

Clustering Julia Set Examples to Enhance Evolution of Fractal Parameters

Andrew Dong and Daniel Ashlock

Abstract—This study updates a novel technique for evolving parameters that specify fractal images. Example parameter sets are provided as an information resource to evolution, following an earlier study. Instead of choosing parameters with high average compatibility with all other parameters, this study clusters the parameters using a graph clustering algorithm within a network where the adjacency relation of the network is derived from co-fertility, i.e. genetic compatibility values. The result of using the new types of sets of parameters as information resources is studied and compared to evolution that uses the previous type of information resource. The new technique of selecting information resources presented here yields higher fitness values. The new results are on the high end of the fitness distribution, and so the new information resources tested give similar improvements in fitness. However, their variability vary substantially and the resulting fractals have different appearances.

I. INTRODUCTION

Evolved art is a subfield of evolutionary computation in which digital evolution, in some form, is used to create an appealing image. The two most critical components to determine in evolved art are:

- 1) a representation that encodes images in a compact, evolvable form; and
- 2) an objective function that denotes an acceptable approximation to what human observers may consider appealing.

Fractals [19] are a natural target for evolved art as they are objects with fractional dimension and complex structure. Many classes of fractals such as Mandelbrot sets, Julia sets, or fractals generated with Newton’s method [17] are all specified by a short list of numerical parameters and thus have a choice of representations that are well-studied in the literature of evolutionary computation.

An earlier study [5] adopted a real parameter optimization approach to locating appealing Julia sets, which are carefully defined in Section II using a fitness function that measures the complexity of the resulting fractal. The iteration numbers, defined in II, at a grid of sample points within the image were taken; the fitness was defined as the Shannon entropy of the distribution of these values. The focus of the paper was *fertility*.

Informally, two sets of parameters are *fertile* if the expected fitness of their potential crossover is high. In order

to enhance evolution, 1000 sets of Julia parameters were produced via independent runs of an evolutionary algorithm. These parameter sets were evaluated for their average fertility f_{av} with one another, and the 50 with the highest values of f_{av} were used as sources of genetic information. This was done using *single-parent crossover* [10], a novel variation operator in which population members are crossed over with copies of example genes. This technique provides continuous injection of the alleles in the examples back into the evolving population.

This study re-examines the choice of parameters with high average fertility as genetic information sources. Two genes that have high average fertility with other genes in a group of 1000 genes may not have high fertility with one another. This means that the single-parent crossover techniques may be constantly injecting counterproductive information into the population. In this study we use diffusion characters [11] to create a map of the compatibility of the example parameter sets from their pairwise fertility values, and then apply multi-clustering [18] to find small groups of co-fertile genes to use in evolution based on single-parent crossover. The hypothesis is that using small groups of mutually fertile examples will outperform the types of examples used in the earlier study.

Computer generated fractals are early foci for computational creativity in art. Mandelbrot [20] highlights the development of mathematical algorithms into images and then links them into artistic structures. A number of artistic works prior to the discovery of fractals have been found to have fractal properties. For example, the paintings of Katsushika Hokusai, such as *The Great Wave off Kanagawa* [19], and Jackson Pollock [22] have been found to incorporate fractal elements. Artists’ and researchers’ works use or have been inspired by fractals for the creation of poetry [15], photomosaics [16], and song lyrics winning the 2014 Academy Award and 2015 Grammy [1]. The technique of *generative adversarial networks* [21] can be used to learn a style or type of image and generate new examples of images with a similar style.

A number of other studies have evolved fractals in the past. These include other techniques for evolving Julia sets [7], [5], and several attempts to search the Mandelbrot set for interesting fractal views [6], [3], [8], [4]. Genetic programming has been used to search for fractal iterator functions that generalize Mandelbrot and Julia sets [12]. An attempt has been made to use on-demand fractal generation as a method of providing an open-ended photo library for photomosaic generation [16]. Evolved iterated function system fractals

Andrew Dong is an independent researcher in Guelph, Ontario, Canada
Daniel Ashlock is with the Department of Mathematics and Statistics
at the University of Guelph, in Guelph, Ontario, Canada, N1G 2W1,
dashlock@uoguelph.ca

The authors thank the University of Guelph for its support of this work.
978-1-7281-6929-3/20/\$31.00 ©2020 IEEE

have also been used in bioinformatics to create sequence visualizers [13], [14], [9].

The remainder of this study is organized as follows: section II will give the mathematical background and define the objective function and the notion of fertility used in this study. Section III will explain how the information resource parameter sets were chosen and give the experimental design for evaluating their effectiveness. Section IV will give and discuss results, and Section V will draw conclusions and outline possible next steps.

