

Algorithmic Art -- Composing the Score for Visual Art

by Roman Verostko



Algorithmic design detail, Hispano-Moorish tile (azulejos), 14th Century, Nasrid Palaces, Granada, Spain. Photo: RV

Algorithm (variant: algorism). For term origins, spelling and example see [1]

What is an algorithm? An algorithm may be viewed simply as a detailed recipe for carrying out a task. The term has its origin in mathematics as the step-by-step procedure for solving a problem. The commonplace procedures we use for multiplying and dividing numbers are algorithms. With precise details for each step, the procedures yield the same result whether executed by a computer or by a human, which is why robots are able to handle many tasks that were once done only by humans.

Many view an 'algorithmic procedure' as a strictly mathematical operation. Today we are inclined to understand any well-defined procedure as an algorithm. A recipe for baking bread is an algorithm. Follow the recipe faithfully and you will duplicate the kind of bread made by the person who wrote the recipe.

Applications. Machines can also be made to follow recipes. The programmed circuitry in our bread-making machine directs the machine's mechanism on precisely how to mix ingredients, knead the dough, and bake the bread. It succeeds every time!

Within the past quarter of a century, operational instructions have been imbedded in the design of many industrial and household utilities. They implement our daily use of telephones, automobiles, cameras, TVs, and radios. Our hospitals, factories, banks, and shopping centers all depend on the algorithms that control inventories, transactions, communications and security. They are ubiquitous and our mass culture would collapse without them.

Algorithmic procedures are also imbedded in the digital tools used in the arts. Use of these tools influences form in the practice of film, architecture, photography, music, printmaking, and all types of electronic sound and image. The drudgery of executing algorithms,

which would require immense time, or would even be impossible to execute without computing power, has been handed over to the machine -- leaving humans more freedom to focus on the creative part of their work. For the artist, this means improving and improvising the art-making procedure. For the algorist, work on the algorithm is work on the procedure.

History & breadth. Although the term derives from the name of a 9th century mathematician, the use of algorithms dates from prehistoric times. Study of the stone circles at Stonehenge (c. 2000 BC) reveals an algorithmic arrangement based on phases of the moon and the annual movement of the sun. While we cannot know the meanings that the builders attached to the structure, we are able to discern something of the 'rules' for stone positions. Their alignment relates to the annual movement of the sun and the moon.



Stonehenge, c. 2000 BC, ff. Algorithmic arrangement, Salisbury plain, England. Ca. 24 ft high. Photo: RV

Clearly, early civilizations developed procedures for counting and measuring. They also created procedures for weaving, grinding, making fire and cooking. Any of these procedures, when well defined, could be viewed as an algorithm. Indeed, weaving technology played an important role in the history of computers. If we can spell out the procedure for any given task then, given all the necessary materials and skills, we should be able to carry out the task.

Architectural plans, musical scores and dance notations [2] bear one feature in common -- they are all recipes for carrying out a task. From this perspective, a broad range of notational systems can be viewed and studied as algorithmic procedure. Algorithmic procedures for generating artistic forms enjoy a rich and varied tradition, even though we have used other terms to describe them.

Algorithms and art



Gregorian Chant, Missal, c.1200. Detail from the *Sursum Corda*. Musical scores, viewed as instructions, are algorithms for performing music.

In Art History. A history of algorithms in the visual arts would be voluminous touching many phases in every culture at every turn -- the Egyptian canons for drawing the human figure, the infinite geometric play in Islamic art and the role of both linear perspective and proportion in Renaissance art. In China, we would find the Mustard Seed Manual and, in Byzantium, the conventions for icon painting. In Europe, we would find extremely sophisticated algorithms for plotting the dizzying perspectives imaging the passage from earth to heaven by the 17th century.

