Procedural Organic Modeling

David Bachman

SIGGRAPH '19 Educator's Forum , July 28 - August 01, 2019, Los Angeles, CA, USA
ACM 978-1-4503-6782-0/19/07.
10.1145/3326542.3328013

About the Presenter

David Bachman
Pitzer College
bachman@pitzer.edu
pzacad.pitzer.edu/∼dbachman
davidbachmandesign.com
mathartblog.com

David Bachman is a professor of Mathematics at Pitzer College in Claremont, CA, where also teaches computer science, and has co-taught classes on design with faculty in the art department. He received a PhD in 1999 from the University of Texas at Austin, and has since published over 20 research articles, three books, and received two grants from the National Science Foundation. For the last nine years David has been combining techniques from mathematics, biology, and computer science to produce 3D printed artwork, which has been shown in galleries across the country.

Course Outline

1. Organic vs Inorganic forms in nature
2. Why algorithms?
3. Phyllotaxis
4. Reaction-Diffusion
5. Differential Growth
6. Diffusion-Limited Aggregation
7. Vein structures
8. L-systems

Organic vs. Inorganic Forms in Nature

Inorganic

- Mountain ranges, Coastlines
- Snowflakes
- Liquid flow
- Clouds

Organic

- Branching of trees, veins, etc.
- Skin/fur patterns (spots, stripes, fingerprints)
- Brains, Coral, etc.
- Leaf/seed arrangements

Why Algorithms?

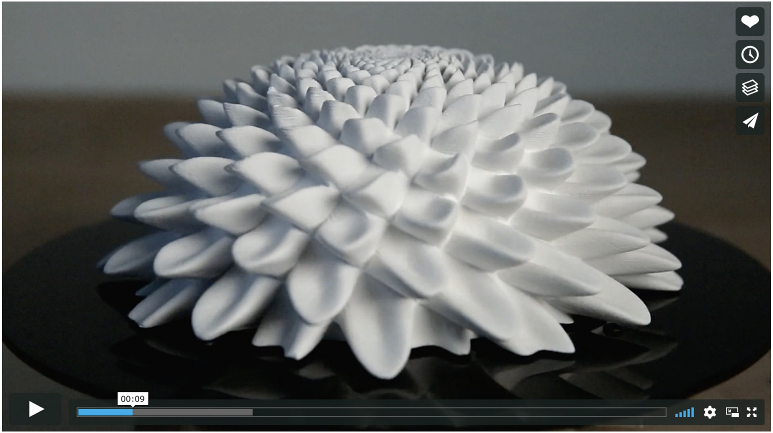
Advantages of Procedural Growth

- Arbitrary complexity possible (complexity = computation time/power)
- Less reliant on artistic ability
- Realistic results
- Can help us understand biological processes
- Tweaking models allows us to "discover" alien worlds

Phyllotaxis

... also pineapples, cactus needles, flower petals, palm leaves, etc.

Phyllotaxis in art



John Edmark's *Blooms*

https://www.youtube.com/watch?v=Qx2i9RYmsoo

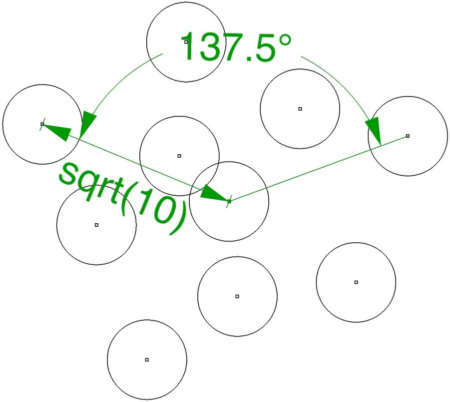
How it works (2D version)

- "Seeds" are added radially around a central point.
- The angle between each successive seed is the *Golden Angle*:

$$360^\circ\left(1-\frac{1}{\phi}\right), \text{ where } \phi=\frac{1+\sqrt{5}}{2}$$

- Distance from center point to new seed is proportional to the **square root** of seed number (keeps spacing between seeds relatively constant).

