MASS. INST. OF TECHNOLOGY
NOV 29 1967
LIBRARIES

COMPUTER DISPLAY OF PROTEIN ELECTRON DENSITY FUNCTIONS

by

DAVID E. AVRIN

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
August, 1967

MASS. INST. OF TECHNOLOGY
MAY 22 1968
ENGINEERING LIBRARY

Signature of Author . . . . . .
Department of Electrical Engineering, August 21, 1967

Certified by . . . . . .
Thesis Supervisor

Accepted by . . . . . .
Chairman, Departmental Committee on Graduate Students

COMPUTER DISPLAY OF PROTEIN ELECTRON DENSITY FUNCTIONS

by

DAVID E. AVRIN

Submitted to the Department of Electrical Engineering on August 21, 1967, in partial fulfillment of the requirements for the Degree of Master of Science.

ABSTRACT

Through X-ray diffraction techniques, it is possible to obtain a synthesized electron density distribution of a protein crystal. This thesis develops a procedure, stressing real time, man-computer interaction, for using an on-line, simulated three-dimensional display system as an aid for the visualization, determination, and modelling of protein structure from electron density information. This procedure was developed using the facilities of Project MAC, and in particular, the Electronic Systems Laboratory Display Console.

We developed a procedure, using the mode of contouring, that can meaningfully present regions of a continuous, three-dimensional scalar function to a human observer as simulated surfaces of constant density. A routine was developed to extract and display a contiguous volume of high density from the confusion of its surroundings, upon initiation by light pen identification performed by the observer. This routine appears to be a promising aid for determining the pathway of the polypeptide chain. In effect, it performs a pre-processing to enable the observer to more effectively use his pattern recognition abilities.

We also developed a simple routine to allow the user to operate on and model the patterns he observes by drawing three-dimensional line segments.

The thesis includes discussion of the algorithms of the above procedures.

An operational procedure was devised to effectively apply the computer routines developed to an investigation of a protein structure.

A test was performed on the electron density distribution of myoglobin, a protein whose structure has been well established, with the intention of evaluating the methods developed. Fair but encouraging results were obtained. There was a strong indication of the necessity for a display system capable of plotting many more line segments with less flicker, and for use of a synthesized electron density function of higher resolution.

The user interaction facilities developed operated smoothly and conveniently. Furthermore, the capability to contain in core memory storage the quantity of information required to specify the electron density distribution of a protein crystal, as well as the programs required to transform this three-dimensional scalar function to a meaningful display and provide for effective user interaction was demonstrated.

Thesis Supervisor: Cyrus Levinthal
Title: Professor of Biophysics

ACKNOWLEDGMENT

The author hopes that the limited, but encouraging results of this thesis can in some way express his gratitude to Professor Cyrus Levinthal. Professor Levinthal formulated the problem investigated herein, provided ideas, criticism, encouragement, and a large degree of freedom.

This thesis is a beginning, and like most partially successful beginnings, it brings together the efforts and ideas of numerous people. The basic problem attacked herein was also investigated by William Brody (Computer Display of Three-Dimensional Scalar Functions, M.I.T. MS 1966). His suggestions for a better approach were as invaluable as they were correct. Programming assistance, criticism, and biological advice of Martin Zwick, Thomas Warner, and David Barry were essential.

Two other individuals, Daniel Thornhill and Harold Levin of the Electronics Systems Laboratory at Project MAC devoted almost their entire summer to completing the software for the new Electronic Systems Laboratory Display Console in time to be used for this thesis. Their "factor of two" enlargement of the display space spelled the difference between mediocre and encouraging results--between seeing structure and not seeing structure.

Myoglobin X-ray diffraction data (intensities and phases) was obtained from J. C. Kendrew and H. Watson, Medical Research Council Laboratory, Cambridge, England.

X-ray data was converted to the electron density function using

the MIFR 2-A Crystallographic Fourier Program for the IBM 7094, written by D. P. Shoemaker, W. G. Sly, and J. H. van der Hende.

Work reported herein was supported in part through the National Institutes of Health, Grant GM 13813-02, Computer-Controlled Displays in the Study and Teaching of Molecular Biology, and in part by Project MAC, an M.I.T. research program sponsored by the Advanced Research Projects Agency, Department of Defense, under Office of Naval Research Contract Number Nonr-4102(01). Reproduction in whole or in part is permitted for any purpose of the United States Government.

The contact the author has had over the past two years with Professor Levinthal, stimulating lecturer, enthusiastic scientist, imaginative advisor, and generous man, has been a most rewarding and pleasant experience.

And to Cynthia--not an enthusiastic scientist, engineer, programmer, or biologist--but a photographer, typist, and indispensible factor of two . . .

TABLE OF CONTENTS

LIST OF FIGURES

I. INTRODUCTION

This thesis develops a procedure for using an on-line, simulated three-dimensional display system as an aid for the visualization and determination of protein structure from electron density information. Our method has its basis in the traditional means used in protein X-ray crystallography for visualizing molecular structure in electron density data. However, the use of a digital computer with an on-line electronic display system, with facilities for user interaction, has allowed us to make significant departures from the methods used in this area previously.

Restricted by the limitations imposed by the hardware available, we sought to develop an optimum system. We started with only a general notion of the type of display we desired, but quickly began to take advantage of the extensive flexibility offered by a display system such as the Electronic Systems Laboratory Display Console at Project MAC.

The procedure that subsequently evolved is described, demonstrated, and evaluated in the following pages. Essentially a continuation of work begun by William Brody[1], this thesis devises variations on one particular mode of presenting three-dimensional scalar functions: that of contouring. It stresses the development of user interaction provisions to increase "the usefulness of the man-computer combination in solving real problems in molecular biology."[2]

II. THE THREE-DIMENSIONAL SCALAR FUNCTION PROBLEM

2.1 General Considerations

Although many of the procedures developed in this thesis are applicable to investigation of any three-dimensional scalar function, the particular problem to which we addressed ourselves is that of determination of protein structure from X-ray crystallographic data. For a discussion of the process of X-ray crystallography, consult the reference by Wilson.

For the moment, consider the general problem of three-dimensional scalar functions. The visualization of these functions poses severe problems. The difficulties are much more complicated than the simple analogy of looking for a brick in a cloud. More often, the problem is one of visualizing subtle changes in the density of the cloud, recognizing a pattern in these variations, or identifying specific shapes or continuities.

Our visual world is composed primarily of opaque surfaces which have boundaries, obscure objects behind them, and are in turn obscured by objects located in front of them. In a continuous three-dimensional scalar function such opaque surfaces do not exist. Such a function contains more, and more subtle information than a function which possesses discrete surfaces. All the gradual variations in the cloud are seen at once by the observer, and spatial relationships are difficult to determine. Furthermore, the shapes of objects are difficult to define.

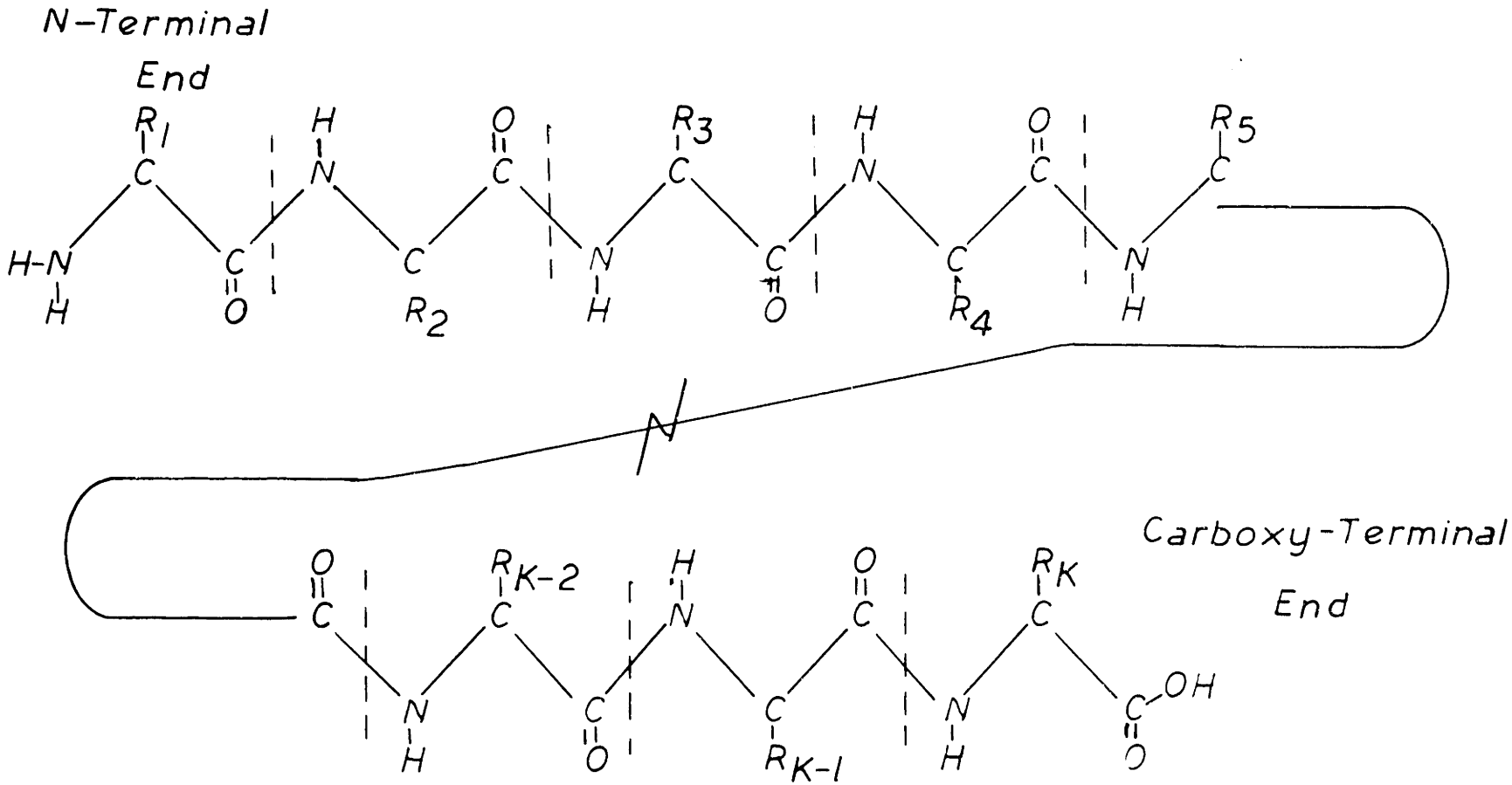
There are two broad classes of approach to the above dilemma. The first seems to involve improving the ability of the observer or aiding

him in some way to work directly with three-dimensional cloud patterns. However, even with assistance, the human seems to be incapable of operating in this visual realm. The second class of approach involves a display system which performs some type of transformation or information reduction scheme that will make the necessary leap from the continuous three-dimensional scalar function to the more familiar visual world of discrete objects, lines, and surfaces. There are many types of transformations which can be made. However, before discussing the procedure that we have developed, let us consider the special problem of electron density distributions of protein crystals.

2.2 Special Problem of Protein Crystal Electron Density Functions

The following comments are not intended to be a thorough discussion of protein structure, but rather a general discussion of some pertinent features of protein structure which affected the type of procedure we developed.

In general, a protein is a linear polymer of amino acids, as shown in Fig. 1, known as a polypeptide chain. The number and variety of amino acid units varies from protein to protein. The number of units is of the order of 150, which results in an extended polypeptide chain length of about 500 Angstroms. The molecular weight of such a molecule is between 15,000 and 20,000 and consists of approximately 1,000 to 1,500 atoms excluding hydrogen. In a biologically active protein, this linear chain assumes a specific three-dimensional conformation. This three-dimensional structure, known as the tertiary structure, is intimately related to the biological function of the pro-



(II)

Figure 1. Primary Structure of a Protein

tein. Hence this tertiary structure is a question of extreme importance to the molecular biologist or biophysicist.

The technique of X-ray crystallography makes these three-dimensional structures accessible to investigation. A purified protein solution is crystallized and the resulting crystal subjected to X-ray diffraction studies. The electron cloud of the macromolecule scatters the X-rays. Through various tricks known as isomorphous replacement and through Fourier mathematics, it is possible to obtain a synthesized electron density distribution of a unit cell of crystal.

> Consider a continuous function $p(x,y,z)$ whose value at every point in the crystal is a measurement of the electron density at that point. Because electron dense regions correspond to the position of atoms, a sufficiently detailed knowledge of $p(x,y,z)$ suffices for the localization of all the atoms in the unit cell.[3]

The procedures which we have developed are primarily concerned with investigating this three-dimensional electron density function. A satisfactory display system must enable the user to determine the location of the protein molecule in the unit cell and the general shape of the molecule, to follow the pathway of the backbone of the polypeptide chain in the synthesized electron density function of the unit cell, and to construct a model of this pathway. This is obviously a taxing problem in human pattern recognition.

2.3 Previous Approaches

The most common and successful approach used to date by protein crystallographers is the "Perspex" model. The electron density function can be thought of as being comprised of sequential planes

of two-dimensional scalar functions. The Perspex model is constructed from a series of parallel Lucite sheets, each of which has drawn upon its surface an ensemble of contours of varying density levels obtained from a planar section of the unit cell data. Each Lucite sheet represents one two-dimensional plane in the data array. There are numerous difficulties with this approach. Such models are difficult and time-consuming to construct, and the difficulty of visualizing regions in the center of the model remains. Furthermore, planar constraints or effects are imposed upon the density array by the fact that contouring occurs only in a single planar orientation. For example, a rod-like structure such as an alpha helix will appear differently to the observer when positioned in different orientations with respect to the contouring planes. A planar molecular group such as the heme region in myoglobin will appear differently depending upon whether the contouring planes are perpendicular to the plane of the heme or parallel to it. A third obstacle is that it can be difficult to trace the pathway of the polypeptide chain between layers of the model.

