# XaoS 4.0 and beyond: Fractals in mathematics education

Anurag Aggarwal, Zoltán Kovács, J. B. Langston and Jonathan Wolfe

**Abstract.** We report on recent improvements on the free fractal software tool XaoS, including the latest features added during the Google Summer of Code 2020 project.

**Keywords.** Fractals, mathematics education, Google Summer of Code, Qt.

## 1. Introduction

XaoS is a computer software tool that provides displaying various fractals and fast real-time zooming inside them. It was started by a young American programmer, Thomas Marsh and a Czech secondary school student, Jan Hubička in 1996 and further developed by various contributors world-wide, including the authors of this paper.

XaoS is well-known in the open source movement. It is a part of the Debian Linux system for several years. Its development was seemingly stopped in 2013 at version 3.6, but then, in 2019, big efforts were made to put in some technical improvements by the third author. Later, in April of 2020, version 4.0 was released that provides a professional graphical user interface based on the Qt library.

This paper first summarizes how XaoS was used during the last two decades in some educational settings:

1. We give an overview of usages of XaoS according to the available scientific literature.
2. We enumerate uses that are related to social communities, reported on the Internet.

The second part of the paper lists some recent improvements made on XaoS during the Google Summer of Code 2020 programme, implemented by the first author with the supervision of the second author. With these changes XaoS 4.1 and 4.2 were released in July and September 2020, respectively.

The third part of the paper points to possible future work.

## 2. The past

The early 1990s can be identified as a 'fractal boom' in the history of mathematics. Popularization of fractals was largely the consequence of the free availability of the software package *Fractint* [7], developed by Bert Tyler, and later many others. Fractint quickly became the de facto database of fractal formulas, and its open-source access ensured that even non-mathematicians were able to understand and visualize several features of fractal geometry.

As a next step, Thomas Marsh and Jan Hubička started to develop a free fractal zoomer in 1996, *XaoS*. It became popular because of Hubička's revolutional algorithm [2] that allowed real-time zooming even on slow computers.

### 2.1. Escape time fractals

*Escape time fractals* are fractals generated using escape time algorithms. *Escape time algorithms* compute fractals by repeatedly applying transformation to a given point in the plane. The obtained series of transformed points is called the *orbit* of the initial point. An orbit diverges when its points grow further apart without bounds. A fractal can then be defined as the set of points whose orbit does not diverge to infinity.

Consider the transformation $z' = z^2$ in the complex plane $\mathbb{C}$. Recursively applying this formula to a complex point $z$ an orbit is generated that converges to 0 when $|z| < 1$, diverges when $|z| > 1$, and forms a circle at $|z| = 1$. In this case, a simple "fractal" would be the set $|z| \leq 1$, that is, the unit disk.

Things become more complicated when an arbitrary constant is added to the transformations. Consider $z' = z^2 + c$ (where $c$ is a complex number). For such transformation, it is impossible to predict for which points their orbit will diverge to infinity or not.

A *naive escape time algorithm* uses the coordinates of each point as starting values in a brute force iterating calculation. The result of each iteration is used as the starting value for the next. These generated values are checked if they diverge. One can mark the points as white if they are diverging to infinity (escape) and black if they are not (they remain within a bailout value). A colorful image can also be generated by making the color of a pixel correspond to how fast its orbit diverges, i.e. the number of iterations required to escape. This gives a visual representation of how many cycles were required before reaching the escape condition.

The *Mandelbrot Set* (Fig. 1) is one of the most popular fractals. Its general equation is defined as

$$z' = z^2 + c$$

where the initial value of $z$ is 0, and $c$ corresponds to the represented complex number for each pixel. In other words, the Mandelbrot Set is defined as follows:

$$\mathcal{M} = \{c \in \mathbb{C} : z_0 = 0, z_{n+1} = z_n^2 + c \ (n \in \mathbb{N}), |z_n| \not\to \infty\}.$$

