UNCL

SECURITY C

AD-A197 526 DOCUMENTATION PAGE

DTIC FILE COPY

(4)

1a. REPORT NA	1b. RESTRICTIVE MARKINGS NA
2a. SECURITY CLASSIFICATION AUTHORITY NA	3. DISTRIBUTION/AVAILABILITY OF REPORT Distribution Unlimited; Approved for Public Release
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE NA	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) INDU/DC/GMH/TR-88-32	5. MONITORING ORGANIZATION REPORT NUMBER(S) NA
6a. NAME OF PERFORMING ORGANIZATION Indiana University — 6b. OFFICE SYMBOL (If applicable) NA	7a. NAME OF MONITORING ORGANIZATION ONR
6c. ADDRESS (City, State, and ZIP Code) Department of Chemistry, Bloomington, IN 47405	7b. ADDRESS (City, State, and ZIP Code) 800 N. Quincy Street, Arlington, VA 22217
8a. NAME OF FUNDING/SPONSORING ORGANIZATION — 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER Contract N00014-86-K-0366
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. — PROJECT NO. — TASK NO. R&T Code 4134006 — WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)

Applications of Cellular Automata: Attractors and Fractals in *ANALYTICAL* Chemistry?

12. PERSONAL AUTHOR(S)

Robert A. Lodder, Mark Selby, and Gary M. Hieftje

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical	FROM ____ TO ____	July 11, 1988	21

16. SUPPLEMENTARY NOTATION

Accepted for publication in Trends in Analytical Chemistry

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Cellular automata, Fractal mathematics, Chemometrics.

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A cellular automaton is a discrete dynamic system of simple construction, yet capable of exhibiting complex self-organizing behavior. A cellular automaton can be used to model differential systems by assuming that time and space are quantized, and that the dependent variable takes on a finite set of possible values. Cellular-automaton behavior falls into four distinct universality classes, analogous to (1) limit points, (2) limit cycles, (3) chaotic attractors (fractals), and (4) "universal computers". The behavior of members of each of these four classes is explored in the context of digital spectral filtering. The utility of class 2 behavior in experimental data analysis is demonstrated with a laboratory example.

DTIC ELECTE AUG 1 6 1988 S E D

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Distribution Unlimited	
22a. NAME OF RESPONSIBLE INDIVIDUAL Gary M. Hieftje	22b. TELEPHONE (Include Area Code) (812) 335-2189	22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

OFFICE OF NAVAL RESEARCH

Contract N14-86-K-0366

R&T Code 4134006

TECHNICAL REPORT NO. 32

APPLICATIONS OF CELLULAR AUTOMATA:

ATTRACTORS AND FRACTALS IN *ANALYTICAL* CHEMISTRY?

by

Robert A. Lodder, Mark Selby, and Gary M. Hieftje

Accepted for Publication

in

TRENDS in ANALYTICAL CHEMISTRY

Indiana University
Department of Chemistry
Bloomington, Indiana 47405

11 July 1988

Most chemists, if they have heard of cellular automata at all, are probably familiar with them only as a means of synthesizing complex computer-graphic images. The public's view of automata has been shaped by cinematic applications like the Lucasfilm Computer Graphics Laboratory's creation of the "Genesis bomb" explosion in *Star Trek II - The Wrath of Khan*. The perception of automata as an arcane mathematical construct has prevented their application to a number of problems for which they are naturally suited, and has left many researchers with the public's impression that cellular automata are best left in the hands of Hollywood's high-tech artists.

Nevertheless, cellular automata have been applied to both theoretical problems and experimental data analysis. For example, workers at the University of Toronto have used such constructs to theoretically model the formation of stars and galaxies[1], and cellular-automaton-based computers have been used to analyze data transmitted to earth from Landsat satellites[2]. The purpose of the present review is to outline what a cellular automaton is and how it works, and to show how automata can be used to study real systems. The cellular automaton as a problem-solving machine will be compared to the more common Turing-machine approach in the course of this discussion.

In essence, the comparison between Turing machines and cellular automata is a comparison between sequential and parallel methods of problem solving. Turing machines and cellular automata are both mathematical constructs that can be used to investigate the process of computation. Modern computing is based on theories developed using the

Turing machine, and the use of parallel constructs like cellular automata represents a fairly recent development in computing.