An approach at satisfactory computer generated dot-density display of a scalar function was attempted by William Brody.[4] He attempted to generate a dot-density display and simulate three-dimensionality through real time control of the display of serial sections. The results of his thesis were not very encouraging for intensity-coded displays, but yielded invaluable advice concerning more promising approaches. Although he felt that the particular type of display he attempted was not very successful in being a satisfactory solution to the problem, he did voice a belief that "suitable methods for displaying three-

dimensional scalar functionswith the assistance of the digital computer do exist."[5]

It should be added that two-dimensional contouring programs for simplifying the construction of Perspex models do exist.[6]

III. PROCEDURE DEVELOPMENT AND DESCRIPTION

3.1 Objectives

Our approach went forward from the work done by Brody by first back-tracking one step. Due to the limited success of the intensity-coded serial section displays (essentially an attempt to reconstruct directly the density information as an intensity-coded display of successive two-dimensional sections), we chose to generate a contour display, similar to the Perspex model but which would be generated by a digital computer and be viewed on a simulated three-dimensional display system such as the Electronic Systems Laboratory Display Console at Project MAC. Our original goal was to compare the advantages and limitations of a computer generated, on-line contour display and to explore what could be done with such a system that could not be done with a physical model. This is really a two part question. The first part concerns the type of picture the computer can generate and meaningfully display, while the second part concerns to what extent the user may interact with such a display and with the computer generating it in order to bring his pattern recognition abilities and protein knowledge to bear upon elucidating the protein structure under investigation.

3.2 Available Hardware and Software

This project has been developed and performed using the Compatible Time-Sharing System (CTSS) and the Electronic Systems Laboratory Display Console (ESLDC) at Project MAC at M.I.T. CTSS

presently operates with an IBM 7094. The system can service up to thirty on-line users simultaneously who have access to the computer at any one of several hundred teletypewriter stations through a Dataphone link. The system is extremely reliable and has such useful conveniences as on-line program editing, execution, debugging, and interaction. The core memory of the CTSS 7094 is divided into two sections: A-Core (for the CTSS supervisor system) and B-Core (for the user's programs).

The bulk of the project was executed using the ESLDC. "The purpose for the development of the ESL Display Console was to provide direct, fast, computer-controlled display plus a flexible set of input devices including a light pen."[7] It is connected to the Direct Data Connection of the CTSS IBM 7094 Data Channel. Its first useful feature is its ability to simulate the display of three-dimensional line segments through user-controlled, real time, simulated rotation. The ESLDC calculates the isometric projection of the line segments onto any two-dimensional plane through multiplication by a rotation matrix which specifies the orientation of the two-dimensional plane with respect to the original coordinate system of the line segments. This rotation matrix is dynamic; the user can control its rate and direction of rotation with a "globe" (a plastic hemisphere approximately six inches in diameter, gimballed about three orthogonal axes). The user twists or pushes the globe in the direction of the desired rotation, and the globe controls in real time the rate of updating of the rotation matrix in the desired direction. Actually, the plane of projection is

fixed (the face of the display screen) and, consequently, the three-dimensional array of line segments appears to rotate with respect to the face of the display screen. The sequential projections repeat at a rate dependent upon the number of segments in the picture. Except for very large or detailed pictures, this rate is above the perceptible flicker rate, and a smooth, continuous rotation is observed.

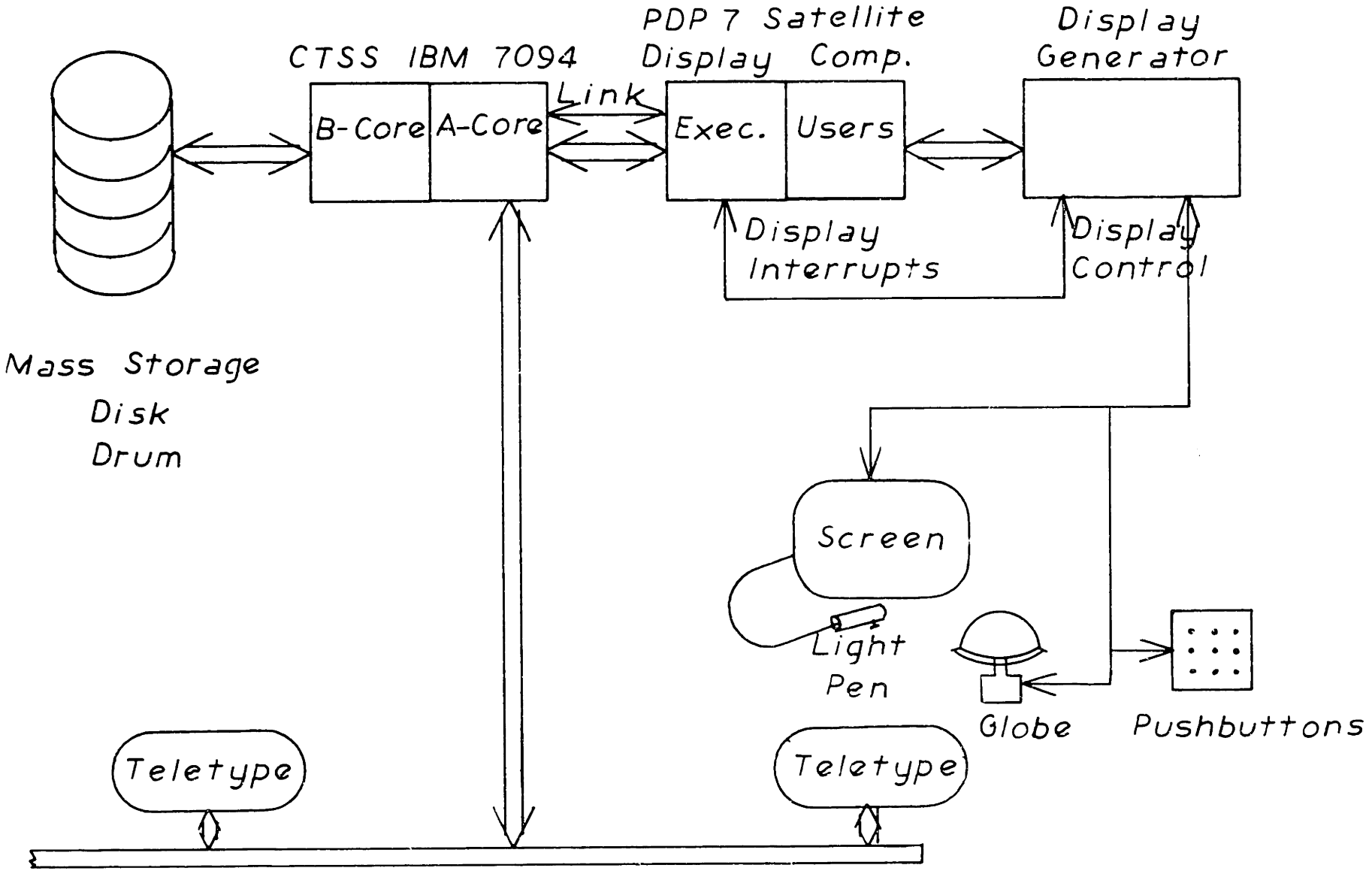
> The brain of the human viewer readily constructs a three-dimensional image from the sequential display of the two-dimensional images. Such sequential projections seem to be just as useful to the brain as stereoscopic projections viewed by two eyes. The effect of rotation obtained from the continuously changing projection nonetheless has an inherent ambiguity. An observer cannot determine the direction of rotation from observation of the changing picture alone. However, with a little practice the coupling between the hand controlling the position of the globe and the brain becomes so familiar that any ambiguity in the picture can be resolved.[8]

Other useful features include interaction facilities through push buttons and a light pen. The light pen operates in two modes: to identify a particular line segment or segments on the display screen or to move a tracking cross about the face of the screen. The number of a push button depressed, the location in the display list of a particular line segment identified with the light pen, and the horizontal and vertical coordinates of the tracking cross are available as output from the ESLDC for input to a program under execution in B-Core of the CTSS 7094.

The display screen itself is a Type 340 Incremental CRT made by the Digital Equipment Corporation.

It should be added that the work done in the latter half of the summer was performed on a new configuration of the ESLDC which operates

Figure 2. Hardware Configuration



by being interfaced with the 7094 through a PDP-7 satellite computer used as a display system buffer. This new system performed the same functions as the old system, but much more efficiently. Most important, the new system reduced picture flicker and more than doubled the size of the available display space.

The complete hardware system is diagrammed in Fig. 2.

Extensive use was made of the numerous provisions of the CTSS supervisor system (A-Core). In particular, use of the ESLDC involved extensive use of the portion of A-Core concerned with driving the display system.

The Biology Group of Project MAC has developed a library of useful procedures. We used these procedures whenever possible. Certain other B-Core routines concerned with operating the ESLDC system commands were extracted from the Biology Group library and modified with the aid of Thomas Warner. Modifications were necessary because of the particular interaction procedures developed.

All program development for this project was performed on the CTSS 7094. The great majority of the programming is done in MAD (Michigan Algorithm Decoder) for the convenience of quickly obtaining working versions of various procedures and algorithms. Some system programming for usage of the display system commands was done in FAP (Fortran Assembly Program).

3.3 Consequences of Hardware and Software

It is helpful to consider two of the consequences of the hardware

and software available: three-dimensional simulation and user interaction.

The objective of this project has been to develop a system that would allow a molecular biologist to investigate a three-dimensional electron density function representative of a protein molecule. It is necessary to create a satisfactory three-dimensional effect to enable a person to establish spatial relationships and to perform three-dimensional pattern recognition. User controlled real time simulated rotation of a line segment represented in three space as $(x_2-x_1, y_2-y_1, z_2-z_1)$ with respect to a projection on to a viewing plane (the face of the display screen) as (h_2-h_1, v_2-v_1) appears to be a dimension-decreasing transformation. However, when the display is "rotating" the transformation is actually dimension preserving:

$$(x_2-x_1, y_2-y_1, z_2-z_1) \longrightarrow (h_2-h_1, v_2-v_1, \text{time})$$

with the transformation being achieved by a dynamic rotation matrix controlled by the globe as previously described. The three-dimensional simulation occurs because the mind is able to use the time-encoded horizontal-vertical patterns to reconstruct the three-dimensionality of the picture.

Convenient provisions for the user to interact with the display should enable him in some way to keep track of the portions of the molecule that he has investigated, to use light pen identification to extract continuous regions of high electron density from the confusion of their surroundings, and to construct directly a model of the structures observed.

3.4 Development

Since it was decided to create displays through the mode of contouring, the first programming necessity was the development of a general, simple contouring program. The algorithm, to be described later, was provided by Professor Cyrus Levinthal. The output from the contouring program was displayed on the ESLDC through a modified version of the dipslay routine developed for the Biology Group Protein Structure Program Package. The necessary system programs were obtained from the Biology Group library routines.

The contouring program was successful in transforming density data from several sections of successive parallel planes of the density array into a stack of contour maps. The line contours of constant density were broken-line approximations to a curve. The ability to rotate the generated contour display in real time, combined with the ability to simultaneously display an ensemble of contours of several density levels enabled us to achieve a limited approximation of the effect of a Perspex model. However there were two severe restrictions.

The first limitation was a consequence of the display system. Due to the size of the A-Core space allotment for the display list (display space), our display originally could be constructed from no more than 450 straight line segments (visible and invisible). This was a very stringent limitation considering the complexity and size of the density distributions we wanted to look at. Although this display space limitation, even with the new ESLDC PDP-7 buffer configuration, has

proved to be our most persistent and frustrating difficulty, paradoxically, the development of our most powerful procedure resulted primarily from an effort to overcome this problem.

The second disadvantage was the planar constraints imposed upon the appearance of various density formations due to orientation with respect to the contouring planes. This effect was most noticeable when looking at a planar structure such as the heme group of myoglobin as discussed earlier. Similarly, rod-like distributions of high density appeared quite differently if they were parallel rather than perpendicular to the contouring planes.

At first the simplest solution to the latter difficulty seemed to be to write a program which could arbitrarily rotate the scalar density matrix upon which the contouring program operated by interpolating its values on to a matrix of different orientation. Then we could look at the same density distribution contoured in planes of an arbitrary orientation. However, this attempted solution did not alleviate the central problem of planar constraints being artificially imposed upon a three-dimensional distribution. In fact, rotating the matrix from the coordinate axes of the unit cell introduced even more problems. Also, the operation could not be performed in real time.

One significant benefit did result from this effort to eliminate planar constraints from the display. In an attempt to simplify the work of the matrix rotation and interpolation routine, it was decided to generalize the contour routine so that the planar contouring operation could be performed in any one of the three orthogonal sets of

parallel planes of the three-dimensional data array. This improvement was a simple modification of the contouring program causing the algorithm to operate in horizontal and vertical coordinates at a specified depth. A control variable caused the proper correspondence to be made between (horizontal, vertical, depth) and (x,y,z) to obtain the proper value of the density function at a desired point (h,v,d) from the density array for the contouring program.