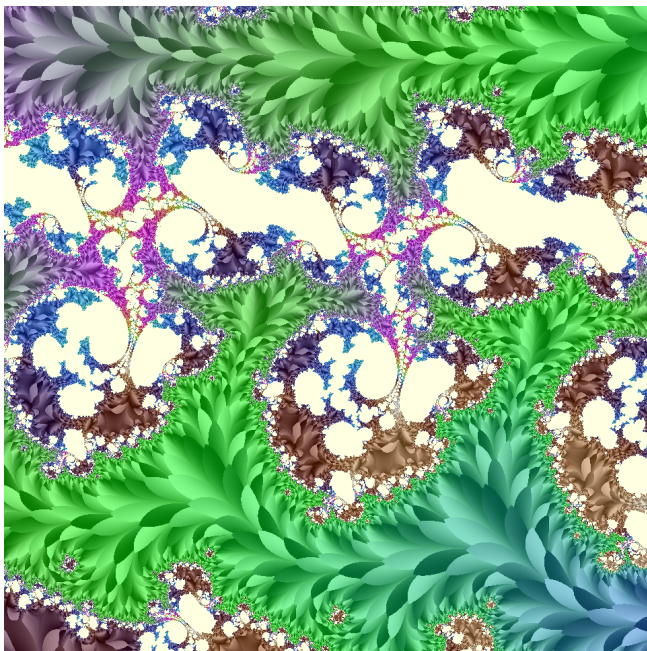


Fig. 1. An evolved generalized Julia set with three complex parameters.

II. BACKGROUND

This study attempts to locate parameter sets for fractals known as *generalized Julia sets* [7]. Figure 1 gives an example of a rendering of one of these fractals. The actual fractal is approximated by the white regions of the picture and is comprised of a subset of the complex plane. The fractal depends on N complex parameters, z_0, z_1, \dots, z_{N-1} , each of which has a real and imaginary part (thus $2N$ real parameters are used). For $N = 1$, the generalized Julia set is a standard Julia set.

For any complex number w in the complex plane, the following sequence is generated:

$$\begin{aligned} w_0 &= w \\ w_{k+1} &= w_k^2 + z_{k \bmod N} \end{aligned}$$

If the sequence associated with the complex number w contains an element $|w_i| \geq 2$, then w is not a part of the fractal; the smallest such index i is returned as the *iteration number* I of w . Otherwise, if there is no index i with $|w_i| \geq 2$, then w is part of the fractal.

In practice, an upper bound of 120 iterations is imposed. Points whose corresponding sequences satisfy $|w_i| < 2$ for

all $i < 120$ are considered to be part of the fractal, and must in fact be very close to points in the fractal. The iteration number of these points are reported as $I = 120$.

A. The objective function

As stated in the introduction, finding a fitness function that appropriately evaluates whether or not a photo is “appealing” or “beautiful” is one of the substantial problems in evolved art. The fitness function used in this study starts with an equally-spaced 11-by-11 grid of points in a square with corners $-1.6 - 1.6i$ and $1.6 + 1.6i$ in the complex plane. Then, each of the 121 points is assigned to one of 16 bins, numbered from 0 to 15. A point whose iteration number is I is assigned to bin number $\lfloor 16I/121 \rfloor$. At the end, bin k contains some number c_k of points, and we normalize these values to form an empirical probability distribution $(p_0, p_1, \dots, p_{15})$ given by $p_k = c_k/121$.

The objective function is

$$E = - \sum_{k=0}^{15} p_k \log_2(p_k), \quad (1)$$

which is the *Shannon entropy* of the distribution of the iteration numbers. Shannon entropy is maximized when a probability distribution is as close to uniform as possible. Hence, this objective function rewards an even distribution of the sample points over the different iteration-number bins. This does not guarantee beauty or appeal, but it does ensure complexity and therefore enriches the set of fractals located with interesting and potentially appealing fractals.

B. Selection of example fractals

Section III requires a method of selecting fractal parameters that will be used as sources of genetic information. Using uniform crossover on chromosomes with N loci each yield 2^N possible offspring, two of which are clones of the parents.

Definition 1: Given a fitness function, the **fertility** of two genes, each with N loci, is the average fitness of the $2^N - 2$ offspring which are not clones of the two genes themselves.

In the earlier study [5], the selection used was high average fertility with all other genes among 1000 best-of-run sets of genes (fractal parameters) with $N = 3$, selected with an evolutionary algorithm using the entropy-based objective function (1).

C. Rendering the fractals

To render the fractal, the square in the complex plane with corners $-1.6 - 1.6i$ and $1.6 + 1.6i$ is mapped onto a square image. At each pixel, the complex number corresponding to the centre of the pixel is computed and its iteration number I is calculated. This value I is then mapped to a color using the RGB values $R, G, B \in [0, 255]$ determined by the following