The Turing machine and its operation have been described thoroughly[3], and only a brief discussion is necessary here. The Turing machine is usually thought of as a mechanical device, although few would bother to actually build one. (The operation of the machine is much more efficiently simulated on paper.) The "machine" is composed of a tape (containing "ones" and "zeroes") and a tape scanner, and is really quite similar to a cassette recorder. The only real difference between the Turing machine and the tape recorder is that the Turing-machine head reads and writes in both directions on a single track. The direction of the head's movement and its read/write mode are determined by the state of the machine and the information configuration of the data tape. The head moves in discrete steps and operates on only one value on the tape at a time. The head's next step is determined by the current value on the tape (at the head) and the current state of the machine (a configuration of "switches" that is itself determined by the previous value at the tape head). Despite its seemingly simple construction, the Turing machine has been shown to be capable of performing the most complex calculations, given enough time.

The simplest cellular automaton resembles the Turing machine in that it also starts with a one-dimensional array of data values. In cellular automata, however, each site can directly communicate with some of the others, and the values at all of the positions along the array are updated simultaneously instead of one at a time (sequentially). A cellular automaton is perhaps the simplest architecture for a massively parallel processor.

In a typical cellular automaton, each cell in the array contains one member of a finite set of possible cell values. The values in all of the cells change according to the same set of rules (a program). These rules describe the state of a cell as a function of the previous value in the cell and the previous values of the cells in some defined neighborhood around the central site. The values in all of the cells are synchronously updated. In the simplest case, a cellular automaton is a one-dimensional array of cells arranged on a line so that each cell has only two nearest neighbors. A more common computational configuration has the cells in a 2-D square array, giving each cell eight nearest neighbors. Mathematical analyses of cellular automata treat the cells essentially as "mailboxes", and describe the temporal evolution of structure in the cell-value configuration. The computational implementation of cellular automata is a bit more complex, with each cell representing a simple processor, some memory, and an I/O device. (As a matter of historical interest, the name "automaton" came about as the result of John von Neumann's attempts to construct a machine that could reproduce itself. When Von Neumann adopted Stanislaw Ulam's suggestion to move the search for self-reproducing systems to the mathematical level, and Von Neumann discovered geometric configurations that spontaneously replicated in an array, cellular automata were born.)

The difference between conventional computing and computing in a cellular-automaton framework is essentially the difference between parallel and seqential methods of problem solving. In sequential problem solving, of course, a problem is broken into steps, each one of which must be solved before the next step can be attempted. In parallel problem solving, however, the problem is broken into pieces somewhat differently, so each piece can be solved independently of the others. (As mentioned above, cellular automaton sites generally communicate with a certain neighborhood of nearby cells, so the calculations are not totally independent.) Turing worked to prove the sequential method capable of universal computing, while Conway and Wolfram did the same for the parallel method using cellular automata.

Cellular-automaton behavior (in terms of cell-value configurations) falls into four distinct universality classes[4]. Because cellular automata can be considered approximations of differential equations (with large numbers of discrete degrees of freedom), it is not too surprising that these configuration classes parallel the kinds of behavior of nonlinear ordinary differential equations. Each configuration class can be categorized with respect to its self-organizing behavior, and in particular to its attractor. A good appreciation of each of these classes of behavior can be obtained when one examines their performance in a digital-filtering context. To simplify the following discussion, a synthetic digitized "spectrum" of 100 readings was created with full-scale peaks (given a value of 1) and a noiseless baseline (given a value of zero). The 100 synthetic spectral values that were created are: 011001101100111101100101111111101100101010001000010101000000010000111000011 001101111101011100101101010. The plots that are Figures 1-4 depict contours of a 100-by-100 element array denoting transitions between the 0 and 1 readings. The original "spectrum" (which is the same for Figures 1-4) occurs above the top row of each figure. The subsequent rows in the figures represent 100 successive synchronous steps in the temporal evolution of the 1-D automata. Each successive synchronous step is equivalent to a filtering pass on the previous spectrum, producing a new sequence of readings.

Figure 1 depicts a cellular automaton rule acting on the spectrum to produce class 1 behavior. Class 1 behavior is characterized by a limit-point attractor: temporal evolution from all initial states tends toward a homogeneous final state. In other words, a class 1 filter acting on a spectrum tends to reduce all of the peaks to a flat baseline (with a DC offset, perhaps). Running-means filters (among other rules) have this effect.

Figure 2 depicts a class 2 cellular-automaton rule acting on the same initial spectrum used in Figure 1. Class 2 rules are characterized by limit-cycle behavior: temporal evolution from all initial states tends to propagate the value-structures around some sites indefinitely while extinguishing the values at the remaining sites. The value of a particular cell after a large number of time-iterations depends upon the initial values at a definite number of the "original" cells. Running-median and certain other nonlinear filters show this kind of behavior, which is desirable from a spectral-filtering or image-enhancement standpoint, because all values that are not part of a target structure are eventually extinguished.