Even so, notational systems for the visual arts played a limited role when compared to notational systems for music. A gifted composer could compose a score for a profoundly moving musical passage that could be played hundreds of years later by a skilled virtuoso. Not so for the painter. While Leonardo could easily compose an algorithm for creating the perspective space in the Last Supper, he could not, at that time, compose an algorithm for rendering the face of Judas.

The 20th Century. A 20th century history would find some interesting pre-algorist examples in the 1960s and 1970s. Fluxus, minimalist, and conceptual artists employed various methods of procedural specification challenging traditional conceptions of art. For example, George Brecht's early works were primarily very lean instructions printed on cards; the instruction became art. But it was not until artists gained access to computing power that they were able to compose form-generators for the investigation of form.

TWO VEHICLE EVENTS

- start.
- stop.

George Brecht, 1961, *Two Vehicle Events*, Detail of a 3.5 " by 4.5 " white card with printed text.

Summer, 1961

George Brecht sent this card, along with others, to RV in 1961. This is one of many examples from the 1960s where artists employed detailed instruction for con-

structing an experience. In one such work, "*Vehicle Sundown Event*," GB published 50 cards reproduced in sets for each participant. Each card held an instruction to be performed with a vehicle. Vehicles with drivers were instructed to assemble at sundown in a parking lot and randomly park their vehicles. Then each driver, with a shuffled deck of instructions, performed 50 events such as "turn on lights," "start engine," "stop engine," "open window." This work was performed at St Vincent College under the direction of Stephen Joy in 1963.

The Algorists. As computers became more accessible to artists in the 1970s and 1980s, some artists began to experiment with algorithmic procedure. The new technology offered them methods of working algorithmically that were unavailable before the advent of computers. By the 1980s, a number of these artists were working with the pen plotter, a machine with a 'drawing arm.' By the end of the 1980s, algorists like Harold Cohen, Mark Wilson, Manfred Mohr, Jean Pierre Hebert and myself had already achieved a mature body of work. Each in their own way had invented algorithmic procedures for generating their art and created their own distinctive style. Clearly, style and algorithm were linked in a very important way. [8]

Simply put, algorists are artists who introduce and control original algorithms in the creation of their work. Jean Pierre Hebert, a master algorist himself, has written an algorithm that identifies an algorist as one who uses one's own algorithms for creating art objects. Hebert's algorithm, lean and beautiful in itself, identifies the essential features of algorist art. As the JPH algorithm makes clear, an algorist, in the proper sense of the word, employs her own algorithms in the process. Even so, all artists, including algorists build with and upon the algorithms, namely the procedures, set forth by our predecessors and colleagues. Like all survivors, we algorists stand on the shoulders of those who preceded us.

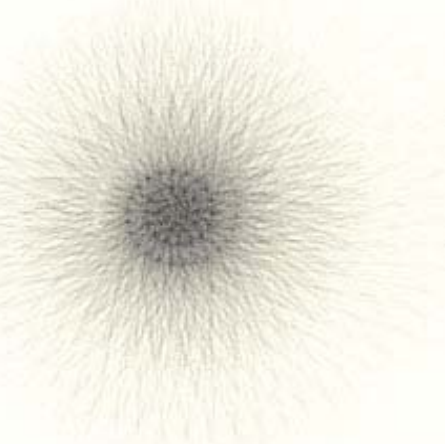
The Algorists, historical notes



Peter Beyls, 1988, Ghent, Belgium
Untitled algorithmic pen plotter drawing tinted with watercolor.
11.5" by 16.5". Artist's software
Hardware: Symbolics 3600 & HP plotter

The following are brief notes on the origin of the algorists with pointers to related theory and practice. Jean Pierre Hebert maintains several algorist pages on his web site at <http://hebert.kitp.ucsb.edu/studio/algorists.html>.

Who are the algorists? Simply put, algorists are artists who create art using algorithmic procedures that include their own algorithms. A key issue in discussions on algorithmic art centers on the relationship between art forms and the procedures employed in achieving those forms. With the advent of computers this relationship became crucial.