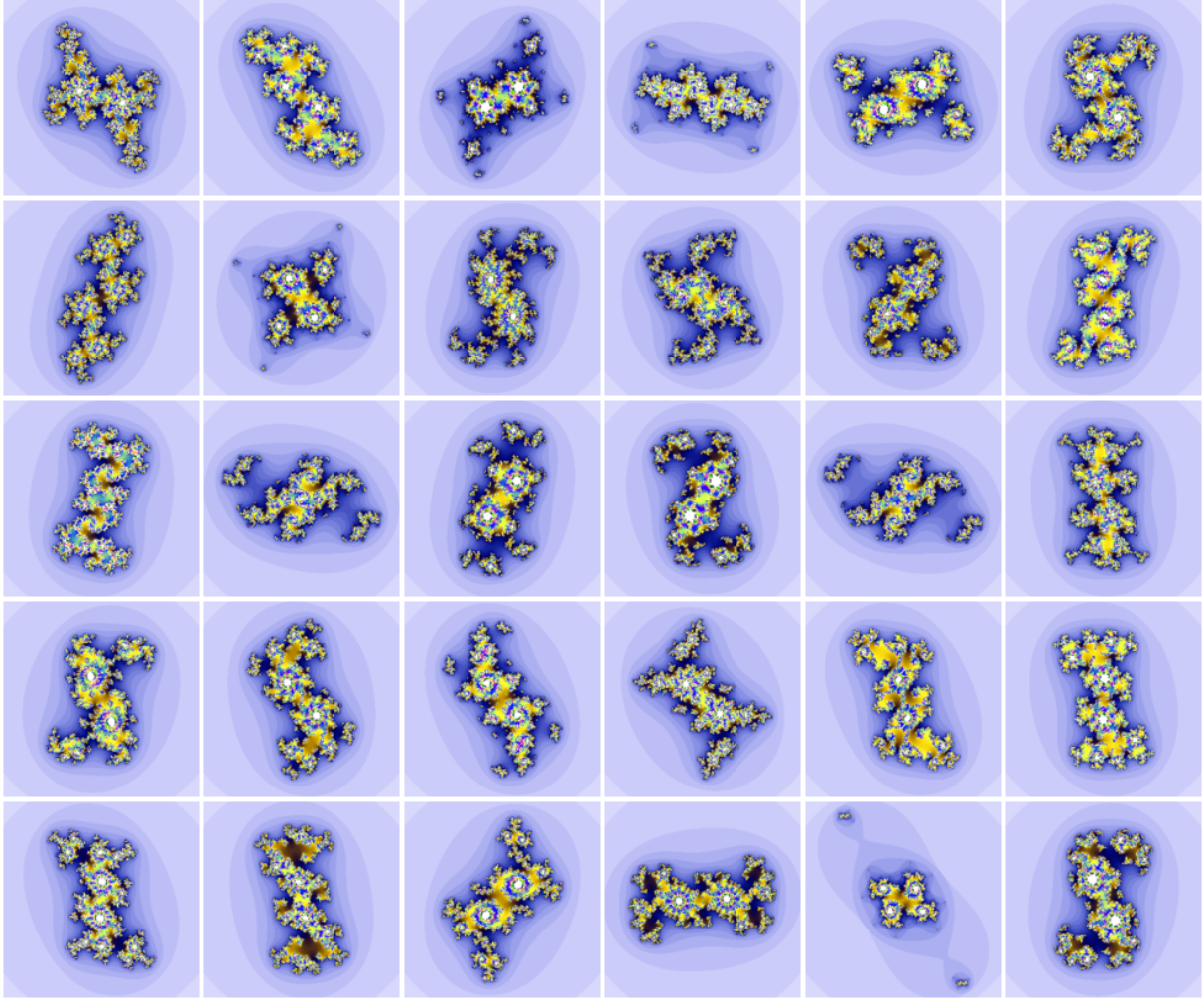


Fig. 2. All 30 images generated by the re-evolved genes using cluster twelve, with five members, from the $\lambda = 0.7$ extraction as the information resource.

equations, with 0 representing no color and 255 the most intense shade available:

$$R = 127 \cos(0.138 \cdot I + 0.8) + 128$$

$$G = 127 \cos(0.127 \cdot I + 0.8) + 128$$

$$B = 127 \cos(0.092 \cdot I + 0.2) + 128$$

These values are rounded to the nearest integer. Varying the period and phase shift by changing the two numerical values inside the cosine function permits an easy way to select from of a wide range of coloring palettes for the fractals.

Figure 1 gives an example of a rendering of one of these fractals.

III. DESIGN OF EXPERIMENTS

Compared to [5], a more elaborate procedure was used to select the information source from the same set of 1000 evolved fractal parameters used in the earlier study.

First, an unweighted, undirected graph was constructed where each node represented a gene, i.e. a set of Julia parameters. The fertility of all $\binom{1000}{2} = 499500$ pairs of parameter sets was computed and an edge was added between

two nodes whenever their corresponding genes had a fertility value exceeding a threshold τ . Selection of effective values of τ is a part of the research.

The diffusion character algorithm (detailed below) was then applied to this graph to yield a new association measure indicating closeness or fertility similarity in the fertility space of fractal parameters. Finally, K-means multiclustering [18] was applied to the fertility similarity relationship, yielding a set of clusters. The number of clusters obtained depends on the association strength λ used as the cut value for K-means multiclustering. These clusters of genes are the information resource for the evolutionary algorithm in this study.

Diffusion characters [11] are computed as follows: each vertex is associated with a different gas. In each time step,

- one unit of each gas is added to its home vertex;
- all gas at each vertex is divided evenly between its neighbours;
- the total amount of gas of each type at each vertex is multiplied by a decay factor $\omega < 1$.

Since gas is added arithmetically and decays exponentially (as the decay factor is $\omega < 1$), this process converges to a