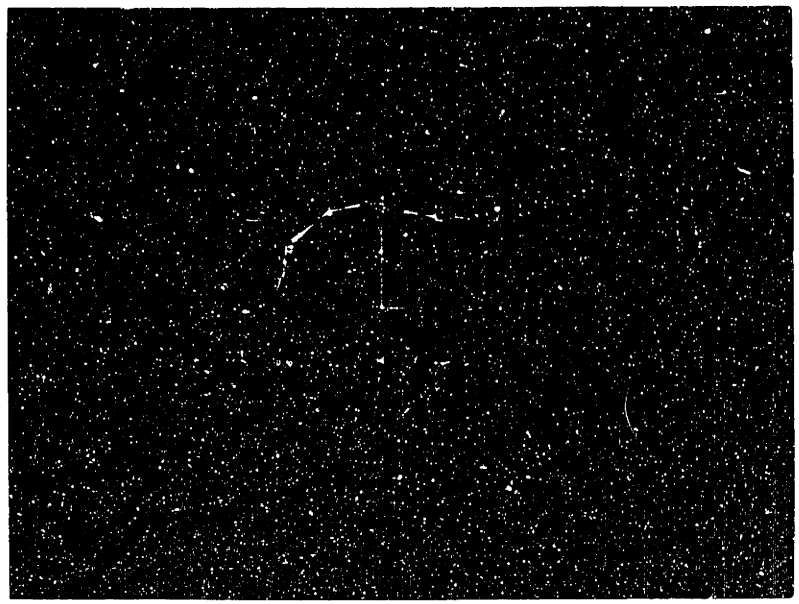
Out of curiosity we simultaneously displayed contours in x-y planes of constant z, y-z planes of constant x, and z-x planes of constant y. Only one contouring density level was used. The psychological effect of this display was dramatic. Display of contours in sequential parallel planes in effect reduces a three-dimensional scalar function to a series of two-dimensional scalar functions. While a two-dimensional scalar function can be simulated by an ensemble of contour lines of constant density, by extension, a three-dimensional scalar function is best simulated by an ensemble of surfaces of constant density rather than by a stack of lines of constant density. This display created an excellent impression of a surface of constant density, as demonstrated in the series of photographs in Fig. 3. Planar constraints were essentially removed; similar density distributions no longer appeared so different at different orientation in the matrix. The picture never degenerated when rotated, as contours of a single set of parallel planes would do when viewed edge-on. It was also observed that just two of the sets of orthogonal contours produced almost as profound an effect as all three sets, although the picture was

Figure 3. Example of Surface Contouring

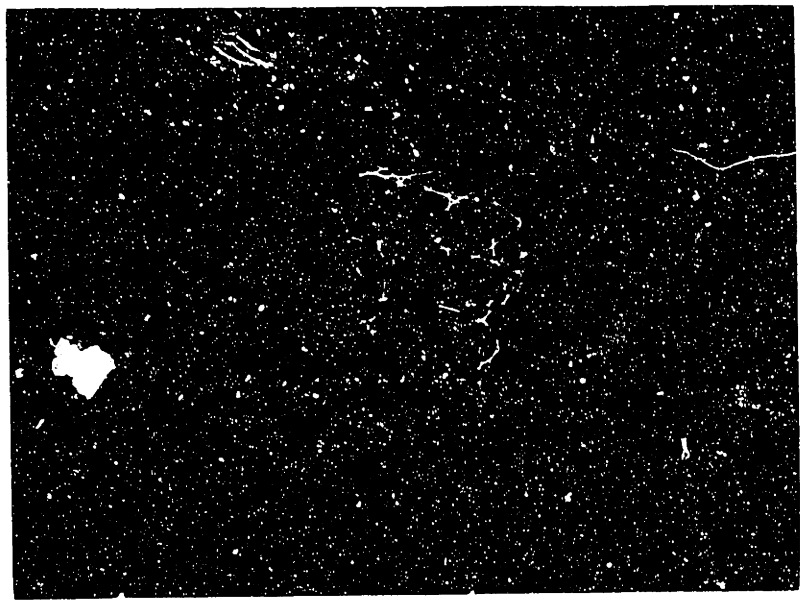
The following four pictures illustrate surface contouring. They are of the same cubical volume, eight Angstroms on a side, viewed from various angles of observation near the iron atom at the center of the heme, surface contoured at approximately .15 el/A^3 above the mean electron density.

a) x-y projection. The vertical lines are y-z contours of constant x viewed edge-on. The horizontal lines are z-x contours of constant y.

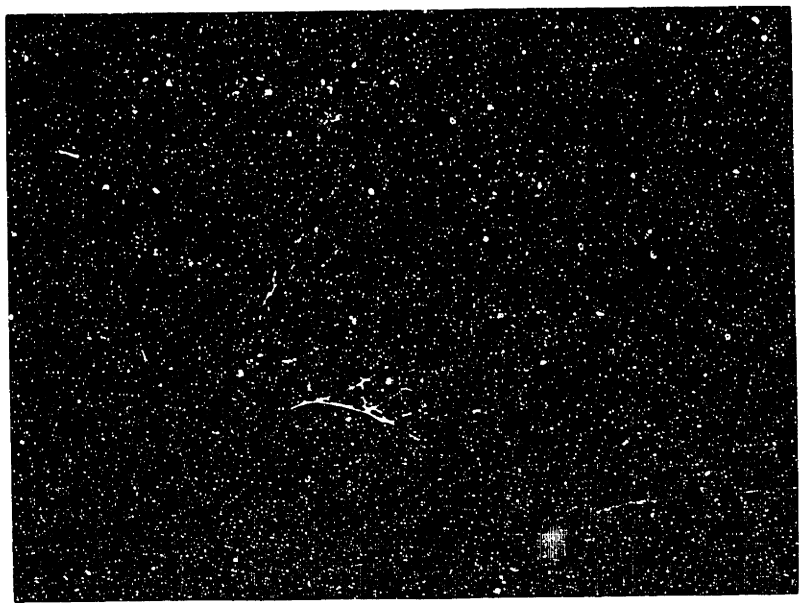
b) z-y projection.



c) Random orientation.



d) Random orientation.



subject to degeneration from two points of observation. However helpful this new display appeared, it complicated the already severe problem of the display space limitation.

It should be emphasized that it was at this point that our display system began to depart from being merely the electronic analog of the plastic Perspex model. We had generated a display which is practically impossible to build in plastic or any other media other than soldering wires together. However, we were still not using the ESLDC system to full advantage, for we were operating only in one direction, from our programs in B-Core of the CTSS 7094 to the display system, and, other than the real time rotation globe and the teletype, we were not taking advantage of the powerful provisions of the ESLDC for user interaction.

The first type of user interaction developed was suggested by Brody. "I think the user should be able to have the light pen at his disposal to construct a three-dimensional line segment model of the protein while he is looking at (the display) of the electron density."[9]

A simple routine was developed for drawing a series of straight line segments of arbitrary length and number of segments in three dimensions by using the light pen and the light pen tracking cross. It was intended that this line could be passed or threaded through continuous regions of high density when such regions were observed by the user. It was optimistically anticipated that such a broken line could be passed through the length of the peptide chain. The user could then call for display of an adjacent

cubical region and attempt to continue the line in sequential fashion from cube to cube.

However, we were forced by display space considerations to look at quite small portions of the unit cell density array--approximately a cubical region eight data units per side which corresponds to a volume of approximately eight Angstroms per side. This is certainly not a very large region considering the viewer hopes to follow a chain that is approximately 500 Angstroms in length and that winds into a globular molecule up to forty-five Angstroms in diameter. An overview of even secondary structural features is impossible.

Although the drawing routine was developed to a satisfactory level, the process of attempting to follow regions of high density from cube to cube in this manner proved extremely disappointing.

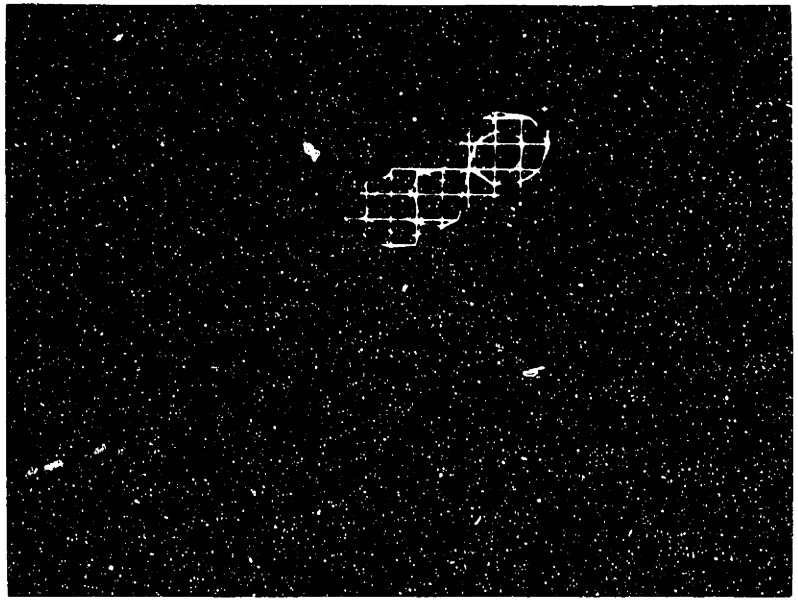
The drawing routine will be briefly discussed now, because a significant psychological effect is involved and had to be dealt with. To draw a straight line in three dimensions, one has to specify the (x,y,z) coordinates of the end points. However, the light pen and its tracking cross can provide the B-Core program in the 7094 with only the horizontal and vertical coordinates of the tracking cross position. This difficulty can be resolved by using the light pen and cross twice for each end point as indicated by the series of pictures in Fig. 4 .

First a button is depressed which causes an x-y projection to be displayed on the screen. The tracking cross is then moved with the light pen by the user to the desired location of the end point

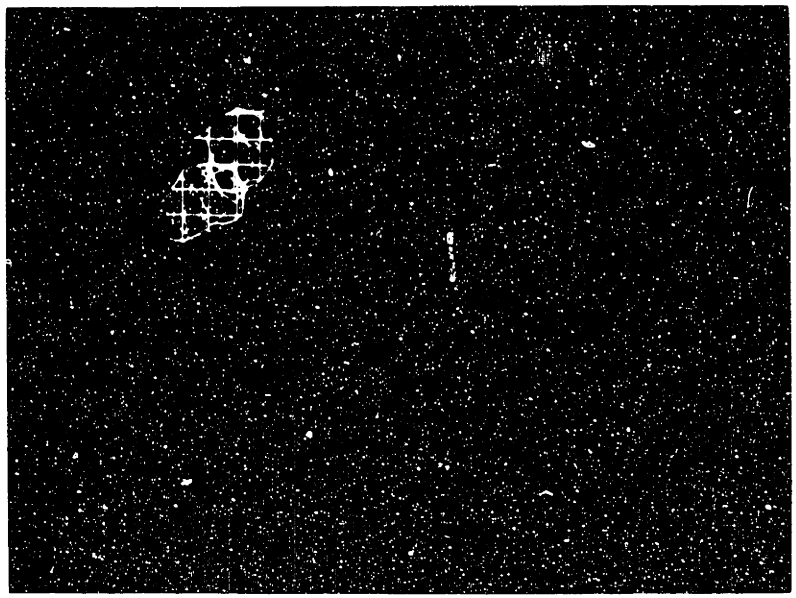
Figure 4. Line Drawing Procedure

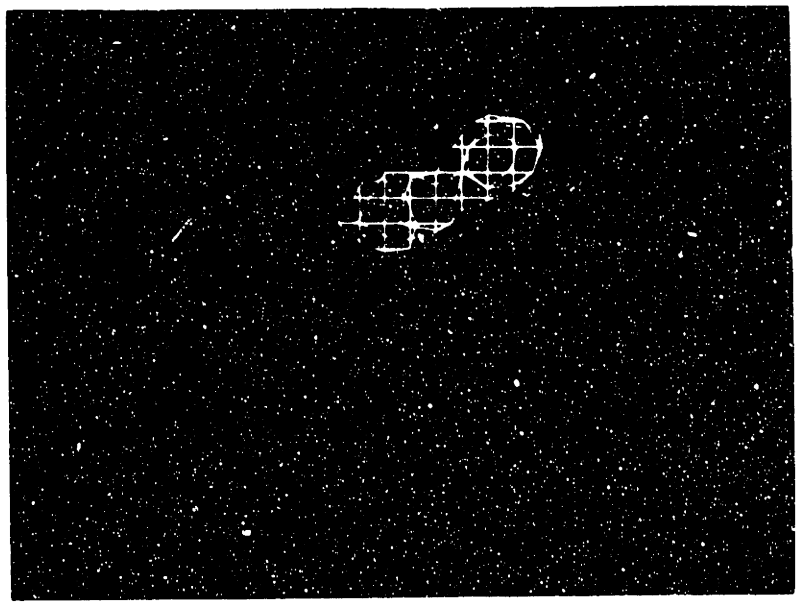
Line drawing with light pen and tracking cross, to model a tubular section of the F helix.

a) Setting first end point. x-y projection. Note tracking cross in upper right-hand corner.

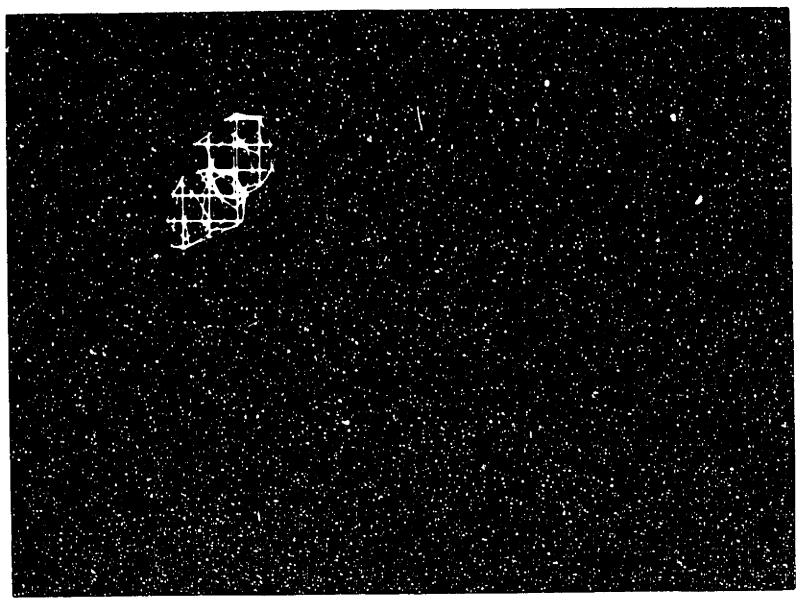


b) Setting first end-point. z-y projection.



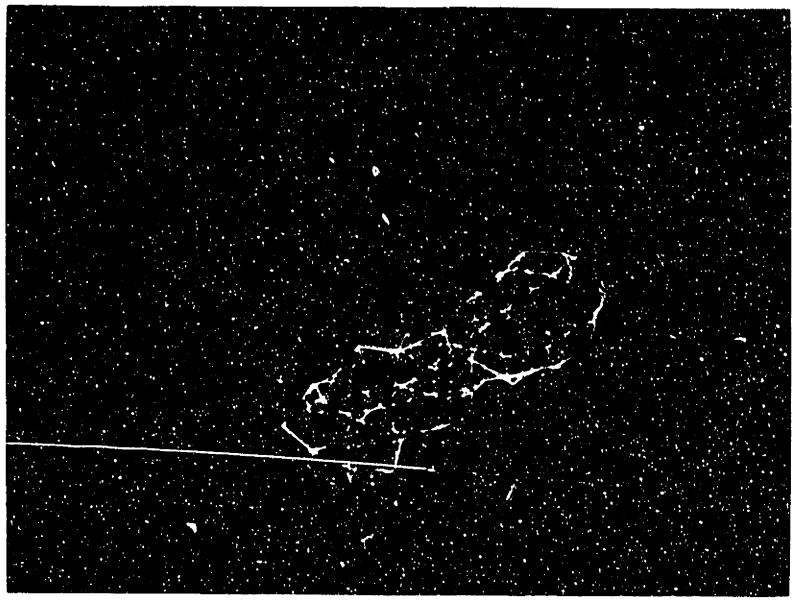


c) Setting second end point. x-y projection.