Jean Pierre Hebert, 1999. Santa Barbara, CA.
Artist's coded procedure from 100 views of a metagon
(<http://hebert.kitp.ucsb.edu/hv/hv.html>)
Giclee print 8" by 8" image on paper, Somerset Book, measuring 12.75" by 19".

In the last twenty five years of the 20th century, various symposia and conferences began to include exhibitions and papers related to the use of computing procedures in the arts. [8] By the late 1980s, a number of artists using original algorithms had achieved distinctive styles, each with a body of mature work. Working independently of each other, several found that they shared similar experiences over the years. Following the 1995 panel on "Artists and Algorithms" in Los Angeles, Jean Pierre Hebert, Ken Musgrave and I briefly discussed forming an informal group of artists who shared similar interests in algorithmic procedure. [9]

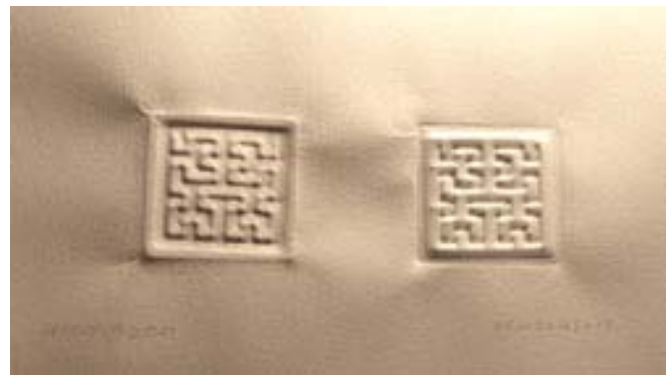
For several months, we corresponded in search of a term with possible ways to share views. Eventually we settled on the term "algorist" as defined by Jean Pierre. He wrote an algorithm that identifies an algorist as one who uses one's own algorithms for creating art objects. The classic Hebert algorithm, as quoted here, dates from correspondence in September 1995:

```
if (creation && object of art && algorithm && one's own
algorithm) {
include * an algorist *
} elseif (!creation || !object of art || !algorithm || !one's
own algorithm) {
exclude * not an algorist *
}
```

This definition identifies an algorist as one who creates an object of art employing algorithms that include one's own algorithm.

In the course of our work, with or without computers, we all employ algorithms created by our predecessors and colleagues. The use of algorithms in and of itself does not constitute algorist work. As defined in the Hebert algorithm, it is the inclusion of one's own algorithms that makes the difference.

Finally, one aspect of the algorist definition remains open to interpretation. An algorist, by definition, creates an 'object of art.' One might employ original algorithms in creating a scientific visualization that some may view as an 'art object.' Yet the visualization may not fit another's conception of art. Or one might employ original algorithms and create work that one person may consider a work of art while another may consider it unacceptable as a work of art. The definition does not attempt to identify what constitutes an 'object of art.'



The sculptor Helaman Ferguson, an algorist pioneer, made this impression directly from his coded carvings in Verostko's studio in 1997. signed with code: 961026165417

Later, in 1996, Hebert introduced several webpages dedicated to the algorists on his web site (<http://hebert.kitp.ucsb.edu/studio/algorists.html>). These initial pages identified an informal group of artists who were active algorists. Clearly there would have been many more algorists whom we did not know but whose practice would fit this definition. It was understood that a deeper understanding of algorist theory and practice, including its practitioners, would emerge in the years ahead.

The history of algorist art in the last quarter of the 20th Century presents many interesting questions on the

very nature of art. We may expect that -- as more detailed studies of late 20th century exhibitions and catalogues emerge -- that unknown algorists with surprisingly interesting work and perspectives will surface.