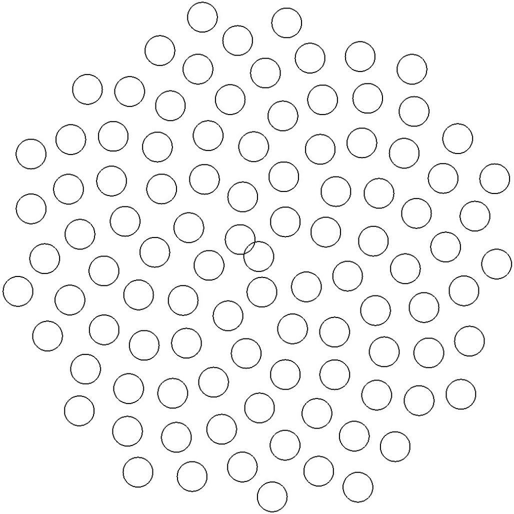
How it works (2D version)



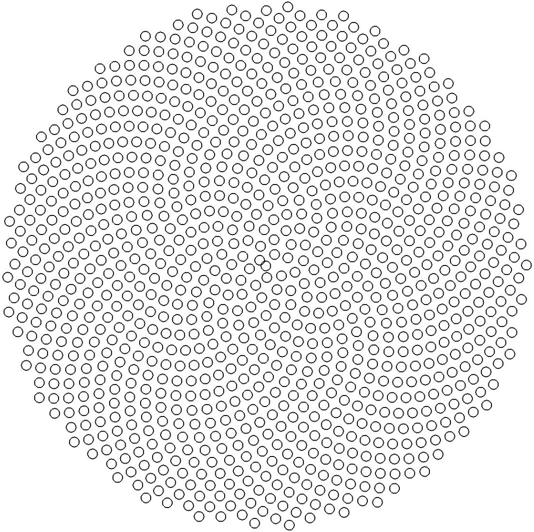
Grasshopper code available at `pzacad.pitzer.edu/∼dbachman/demos`.

[DEMO]

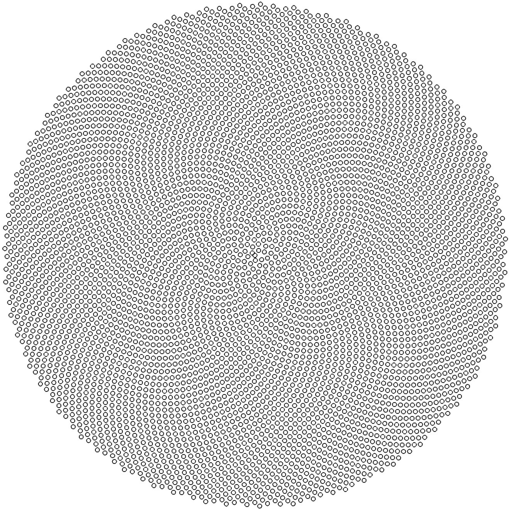
100 seeds



1000 seeds



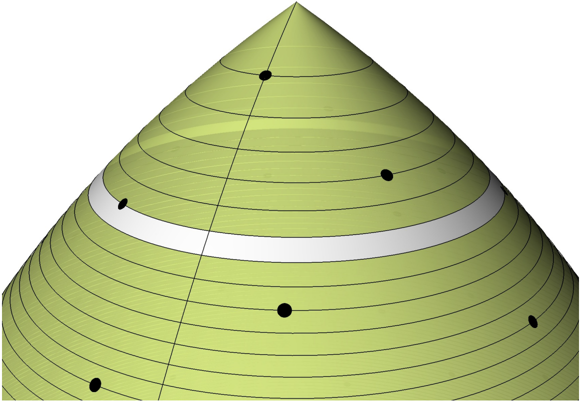
5000 seeds



How it works (3D version)

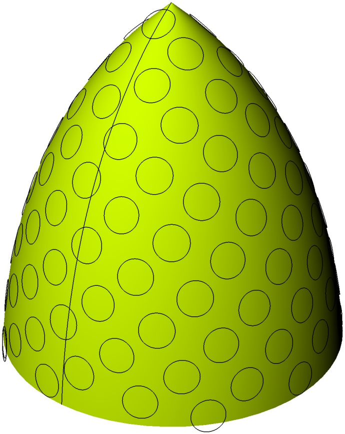
- “Stem” shape is modeled as a surface of revolution.
- “Leaf” is copied to various locations around stem.
- Angle around stem is determined by multiple of Golden Angle (as in 2D).
- Distance from stem tip is determined so that...