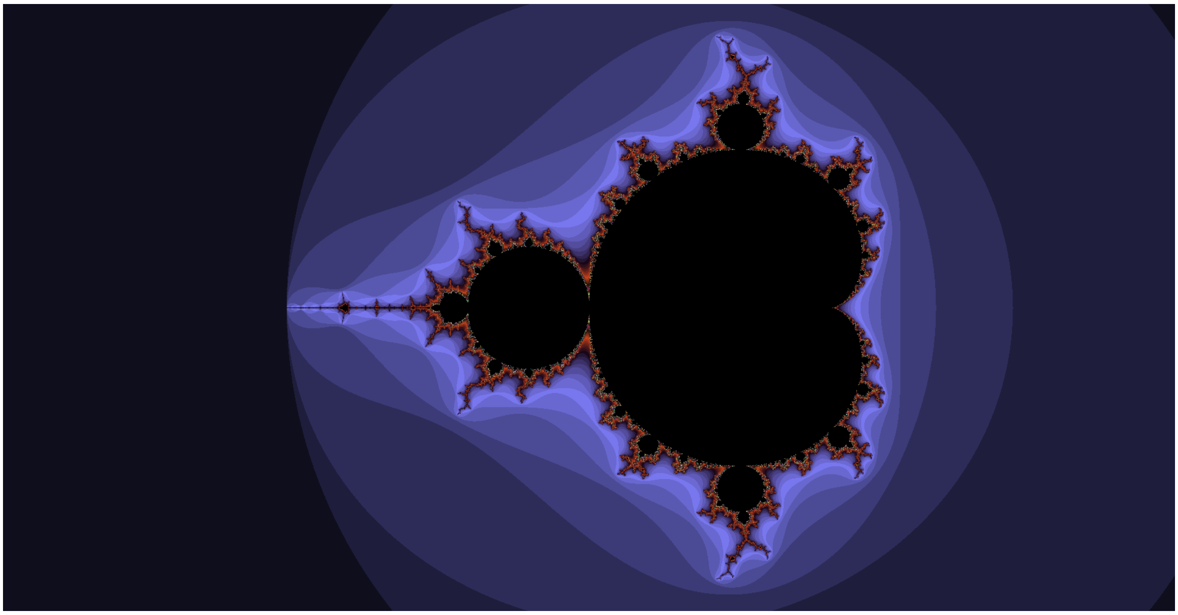

Figure 1. The Mandelbrot Set displayed in XaoS

It can be shown that 2 is a bailout value for the Mandelbrot Set, that is $c \notin \mathcal{M}$ if and only if $|z_n| > 2$ for some $n \in \mathbb{N}$. (See [4] for a proof that can be presented for young learners as well.) We note that XaoS uses the programmatic notation $z = z^2 + c$ (or in another form, `z^2+c`) to express that the value of $z$ is computed by iteration.

XaoS is capable of displaying several other fractals that are of escape time type, including different variations of the Mandelbrot Set, the Sierpinski Triangle, the Koch Snowflake, the Newton fractal, Barnsley's formulas, and many others.

### 2.2. XaoS: A Debian package

Wide-spreading of XaoS was boosted up when the freely available Debian Linux system added it as an official package. According to `https://sources.debian.org/patches/xaos/`, XaoS 3.0 was already an official package of Debian GNU/Linux 2.0 (Hamm), being released on July 24th 1998.

Being a Debian package, XaoS became popular among open-source contributors, and several minor improvements were added during the next years. As the primary authors did not add new functionalities in the 2000s there were not major features added, and development of XaoS remained as a maintenance-only work.

It is difficult to be objective to tell if XaoS is still among the most popular fractal software nowadays, but, according to some unbiased data from the Debian project we can support this opinion.

Debian's popularity contest 'popcon' results, available at `https://qa.debian.org/popcon.php?package=xaos` shows XaoS usage statistics for users who participate in popularity contest survey. Similar data can be obtained for other popular fractal visualizing tools, including *Mandelbulber*, *Mandelbulber2*, *Xfractint*, *Fraqtive*, *FractalNow* and *Fractgen*.