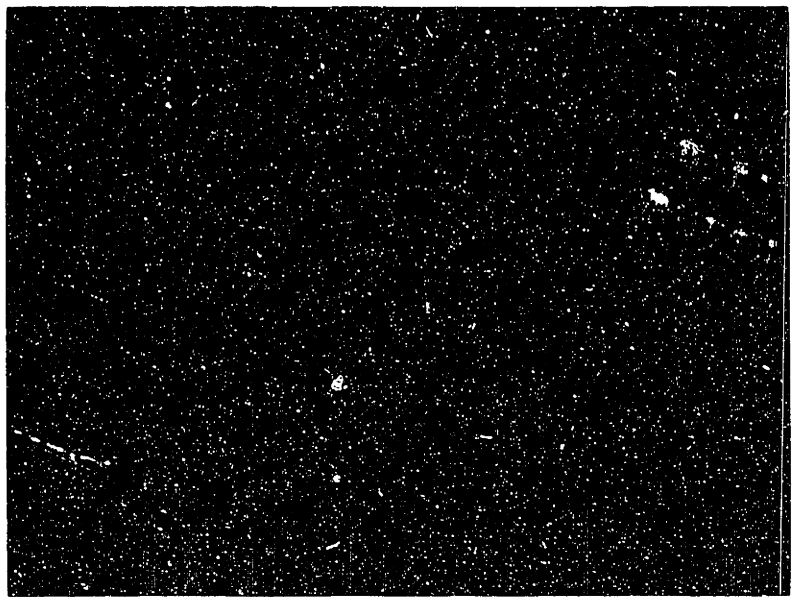


d) Setting second end point. z-y projection.

e) Blow-up of resulting line and surface contours.



f) Blow-up of line alone.



of the straight line segment on the display screen. A second button is then depressed to send the (h,v) coordinates of this point to the B-Core program. Then a third button is pushed which causes the picture to be displayed as a z-y projection. The tracking cross is moved to the new location of the same desired first end point on the display screen. The second button is depressed again to send this second pair of (h,v) coordinates to the B-Core program. The first h coordinate corresponds to the x position of the end point, and the second h coordinate corresponds to the z position. The two vertical coordinates should be very nearly the same, since both represent the y position. The y position is obtained by averaging the two vertical coordinates. In this manner the two sets of (h,v) coordinates are converted to one set of (x,y,z) coordinates, and thus the first end point locus has been established. The process is repeated again for the second end point, and for as many end points as necessary to determine a series of connected straight lines consisting of the specified number of segments.

The psychological effect mentioned above results from the following situation. In order to use the light pen and tracking cross to determine the locus of the end point in three dimensions, this end point must be closely associated with some object or portion of surface in the display in order for the user to be able to locate this same point in x-y and z-y projection. It was found to be necessary that the user be able to rotate the picture slightly from the x-y or z-y projection to reassociate what he sees

in projection with his prior visual familiarity with the picture being displayed. In other words, if the user becomes familiar with the spatial relationships in a given picture by rotating it, he may not be able to correlate a static x-y or z-y projection of the picture with his previous visual experience. However, a slight rotation from the x-y or z-y orientation succeeds in reestablishing the proper correlation between the projection and the user's familiarity with the picture.

At this point, although we were able to generate convincing surfaces of constant density, we were still up against several walls, a few of which have not even been mentioned yet. Our display was absolutely worthless unless we could show the user what he wanted to see. It was decided that a more efficient way of using the limited display space was necessary. The method, which has powerful implications that will be considered later, was to allow the user to select or identify with the light pen any one of several "pieces of surface" found in a particular cubical section. The object was to create a display consisting only of the complete surface of constant density of which a piece was identified, using the entire display space for this single continuous surface object. In this way the user would not be impeded by artificial boundaries imposed by contouring and displaying a small cubical section, but rather would see a "surface object" enclosing a particular contiguous volume of density above a specified level.

This type of display appeared to be helpful in several ways.

First it conserved the display space while showing the user a picture that was much more valuable to him. It isolated a particular high density pattern from the complexity of the surrounding regions. In particular, with regard to the protein problem, where one of the first objectives of the user is to follow the pathway of a long polymer, it appeared hopeful that this routine could, with the user's assistance, find and display lengthy tubular patterns of high density, isolate them, and thus perform a limited pre-processing for the user, enabling him to focus his attention on the three-dimensional shape of the delimiting surface he obtains through interaction by identification with the light pen. The user could then apply his molecular knowledge and pattern recognition abilities much more readily.

Of course the shape, continuity, and degree of connectedness of the surface the user observes is highly dependent upon the value of the surface contouring density "threshold" level specified and the resolution and sampling frequency of the electron density function. Hopefully, connectedness of two regions which appear isolated at one density level could be established by investigating the region between them at a lower density level. This approach will be returned to later.

In effect, the entire procedure is performing two additional transformations on the three-dimensional scalar information. Density information, rather than being presented as a continuous function in three dimensions, is being separated into displays of discrete surfaces of varying density levels, and into individual surface

displays of various regions of the molecule. Particularly when attempting to establish connectedness or continuity, it appears that with an aid such as line drawing, the mind is extremely capable of reassembling this temporal ensemble of surfaces into a working knowledge of patterns, continuity, and spatial relationships in the density function.

Although we had found and exploited several intriguing effects, we were still plagued by the twin problems of B-Core memory space and display space. It became obvious that a satisfactory working version of the above procedure would necessitate some form of data reduction and/or condensation.

To appreciate the magnitude of these difficulties, consider the following facts. 77,777 octal locations or 32,768 decimal locations of thirty-six bit words are available for the user's programs and array storage in B-Core of the CTSS 7094. The scalar density array of the Fourier synthesis of the myoglobin cell was evaluated on a sixty by thirty by thirty grid, for a unit cell whose dimensions were:

$a = 64.5$ A, $b = 30.86$ A, $c = 34.7$ A

i.e., the data were originally evaluated at approximately one Angstrom intervals. Fourier intensity and phases out to four Angstroms were supplied to the Fourier program. The value of the synthesized electron density function ranged from zero to 796 and was evaluated for 54,000 points within the unit cell.

The first approach attempted was to scale the density array from

zero to sixty-three instead of zero to 796. A test was performed by superimposing two pictures of a plane of data, one picture derived from contouring the full-valued plane at one level, and the other obtained by contouring the plane of scaled-down data at a correspondingly scaled-down contouring density level. The pictures superposed almost perfectly. One or two lines out of approximately 200 were discernibly different, and only by an amount corresponding to one or two increments on the face of the display screen. The conclusion was that a scaled-down version of the data would provide essentially the same picture. We were now in a position to pack the scalar array six six-bit words to a thirty-six bit word of memory. In other words, the entire unit cell could be contained in 9,000 words of B-Core storage. Considering, however, that the two halves of the unit cell of myoglobin are related by a two-fold screw axis ($P2_1$), and that most protein crystals possess some such symmetry relation, the number of words of core storage required for the complete density array information could be dropped to below 5,000 words for myoglobin and for similar sized proteins at four Angstrom resolution.

This, however, offered absolutely no improvement for the problem of display space limitation. Due to the observation that, although the broken line contours did not look like continuous curves, they were satisfactorily smooth, we began to consider the problem through the following reasoning. The number of lines required for display of a particular volume is proportional to the cube of the frequency of sampling of the electron density distribution of the unit cell. In

other words, suppose the data array sampled the unit cell electron density distribution with a fifteen by fifteen by thirty matrix rather than with a thirty by thirty by sixty matrix. Hopefully such a matrix would create a similar picture to the original matrix, but using $(\frac{1}{2})^3$ or one-eighth as many lines. This approach was attempted, and the results were extremely unsatisfactory. There appeared to be very little correlation between the picture of the array sampled every two Angstroms and the picture of the original array sampled every Angstrom at the same density level.

A satisfactory explanation was found, and that answer led us to make a modified, slightly less drastic attempt at condensation which proved to be relatively successful. There is a theorem concerning data sampling known as the Nyquist Sampling Theorem.[10] Its result states that in order to preserve all the information contained in a signal when sampling it, it is necessary to sample the signal at a rate equal to twice the frequency of the highest frequency component contained in the signal. The contours we display are essentially a straight line reconstruction of a three-dimensional array of sampled data. The highest frequency component in the trial density array we used was four Angstroms. When we condensed it to a fifteen by fifteen by thirty array, the signal was sampled every two Angstroms, which is exactly the minimum Nyquist rate for the four Angstrom information in the picture. Consequently, this four Angstrom information broke down. For the type of molecular structure with which we are dealing, this four

Angstrom information is necessary for satisfactory visualization of the structure. Furthermore, a straight line reconstruction of sampled data is hardly an optimum method of reconstruction.

Feeling reasonably confident of the validity of the above explanation, we attempted a second condensation to a twenty by twenty by forty matrix corresponding roughly to a one and a half Angstrom sampling interval. The advantage factor in display space limitations was $(2/3)^3 = 8/27$, that is, approximately 3.5 times as much volume could now be seen.

The qualitative results were as desired. Although the contours of the condensed array were substantially rougher in appearance, the contour surfaces at first seemed to display essentially the same continuities and shapes apparent in the original array. Most significant, the number of lines required to display the surfaces of constant density in a particular region were reduced by a factor of approximately 1/3 as predicted. This represented a significant gain, and relaxed considerably the display space limitation. We were now in a position to be able to see satisfactorily large sections of the type of regions and surfaces for which we were looking. However, it remained to be proven to what degree the continuity of high density regions was maintained in this condensed array. Eventually, however, even this condensation proved to be unsatisfactory.

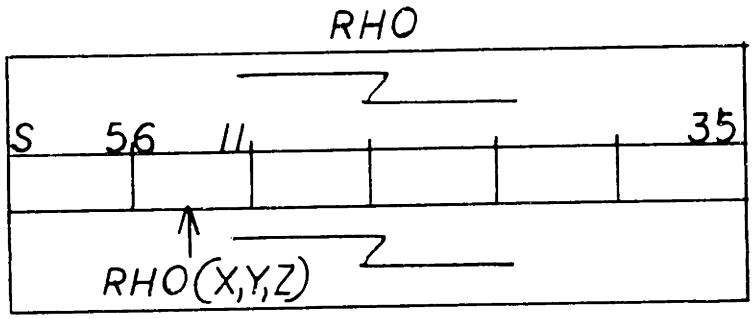
Two other instances of word packing to conserve B-Core space were found to be necessary. First, a Boolean array was needed to keep track of the locus of contours already determined in order to avoid

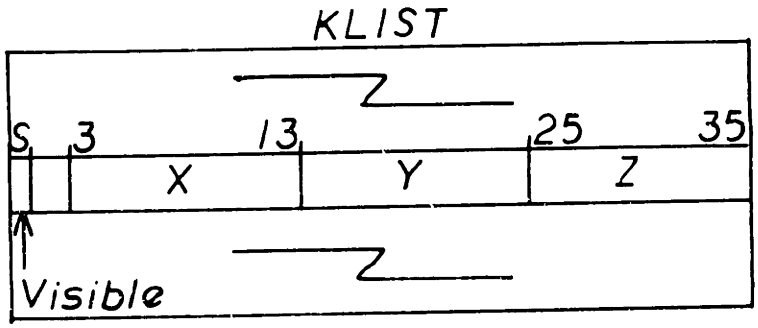
redundant contouring. The details of the information necessary to specify this situation, as well as the format of this Boolean array, will be described in detail later.

Second, whenever a point in (x,y,z) space was determined as one of the points along the broken line contour, its x, y, and z coordinates were packed into a single thirty-six bit word. These coordinates are integers allotted sixteen integer increments per data unit. In other words, the x coordinate at a data point at x = 2 in the data array is listed for display purposes as x = 32 = 2 * 16. The packing format is indicated in Fig. 5. The sign bit is reserved to indicate whether the line to the point from the previous point is to be visible or invisible. Bits one and two are left vacant. Each x, y, and z coordinate is allotted eleven bits. Therefore each coordinate may be represented by an integer from zero to 2,047. Considering the field of the display screen to be 1,024 bits in width and 1,024 bits in height, eleven bits per coordinate will be able to give a satisfactorily fine specification of the point location on the display screen, because the point will be specified more accurately than the digital display system can plot it.

The final version of the display package contains three subroutines which alleviate several problems not previously mentioned. The first problem concerns the fact that, given a unit cell of protein molecule, portions of the molecule may extend past the boundaries of the unit cell regardless of how the unit cell boundaries are specified with respect to translation along any one of the three crystallographic

Figure 5. Packing Format of Density and Contour Arrays





axes of the unit cell. Since we desired to see all portions of a single molecule in their proper spatial relationships, and definitely did not want the boundaries of the unit cell to impose any artificial constraints upon the generated display of the protein molecule, we defined a working volume of three by three by three unit cells, twenty-seven in all. We would operate primarily on a molecule in the central cell, with the understanding, knowledge, and intention that our investigations of this molecule could carry us into any of the surrounding twenty-six cells. It was obviously redundant and (with respect to B-Core space) impossible to define the data array over these twenty-seven unit cells by repetition of the array of a single cell. Instead, a simple routine was written to translate the coordinates of any point in the working volume back to the central unit cell for the purpose of obtaining the proper value of the density at that point and to perform the Boolean operation mentioned above for avoiding redundant contouring.

One further point should be added. Very often in protein crystallography, a unit cell will be comprised of two or more asymmetric units. A single asymmetric unit contains all the electron density information needed to construct the entire unit cell. The individual asymmetric units in a unit cell are related by a symmetry operation. For example, in myoglobin, the entire unit cell may be constructed from the asymmetric unit by performing a two-fold screw-axis rotation ($P2_1$). The symmetry operation for a particular protein crystal may be added to the translation function described above and thereby

reduce the size of the array needed to store the electron density information by at least a factor of two, because only the data of the asymmetric unit must be specified.

This symmetry operation, apart from certain dimension and initial value declarations, is the only non-general part of the procedure we have developed. In other words, the procedure we have developed can be readily applied to any protein crystal, with modifications in unit cell dimensions and symmetry operation.