How it works (3D version)



... surface area between horizontal levels is constant.

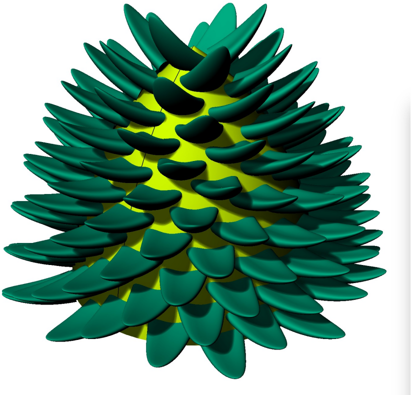
How it works (3D version)

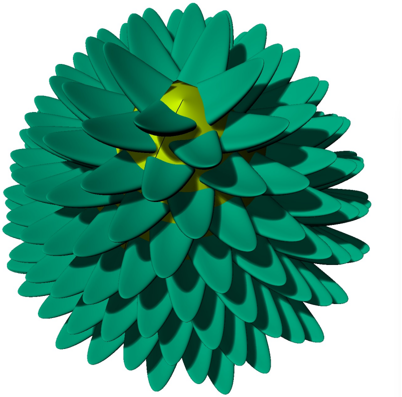


Grasshopper code available at `pzacad.pitzer.edu/∼dbachman/demos`.

[PAUSE FOR LIVE DEMONSTRATION]

How it works (3D version)





Once "grow sites" are determined, "leaves" can be copied to each location.

Phyllotaxis (Biological explanation)

Theories:

- Golden angle maximizes sunlight exposure to leaves.
- Golden angle minimizes “energy” between newly born leaves.

Phyllotaxis and Fibonacci

One can prove ϕ is the limit of the ratios between consecutive elements of the *Fibonacci sequence*:

$$1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...$$

That is,

$$\frac{1}{1}, \frac{2}{1}, \frac{3}{2}, \frac{5}{3}, \frac{8}{5}, \frac{13}{8}, ... \to \phi$$

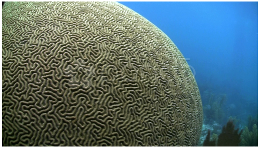
One consequence is that the apparent number of spirals in each direction is a consecutive pair of Fibonacci numbers. However, *which* pair you see changes with more points! This is because more points "shows" a better approximation of ϕ, corresponding to a rational with higher numerator/denominator.

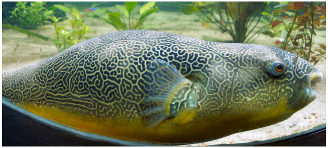
Reaction-Diffusion

Useful for modeling many 2-D organic textures, such as:

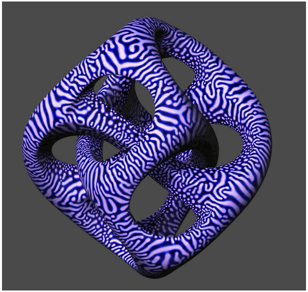
1. Tiger and zebra stripes
2. Giraffe and leopard spots
3. Fingerprints
4. Coral

Reaction-Diffusion in Nature


Reaction-Diffusion in Art



Bathsheba Grossman

Pendarestan @ Shapeways

The Reaction Diffusion Equations

1. System of two Differential Equations
2. Models the concentrations of an organism B, and its food A.
3. Both A and B are functions of position (usually 2-D) and time

Modeling Assumptions

1. Both food A and organisms B diffuse from regions of higher concentration to lower.
2. When two "parent" organisms (B) encounter one food (A), they consume it and produce another organism.
3. Small amounts of food A are constantly added to the system.
4. The organism B has a finite life span, and hence dies at some rate.