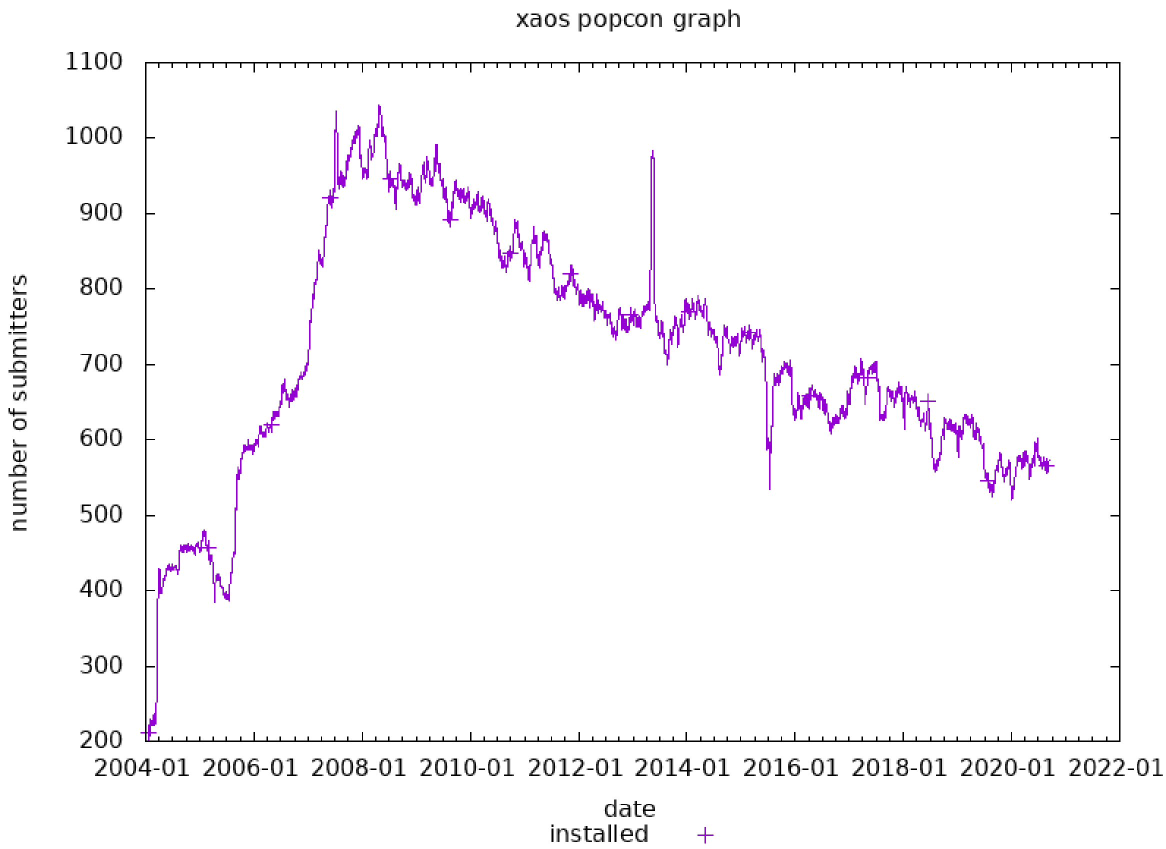


FIGURE 2. Popcon results for XaoS

According to Fig. 2:

1. The number of users were highest during the year 2008-2009.
2. The number gradually decreased until 2014 after peak in 2009.
3. After the release of XaoS 3.6 in 2014, number of users suddenly increased again.
4. The average number of users is around 600 in 2020.

On the other hand, the number of users participated in the popcon survey is currently 572 for XaoS, 72 for Mandelbulber, 173 for Mandelbulber2, 157 for Xfractint, 197 for Fraqtive, 128 for FractalNow, and 107 for Fractgen.

These statistics, of course, do not contain data from usages of fractal software in other Linux variants, and in more popular operating systems like Windows and MacOS.

## 3. Didactic use of fractal software

Fractal geometry is seldom part of secondary school curriculum, even if it was envisioned in some papers in mathematics education in the 1990s (see [5, 8, 3]). Its reason has several roots, mainly its novel character, the difficulty of precise definitions, and some technical requirements. Uncovering the full background is out of our scope in this paper, but we would like to highlight some examples of using fractal geometry to improve math education.

- Ben Sparks, well known as mathematician, musician and speaker at University of Bath, United Kingdom, uploaded his first applets to GeoGebra's free online database on exploring the Mandelbrot Set, back in 2013. They include a Mandelbrot painter and an applet that shows the iteration orbits (see `https://www.geogebra.org/m/BUVhcRSv`).
- Bernie Innocenti, a Linux systems engineer, ported XaoS to the OLPC (One Laptop Per Child) project in 2008 (see `http://wiki.laptop.org/go/XaoS`). On a feedback page `http://dev.laptop.org/ticket/8026` it was recorded that some users were complaining that Spanish translation was not included in that version, but this would have been mandatory in Peru. This variant of XaoS already included an option to enter user formulas that was also added to the mainstream version 3.3.
- Second author of this paper lead two series of lectures in 2019 and 2020 at Epsilon Camp (`https://epsiloncamp.org`) for gifted students at age 11-12. XaoS was used to visualize several fractals, including the Mandelbrot Set, the Koch Curve and Snowflake, and the Sierpinski Triangle and Carpet (see [4] for a report on the camp in 2019).

### 3.1. The Fractal Foundation

The XaoS program was instrumental in the formation of the *Fractal Foundation*, an educational nonprofit organization that officially began in 2003 in New Mexico. Beginning in 2001, fourth author had been invited by a friend to visit her kindergarten classroom to teach the students about fractals. Borrowing a video projector, he used a laptop and XaoS to take the children on a journey zooming deep into the Mandelbrot Set. He then compared the spirals and branching patterns in the algebraic fractals to slides of naturally occurring fractal patterns such as trees, lightning and seashells—and the children were captivated. The teacher then told other teachers about the "amazing, incredible fractals" and these teachers began inviting Wolfe to visit their classrooms to share the exciting fractal content with many more students. Within two years, this had become such a successful phenomenon that Wolfe and a few of his colleagues incorporated the business as a nonprofit organization to be able to apply for grant funding to expand their reach.