A second subroutine performs the Boolean check mentioned above. Once a contour of a specific density level has been determined to pass between the two points of the data array, and its locus has been interpolated between those two points, it is essential that a second contour, co-planar with the first contour is not passed between those two same points. From the conditions of this problem it follows that no less than six bits are required for this violation check for every point in the data array, as will be described in detail later.

The third subroutine merely extracts the correct six-bit word containing the density value at a particular point in the data array and supplies it to the contouring program.

Before describing specific details of the display procedure, the impact of the user interaction provision should be emphasized. First, three-dimensionality can be effectively simulated by real time control of rotation. Second, points can be specified with the

light pen in the three-dimensional working volume for the purpose of tracing lines or indicating the center of the cubical region to be examined. Line drawing enables the user to keep track of where he has been and where he is going in the working volume as well as construct a model of structures he has observed. Third, the light pen may be used to identify a portion of the surface in a cubical region for the purpose of initiating a routine which extracts and displays the single, complete, contiguous surface connected to the point identified, enclosing a continuous region of high density (i.e., density above the specified contouring level).

IV. ALGORITHM DETAILS

4.1 Organization and Content of Procedure

The procedure was written primarily in MAD (Michigan Algorithm Decoder) and consists of a main control program (PEPTID), nine external functions, and two adapted FAP Biology Group library routines. Each external function contains several internal functions. First the organization of the program will be described, and then details of the several key routines will be elaborated.

It must be emphasized that the core of the entire procedure is the contouring algorithm contained in the external function FOLLOW. This algorithm, when operated in all three orthogonal sets of parallel planes of the three-dimensional density array, performs a transformation on the array. It makes it possible for the user to effectively visualize this three-dimensional scalar function by creating a temporal ensemble of surfaces of constant density. This contouring procedure may be called through one of two modes.

The entire procedure is organized about a READ DATA statement in PEPTID. This statement is followed by a series of conditionals which direct the program into one of several modes of operation specified by the user through the teletype. First, the user can request to see a display created previously and stored on pseudo-tape, or he can request a currently displayed picture to be saved. Second, he can request a particular cubical volume to be surface contoured and displayed by providing the boudaries of the volume, and specifying the density to be used for the contouring operation. Third, he can put

the program into "input-wait" status by entering CONTIG. In this situation the B-Core program in the CTSS 7094 continues operation upon receiving a light pen identification from the ESLDC through the "get attention" subroutine GAT. CONTIG then assembles a contiguous surface of constant density connected to the point identified by the light pen. Fourth, he can enter the routine STICKS to perform line drawing, observe or erase lines previously drawn, or use the SUROND option to observe the surface contours of a cubical region centered about a point identified by the light pen. All these display system interactions require the use of the light pen, tracking cross, and push buttons, and consequently involve the use of the routine GAT to convey this information from the ESLDC to the procedure under execution in B-Core.

4.2 FOLLOW

4.2.1 Core of the Contouring Algorithm

We developed an extremely general contouring algorithm. The following example should serve to illustrate the core of the contouring algorithm. Consider four co-planar points, each located at one of the vertices of a square as illustrated in Fig. 6. Each point corresponds to a point in a three dimensional (x,y,z) scalar data array. The value of the scalar function at each of these points may either be greater than, equal to, or less than a specified value. To avoid certain ambiguities, we will define the value of the scalar function at a point as being greater than the specified value if it

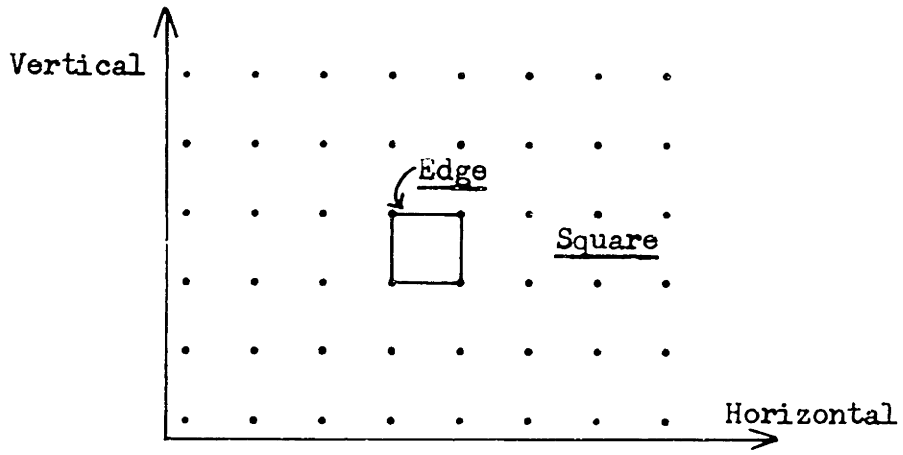


Figure 6.

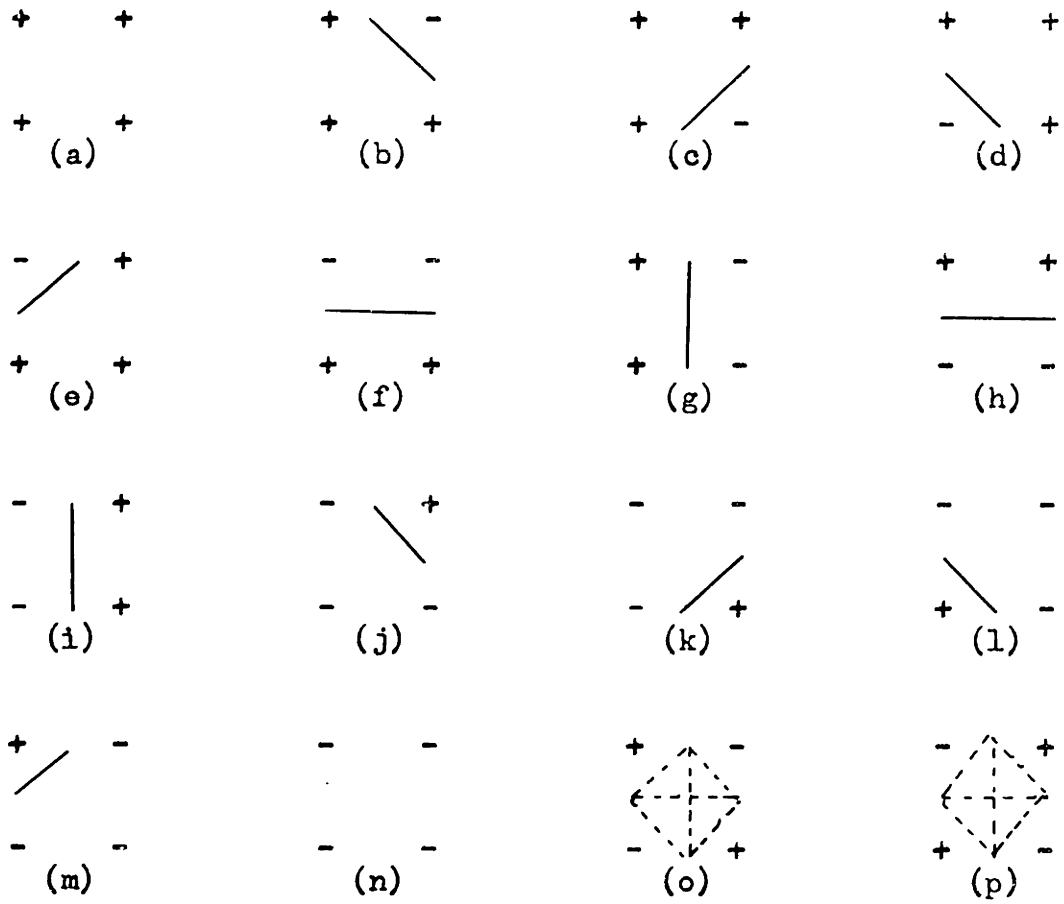


Figure 7.

is equal to the specified value. Then, any one of the sixteen possible combinations in Fig. 7 can result, where "+" indicates that the value of the scalar function is greater than the specified value and "-" less than the specified value.

Certain conditions may be determined from the complete set of possible combinations in Fig. 7. An edge is defined as the line segment between any two directly adjacent data points; a good edge is defined as an edge whose end points have opposite sign (i.e., the values of the function at the two points straddle the specified contouring level).

1) A contour will cross only good edges.

2) A contour will either enter or not enter a given square. It will not enter a square only if none of the edges are good edges, as in Fig. 7(a) and 7(n).

3) There must be an even number of good edges: zero, two, or four.

For two of the cases illustrated in Fig. 7, namely (o) and (p), there exists an essential ambiguity. These are the only two squares which contain two contours. Furthermore, these two squares represent identical situations in that the value of the scalar function at the four points alternates "+" and "-" around the vertices of the square with respect to the specified contouring value. Two facts should be noted for this situation. Two contours pass through the square, and three different but completely satisfactory sets of contour loci can be established as indicated in Fig. 8. Our algorithm resolves this ambiguity by arbitrarily allowing 8.a.1., 8.a.2., 8.b.1., 8.b.2., but not 8.a.3. or 8.b.3. Our

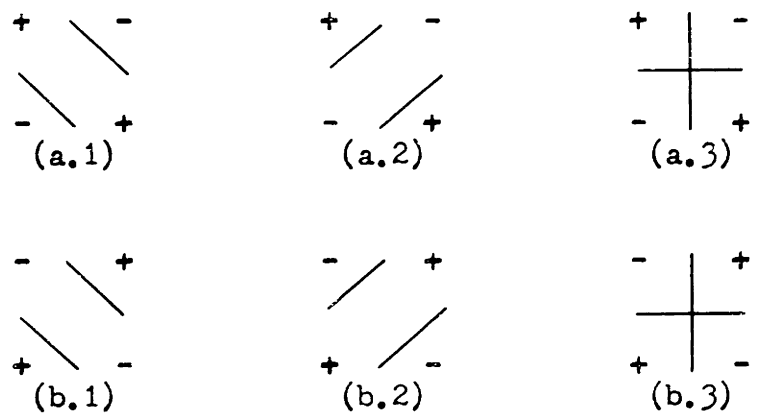


Figure 8.

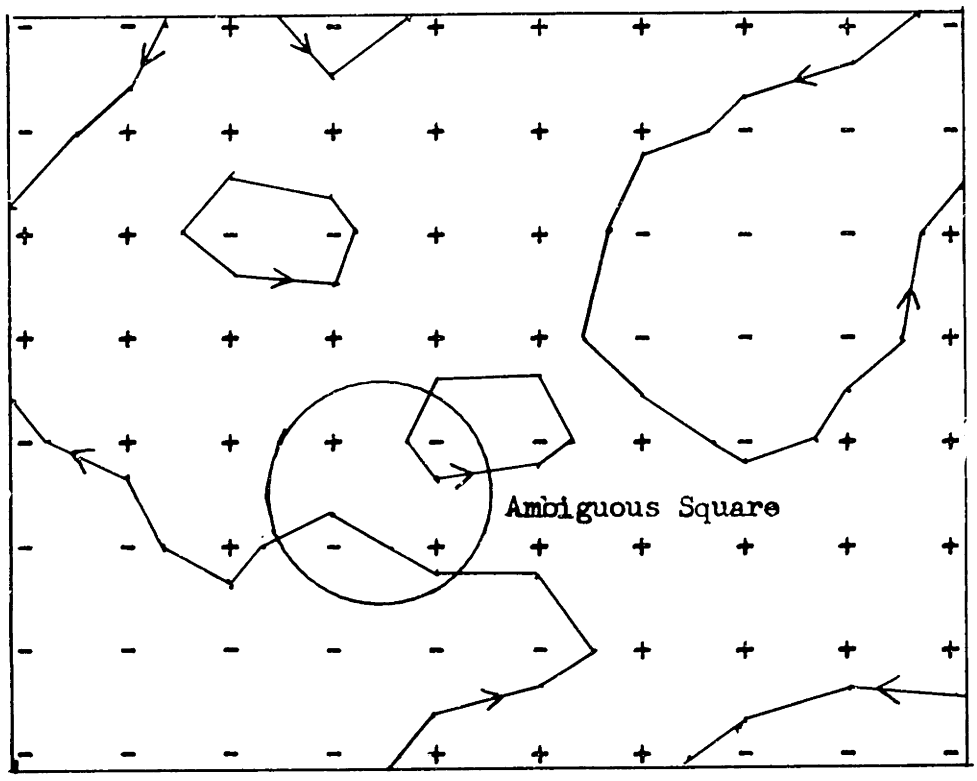


Figure 9.

contours will not cross in the interior of a square, but will instead create a saddle point. The reason for the above situtation is that once a contour has been determined to cross a good edge of a square, the remaining edges of that square are examined in a clockwise fashion from the entrance edge to determine where on which edge the contour exits the square. Obviously for the ambiguous case the next adjacent edge will be determined to be a good edge, and the contour will exit. Consequently, this first contour will pass in and out of the square through adjacent edges. Furthermore, whenever a contour crosses a good edge, this edge is registered as having already been used for contouring, and a second contour of the same density level will not be placed through it. The second segment of contour to pass through this ambiguous square can only pass through the remaining two edges which will also be adjacent. Hence the two contour segments will not cross in the interior of the square. See Fig. 9.

4.2.2. Following a Contour

FOLLOW performs the process of following a single contour from square to square in a two-dimensional plane of data. This process derives directly from the above discussion. In general, if a contour exits a square through a good edge, it must enter the adjacent square through the same edge. In other words, the exit edge from one square becomes the entrance edge to the next square as the locus of the contour is followed in the plane of data. Whenever the contour is found to cross a good edge, a linear interpolation is performed along the edge to determine the location of the end point of the

broken line segment, approximating the path of the contour through this square. FOLLOW packs the (x,y,z) coordinates of this end point into a single three-dimensional bit word of the KLIST according to the format previously described.

The straight line segment approximation to the contour is achieved by constructing straight line segments between consecutive interpolated end points. That is, a series of straight line segments are constructed, one in each square, from the interpolated point on the previous square's exit edge (the current square's entrance edge) to the interpolated point on the current square's exit edge. This broken line approximation to a curved contour can be quite satisfactory if the data points are close enough (with respect to the wavelength of the highest spatial frequency component of the data array), for then the broken line approaches the curve that it is approximating.

The routine FOLLOW, which performs the labor of contouring, must be provided with a starting point. This starting point may be specified through two vastly different modes of operation, PRISM and CONTIG.