Gray-Scott Model

$$\frac{\partial A}{\partial t} = r_A \nabla^2 A - AB^2 + f(1 - A)$$

$$\frac{\partial B}{\partial t} = r_B \nabla^2 B + AB^2 - kB$$

Gray-Scott Model

$$\frac{\partial A}{\partial t} = r_A \nabla^2 A - AB^2 + f(1-A)$$
$$\frac{\partial B}{\partial t} = r_B \nabla^2 B + AB^2 - kB$$

r_A and r_B are the diffusion rates of A and B.
∇^2 is a Laplacian function, giving the difference in the concentration in the surroundings of a particular point, and at the point itself.

Gray-Scott Model

$$\frac{\partial A}{\partial t} = r_A \nabla^2 A - AB^2 + f(1 - A)$$
$$\frac{\partial B}{\partial t} = r_B \nabla^2 B + AB^2 - kB$$

AB^2 is the probability, at each point, that two organisms encounter one food. This term is computed pointwise.

Gray-Scott Model

$$\frac{\partial A}{\partial t} = r_A \nabla^2 A - AB^2 + f(1 - A)$$

$$\frac{\partial B}{\partial t} = r_B \nabla^2 B + AB^2 - kB$$

f is the feed rate of A

k is the "kill" rate, or rate at which B dies "naturally." Normally we use $k \geq f$.

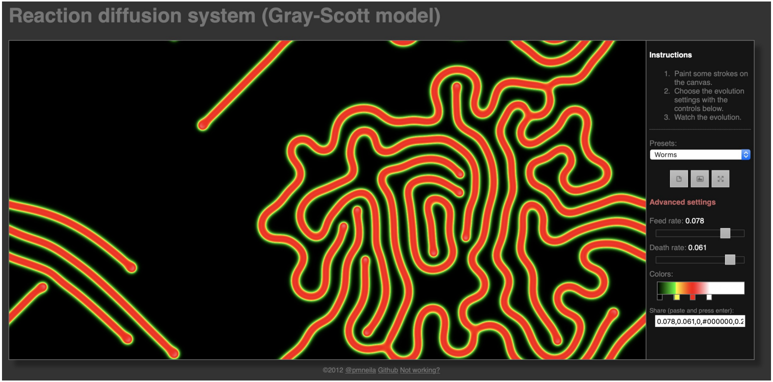
Discretizing Gray-Scott on a grid

1. A and B take on values only on a grid.
2. Replace $\frac{\partial A}{\partial t}$ with $\frac{\Delta A}{\Delta t} = \frac{A^* - A}{\Delta t}$, where A^* is the concentration of food after Δt units of time, and A was the concentration before.
3. $\nabla^2 A$ is computed at each point by taking the difference between A and a weighted average of the values of A in neighboring cells. (Typically E, W, N, and S neighbors are weighted with 0.2, and NE, NW, SE, and SW neighbors are weighted with 0.05.)

Discretizing Gray-Scott on a triangulated mesh

$\nabla^2 A$ is computed at each vertex v by taking the difference between the value of A at v and the average of the values of A at each vertex that is connected to v by an edge.

Reaction Diffusion in action

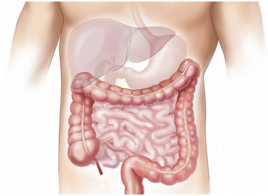


`https://pmneila.github.io/jsexp/grayscott/`

For more information, see `https://www.karlsims.com/rd.html`

Differential Growth

For modeling:



Intestines (1-D growth)

Kale (2-D growth, with boundary)

Brains (2-D growth, no boundary)