During the subsequent years, the Fractal Foundation created a significant body of didactic material, including presentations, lessons, and interactive "Fractivities" that are targeted at students ranging from kindergarten to high school. The Fractal Foundation educators have taught fractals to over 60,000 children and over 700 teachers. The XaoS program is still at the heart of all the presentations they do in schools. A standard presentation begins with several minutes of projected XaoS fractal zooms, generated in real time, while the students—most of whom have never seen anything like this—look on in amazement. The presenter takes the class on a variety of journeys into different parts of the Mandelbrot Set, showcasing the diversity of patterns that are discoverable in the fractal. The students are enthralled by the journey into the infinite, and demand to learn how it works. With this motivational context, the presenter then explains the simple repetitive process responsible for generating the Mandelbrot Set. As soon as children know how to square numbers, they can understand the basic mechanism that generates the fractal. The key insight involves determining the fate of different starting values after repeated iteration through a nonlinear formula. Even without referencing complex numbers, children can easily understand the difference between starting values that get ever larger when repeatedly squared ($> 1$) versus those that get smaller (fractions). Exposing young people to algebraic formulas before algebra is normally taught in the curriculum is a powerful way to instill a positive attitude toward learning math. The students learn to view algebra as something exciting and empowering that they look forward to learning, rather than as something scary and difficult as is often currently the case.

Fractals also provide exciting educational opportunities outside the classroom. One of the more interesting nontraditional educational outreach activities is to accompany live musical performances. The Fractal Foundation uses the XaoS program to create and project real-time fractal animations choreographed to live music. The software allows the explorer to adjust various parameters including zoom speed, rotation speed and color cycling frequency such that they can be roughly synchronized with the rhythms of the music. This can be done with a variety of musical styles ranging from rock and roll to jazz, hiphop, electronica, and classical. In any of the venues where this fractal "VJ'ing" has been performed, ranging from concert halls to nightclubs to raves, audience members invariably come up to the projectionist after the show and are overcome with curiosity to learn about what they are seeing. Engaging conversations about math and science often occur. One of the most commonly expressed sentiments is: "If I had seen this sort of thing when I was in school, I would have loved math!"

One of the most impactful ways the Fractal Foundation uses XaoS is the Fractal Challenge, a project based in New Mexico and funded by the city of Albuquerque. Students are invited to create their own fractal artworks using XaoS and to submit their favorite image to the Fractal Challenge. The project organizers convene a committee to select the eight most interesting and beautiful images which are declared the winners. These chosen artworks are then rendered at extremely high resolution and reproduced at large scale and mounted on the sides of buildings (see Fig. 3 and 4). These images are about 10 meters on a side and decorate schools, museums and parking structures, earning Albuquerque the nickname "the Fractal Capital of the World." More importantly, the project elevates the student winners into role models and heroes in their schools and communities. Elisabeth, a 9 year old winner of the Fractal Challenge, said "I can't wait to learn algebra, so I can make better fractals!"

Finally, the Fractal Foundation engages in a very impactful public relations campaign to change the perception of algebra using fractal artwork. The highest visibility of these are full sized hotair balloons printed with high resolution imagery of the Mandelbrot Set. These balloons use dye sublimation technology to print the high resolution imagery onto the balloon fabric before assembly, creating a dramatic visual effect. The Fractal Foundation has thus far produced two of these flying fractal art balloons which appear at schools and festivals around the world, exposing hundreds of thousands of people to the inspiring beauty of fractal mathematics (Fig. 5, 6).

## 4. XaoS 4.0 and beyond

**Technical overview in a historical perspective**

XaoS is quite an old codebase, originating in 1996. It was originally designed for extreme portability, supporting Linux, DOS, Windows, MacOS, and even more obscure operating systems like OS/2, BeOS, and Plan 9. The lower level code was encapsulated in a modular driver system that allowed a minimal amount of code to be written in order to port it to a new operating system.

When the third author initially got involved with the project in 2006, he found this architecture quite elegant, and it allowed him to quickly write several new drivers for Mac OS X, GTK+ and

Figure 3. A 3rd grade winner of the Albuquerque Fractal Challenge watches as her winning artwork is installed on her elementary school

Qt, with minimal effort. The downside is that XaoS suffered from the “lowest common denominator”, making it difficult to add features that are not supported by every platform that it targets. This means it has not kept up with modern expectations and the user interface is sometimes unusual. Also, the code was riddled with `#ifdef`s, which made it hard to understand and modify.

Thus, when being reengaged with the project in 2019, there was a decision made to abandon the goal of extreme portability in favor of standardizing on Qt, which will allow current developers to write a modern GUI targeting the three major desktop platforms (Linux, Mac, and Windows) with less code and less effort. Qt also opens the door to mobile platforms, although App Stores present licensing challenges with the GPL. We remark that, among others, the Raspberry Pi platform is also supported.