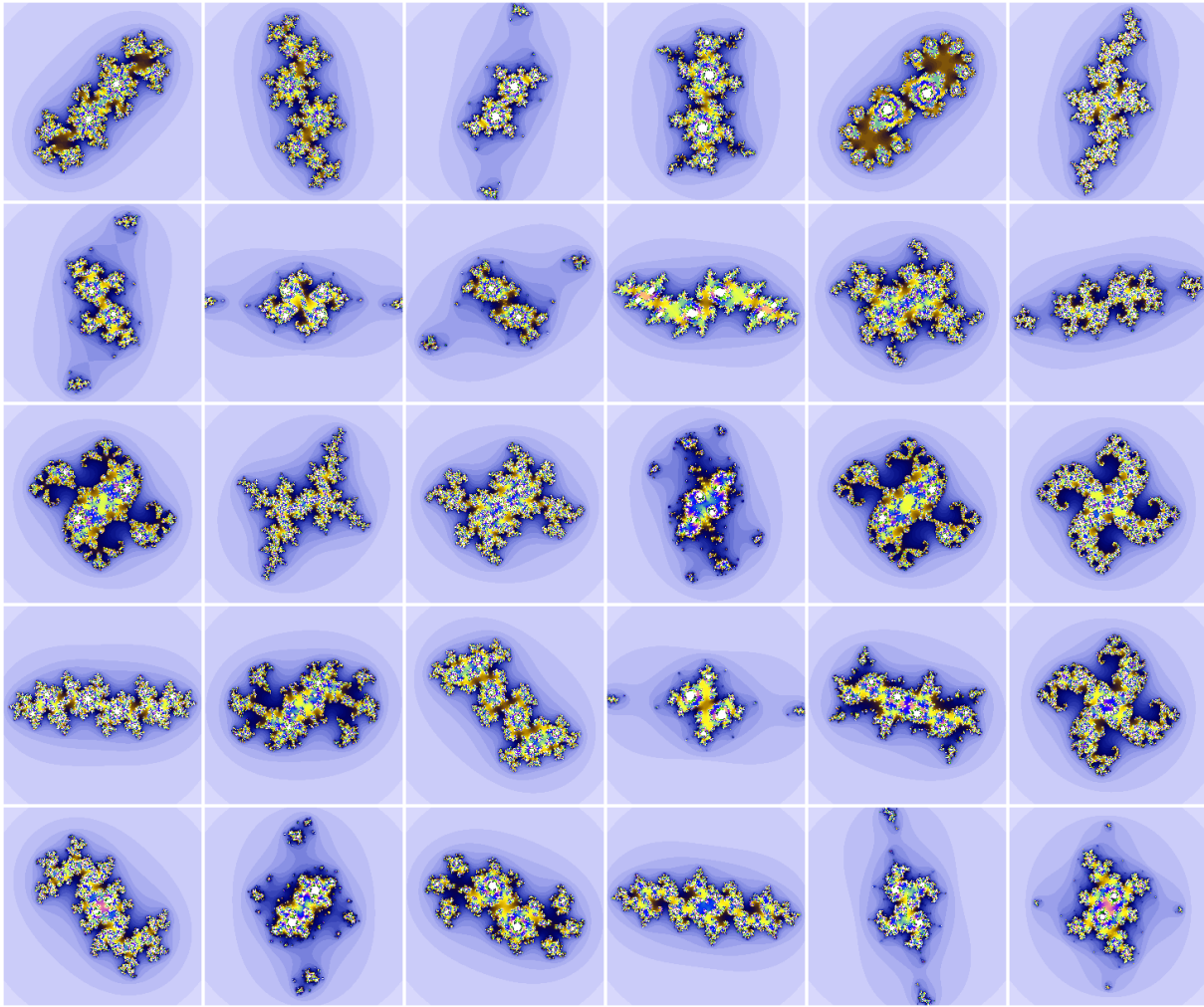


Fig. 3. All 30 images generated by the re-evolved genes using cluster sixteen, with three members, from the $\lambda = 1.0$, $\tau = 0.701$ parameter experiment as the information resource.

stable distribution of the gasses on the vertices. This vector of stable gas concentrations at each vertex is transformed by applying the map $x \mapsto -\ln x$ to each coordinate. The result is an injection of the vertices of the graph into Euclidean space so that the distance between points corresponding to vertices is proportional to the ease of moving between those vertices along any path; the gas analogy traces out those paths.

K-means multiclustering [18] operates as follows: the K-means algorithm is applied to a data set, in this case sets of fractal parameters with similarity measured as co-fertility, a large number of times. The value of K , the number of clusters, is varied in the range 10 to 60. Each time two sets of fractal parameters are in a cluster together as the result of K-means clustering, their association strength is increased by one. These strengths start at zero. After all the association strengths have been computed, they are normalized by dividing by the number of K-means clusterings performed, yielding association values in the range $[0, 1]$. A *cut value* λ is then used to build a second undirected graph in which there

are edges only between pairs of fractal parameters whose association strength exceeds λ . The clusters are the connected components of this graph.

The hypothesis tested in this study is that taking the information source to be clusters generated from the diffusion character algorithm and multiclustering, i.e. sets of relatively-highly co-fertile genes, will yield a higher average fitness than taking the fifty highest average fertility sets of fractal parameters, which was the procedure followed in the earlier study. This method of selecting the information resource sets of fractal parameters was conjectured to yield higher-fitness fractals because the co-fertility of genes close to one another in the fertility graph would yield more focused search than purely high-average fertility sets of fractal parameters.

To keep the members of an information resource relatively co-fertile, only clusters with between 3 and 60 sets of Julia parameters were tested. An example of a collection of renderings of thirty fractals evolved with a highly co-fertile information resource is shown in Figure 2; another is shown in Figure 3.

A. The evolutionary algorithm

The evolutionary algorithm used in this study is a simple real parameter estimation algorithm. It operates on a population of ten sets or Julia parameters. The mutation operator adds a normally-distributed value with a variance of $\sigma = 0.1$ and mean $\mu = 0.0$ to from one to three of the six real parameters that define the fractal. The number of positions modified by mutation is selected uniformly at random. Relevant complex numbers v for determining Julia sets satisfy $|v| < 2$, so the small value for the variance of the mutation operator is sensible.