References:

Note 1. Algorithm, a variant term for algorism, most probably descended from the name of an Arabian mathematician who was active around 820 AD in the court of Manun in Baghdad. This mathematician, Abu Ja'far Mohammed Ben Musa, a native of Khwarasm, surnamed al-Khowarazmi, wrote treatises on Hindu arithmetic and algebra. The title of one of his works, al-jabrawa al-maqa'bala, is taken as the source for the term algebra. It is also believed that his name, al-Khowarazmi is the source for the term algorism. The use of the term algorism appeared with various spellings in several languages and often with latinizing influence as in algorismus. In English the term algorism came to be replaced with the term algorithm which is commonly used today. The transformation to algorithm may have been influenced by classical learning since the Greek term for number is 'arithmós, the root for our English term arithmetic.
Terminology briefing and example

Algorithm - a precisely detailed procedure for carrying out a task. Example:

(1) Identify two random points on a 100 unit square plane.

(2) Draw a line connecting these two points.

Software - Software consists of algorithms designed to execute specific tasks. The software (algorithms) must be coded in a computer compatible language. The example above could be coded for most computers with the following line of code in elementary BASIC:
window (0,100)-(0,100)

```
for n=1 to 4
p(n)=(rnd*101)
next n
line(p1,p2)-(p3,p4)
```

Pen Plotter - Designed primarily for engineering and architectural drawing, a pen plotter draws on paper with ink pens. These machines receive their instructions from software programs designed for architects & engineers. First generation plotter artists created their own software.

Note 2. The Greek origin of the term choreography, to write down (graphein) the dance (choreia), reveals its algorithmic nature.

Note 3. First generation pioneers included Piet Mondrian, Wassily Kandinsky, Kasimer Malevich, and the brothers Naum Gabo and Antoine Pevsner.

Note 4. Those drawn to view culture with neo-Darwinian spectacles will relish the evolution of this art. See Daniel Dennet's *Darwin's Dangerous Idea* (NY 1996) for an engaging discussion of the hyperspace of all possible books (Library of Babel, p 107 ff) and that of all possible genomes (Library of Mendel, p 111 ff). Writing on the new biology of machines Kevin Kelly identified The Library of Form, a frontier hyperspace of form being pioneered by Karl Sims (Chapter 14, *Out of Control*, 1994). I propose to identify the parameters for a Gallery of D'Arcy Thompson to embrace computable abstract art that is rigorously non-representational, i.e. non-objective, concrete, pure abstract art. Unveiling art within the hyperspace of forms with these parameters was certainly the dream of artists like Frantisek Kupka.

Note 5. The term epigenesis, borrowed from biology, refers to the process whereby a mature plant (phenotype) is grown from a seed or genotype (DNA). By analogy, the art work (phenotype) is grown from the software (genotype). The procedures for growing the work may be viewed as epigenetic. The code (genotype) for each series of works is capable of generating a family of forms with each being one of a kind. This procedure was employed in the limited edition of Boole's *Derivation of the Laws*. My 1988 Utrecht paper, "*Epigenetic painting: software as genotype*" (<http://www.verostko.com/epigenet.html>) outlines procedures developed up to that time.