Differential Growth in Art

Nervous System

Daina Taimina

Mediated Matter group @ MIT media lab

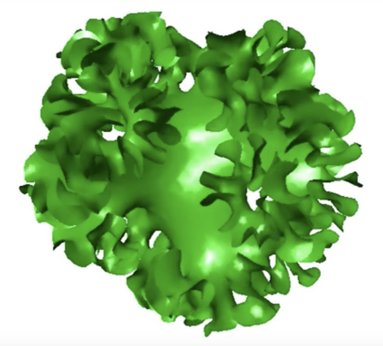
Differential Growth in Art

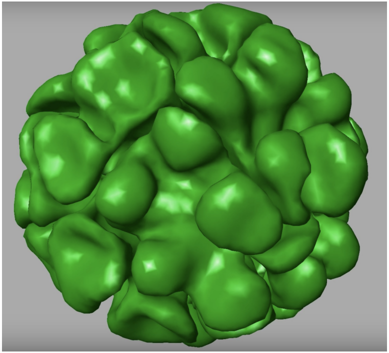
A Differential Growth Algorithm

1. Begin with mesh (or polygonal curve) with uniform edge lengths (normalized to 1)
2. Replace each edge with a spring with rest length 1
3. Each vertex is given a weak repulsive "charge," forcing vertices to spread apart
4. When the repulsive vertex forces make some edge length longer than two, divide it (and the incident simplices).
5. Repeat!

Notes
–1-D growth is often confined to a surface.
–It is helpful to add some smoothing forces to the mesh/polyline to obtain more organic looking results.

Differential Growth Demos





http://mathartblog.com/?p=464

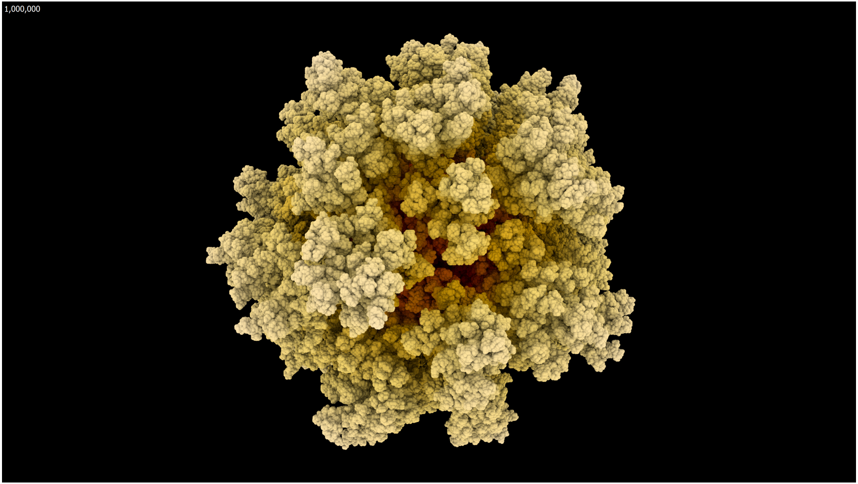
Diffusion-Limited Aggregation

Good for modeling:

1. Lichen
2. Neurons
3. Lightning (I know... its not organic)
4. tentacles

Lichen

DLA art



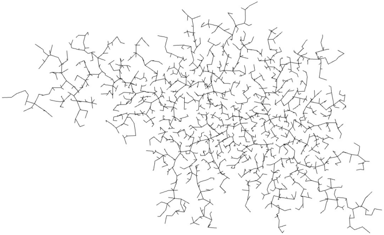
Jason Rampe (Visions of Chaos)

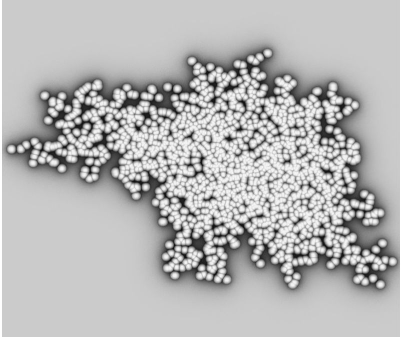
The DLA Algorithm

1. Begin with a (set of) fixed "seed" particle(s) as the initial aggregate.
2. Introduce a single particle near the boundary of the environment.
3. Subject new particle to brownian motion until it exits the environment, or comes into contact with the aggregate.
4. When a particle comes near enough to the aggregate it "sticks" to it, and a new particle is launched.