Two crossover operators are used.

- For crossover between members of the population, two-point crossover is employed.
- Otherwise, a uniform single-parent crossover with a copy of one of the information-resource parameter sets is performed. In this case, the information resource used is selected uniformly at random from those available to the algorithm.

Selection and replacement are performed with single tournament selection of size seven [2]. This model of evolution selects seven members of the population. The two best are copied over the two worst and then the copies are subjected to crossover and mutation. A run of the algorithm consists of 100,000 instances of tournament selection and the algorithm is run 30 times with distinct random number seeds for each set of algorithm parameters tested. The frequency of single-parent crossover is a novel parameter is 0.05, adopted based on the parameter study in the first fertility study. The rate of single-parent crossover controls the degree to which evolution is directed toward members of the information resource. The relatively low rate is intended to reduce the likelihood of cloning of one of the information resources.

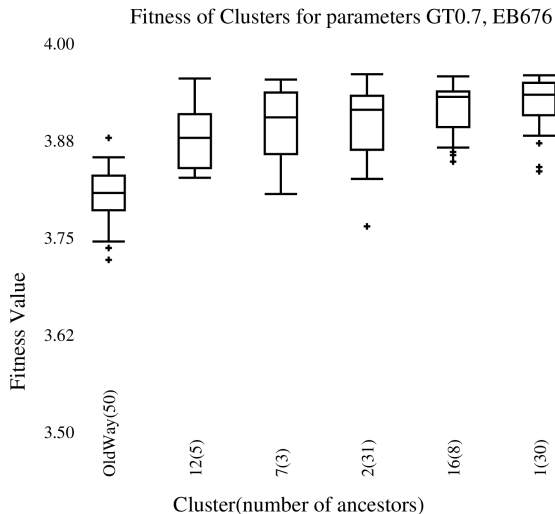


Fig. 4. Shown are the distribution of fitness for fractals evolved using ancestors from clusters 12, 7, 2, 16, and 1. The clustering was performed with an graph creation threshold of $\tau = 0.676$ and a cut value of $\lambda = 0.7$ for selecting clusters. The baseline fitness from using the 50 ancestors with highest fertility is shown at the left in each plot.

IV. RESULTS AND DISCUSSION

Figure 4 compares the fitness of the technique from the original study with the fitness resulting from using different clusters as information resources. These clusters were selected from the 20 available to demonstrate the range of fitness variation. While the authors expected more variation between the results arising from different clusters, these results support the hypothesis that the method of selecting information resources in this study yield higher fitness. The expectation is that this improved fitness results because of better co-fertility within the information resources.

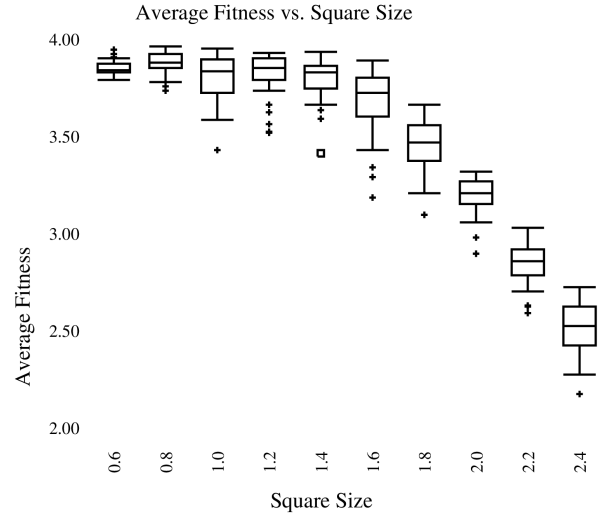


Fig. 5. The range of fitness values when varying the square size used for the grid of sample points in the fitness function. A set of 30 fractal parameters was used as an information resource while generating these fitness values; to generate this set, the graph threshold was $\tau = 1.0$ and the extraction bound was set to $\lambda = 0.676$.

Figure 6 also compares the results of the original study with fractals evolved using eleven different clusters as information resources. While there is a modest increase of variability of the fitness ranges for the runs using distinct clusters as information resources, the only significant difference is between the original study and all the runs performed with the new method of assembling information resources. The difference between the results obtained with distinct clusters is largely in their variability, not as much their fitness level.

Several values were adjusted during the course of the study:

- the square size used to contain the 11-by-11 grid of sampling points;
- the threshold for generating the graph on which the diffusion algorithm was run τ ; and
- the boundary λ for association strength, used as the cut value in multiclustering.

Varying the sampling square size (side length) caused substantial changes to the resulting fitness of the re-evolved fractals. The graph threshold τ had virtually no effect on the fitness values. The cluster extraction bound λ was adjusted to control the number of clusters found and, as a consequence, their size.

A. The sampling square

The square side length s was the largest determining factor in the average fitness f_{av} of evolving fractals. Figure 5 shows the fitness vs. the square size s . The value chosen for the results shown throughout this paper is side length $s = 0.8$, which yielded a relatively high value of f_{av} while maintaining a fairly high diversity of fitness values. In comparison, the square size used in the earlier paper was side length 3.2. These higher fitness values are the result of tuning the fitness function to be more effective (higher resolution), but they are not salient to the hypothesis being tested. All fractals evolved in this study use the square length 0.8.