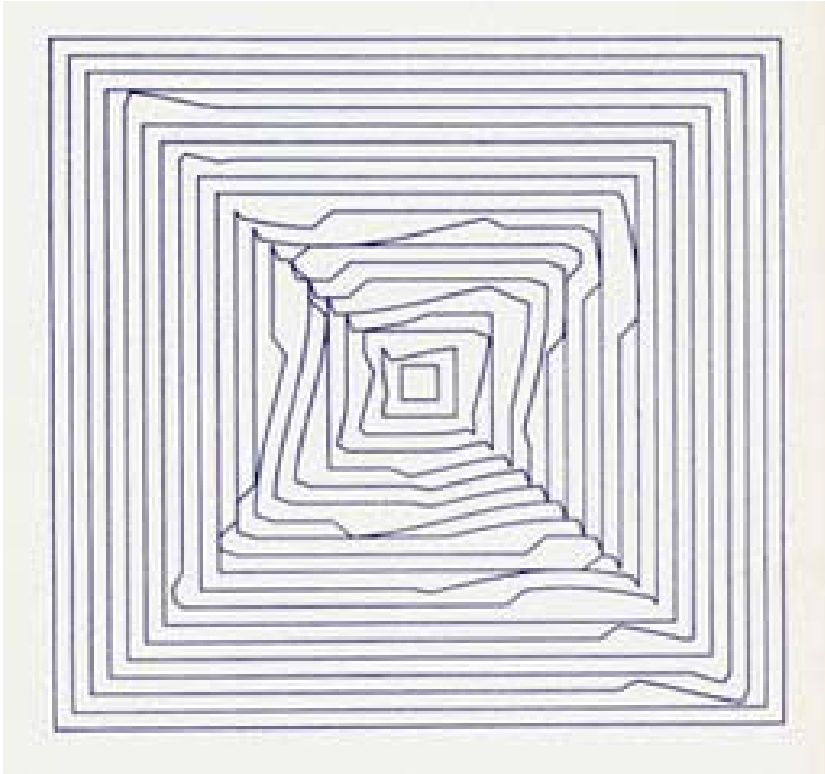
Note 6. See Art and Algorithm, (<http://www.verostko.com/alg-isea94.html>; ISEA '94, Helsinki), addresses procedures and issues related to an artist's use of algorithms.

Note 7. For example, Harold Cohen's early algorist work displayed form qualities similar to his pre-algorist work as a painter. The link between form and procedure remains one of the most important links to be explored in algorithmic art.

Note 8. By the late 1980s, the established symposia and exhibition venues that were known to me included: The Inter-Society for Electronic Art (ISEA), SIG-GRAPH, and Ars Electronica. The annual Small Computers in the Arts conference (Philadelphia, 1980 ff) was also an important venue in the U.S., recognizing the impact of the PC for individual artists. At all of these conferences, artists could see the work of others and share mutual concerns. The exhibitions, papers, panels, and publications of these venues provided an overview of what was generally called 'computer art' but there was no single venue for specifically 'algorist' work.

To address algorithmic procedure in the arts, I organized a small symposium at the Minneapolis College of Art and Design in 1991 (February 23/24). This symposium

sium *Art & Algorithm - Mind & Machine*, included an audio visual show, "Images of the Unseen From the Worlds of Art & Science." The presentation of video, sound tracks and slides included the work of 23 artists and scientists from 6 countries.



Algorithmic drawing, Vera Molnar. 74.338/14.29.00. c.1990
Courtesy of the artist for the 1991 symposium on Art & algorithm...

Artists and scientists whose work was shown included: Stephen C.G.Bell (UK), Donna Cox (US), Charlotte Davies (Canada), Hans Dehlinger (Germany), Helamen Ferguson (US), Samia A. Halaby (US), Bruce Hamilton (US), Jean Pierre Hebert (US), Yoichiro Kawaguchi (Japan), William Latham (UK), Vera Molnar (France), Jim Otis (US), Clifford Pickover (US), Jeffrey Ventrella (US), Mark Wilson (US), Toshifumi Kawahara (Japan).

Note 9. Peter Beyls (Belgium) and I laid plans in 1993 for a panel on Algorithms and the Artist for the Fourth International Symposium on Electronic Art (Helsinki, September 1994). The panel, with Peter as chair, included Brian Evans (US), Steve Bell (UK) and myself. With growing interest in these issues, Peter proposed and chaired a similar panel, with the addition of Jean Pierre Hebert and Ken Musgrave, at SIGGRAPH in LA the following year (1995). Following this LA panel we initiated correspondence for establishing a common identity. We adopted the term *algorist* as proposed by Jean Pierre Hebert.

Original text 1999, Roman Verostko. Compiled and edited with the kind permission of the artist.