Figure 3 shows a class 3 rule acting on the same initial spectrum used in Figures 1 and 2. Class 3 rules are characterized by chaotic (strange) attractors: temporal evolution from all initial states leads to reproduction of the original structures at seemingly random locations and scales. This random scaling is typical of fractals, and in fact, fractal dimensions can be calculated for cellular automata[4]. As the number of time-iterations increases under a class 3 rule, the value in a particular cell is determined by an ever-increasing number of the initial cells. The chaotic pattern that evolves has a fractal self-similarity that is undesirable from a spectral-filtering point of view, because the pattern means that peaks would spontaneously replicate in the spectrum during filtering, and would constantly change location and magnitude.

Figure 4 illustrates the fourth class of cellular-automaton behavior, again using the same initial spectrum as the previous three figures. Class 4 rules are characterized by the lack of a definite attractor: temporal evolution from all initial states leads to configurations that may continue to change indefinitely, or may become extinct after the next time iteration. The value of a particular cell after a number of time iterations depends on the values of an indeterminate number of initial cell values. In other words, there is no general finite algorithm capable of predicting whether the cellular automaton will evolve to a homogeneous state after a certain amount of time, or whether some structures will continue to exist after an arbitrarily large number of time iterations. The final configuration of a class 4 cellular automaton is thus formally undecidable, except by the explicit simulation of every step in the evolution of the cellular automaton. The undecidability of the parallel-processing problem in cellular automata corresponds to the insolubility of the halting problem in universal Turing machines (the logical construct upon which modern CPU-based computers are predicated)[4]. Class 4 cellular automata can be considered as models for parallel processing in general, and some work indicates that cellular automata may be better models for computing at certain levels than the more conventional Turing-machine construct[5].

Parallel processing using cellular automata has many real applications, both inside and outside of chemistry. Mackay[6] has proposed cellular automata as a way of developing a unified theory of crystal formation capable of describing the 230 space groups as well as the occurrence of pseudosymmetry and local symmetry. Burks and Farmer[7] at the Los Alamos National Laboratory have been working on modeling the evolution of DNA sequences using cellular automata. Axelrod and Hamilton[8] have described the evolution of cooperation in a Darwinian world using elegant experiments involving cellular automata and a variety of rules contributed by invited theorists and by people responding to a magazine advertisement. At Indiana University we have recently been applying cellular automata to the three-dimensional imaging of surfaces by means of near-infrared reflectance spectrometry. The types of surfaces we have chosen to investigate are painted, specifically, the painted walls in historic buildings that are being restored[9].

Over time, the accumulation of dirt and smoke on historic surfaces, as well as the aging of overcoat layers like varnishes, gradually obscures both the color and pattern of paintings, murals, and walls. Typically a general darkening of the surfaces gradually destroys their color and contrast. Later, restorers may repaint these surfaces to match their present (decadent) condition, and might consequently conceal entirely detail that was already becoming obscured by the aging process. Even very few such cycles of aging and restoration are sufficient to alter fundamentally the nature of a surface covered with decorative designs. Finally, political and aesthetic considerations often result in the deliberate and total concealment of original surfaces in historic buildings. In these cases the circumstances and even locations of such alterations are often long forgotten.

The usual method of investigating historic surfaces employs stereomicroscopic analysis and manual dissection of individual paint layers to detect the presence of concealed images and designs. This method is tedious and expensive, and it unnecessarily damages the areas that do not conceal interesting subsurface patterns. A spectroscopic method is really simpler and less destructive.

In our work, a target surface was divided (not necessarily physically) into pixels (see Figure 5), the near-infrared spectra of these pixels was loaded into a cellular automaton, and the cellular automaton rules were selected. The proper choice of (class 2-type) rules forces the temporal evolution of the automaton to converge on an attractor that is the image of the subsurface design (see Figure 6). The use of this nondestructive spectroscopic method of reconstructing subsurface images might permit restorers to choose intelligently either to restore the original designs they find, or to duplicate them on a freshened surface with new stencils.

Cellular automata have contributed much to computer graphics, and they have much to contribute to chemistry and other sciences as well. Major changes in parallel processing and the implementation and role of pattern recognition are now underway[10]. The cellular-automaton model suggests that more than just the process sensors used in pattern-recognition methods can benefit from simplification: the computers, and even the calculations themselves, can benefit from a union of simplification and parallelism. Future work, particularly in the area of parallel algorithms and the design of instruments optimized for use with such algorithms, will open up a range of applications that have yet to be imagined. K[illegible] Fractal Mathematics, Chemometrics.

