of  $z \rightarrow \exp(1/z)$ ,  $z \rightarrow \cos(1/z)$ , and  $z \rightarrow \sin(1/z)$ . Each of these has an essential singularity at zero. By Picard's theorem, each of these assumes every complex value (with at most one exception) infinitely often in every neighbourhood of zero. Upon first encounter most students, just as the author, react with disbelief: How could that be? What does that look like? We implemented a simple JAVA applet that allows one to successfully zoom in into these singularities, using colormaps as above, compare figure 3 for sample images.



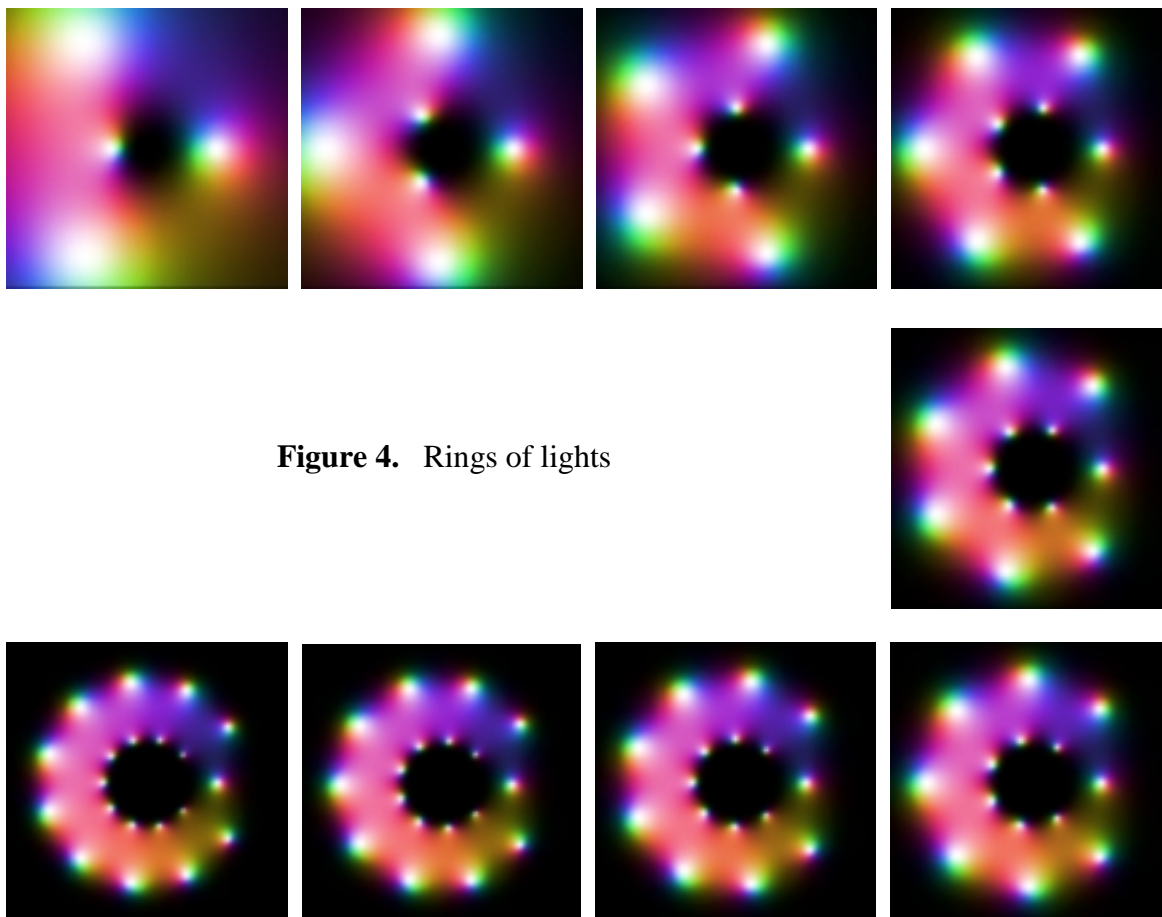
**Figure 3** Zooming in on the essential singularities of  $\cos(1/z)$  and  $\exp(1/z)$