Since making this decision, a major refactoring effort was undertaken. First, the old drivers for everything but Qt were deleted. Next, the Qt driver code with the generic UI code was merged, slowly moving more functionality into Qt’s purview. The external dependencies (`libpng` and `gettext`) were replaced with Qt equivalents and modern text rendering was added which supports TrueType fonts and non-Latin scripts. Lots of old cross-platform code have been removed that is no longer needed when POSIX compatibility is much more universal (even on Windows), as well as features that are no longer necessary, such as the “Ugly UI” and dithering support for non-truecolor visuals. All the existing C sources have been changed to C++ and any compilation errors that ensued were fixed.

These changes have reduced the overall lines of code in XaoS from 86,000 at its peak to 29,000 today, while still retaining all the important functionality. Thus far, these changes have not resulted in a lot of user-visible features, but—we believe—they are laying the groundwork to much more productive development of features going forward.

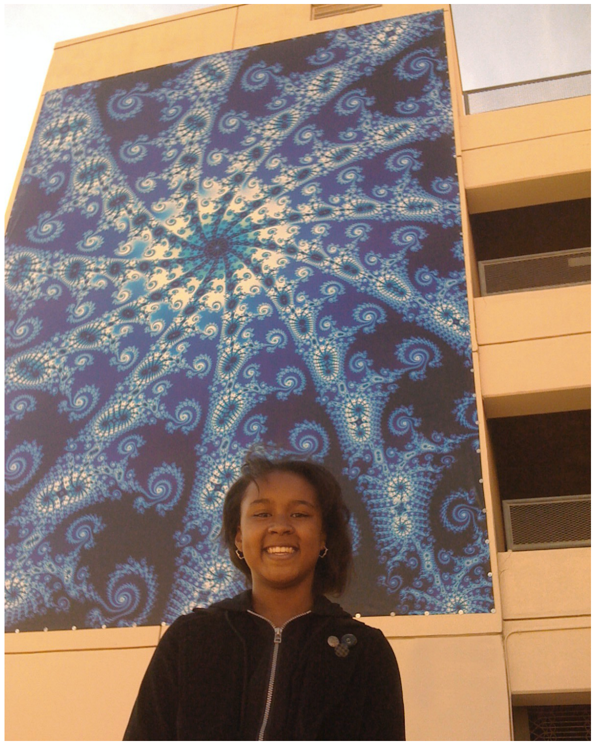

FIGURE 4. A 5th grade winner of the Albuquerque Fractal Challenge standing in front of her winning artwork printed on a parking structure

**Version 4.0**

XaoS 4.0 included just a few new features that had mathematical or didactic relevance, because it focused on finalizing the graphical user interface (based on the Qt library, see `https://www.qt.io/`). They include:

- Standardized menu entries. This was helpful for users familiar with other software, to get familiar with XaoS as well.
- TrueType text rendering started full Unicode support and therefore non-Latin alphabets and possibility to show texts in many new languages.
- Hindi, Icelandic, Russian, Serbian and Swedish translations.
- Zooming is already possible with the mouse wheel.
- Mouse cursor has been changed to cross.
- Iteration number $N$ can be used in user formulas. This opens new horizons in exploring new fractal types (see Fig. 7).

XaoS 4.0 was released on 1 April 2020, mostly by the work contributed by the third author, and some minor contribution by the first and second authors.

**Version 4.1**

Version 4.1 was released on 1 July 2020, as a first public presentation of the contribution of the first author—he was selected to be a student in the Google Summer of Code 2020 program, by joining the GNU Project. (The role of GNU Project was to be an umbrella organization, to help various open-source projects to contact students and recruit them to be awarded Google's stipend.)

Existing bugs related to stability and broken features persisting after 4.0 were addressed in this release. This version witnessed three major feature enhancements:

Figure 5. “Infinitude”, a flying fractal hotair balloon, featuring a detail of the Mandelbrot Set zoomed in over 7 million times, and containing over 100 billion pixels at over 100 dpi

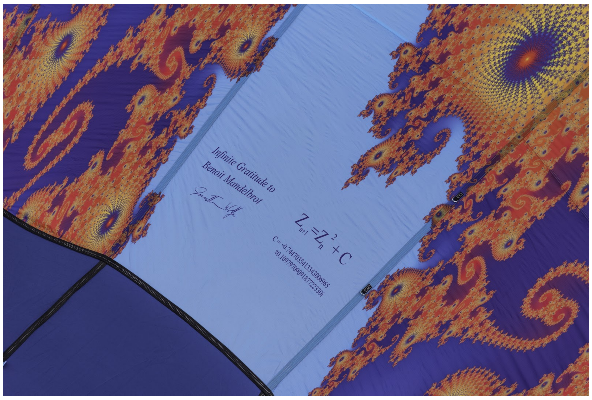


Figure 6. The inscription printed at the bottom of the “Infinitude” balloon acknowledging Benoit Mandelbrot for his contributions, and also showing the equation and the coordinates that generated the image on the balloon

1. **Cartesian Grid**: Cartesian grid added the ability to have an overlay of Cartesian coordinates on the screen. The grid has the capability to dynamically update according to XaoS zooming algorithm. As visible in Fig. 8, the point with coordinates (0.25, 0) marks the cusp of Mandelbrot fractal. The grid automatically zooms on magnifications every 10 times.