ACKNOWLEDGEMENT

This work has been supported in part by the National Science Foundation through Grant CHE 83-20053, by the Office of Naval Research, and by The Upjohn Company.

REFERENCES

1. Freedman, W. L.; Madore, B. F. *Astrophys. J.* **1983**, *265*, 140-147.
2. Peterson, I. *Science News* **1987**, *131*, 28-30.
3. Hopcroft, J. E. *Scientific American* **1984**, *250*, 86-98.
4. Wolfram, S. *Physica D*, **1984**, *10*, 1-35.
5. Carter, F. L. *Physica D* **1984**, *10*, 175- 194.
6. Mackay, A. L. *Phys. Bull.*, **1976**, *27(Nov.)*, 495-497.
7. Burks, C.; Farmer, D. *Physica D*, **1984**, *10*, 157-167.
8. Axelrod, R.; Hamilton, D. *Science*, **1981**, *211*, 1390-1396.
9. Lodder, R. A.; Hieftje, G. M. *Appl. Spectrosc.*, **1988**, *42*, 309-312.
10. Hecht-Nielsen, R. *IEEE Spectrum*, **1988**, *25*(3), 36-41.

FIGURE CAPTIONS

Figure 1. A class 1 rule operating on a synthetic spectrum.

Figure 2. A class 2 rule operating on a synthetic spectrum.

Figure 3. A class 3 rule operating on a synthetic spectrum.

Figure 4. A class 4 rule operating on a synthetic spectrum.

Figure 5. The Indiana University (IU) logo was painted in this format (using red and green acrylic paints) on illustration board. The logo was then covered with one layer of white acrylic paint and one layer of white enamel paint.

Figure 6. The attractor for the cellular automaton formed by the near-infrared reflectance spectra of the overcoat-concealed painting in Figure 5.