DLA display options

Aggregates can be modeled by collections of balls, or points connected by line segments.



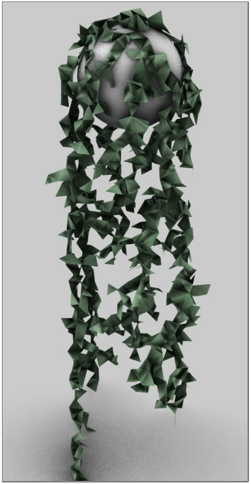


DLA mesh growth

To grow a (mesh) surface, create a new triangle whenever a particle q comes near a boundary edge e of the mesh. The new triangle will be the join of e and q.

DLA mesh growth

You can create other effects by emitting new particles from one direction, and introducing obstacles to constrain brownian motion.

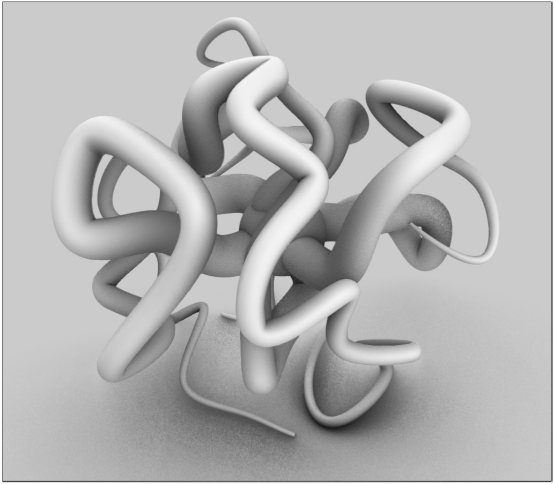


Related Algorithms

1. Begin with a seed aggregate, and a cloud of fixed "food" points.
2. Determine the closest food point p to the aggregate, and the point q of the aggregate that is closest to it.
3. Enlarge the aggregate by connecting p to q with an edge.

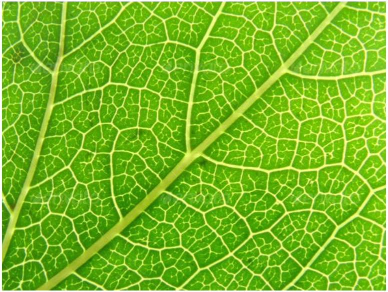
To create "tentacles" instead of branches, restrict to a limited number of "grow sites" on the aggregate. Each time an edge is added between a food particle q and a grow site p, remove p from the list of grow sites and replace it with q.

Tentacles



Tentacles grown by modified DLA algorithm with restricted grow sites.

Vein Structures



Vein Structures

Nervous System

Vein Structures

Algorithm based on:

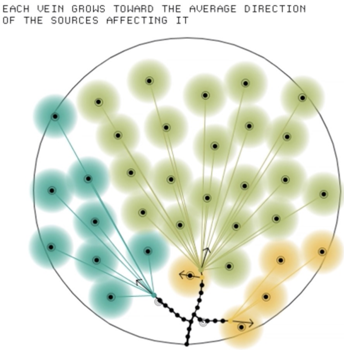
Adam Runions, Martin Fuhrer, Brendan Lane, Pavol Federl, Anne-Gaëlle Rolland-Lagan, and Przemyslaw Prusinkiewicz. Modeling and visualization of leaf venation patterns. *ACM Transactions on Graphics* 24(3), pp. 702?711.

Venation Algorithm

As in the modified DLA algorithm, begin with a "root" point and a cloud of "sources."

1. Compute the shortest path from each source to the vein structure/tree. Each vertex of the tree at the endpoint of at least one such path is a growth site.
2. For each growth site, determine the direction of growth by averaging amongst the directions from that site to the set of sources that it is closest to.
3. When the tree grows close enough to a source, it "eats" it and the source disappears.