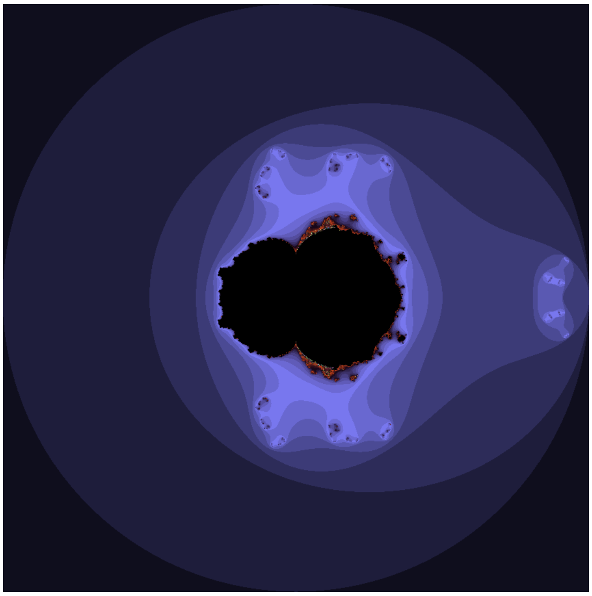

FIGURE 7. The fractal $z_{n+1} = z_n^2 \pm c$, defined with the input formula `z^2+(-1)^n*c` in XaoS 4.0

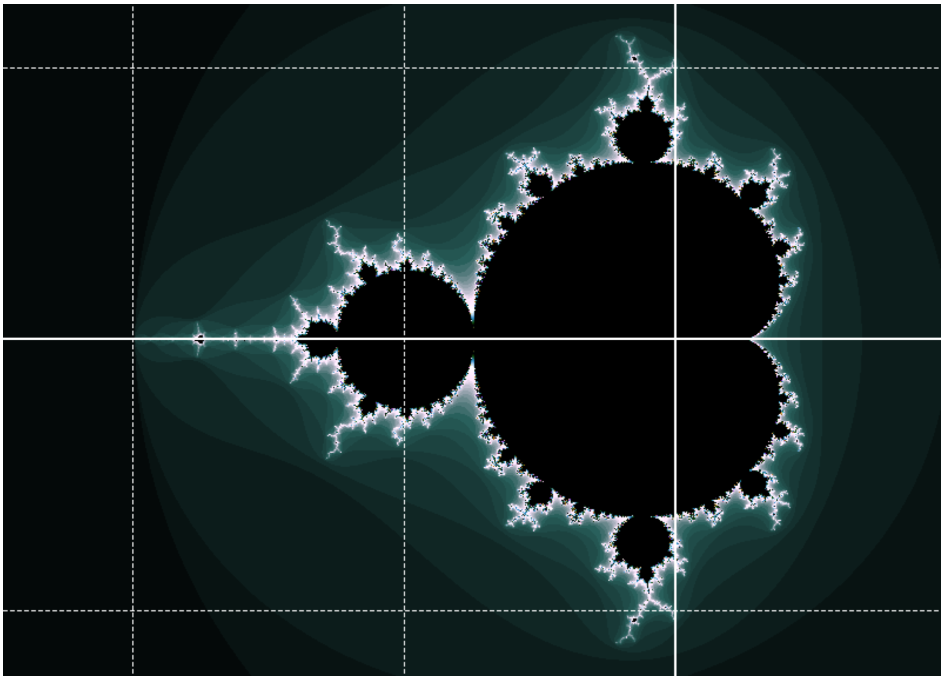

FIGURE 8. Cartesian grid in XaoS 4.1

2. **Custom Palette Visualizer**: The support for custom palettes was limited and dependent on 3 major parameters in 4.0—seed number, algorithm number and shift number. The seed numbers select colors according to 'random seed' value. The shift number shifts the palette colors one step iteration ahead. According to `https://github.com/xaos-project/XaoS/wiki/Reference#custom-palette` the algorithm number decides transition of fractal colors as:

| **Value** | **Meaning** |
|---|---|
| 0 | Default palette |
| 1 | Black to color gradient |
| 2 | Black to color to white gradient |
| 3 | Cubistic-like algorithm |

A problem was that there was no way to know beforehand the potential color of palette without applying these values. With the addition of a custom palette visualizer we were able to predict these values and visualize them.