4.2.3 Plane Orientation Specification

Since the contouring algorithm operates in horizontal-vertical coordinates in planes of specified depth and in any one of three orthogonal planar orientations, it is necessary to relate the (x,y,z) coordinates of the data array to the (horizontal, vertical, depth) coordinates of the contouring procedure and vice-versa. A control variable determines whether contouring is to be performed

in x-y planes of constant z, y-z planes of constant x, or z-x planes of constant y. This control variable makes the correct association between (h,v,d) and (x,y,z) by means of an indexing procedure. This association is necessary to extract the correct value of the scalar function from the data array and provide it to the searching and contouring routines as the value of the function at a position specified by (h,v,d). Also, it is necessary to express the positions of points determined along the locus of the contour by the planar contouring operation in terms of (x,y,z) coordinates of the original array.

4.3 PRISM

It should be apparent that if the user desires to see contours of a rectangular section of the working volume, a line by line, plane by plane scan can be performed in this rectangular volume to determine the crossing of contour loci. Whenever the scan detects a good edge between two adjacent data points and this edge has not been previously contoured in the plane orientation being scanned, FOLLOW is called, and the two adjacent points are provided as a starting position for the contouring operation.

In this situation FOLLOW terminates when one of two conditions is satisfied. The contour will either cross the boundary of the specified region, in which case it will terminate at the boundary, or it will be followed back to its starting point, where it will terminate.

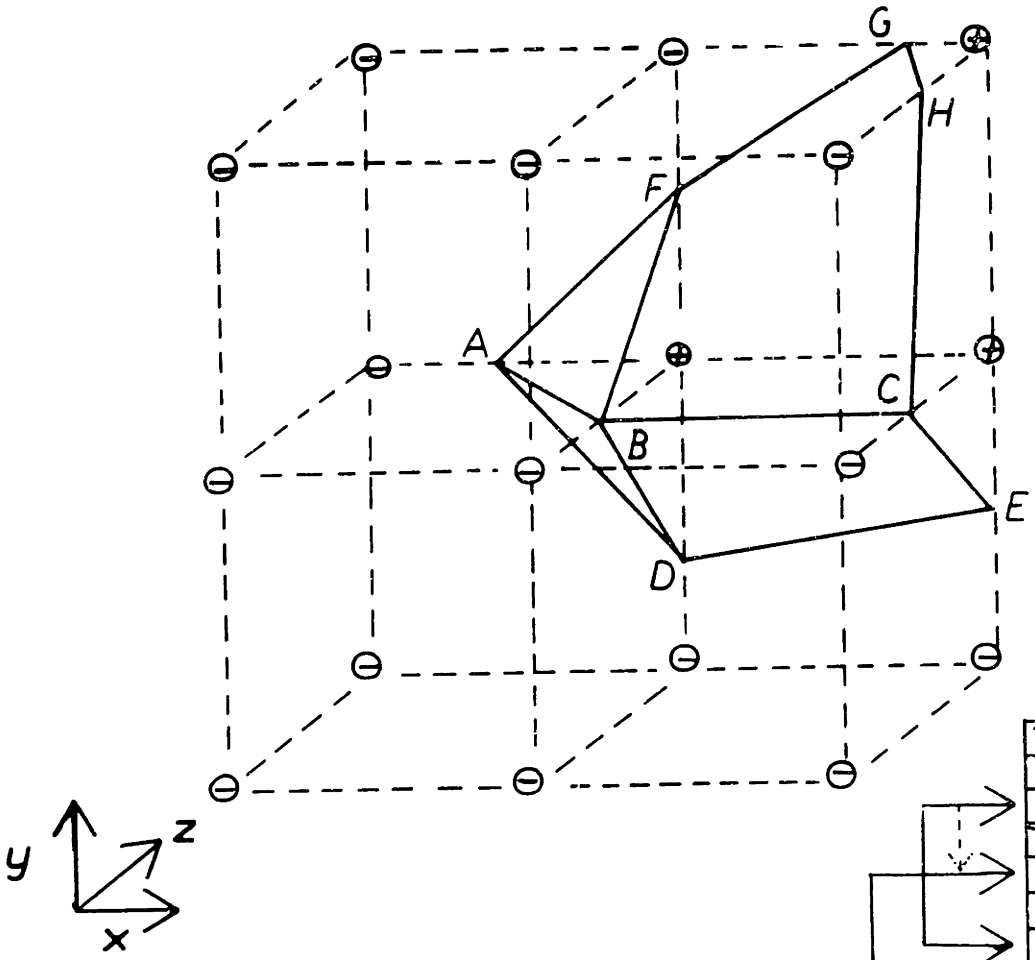
PRISM causes the planar searching and contouring operations

to be performed three times: once in each of the three sets of parallel planes of data perpendicular to each of the coordinate axes of the data array. It is this ability to line contour the three-dimensional array in three sets of orthogonal planes and display the contours thus found simultaneously that enables the routine to simulate display of surfaces of constant density. When all three sets of contours have been determined, control is returned to the main program PEPTID and the resulting KLIST array prepared by the external function DAWIRE for visual display on the ESLDC.

4.4 CONTIG

As mentioned previously, we developed an extremely helpful routine, CONTIG, for extracting and displaying single contiguous surfaces of constant density. The algorithm for constructing these contiguous surfaces is based upon the following idea. Each edge between any two directly adjacent points in the data array lies along the intersection of two orthogonal planes, each of which contains a plane of the data array and is perpendicular to one of the three axes of the array as illustrated in Fig. 10. If the scalar function at the two adjacent points straddles the value of the contouring density level, a linear interpolation can be performed to approximate the locus of a planar line contour passing between these two points. However, when line contouring in all three orthogonal sets of parallel planes, this interpolated point is common to <u>two</u> line contours, each one situated in one of the two perpendicular planes containing this edge as indicated in Fig. 10.

Demonstration of CONTIG Algorithm



KLIST

Vis	X Y Z
0	B
1	C
-	-
1	A
1	B
0	C
1	H
-	-
1	E
1	C
-	-
0	A
1	F
1	G
-	-
1	E
1	D
1	A
0	B
1	F
-	-
1	D
-	-
0	H
1	G
-	-
1	H
-	-

1) Light pen identifies AB.

2) B returned as identified end point.

3) BC . . . AB followed in z-x plane of constant y and inserted in KLIST.

4) Pointer moves down KLIST one word and obtains C.

5) CH . . . EC followed in y-z plane of constant x and inserted in KLIST.

6) Pointer moves down KLIST one word, etc.

Figure 10.

If a single contour in a single plane is identified with the light pen as belonging to a surface that the user desires to see, the construction of the entire surface can be started by using each of the vertices along this broken line contour to initiate the following of the other remaining contour in a plane perpendicular to the original contour that must pass through that point. Each new contour found and added to the KLIST also adds new points to be used for possibly initiating the following of more orthogonal contours. Of course, a point that has already been contoured in two perpendicular planes is not used to initiate a redundant contour. The above process continues until all of the points of the contours found have been checked for possible new contours intersecting that point. Termination will occur either when the complete contiguous surface of constant density has been found or until the display space limitation is exceeded.

There is one ambiguous situation in the above algorithm. If a contour passes directly through a data point because the value of the scalar function at that point is precisely equal to the contouring density level, it is difficult to define "between" which two adjacent points this contour passes. Furthermore, there will be three contours, each located in one of three orthogonal planes which pass through this point. We solved this ambiguity by ignoring the situation, because it is almost assured that this point and these three contours will be located anyway by the above algorithm. The contours comprising the surface all intersect at so many points

that the exclusion of this exceptional case will not affect, except in unusual circumstances (where two surfaces contact at only one point), the complete construction of the surface.

The light pen identification is supplied through the FAP routine GAT. It extracts the index of the line segment identified by the light pen in the display list from the attention buffer in the ESLDC software system, and provides this index to CONTIG. CONTIG then relates this index to the index in the KLIST of the coordinates which specify the second end point of the line segment identified. This index is supplied to an internal function called DISECT, which determines from the (x,y,z) coordinates of the end point the two adjacent data points between which it lies. It then selects one of two possible contouring plane orientations (see Fig. 9) and supplies the necessary starting information to FOLLOW.

This initial contour is followed until its starting point is arrived at again. Once this first contour has been placed in the KLIST, the routine becomes self-generating. A pointer starts from the first word in the KLIST, and uses the (x, y, z) coordinates of each point to follow the second of the two contours passing through this point in the plane perpendicular to the existing contour. With the completion of each contour added to the KLIST, the pointer moves down the KLIST to the next word, that is, the pointer moves along the contour to the next end point of a straight line segment or to the starting point of the next contour indicated.

It DISECTs each point and, if that contour has not been pre-

viously followed, calls FOLLOW to follow the second contour passing through that point. If this contour has been determined already, the call to FOLLOW is skipped, and the pointer moves one more word down the KLIST.

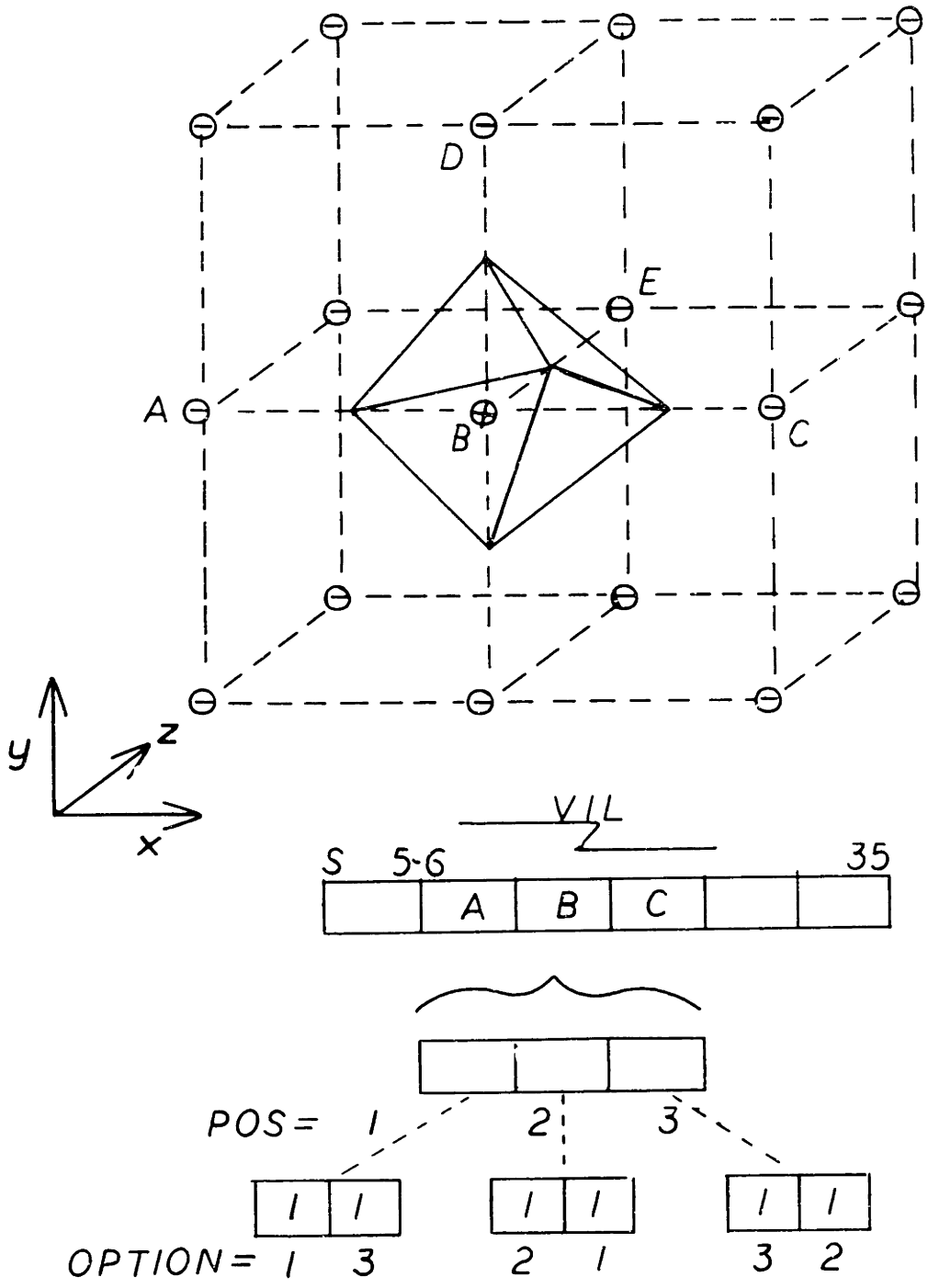
When termination occurs, the resulting picture stored in KLIST is prepared for display by DAWIRE. The picture is sent from the CTSS 7094 down the Direct Data Channel to the ESLDC. Control is returned to the main program PEPTID.

4.5 VIOL

Whenever an edge is examined to determine if it intersects the locus of a new contour or of a contour being followed in a plane of the data array, a check must also be performed to determine whether or not a contour of specified contouring density level has already been determined through this edge. The external function VIOL performs this check. The information required and the related format of the packed Boolean array is derived from Fig. 11. The information supplied as arguments to VIOL are the (x,y,z) coordinates of the two end points of the edge, a control variable indicating the orientation of the contouring plane (i.e., x-y planes of constant z, y-z planes of constant x, or z-x planes of constant y), and a Boolean variable which indicates whether the call is for a check, or to register a new good edge.

Three edges are defined as belonging to each point in the data array, as indicated. Each edge extends from its associated data point either in the direction of positive x, y, or z. Each edge

Description of VIOL Format



	POS
Edge BC	1
Edge BD	2
Edge BE	3

	OPTION
X-Y Planes of Constant Z	1
Y-Z Planes of Constant X	2
Z-X Planes of Constant Y	3

Figure 11.

may contain a point belonging to two contours, each of which is located in perpendicular planes. When a new contour has been determined to pass through a point on the edge between two points (x_a,y_a,z_a) and (x_b,y_b,z_b), the (x,y,z) of the point in the data array with which this edge is associated is determined by

$$(x,y,z) = (\min(x_a,x_b),\ \min(y_a,y_b),\ \min(z_a,z_b))$$

This (x,y,z) location of the point in the data array to which the edge is associated is used to index the correct packed six-bit word containing the Boolean information as indicated in Fig. 11. If the second point is located in the direction of positive x, y, or z, a control variable POS is set equal to one, two, or three respectively. POS locates the proper two-bit segment of the six-bit word since each edge may contain a point belonging to two contours, each of which are located in perpendicular planes. Each bit of the two-bit segment is reserved for one of these two possible contours, as shown in Fig. 11.