The Figure 5 shown contains the data for square sizes $s \in \{0.6, 0.8, \dots, 2.4\}$. The fitness values when the square size ranged from $s = 0.6$ to $s = 1.4$ were relatively constant, with large drops from $s = 1.6$ onward. The majority of the 121 sampled points have a relatively high iteration number when the square is small as the sampled points are concentrated near the centre of the fractal. Since the fitness function groups these iteration numbers into bins, the bin with the lowest iteration numbers (bin 0) will necessarily contain fewer points than the other bins. Thus, on average, the bins will be less evenly-distributed resulting in a lower Shannon entropy, i.e. a lower fitness. Tuning the square yields a fitness function with a greater resolution.

B. The fertility threshold

The diffusion network is run on an graph that is generated based on the pairwise fertility of the 1000 genes. The threshold τ is the minimum fertility value between two genes that will place an edge in the graph between the nodes corresponding to the genes. Thus, if a gene has high fertility with many other genes, its corresponding node will have a high degree in the fertility network.

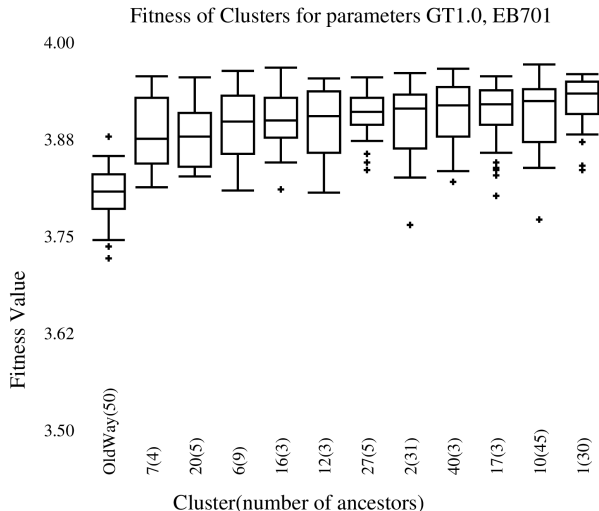


Fig. 6. Results from 11 runs of different clusters as information resources with $\tau = 0.701$ and $\tau = 1.0$. The baseline fifty highest average fitness results are shown at the left.

Cut values $\lambda \in \{0.5, 0.6, \dots, 1.5\}$ were tested with the extraction bound of fertility $\tau = 0.676$. This extraction bound was selected because it yielded a total of 20 clusters. This parameter is not being optimized; it was merely chosen to yield a reasonable number and size of clusters of co-fertile fractal parameters. Only clusters with between 3 and 60 sets of Julia parameters (inclusive) were used as information resources.

C. Cut values for multiclustering

The cut value λ controls the number of clusters that form: a higher boundary yielded more clusters that are, on average, smaller. Varying this parameter did not result in much variation in the resulting average fitness f_{av} , but all runs similarly resulted in a higher value of f_{av} compared to the previous study's technique, where the information resource was taken to be the 50 genes with the highest average fertility.

D. Downstream use of the fractal parameters evolved

The renderings of fractals in Figures 2 and 6 are rendered with a bulk, generic method chosen to permit difference in structure to be seen easily. Once interesting fractal shapes are obtained, they can be rendered with greater care to make more engaging images. Examples of this are shown in Figures 7 and 8.

V. CONCLUSIONS AND NEXT STEPS

This study applied a complex method to cluster a set of 1000 fractal parameters in a manner that depended on the co-fertility of those parameters. In accordance with the study's hypothesis, using these clusters of highly co-fertile fractal parameters as information resources indeed yielded higher fitness results.

Unexpectedly, different information resources, while all superior to the information resource derived in [5], did not exhibit noticeable variation in the average fitness of the sets of fractal parameters evolved. Different information resources located in this study did exhibit substantially different variability of fitness.

The contact sheets, exemplified by Figures 2 and 3, were examined and there was minimal dependence of the appearance of the resulting fractals on their information resource. High fertility means that many of the crossovers of a pair of fractal parameters are also high fitness, but it does not mean they are in a region of the parameter space with a fairly standard appearance. It is a classical result of Mandelbrot that arbitrarily small regions of the parameter space can contain an infinite variety of different appearances. This result implies that the co-fertile information resources, while they enhance fitness, do not give much control over the details of the appearance of the fractals that evolve.

A. Varying fitness functions

Throughout the study, an 11-by-11 grid centred at the origin was used to sample the points that determined the