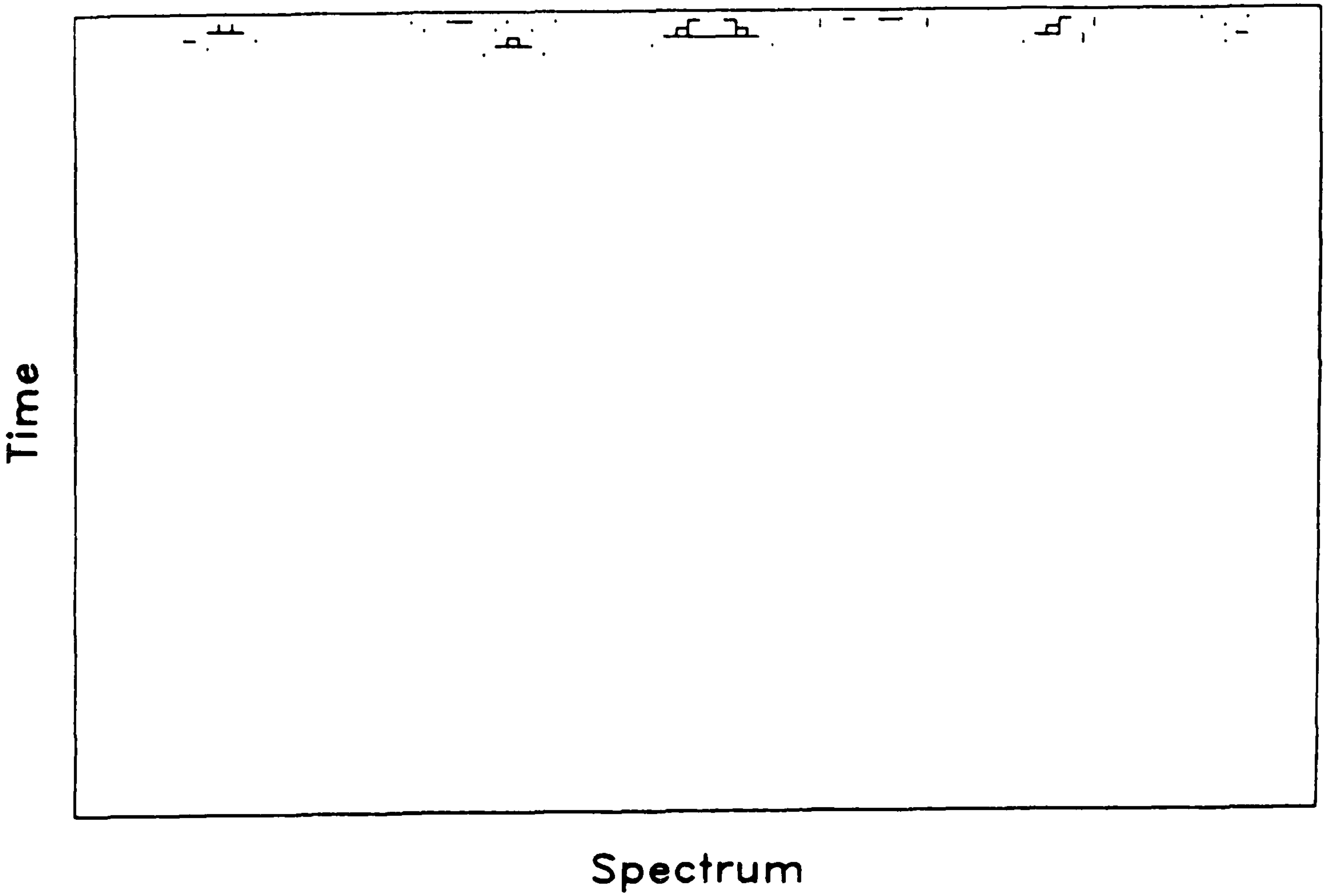

Fig. 1

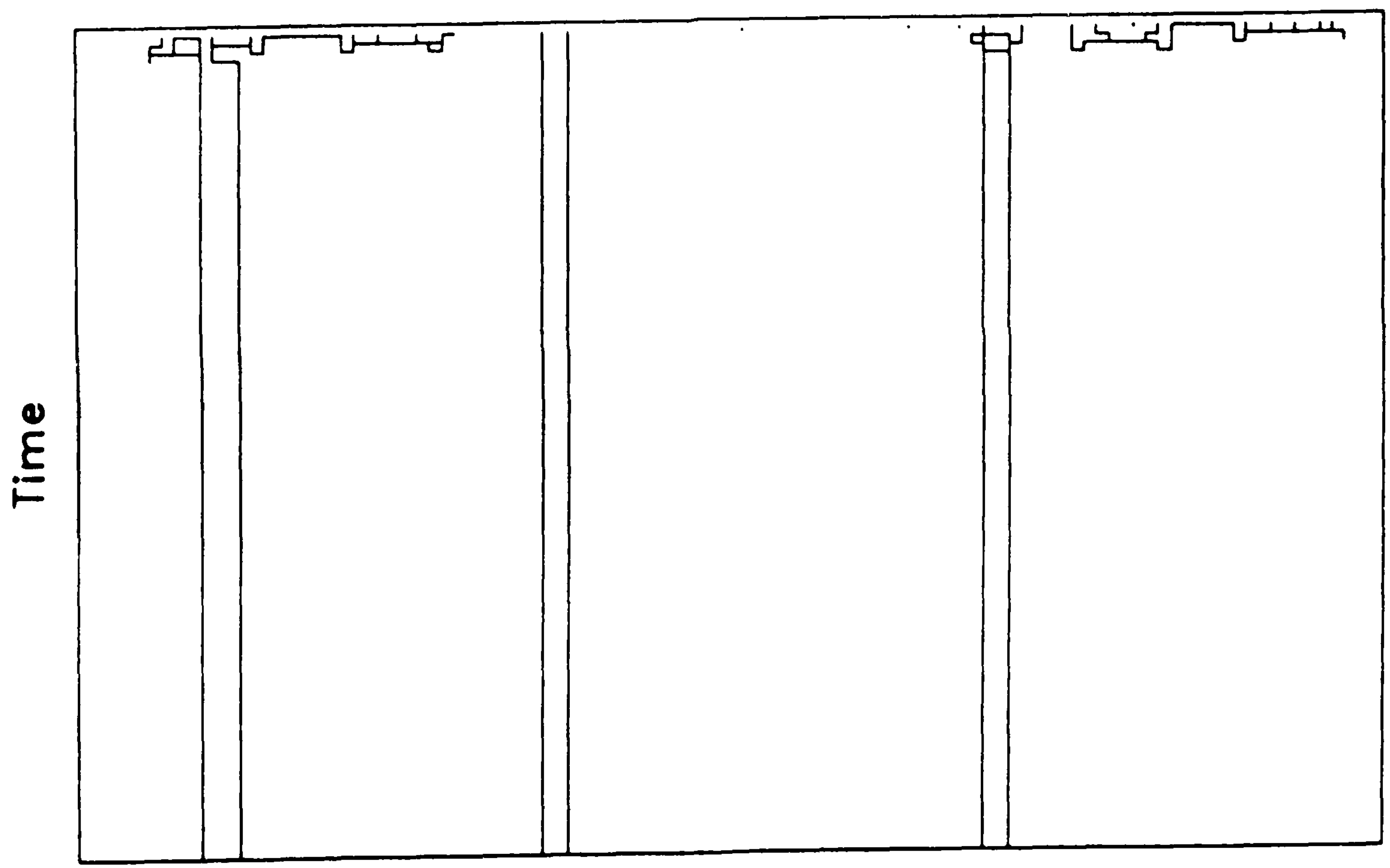

Fig. 2

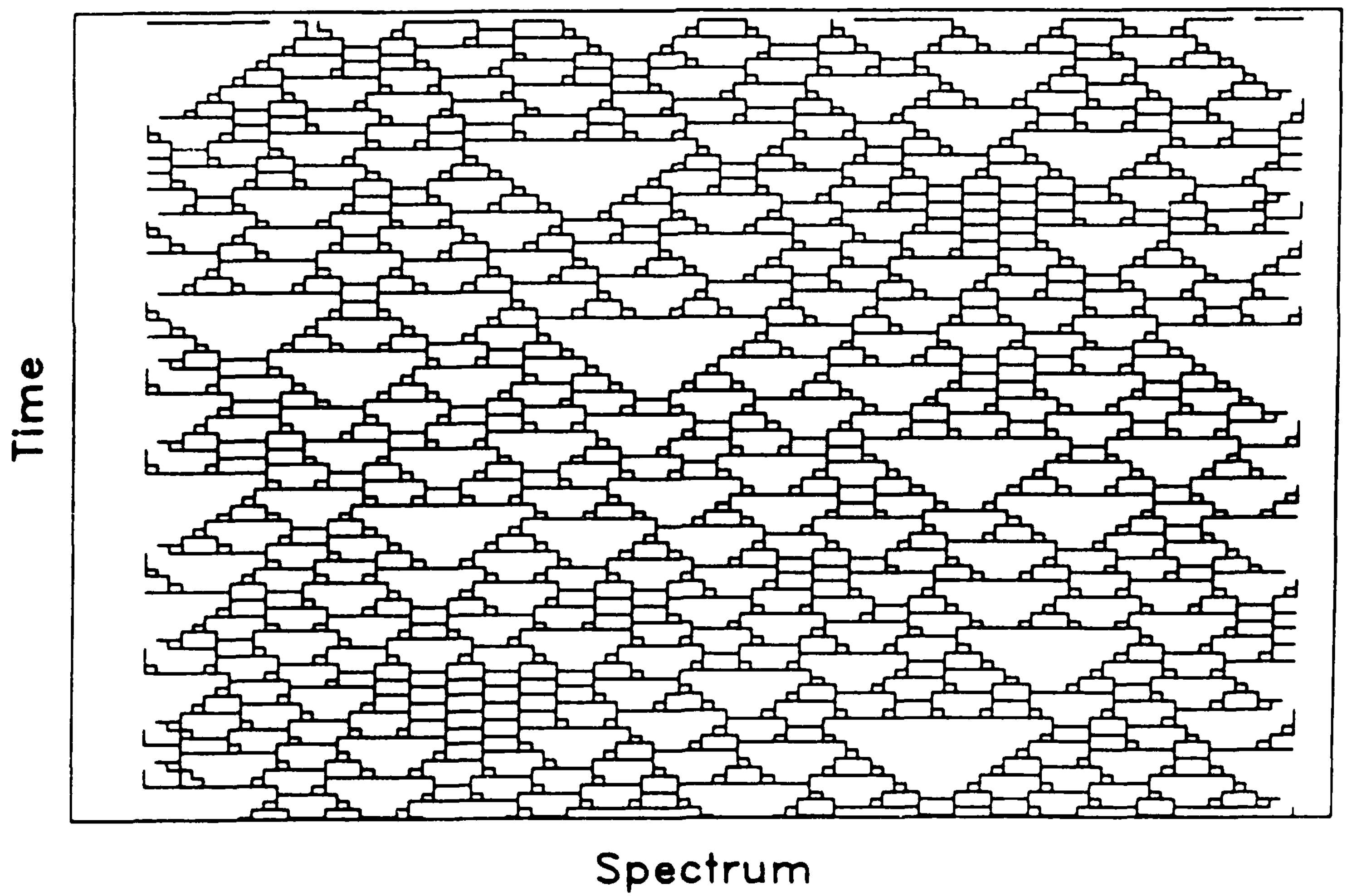

Fig. 3

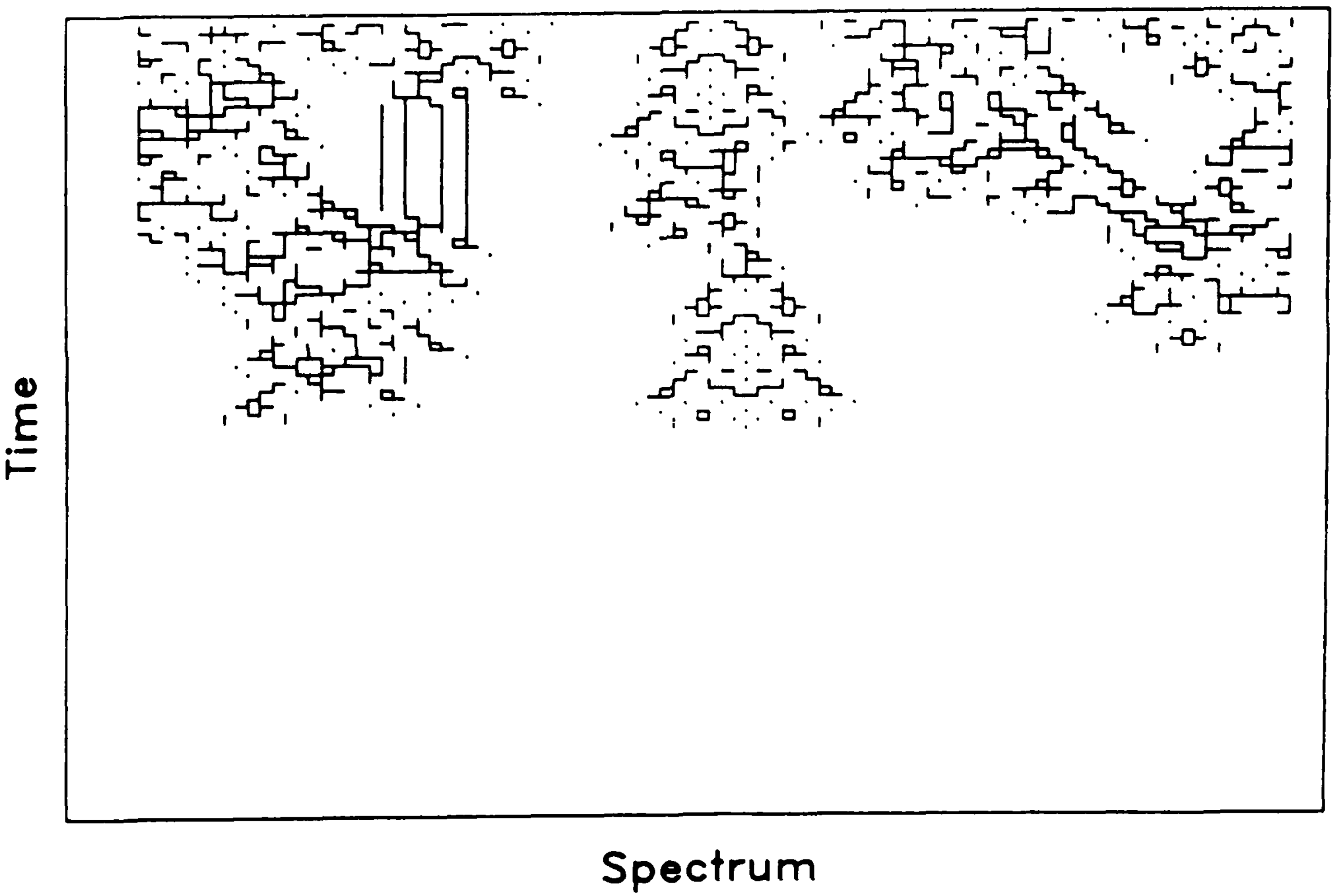

Fig. 4

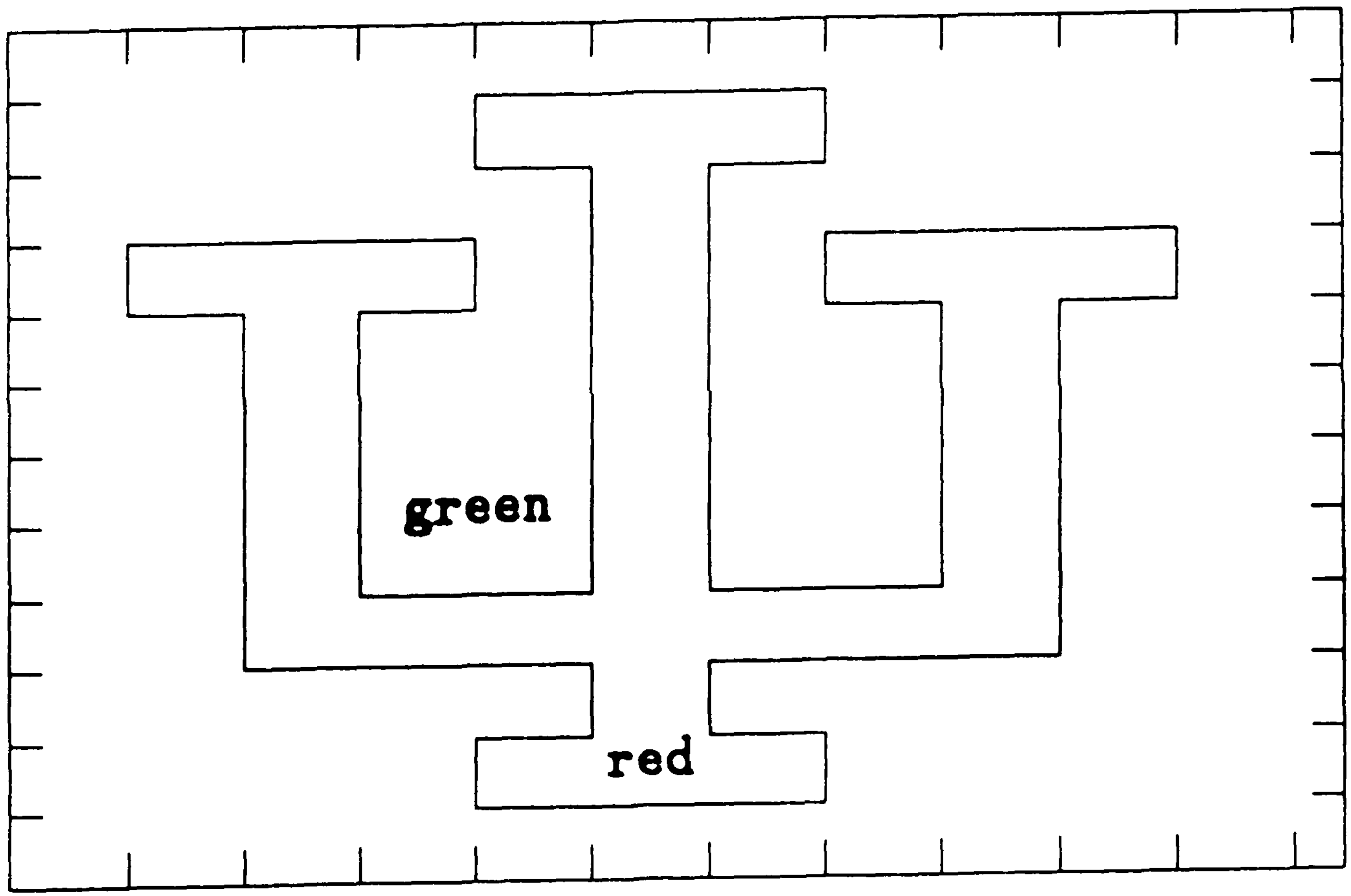

Fig. 5

Fig. 6

TECHNICAL REPORT DISTRIBUTION LIST, GEN

	No. Copies
Office of Naval Research Attn: Code 1113 800 N. Quincy Street Arlington, Virginia 22217-5000	2
Dr. Bernard Douda Naval Weapons Support Center Code 50C Crane, Indiana 47522-5050	1
Naval Civil Engineering Laboratory Attn: Dr. R. W. Drisko, Code L52 Port Hueneme, California 93401	1
Defense Technical Information Center Building 5, Cameron Station Alexandria, Virginia 22314	12 high quality
DTNSRDC Attn: Dr. H. Singerman Applied Chemistry Division Annapolis, Maryland 21401	1
Dr. William Tolles Superintendent Chemistry Division, Code 6100 Naval Research Laboratory Washington, D.C. 20375-5000	1

	No. Copies
Dr. David Young Code 334 NORDA NSTL, Mississippi 39529	1
Naval Weapons Center Attn: Dr. Ron Atkins Chemistry Division China Lake, California 93555	1
Scientific Advisor Commandant of the Marine Corps Code RD-1 Washington, D.C. 20380	1
U.S. Army Research Office Attn: CRD-AA-IP P.O. Box 12211 Research Triangle Park, NC 27709	1
Mr. John Boyle Materials Branch Naval Ship Engineering Center Philadelphia, Pennsylvania 19112	1
Naval Ocean Systems Center Attn: Dr. S. Yamamoto Marine Sciences Division San Diego, California 91232	1