The instantaneous reaction in class was one of intrigue – but this very quickly gave way to many mathematical questions being asked. They start with the difference in the colors at infinity of the three maps (different limits), one white and one black blob for the exponential as opposed to two black blobs for the cosine (hint: the Taylor series expansion of the cosine is an even series!). Everybody asked about the apparent parallel lines of constant color – but this was a standard homework exercise (like proving that the level curves of the real and imaginary parts, or of magnitude and argument are circles or lines etc.) However, when asked to prove their conjecture students react much more positively if this is their own observation as opposed to some assignment like “#34 from the exercises in the textbook”. A little deeper are questions like the conjectured geometric progression of lines of same color, and the “order of contact” of the two black (or the black and white) blobs in the pictures (basically of the level curves of the magnitude): Here alternating “zooming horizontally only” with “zooming in horizontal and vertical directions at the same rates” quickly suggested that the order of contact is two, i.e. like two circles touching each other.

A technical comment: Our applet [10] uses JAVA 2D-Graphics classes, and runs well in NETSCAPE 6, but not under NETSCAPE 4.7. Incidentally, it was a first try to use these classes and it performs reasonably well, delivering a new 256 x 256 image within a second on a typical PC, recomputing all function values and converting them to our LAB-like colormap. However, this speed is still insufficient to allow for slider-controlled continuous zooming or for more computationally intensive images such as in figure 4 (which were produced in MATLAB).

## 5. Convergence

One of the most beautiful sequences of images that we obtained in our initial explorations portrays the convergence of a Laurent series of a rational function. Recall, Laurent series expansions are similar to Taylor series expansions, but they also allow for negative powers. Depending on the choice of the expansion, one finds that the series converges on open disks, open annuli, or the complement of a disk. The behaviour on the boundaries generally can be very complicated.



**Figure 4.** Rings of lights

The series depicted in figure 4 shows truncations (at the same negative and positive orders) of that Laurent series expansion of  $1/((z-1)(z-2))$  that converges in the annulus  $1 < |z| < 2$ . More precisely, the images show the error terms, using different rescalings of the magnitude-to-color-map. (Without such rescalings, the annulus quickly approaches a white color.) What we expected was that as the order of the truncation increases, the regions inside and outside the annulus would become black, with something happening on the annulus itself. But we certainly did not expect the rings of lights. Clearly these are all simple zeros (degree one, single rainbows). Of course such observation, discovery has to be made into a conjecture, which then warrants further analysis as homework/project, trying to prove the statement, and more importantly, finding general hypotheses on the function under which the theorem holds true. Yet in class, we already took pride in the formulation of the conjecture: “As the order of the Laurent approximation increases, the approximating function interpolates the original functions at asymptotically uniformly spaced

points on the circles of convergence.” (Clearly the zero at  $z=3/2$  and the poles at the singularities  $z=1$  and  $z=2$  of the original function are exceptions.)

Technical comment: These images were produced in MATLAB using symbolically generated code from MAPLE and taking advantage of the superior speed for numerics and graphical rendering. (The new releases of the CAS such as MAPLE 7, should now be competitive, too.)

## 6. Classroom experience and conclusion

Student reactions in our class to, even the limited use of interactive visualization has been mostly enthusiastic. From the instructor’s point of view it appeared that, as a result, the class made more and faster progress on the traditional analytic aspects of the course, too. (But we do not have hard data, only anecdotal evidence comparing our class to that of prior semesters.) Our preference is that students (usually in pairs, or one volunteer in the front) directly, interactively explore – but there are plenty of occasions (such as the ring of lights), where even a single still image (transparency) gives rise to lively mathematical questioning.