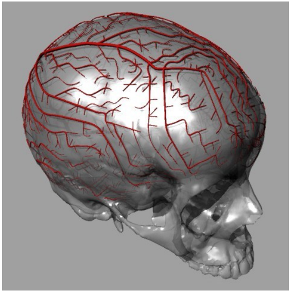
Venation Algorithm

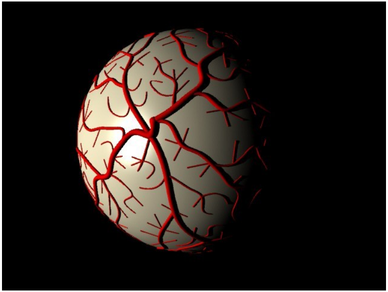


Animated explanation by Nervous System at:

https://n-e-r-v-o-u-s.com/projects/albums/networks-sketches/content/hyphae-growth-process-diagram-in-2d/

Vein Art

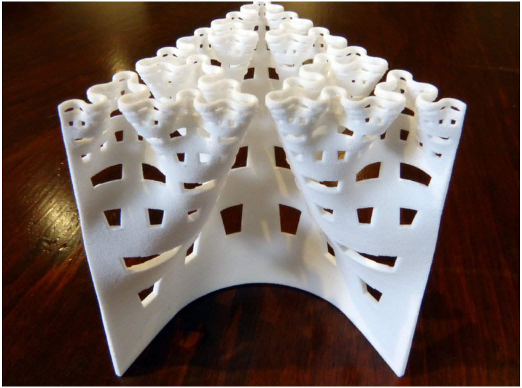




L-Systems

Useful to describe self-similar (fractal) structures with (or without) branching.

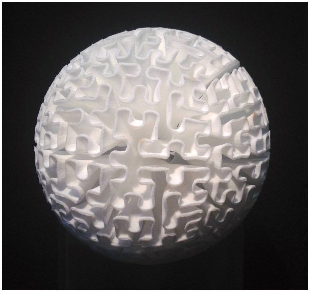
Some L-Systems in Art



Henry Segerman

Robert Fathauer

Some L-Systems in Art



Bachman, Fathauer, Segerman

Bachman

How it works (no branching)

2 step process:

1. *Rewriting System*: Beginning with a symbol string (axiom), replace all characters according to some rule. Then iterate!
2. *Interpretation*: Interpret symbols in string as turtle graphics commands (e.g. “turn left”, “move forward and draw”, etc.)

An Example

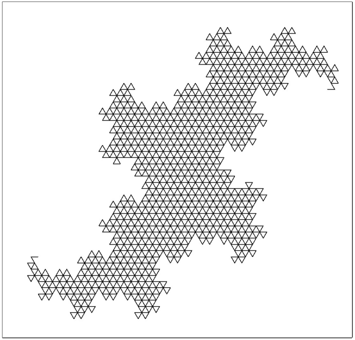
Rewriting system:
Axiom: F
Rule: F $\rightarrow$ F+F$-$F

Step 0: F
Step 1: F+F$-$F
Step 2: $\underline{\text{F+F}-\text{F}} + \underline{\text{F+F}-\text{F}} - \underline{\text{F+F}-\text{F}}$

An Example

Interpretation

- F: Move forward one unit and draw
- $+$: Turn Right 120°
- $-$: Turn Left 120°



[PAUSE FOR DEMO]

An L-System With Branching

Axiom: X

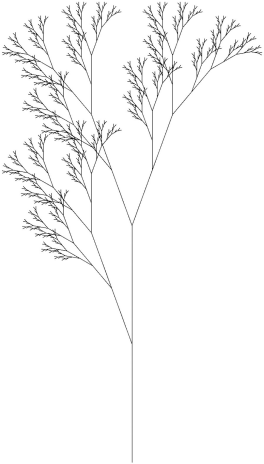
Rules:

1. F $\to$ FF
2. X $\to$ F[+X]F[$-$X]+X

Interpretation:

- F: Move forward one unit and draw
- $+$: Turn Right 20°
- $-$: Turn Left 20°
- X: Do nothing
- [: Store current position/direction
-]: Recall last saved position/direction

An L-System With Branching



[DEMO]

For more information...

https://mathartblog.com