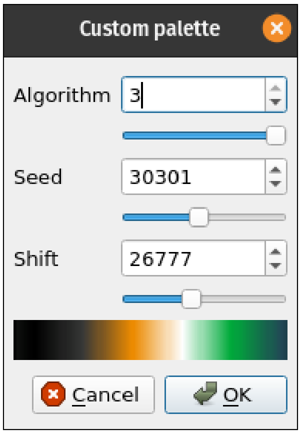


FIGURE 9. Custom palette visualizer in XaoS 4.1

3. **XaoS PNG Images**: The 'Save Image' functionality used to save the current frame as a PNG (Portable Network Graphics) image. Formerly, users had to separately save a XaoS position file (XPF, `https://github.com/xaos-project/XaoS/wiki/File-Format`) file to save the current state of fractal. After the change they could import the .xpf file again and continue zooming from the saved state.

   The 'Save Image' option can now save the XPF data as PNG metadata. These saved images can be imported back in XaoS and users can continue zooming from the saved state. These images can also be opened by any PNG compatible program such as the GNU Image Manipulation Program (GIMP).

The second author summarized these features in a video. The video is available at `https://www.youtube.com/watch?v=3OThpjupsYg`.

**Version 4.2**

Version 4.2 was released on 1 September 2020 concluding the Google Summer of Code (GSoC) 2020 project of the first author.

The release addressed some more bugs mostly related to existing memory leaks. There were six major improvements addressed in this release:

1. **Palette Editor**: Even after the 'Custom palette visualizer' added in version 4.1 there was need to give complete independence to users for customizing palette according to their likeness.

   There are 31 major colors in the default palette mode. The colors in between them are their transitions. Palette editor allows users to customize these 31 colors from outermost to innermost. Palette configuration created using this option automatically gets saved in the .xpf file.

   The applied palette configuration can also be exported/saved as a '.gpl' (GIMP palette format) file using the 'Save Palette config' option.

   The '.gpl' file can either be opened with a text editor or imported in GIMP. The saved palette configuration can also be imported back in XaoS using the 'Load Palette Config' option.

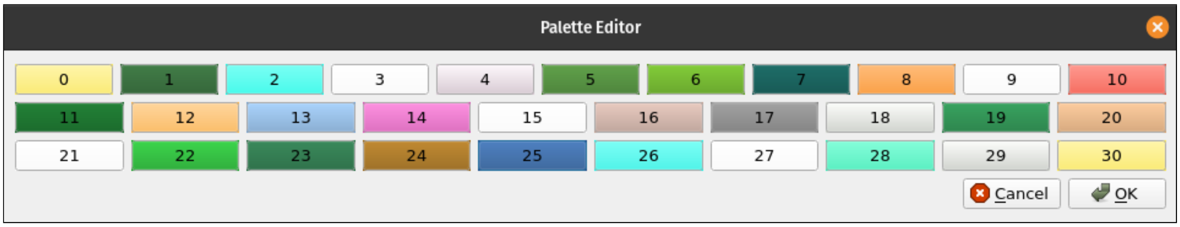


FIGURE 10. Palette Editor in XaoS 4.2

As visible in Fig. 10, there are 31 values that can be manipulated to produce the desired palette configuration.

2. **User formulas history**: Users can define their own formulas. A history of 10 recently defined 'User Formulas' is maintained.
3. **Fractal Information**: To help users learn more about the current fractal, Fractal Information takes users to the wiki page regarding details on the current fractal.
4. **Batch rendering**: There was limited support to render one file at a time using the graphical rendering option until version 4.1. With the introduction of 'Batch Rendering' users can select multiple files to render.
5. **Panning in rotate mode**: Unlike in normal mode, rotate mode had no support for panning during rotation. With the introduction of 'Panning in rotate mode' users can pan zoom and move fractals similar to normal mode.
6. **Unified menu options for supported XaoS file formats**: The 'Open' option can open all the supported XaoS formats (.xpf, .xaf, .png). The new 'Save as' option now contains all the supported XaoS formats in one menu option.

All of these improvements have been communicated in [1].

## 5. Future work

### 5.1. Web Compilation

After XaoS 4.0, the source code was refactored using Qt. Using Qt for WebAssembly `https://en.wikipedia.org/wiki/WebAssembly`, the source code can be compiled for web. This will allow a number of users to use XaoS without installing it on their machines.

In fact, there were already attempts to compile XaoS for the web. One successful result is a NaCl implementation [6] reported by Robert Muth in July 2012. This kind of technology was deprecated by WebAssembly meanwhile.

### 5.2. Support for deeper zoom levels

XaoS does not support very deep zooms because of 80 bit floating point arithmetic currently being used. XaoS globally uses a so-called `number_t` variable for switching between `__float128` and `long double` data types. Unfortunately, currently there is no easy way to switch between these data types at run-time. Thus in the `long double` compiled version we are restricted to $10^{16}$ zoom, and to $10^{32}$ for the `__float128` compiled version. The latter version is naturally slower so we cannot keep the real time zooming capabilities by making it default for early zoom levels. After $10^{32}$ we need to support much higher zoom levels, but—clearly—the use of arbitrary precision would be even slower. Thus we should be able to dynamically switch between `long double` (up to $10^{16}$), `__float128` (up to $10^{32}$) and use arbitrary precision beyond that.