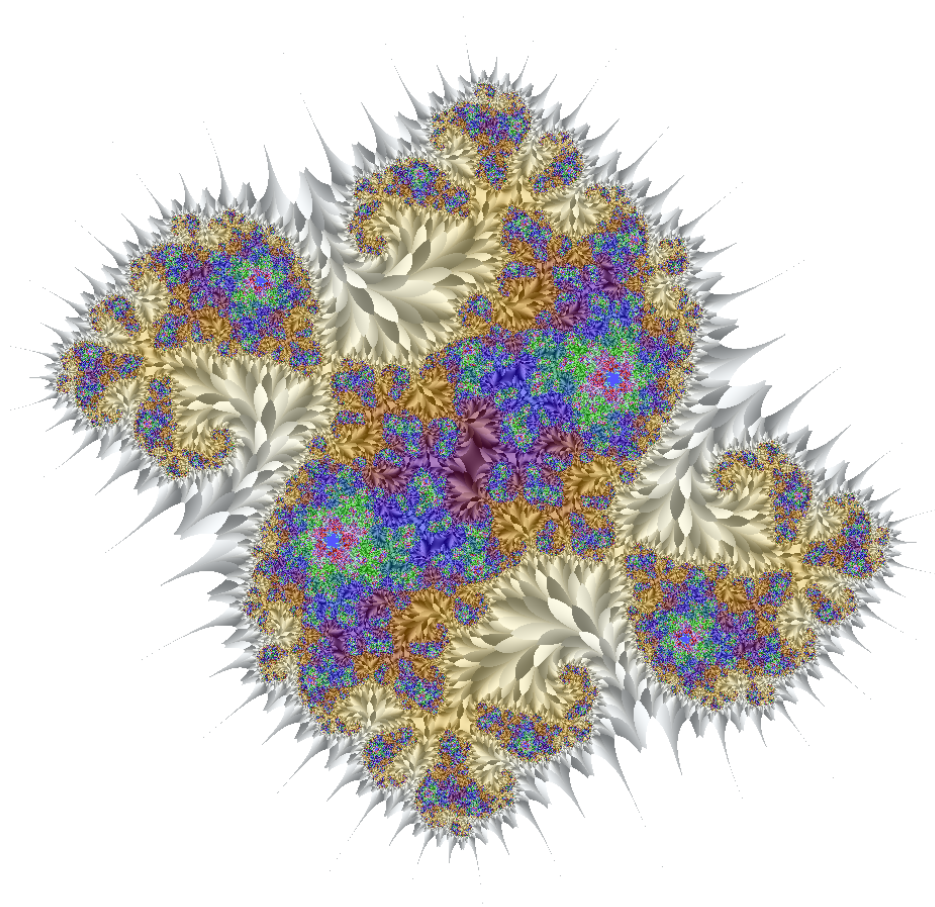


Fig. 7. One of the fractals from Figure 6 rendered with non-generic settings.

fitness of a fractal. Varying the sampled data, by changing the boundaries of the sampling square or by changing the number or arrangement of sampling points, effectively changes the fitness function and can provide control over the locations of the busy regions of the fractal. Sampling from disjoint regions may result in fractals which have a more appealing look. Additionally, the density function or weight of each sampled point need not be uniform. This study used the simplest function in the author's experience that reliably yields interesting fractals and, having selected that function, investigated fertility effect in managing information resources. It may be that fertility effects are different for different fitness functions, an area for future work.

The number of bins used in the fitness function was 16. This places the fitness f in the range $0 \leq f \leq 4$ and, in general, if n bins were used, into the range $0 \leq f \leq \log_2(n)$. In our case, with 121 samples, we have around $121/16 \approx 7.6$ samples per bin. If too many bins are used, fitness values would flatten toward the high end because there would be many bins with almost equal (small) numbers of samples. If we increased the number of sample points the algorithm would slow down in direct proportion to the number of

sample points, but this would also create the potential for a higher resolution fitness function using more bins. The fractal space being searched is highly rugose and increasing the number of sample points would also act to smooth the fitness landscape to a modest degree.

A large number of fitness functions other than the one based on Shannon entropy are possible. Maximizing the variance of the iteration numbers I_k has been tried, but this fitness function has the problem that there is a maximum in which many I_k are equal to 120 (nearly all points are considered to be in the fractal), and another maximum in which many I_k are close to 0 (all points have very low iteration numbers). Both of these cases yield boring fractals. However, this works well if points with the maximum iteration numbers are excluded.

Another possible fitness function modifies either the standard deviation or entropy-based fitness functions by rewarding the situation where the number of points actually estimated to be in the fractal are near some fraction, e.g. 20%, of the pixels in the picture.

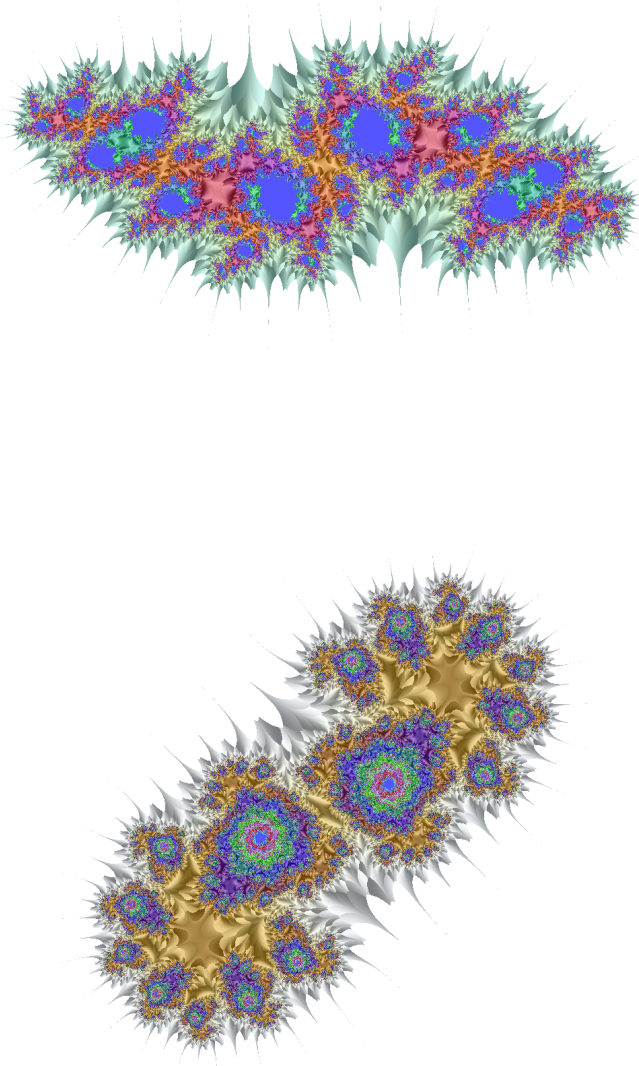


Fig. 8. Two of the fractals from Figure 6 rendered with non-generic settings. They both use different palettes and the upper bound on the iteration number was changed.