While the instructor and author spent much time developing these materials while exploring the subject matter himself, we now expect that with the proliferation of dedicated software, slideshows, CAS worksheets and JAVA applets, one can achieve similar results with only minimal time investment. Moreover, we note that in a typical class only very few minutes, usually at the beginning (and sometimes also near the end) were devoted to graphical explorations. Typically such few minutes already raised so many mathematical questions that could barely be all addressed in the available class-time. The spirit was one of “doing mathematics”, in the truest sense of the word, from exploration, observation, formulating conjectures all the way to hard proof.

In the future we expect that such approach becomes ever more common place. At the same time we expect a further increase in the rate at which new dedicated software is developed, and we foresee a healthy competition between different implementations of related topics that pay special attention to various minute details. The color schemes matter as much as the placement of the buttons, and the choice when to specify input data via the keyboard (e.g. formula for  $f(z)$ , vertices for a rectangular region), sliders (parameter values), or directly with the mouse (drawing regions or “dropping” zeros and poles).

### REFERENCES

- [1] Abdo, G., Godfrey, P., “Plotting Functions of a Complex Variable”, Florida Institute of Technology, <http://winnie.fit.edu/~gabdo/>
- [2] Akers, D., “g(z): A Tool For Visual Complex Analysis”, Brown University, <http://ftp.cs.brown.edu/people/dla/ma126/intro.html>
- [3] Arnold, D., “Graphics for Complex Analysis”, Pennsylvania State University, <http://www.ima.umn.edu/~arnold/complex.html>
- [4] Banchoff, T., Cervone, D., “Understanding Complex Function Graphs”, Brown University and The Geometry Center, <http://www.geom.umn.edu/~dpvc/CVM/1997/01/ucfg/welcome.html>, also: Communications in Visual Mathematics, vol.1, no.1, (1998).
- [5] Bennett , A., “Complex Function Grapher “, Journal of Online Mathematics and Applications (2001), [http://www.joma.org/more/bennettmore.html?content\\_id=19437](http://www.joma.org/more/bennettmore.html?content_id=19437)
- [6] Frank Farris, “Complex Function Visualization”, Santa Clara University, <http://www-acc.scu.edu/~ffarris/complex.html>
- [7] Fishback, P., “Resources for the Teaching of Complex, Grand Valley State University, <http://www2.gvsu.edu/~fishbacp/complex/complex.htm>

- [8] Joyce, D., "Julia and Mandelbrot Set Explorer", Clark University,  
<http://aleph0.clarku.edu/~djoyce/julia/explorer.html>
- [9] Kawski, M., commented MAPLE-worksheet directory for complex analysis,  
<http://math.la.asu.edu/~kawski/MAPLE/commMAPLEindex.html#complex>
- [10] Kawski, M., JAVA and Complex Analysis,  
<http://math.la.asu.edu/~kawski/javaprojects/laurentdemo.html>
- [11] "f(z) - The Complex Variables Program, Lascaux Graphics,  
<http://www.primenet.com/~lascaux/11windem.html>
- [12] Lundmark, H., "Visualizing complex analytic functions using domain coloring, Linköping University, Sweden, <http://www.mai.liu.se/~halun/complex/complex.html>
- [13] Needham, T., "Visual Complex Analysis", 1998, Oxford University Press. <http://www.usfca.edu/vca/>
- [14] Orpen, K., and Djun, M., "Java Complex Function Viewer", University of British Columbia,  
<http://sunsite.ubc.ca/LivingMathematics/V001N01/UBCEexamples/ComplexViewer/complex.html>
- [15] Polya, G., Latta, G., Complex Variables, New York, 1974, Wiley.
- [16] Gluchoff, A., A simple interpretation of the complex contour integral, Amer. Math. Monthly, vol. 98 (1991) 641-644.
- [17] Santa Cruz, S., and Soares Fonseca, P., "BOMBELLI -A Java viewer for arbitrary, user-specified complex functions", Universidade Federal de Pernambuco, Brazil.  
<http://www.dmat.ufpe.br/~pgsf/bombelli>
- [18] Websites related to "Visual Complex Analysis", <http://www.usfca.edu/vca/websites.html>