### 5.3. Rendering large resolution images

For creating an image of width 'W' and height 'H', a space of

$$W \cdot H \cdot 4 \text{ bytes}$$

is required. If anti-aliasing is on, it takes space of

$$16 \cdot W \cdot H \cdot 4 \text{ bytes.}$$

The images are created as a single file inside the memory. For a 32-bit system, the memory limit is 2 GB. For a 64-bit system, this limit is increased, however, there are still limitation as per memory size. In a future version users should be able to split the image up and render each one to disk and then combine them.

### 5.4. Evolving as a platform

While the Fractal Challenge is currently operated manually and limited to New Mexico, the Fractal Foundation envisions a future in which a custom version of XaoS could form the basis of a Fractal Challenge platform for the world. In this design, a classroom could configure a custom instance of XaoS so they could run their own Fractal Challenge. Each student would make their own fractal, and the images would populate an online gallery, where the students could all vote on their favorites, and they would select the winning fractal from their classroom. Simultaneously, the other classes in the school would also run their own web-based Fractal Challenge, and choose their winning fractals, which would then populate a gallery of all the best fractals in the school. All the students would then get to select the best single image from the school. Of course, all the other schools in the city would be participating, and they would each select the favorite fractal from their school, so next there would be a contest where all the participating students could vote on the favorite fractal from the whole city. And this could then move to the state level, the whole country, and the entire world, creating an educational fractal social network.

### 5.5. Technical improvements

Here we give some other ideas that are more technical than didactic:

- The multithreading code requires fixes to avoid deadlocks.
- Fix the frame time calculation. It is not very accurate at the moment, so that zooming motion becomes jerky when rendering gets behind. Some related problems should be fixed in the timer routines.
- Improve (or eventually redesign) the user formula parser routine.

For further details we refer the reader to the XaoS' Developer's Guide at `https://github.com/xaos-project/XaoS/wiki/Developer's-Guide`. Also, GitHub maintains a database of issues on a contributed basis.

## 6. Conclusion

After the release of XaoS 4.0, the XaoS community has successfully continued to remain one of the most popular open source fractal zooming programs available. Versions 4.1 and 4.2 brought many changes that makes use of XaoS convenient, fun and educative. There is a worldwide community of active contributors to the development started by Thomas Marsh and Jan Hubička. We believe that there are promising horizons ahead for future developments in the program.

## 7. Acknowledgments

Second author was partially supported by the grant MTM2017-88796-P from the Spanish MINECO and the ERDF (European Regional Development Fund).

We are grateful to the GNU Project and Google by supporting our work during the GSoC 2020 project.

## References

[1] Aggarwal, A. (2020). GSoC 2020 Final Report for the XaoS project. https://github.com/xaos-project/XaoS/wiki/GSoC-2020

[2] Hubička, J., Kovács, Z., Kovács, Z. (2004) Visualizations on the Complex Plane. In: Sárvári, Cs. (ed.) Computer Algebra Systems and Dynamic Geometry Systems in Mathematics Teaching. Proceedings of Sprout-Selecting Conference, p. 12-27. Pécs, Hungary

[3] Kovács, Z. (1999) Véletlen sorozatok, diszkrét dinamikus rendszerek, káosz. Master thesis (in Hungarian). József Attila University, Szeged, Hungary

[4] Kovács, Z. (2019) Teaching fractals for gifted learners at age 12 by using novel technologies. https://arxiv.org/abs/1912.08703

[5] Krisztin, T. (1992) Káosz (in Hungarian). Polygon 2, p. 11-36

[6] Muth, R. (2012) XaoS Port to Native Client. http://robertmuth.blogspot.com/2012/07/xaos-port-to-native-client.html

[7] Wikipedia contributors. (2020, May 20). Fractint. In Wikipedia, The Free Encyclopedia. Retrieved 11:00, September 14, 2020, from https://en.wikipedia.org/w/index.php?title=Fractint&oldid=957820757

[8] Zsigovics, M. (1996) Kaotikus jelenség a lánctörtekben. Master thesis (in Hungarian). József Attila University, Szeged, Hungary

Anurag Aggarwal
Guru Gobind Singh Indraprastha University
Sector 16 C, Dwarka, 110078 Delhi, India
e-mail: kanurag94@gmail.com

Zoltán Kovács
The Private University College of Education of the Diocese of Linz
Salesianumweg 3, 4020 Linz, Austria
e-mail: zoltan@geogebra.org

J. B. Langston
https://www.linkedin.com/in/jblangston/
e-mail: jb.langston@gmail.com

Jonathan Wolfe
Fractal Foundation
2917 Campus Blvd NE, 87106 Albuquerque, New Mexico, USA
e-mail: jonathan@fractalfoundation.org