When a contour is determined to cross a good edge, the correct bit, which is identified as described above, is set to one. Whenever these two points are encountered again in the same planar orientation and for the same value of contouring density level, a violation is returned to the particular calling program to prevent redundant contouring. The packed violation array VIL is cleared before a new contour picture is created.

4.6 TRANS

TRANS is a trivial subroutine which eliminates the problem caused

by having portions of the molecule extend beyond the boundaries of the unit cell, as previously mentioned. It also conserves the amount of core storage required to specify the density array for the unit cell by employing a symmetry operation to operate upon the data of the asymmetric unit. It utilizes two principles: first, that a protein crystal can be treated as one period of a three-dimensional periodic function, and, second, that this three-dimensional periodic function has internal symmetry relationships. The entire display procedure operates over a working volume of twenty-seven unit cells. However, electron density data and the VIL array are only specified for the asymmetric unit of the central unit cell.

The function of TRANS is to transform the coordinates of a point (x,y,z) in any one of these twenty-seven unit cells into the central unit cell and then into the asymmetric unit of the central unit cell for which the data is provided. It operates by merely shifting the x coordinate of the desired point by the dimension of the unit cell in the x direction until the translated x coordinate is within the central unit cell, and then performs the same operation on y and z. Then, if the point in the central unit cell is located in an asymmetric unit for which the data has not been provided, the correct symmetry operation is performed to relate this point to a point in the asymmetric unit for which the data has been provided. The routine then returns these shifted coordinates to the calling program for the purpose of extracting a value from the density array, or for registering or checking a violation in VIOL.

4.7 DENS

DENS merely locates the correct six-bit word in the packed electron density array to supply the value of the density function to the calling routine.

4.8 STICKS

STICKS is another extremely useful function for user interaction. Its primary purpose is to enable the user to use the light pen with tracking cross and buttons to draw stick models in three dimensions to keep track of portions of the function he has explored.

The user draws a three-dimensional line by specifying with the light pen and the tracking cross the end points of the line in three dimensions. However, it is necessary to perform this task in a slightly round-about fashion because the surface of the display screen is, of course, only two-dimensional.

We modified one of the Biology Group system routines, GAT, to update the real time rotation matrix of the ESLDC to create either an x-y or a z-y projection of the display upon receiving an "attention" from the button box. To specify a point in the picture in three dimensions, it is necessary to identify the point twice with the tracking cross as described previously, once in an x-y projection and once in a z-y projection. The ESLDC sends the horizontal and vertical coordinates of the tracking cross on the display screen to B-Core of the CTSS 7094 through the routine GAT upon sensing an "attention" from button number five.

STICKS takes these two sets of horizontal and vertical coordinates and from them determines the (x,y,z) of the points specified. Other end points may be specified in the same way, and a line between the two end points can be displayed on the ESLDC. A broken line consisting of several straight line segments may be drawn merely by specifying the number of segments to be constructed through the variable LINES, and by specifying (LINES + 1) end points with the buttons, pen, and tracking cross. The coordinates are inserted into a packed array SLIST which has the same format as the KLIST described earlier.

Lines drawn in this manner may be stored external to the program by writing a disk pseudo-tape of the SLIST. All the lines stored in this fashion may be seen by requesting the pseudo-tape to be read into the BLIST array and by requesting the BLIST array to be displayed. The packing format of BLIST is also identical to the format of KLIST. A line previously constructed may be erased when desired by identifying it with the light pen.

4.9 SUROND Option in STICKS

The ability to specify points in the three-dimensional working space as described above has an additional advantage apart from drawing line segments. It provides a very convenient means for requesting a particular cubical volume of the working space to be contoured and displayed. A Boolean control variable SUROND is turned on (set to 1B), and the light pen, tracking cross, and buttons are used to specify the center of the cube desired through the position

identification scheme described above. Using the specified point as center, symmetrical boundaries in (x,y,z) coordinates are determined to define the cubical volume to be contoured and displayed. The extent of the cubical volume specified encompasses a region of seven to nine data points on a side, corresponding approximately to seven to nine Angstroms on a side.

Once the boundaries have been established, the standard procedure for contouring and displaying this cubical region is begun by calling PRISM. This is another extremely helpful application of the user interaction provisions of the ESLDC. It is particularly useful for establishing continuity between two adjacent regions of high density and for selecting pieces of surface for initiating CONTIG to follow regions of high density.

4.10 Saving Display Fields

One further convenience is added to the control program PEPTID. It was desired to have provisions for saving and replaying a display picture that was deemed interesting or significant. This was accomplished by writing an internal function SAV in PEPTID, whose purpose was to write a binary disk pseudo-tape image of the portion of the KLIST currently being displayed upon request. Information concerning the size of the picture and the density level of the surface contours in the pictrue is also included in the first two words of the tape image.

Another internal function, RPLAY, was written to locate a speci-

fied picture image or field on the tape created by SAV, to read it into the KLIST, and to display it.

To allow more efficient switching from picture to picture, we made the KLIST double sized and divided it into two equal sections, or slots. For example, we can maintain a newly created picture undisturbed in one slot while using the other slot to look at pictures that had been previously created, taped, and then replayed.

4.11 DAWIRE

DAWIRE is a modified and simplified version of the Biology Group ESLDC display routine. It converts the packed KLIST array to a display list (DLIST) of ESLDC commands which simulate a three-dimensional display constructed from visible and invisible directed line segments.

This DLIST is sent down the Direct Data Channel to the ESLDC PDP-7 satellite computer which directly drives the real time functions (such as rotation) of the display system.

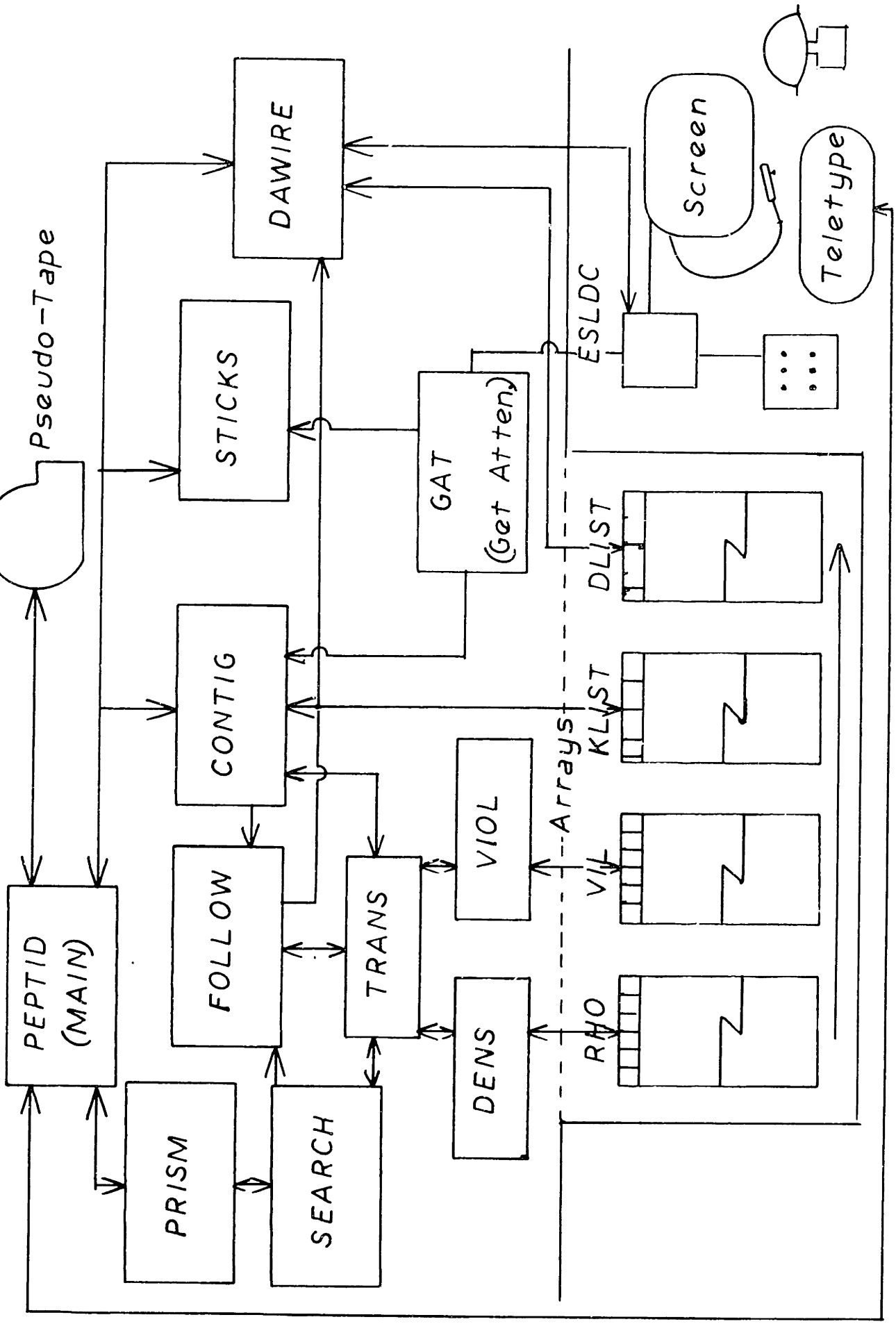


Figure 12. Procedure Block Diagram

V. OPERATIONAL PROCEDURE FOR USER

5.1 Necessity for an Operational Procedure

The computer procedures described in the previous sections were developed through a series of observations and improvements aimed at enabling a person to use a computer driven display system to work with the electron density function of an unknown protein structure. As limitations and difficulties were encountered and often overcome, and possibilities for user interaction exploited, we began to arrive at a more explicit understanding of what was needed in a satisfactory system and of how such a system might be achieved.

However, behind the development of the computer procedures lurked a larger problem. Once a new computer procedure was decided upon, the problem became one of relatively simple logic and programming, but the problem of developing an operational procedure for using this bundle of interrelated computer routines in combination with the knowledge and unmatchable pattern recognition abilities of a human to attack an unknown protein structure is a many-faceted one.

We attempted to develop, try, and adhere to such an operational procedure. The logic and usage of the procedure will now be discussed. The next section of this thesis will elaborate upon the difficulties encountered in attempting to implement the procedure to determine the structure of a known protein, myoglobin. It should be added that we developed our procedure with certain aspects of the structure of myoglobin in mind, and that various features of this procedure will have to be generalized slightly for consideration of other proteins.

5.2 Objective

Our objective was to develop a user's procedure which would enable him to achieve a knowledge of and familiarity with the path of the polypeptide sufficient to construct a model of the pathway. "A molecular biologist's understanding of a molecular structure is usually reflected in his ability to construct a three-dimensional model of it."[11] Much of the procedure was based upon the partial validity of a central assumption.

5.3 Tentative Assumption

Given that data of "sufficient" resolution are available, ignoring disulfide and other interchain bridges as well as non-amino acid moieties, it may be true that the lowest electron density along the covalently bonded backbone of the chain will be greater than the highest electron density in any region between two chains in van der Waals contact (i.e., not covalently bonded). Then there must exist a range of electron density levels such that a contour surface of a density level in this range will encompass the entire backbone. This contour surface will be a long tubular surface and will contain no bridges between adjacent sections of the backbone.

If this assumption is true, it should be possible (ignoring display space limitations) to visualize the entire polypeptide backbone encompassed by a single contiguous surface of constant density through a single judicious application of the routine CONTIG. This is an enticing possibility. The amount of computer time necessary to generate such a surface to obtain a useful visuali-

zation of the protein structure would be negligible compared to the amount of time required to perform the Fourier synthesis.

The logic behind the development of the above assumption is a consideration of a suitable algorithm to determine the pathway of the polypeptide backbone and was based upon experimental observations of protein crystals whose structures have been solved. The process of encompassing the backbone in a surface of constant density appeared to be a satisfactory approach in light of comments such as the following:

> In this synthesis (six Angstroms) the polypeptide chain was visible as a rod of high electron density--folded in a complex pattern--in addition the single haem group with its iron atom, which is much more dense than any other atom in the molecule, could be identified as a disk of high electron density. The shape of the molecule could be determined with confidence.[12]

Before discussing some of the limitations affecting the above assumption, two points should be stressed. First, the computer routines we have developed are capable of simulating, with user interaction, a display of a single, contiguous surface in three dimensions, encompassing a continuous region of high density (i.e., density above the specified contouring density level). In other words, except for current display space limitations, the necessary tools exist for investigating the validity of the above assumption.

Second, it is not necessary that the assumption be entirely true in order for our computer routines and operational procedures to work satisfactorily. This is an indirect consequence of display space limitations, for at present it is often more helpful to see a <u>completed</u> surface encompassing a smaller volume of higher density, than to see only <u>some</u> of the contours of a larger surface

whose construction has been terminated prematurely because of such limitations. The contouring density level should then be set at a value high enough to cause the polypeptide backbone to be visualized as a series of tubular sections arranged end to end. Continuity between adjacent sections could be established by observing the cubical region between them surface contoured at a lower density level.

5.4 Limitations on the Assumption

In the light of previous experimental evidence, there are also several limitations on the previously stated assumption. First, the effective resolution of the Fourier synthesis must be sufficient to resolve gross structural detail: for example, two adjacent but non-bonded sections of polypeptide chain. If the effective resolution is not sufficient there will appear to be false bridges between adjacent lengths of polypeptides. In this situation, an attempt to use CONTIG to create a single surface of constant density may result in an erroneous indication of the pathway. Consider the following comment from one of the early papers on the myoglobin structure:

> If we attempt to trace a single continuous chain throughout the model, we soon run into difficulties and ambiguities, because we must follow it around corners, and it is precisely at the corners that the chain must lose the tightly packed configuration which alone makes it visible at this resolution (six Angstroms). Also, there are several apparent bridges between neighboring atoms.[13]

Second, there is the possibility of low backbone density resulting from one of two reasons. Two or more consecutive small amino acid residues in the chain, such as glycine or alanine, will result in a

local drop in electron density. Furthermore,

> one factor which complicates the interpretation is that similar groups (for example, peptide bonds) in different parts of the structure have substantially different densities. We attribute these differences to variations in the amplitude of thermal vibrations in different parts of the molecule.[14]

Regions which are stabilized by interchain bonding will possess higher densities.