B. Simplifying the information set selection method

The procedure in this study for selecting information resources is fairly complex. It may be profitable to find a simpler technique for selecting these ancestors. An evolutionary set selection algorithm that picks sets of fractal parameters and scores them on mutual co-fertility or minimal co-fertility is a good candidate. The somewhat ornate system used in the current study was chosen in an effort to use global fertility information, a goal that may not be valuable enough to justify continuing to use the current multi-step procedure.

REFERENCES

- [1] Kristen Anderson-Lopez and Robert Lopez. *Let It Go*. Wonderland Music Company, Walt Disney, 2013.
- [2] D. Ashlock. *Evolutionary Computation for Optimization and Modeling*. Springer, New York, 2006.
- [3] D. Ashlock. Evolutionary exploration of the mandelbrot set. In *Proceedings of the 2006 Congress On Evolutionary Computation*, pages 7432–7439, 2006.
- [4] D. Ashlock and J. A. Brown. Fitness functions for searching the mandelbrot set. In *Proceedings of IEEE Congress on Evolutionary Computation*, pages 1108–1115, 2011.
- [5] D. Ashlock, J. A. Brown, and L. Sulancva. Exploiting fertility to enable automatic content generation to ameliorate user fatigue in interactive evolutionary computation. In *Proceedings of the 2018 Congress on Evolutionary Computation*, pages 1392–1398, Piscataway NJ, 2018. IEEE Press.
- [6] D. Ashlock, K.M. Bryden, and S. Gent. Multiscale feature location with a fractal representation. In *Intelligent Engineering Systems Through Artificial Neural Networks*, volume 19, pages 173–180, 2009.
- [7] D. Ashlock and B. Jamieson. Evolutionary exploration of generalized julia sets. In *Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Signal Processing*, pages 163–170, Piscataway NJ, 2007. IEEE Press.
- [8] D. Ashlock and B. Jamieson. Evolutionary computation to search mandelbrot sets for aesthetic images. *Journal of Mathematics and Art*, 3(1):147–158, 2008.
- [9] D. Ashlock, C. McGuinness, and W. Ashlock. Chaos automata for sequence visualization. In *Proceedings of the the 2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology*, pages 1–8, 2015.
- [10] D. Ashlock and S. McNicholas. Single parent generalization of cellular automata rules. In *Proceedings of the 2012 IEEE Congress on Evolutionary Computation*, pages 179–186, Piscataway NJ, 2012. IEEE Press.
- [11] D. Ashlock, E. Shiller, and C. Lee. Comparison of evolved epidemic networks with diffusion characters. In *Proceedings of IEEE Congress on Evolutionary Computation*, pages 781–788, Piscataway NJ, 2011. IEEE Press.
- [12] D. Ashlock and J. Tsang. Evolving fractal art with a directed acyclic graph genetic programming representation. In *Proceedings of the 2015 Congress on Evolutionary Computation*, pages 2137–2144, Piscataway NJ, 2015. IEEE Press.
- [13] Daniel Ashlock and James B. Golden III. Iterated function systems fractals for the detection and display of dna reading frame. In *Proceedings of the 2000 Congress on Evolutionary Computation*, pages 1160–1167, 2000.
- [14] Daniel Ashlock and James B. Goldin III. Chaos automata: Iterated function systems with memory. *Physica D.*, 181:274–285, 2003.
- [15] Chirstian Bök. *Crystallography*. Coach House Books, Toronto, second edition, revised edition, 2003.
- [16] J. A. Brown, D. Ashlock, J. Orth, and S. Houghten. Autogeneration of fractal photographic mosaic images. In *2011 IEEE Congress of Evolutionary Computation (CEC)*, pages 1116–1123, June 2011.
- [17] Robert L. Devaney. *A First Course in Chaotic Dynamical Systems, Theory and Experiment*. Taylor and Francis Group, 2018.
- [18] E.Y. Kim, S.Y. Kim, D. Ashlock, and D. Nam. Multi-k: accurate classification of microarray subtypes using ensemble k-means clustering. *BMC Bioinformatics*, 10(260):1–12, 2009.
- [19] B. Mandelbrot. *The Fractal Geometry of Nature*. W. H. Freeman and Sons, 1983.
- [20] Benoit B. Mandelbrot. Fractals and an art for the sake of science. *Leonardo. Supplemental Issue*, 2:21–24, 1989.
- [21] Xudong Mao, Qing Li, Haoran Xie, Raymond Y.K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [22] Richard P. Taylor, Adam P. Micolich, and David Jonas. Fractal analysis of pollock’s drip paintings. *Nature*, 399(673):422, 1999.