Third, there is the primary effect of interchain bonding, not weak hydrogen bonding but covalent bonding as in the case of disulfide bridges, or other covalent bonds. It is necessary to differentiate between the interchain bonds and the correct path of the polypeptide chain. Hopefully the human factor will make the correct choice. It will not be a blind choice, for information about the sequence as well as a visual appraisal of the three-dimensional geometric shapes and spatial relationships of neighboring and connected surfaces should remove some of the guess work. However, these bridges are not merely troublesome, for they are particularly significant in evaluating a protein structure.

There is also the problem of differences in secondary structure. We have already touched on this point. From previous evidence, it is known that the stabilized alpha helical configuration causes a stronger peak in the Fourier synthesized electron density function than do non-helical regions. However, many proteins are comprised primarily of non-helical structure, and even in a protein that does possess straight sections of helical structure, the backbone must assume a more random structure when turning corners.

The last difficulty concerns non-amino acid moieties such as the heme group in myoglobin. These structures are particularly important as they are often directly related to the function of the protein. They are structurally stabilized through some type of bonding or association with some of the side groups of the polypeptide backbone. If such stabilization occurs at several points, the problem of using CONTIG to follow the pathway of the chain in the vicinity of this non-amino acid structure may prove unsatisfactory. On the other hand, the very existence of such a structure in the protein may simplify the entire problem of structure determination by providing an easily recognizable starting point for following the polypeptide pathway in one or more directions. This does not imply that either the N-terminal end or the carboxy-terminal end of the polypeptide chain are located near this structure.

5.5 Procedure

Assuming that we have an identifiable starting point, such as the iron atom in the heme group of myoglobin (by far the densest point in the electron density map), we decided upon the following procedure:

1) Position the identifiable starting point at the center of the display screen.

2) Investigate several cubical regions in the immediate vicinity of the starting point at various density levels.

3) Apply CONTIG through light pen identification to various surface sections in these cubical regions.

4) When CONTIG creates a display of a surface enclosing a sausage shaped volume, attached in some way to the starting point and extending outward from it, save the picture

and call STICKS to trace the pathway of the tubular region.

5) Repeat (3) and (4) for several lower density levels on the same volume object (be wary of branches) in an attempt to lengthen the tubular region. If the surface object has exhausted the display space, inititate CONTIG again by identifying a contour near the prematurely terminated end of the contiguous surface.

6) Display all lines drawn with STICKS simultaneously with any new surface created to establish a three-dimensional frame of reference.

7) Use SUROND option in STICKS to specify and observe cubical regions at the terminations of newly found tubular shaped surfaces. Observe the same cubical region for several lower density levels in an attempt to establish continuity between a new surface piece in the cubical region and the established tubular shaped region.

8) If such continuity can be established, elevate the density level slightly and use the light pen to initiate CONTIG. Go back and repeat the procedure from (4). Otherwise, apply CONTIG to all the surface pieces in the box (one at a time). Save tubular shaped displays and use STICKS to record these tubular regions. Beware of branch points. Leave continuity indeterminate unless well-established. Repeat the process from (7), applying judgment to deal with the most likely looking surfaces first.

9) If continuity cannot be established between various tubular sections as the user works away from the identifiable initial region, then return to this initial region and explore thoroughly the area surrounding it. As more tubular regions are found and marked by using STICKS, continuity should become easier to establish.

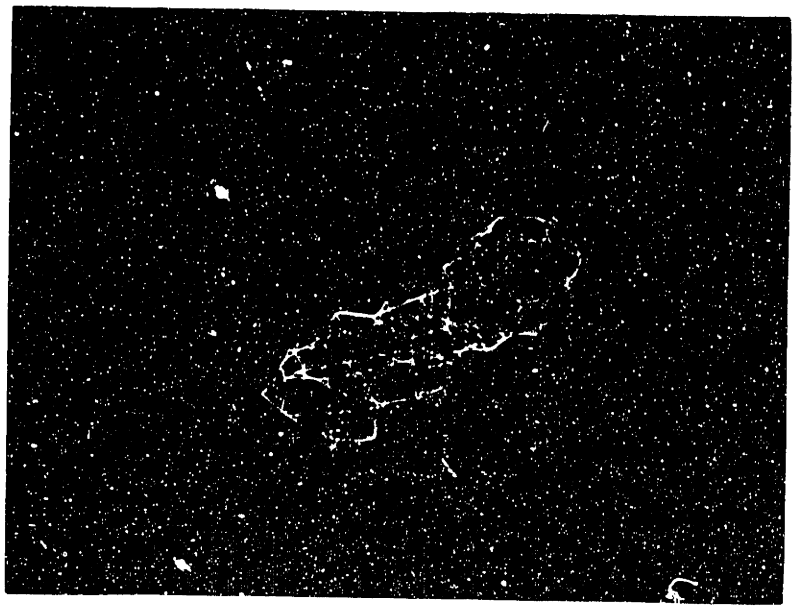
This procedure is illustrated in the sequence of pictures of Fig. 13. It should be stressed that these static, two-dimensional pictures of such abstract forms cannot convey the three-dimensional perception created on the ESLDC. Nevertheless, the following photographs have been oriented to demonstrate the most significant features of these pictures. They should indicate how a procedure such as the one described above could be executed.

Figure 13. Procedure Illustration

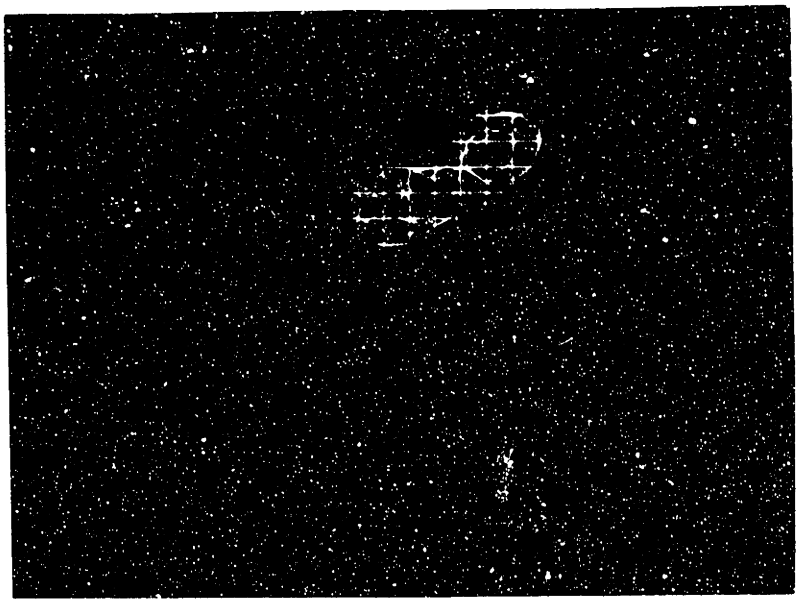
All surface contours approximately .16 el/A^3 above mean electron density.

a) Surface contours of a cubical region.

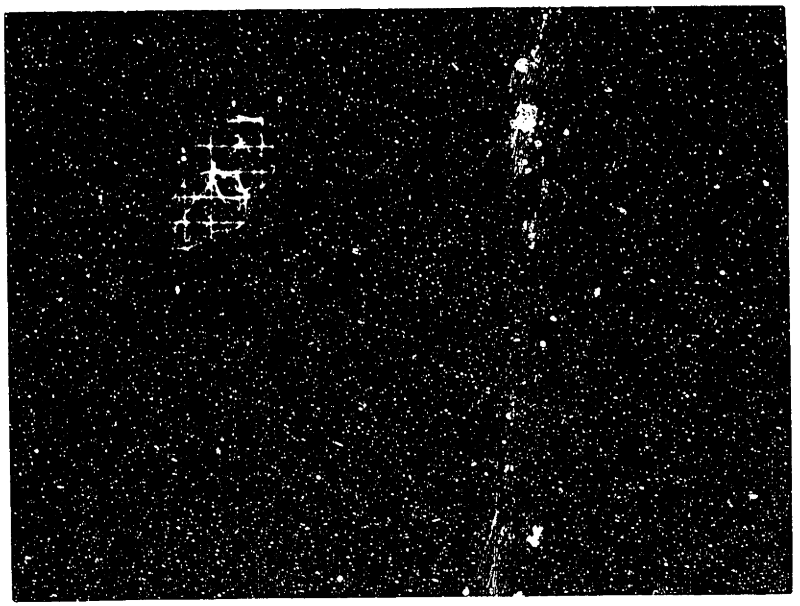
b) Result of initiating CONTIG with light pen by identifying central section of (a).



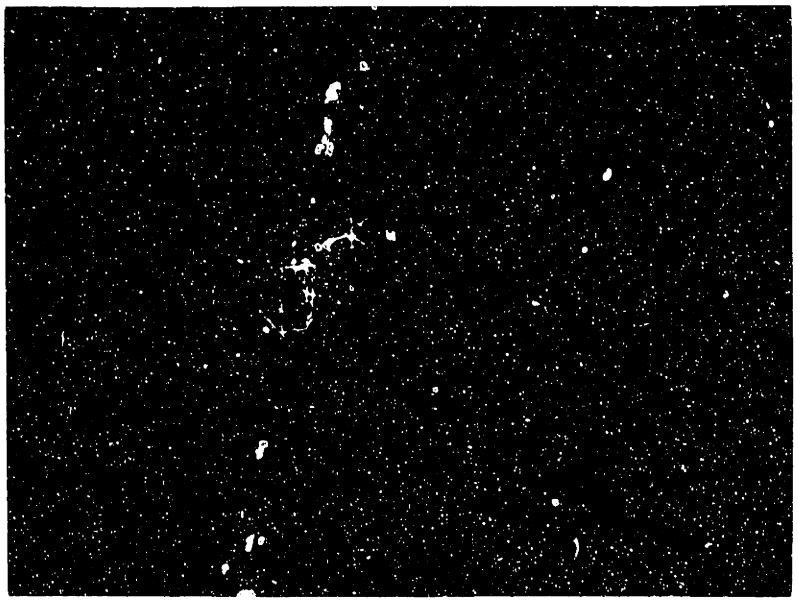
c) SUROND option in STICKS and fixing x-y coordinates of a point off the end of tube observed in (b).



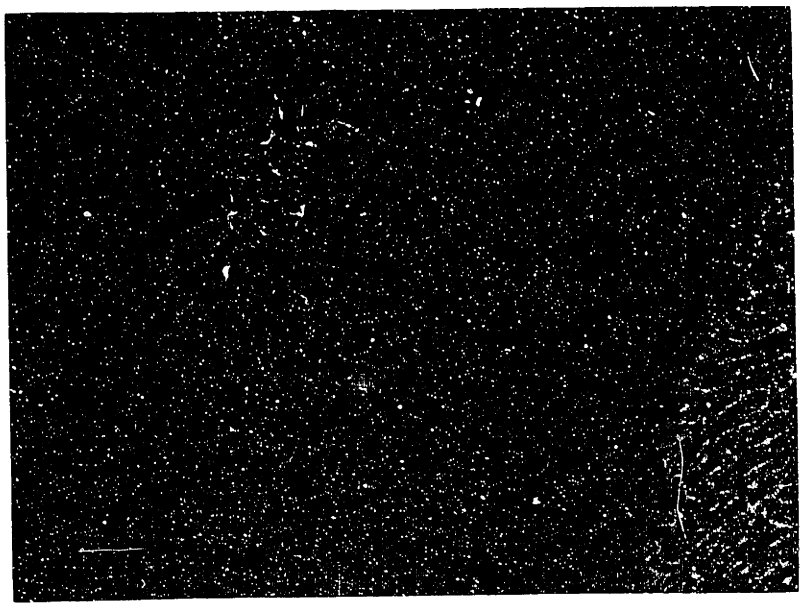
d) Fixing z-y.



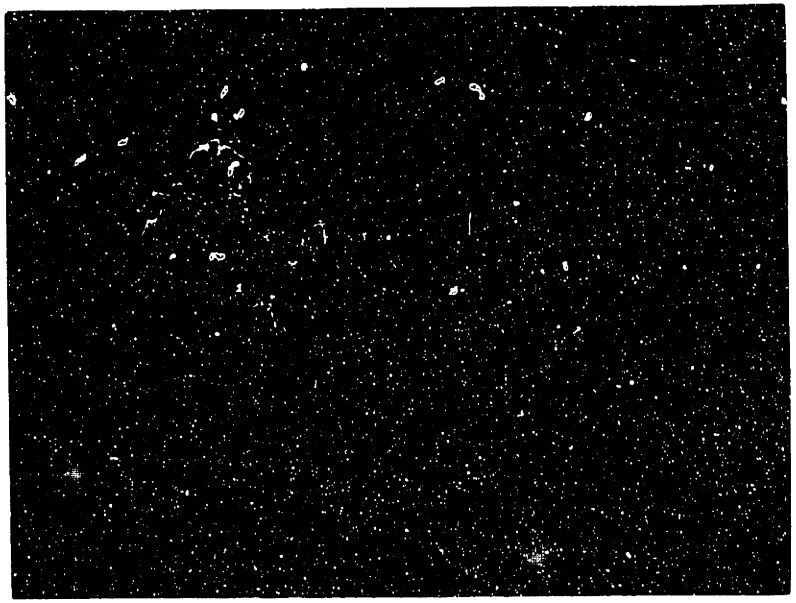
e) Result of (c) and (d).



f) Another example of SUROND.



g) A third example of SUROND.



h) Result of applying CONTIG by identifying upper right hand surface in (g). Display consists of approximately 800 straight line segments.

VI. EXPERIMENTAL TRIAL: MYOGLOBIN

6.1 Objective

It was decided to evaluate the power of the methods developed by applying the procedure to myoglobin in an attempt to construct an approximate, but correct, stick model of the pathway of the polypeptide, as well as identify the significant features such as the heme group, helices, and chain ends.

Only one factor made this trial slightly open to question. Those of us who attempted to apply the procedure to myoglobin were, at minimum, vaguely familiar with some of the features of the molecule. However, if this test did not succeed, then our methods were surely not satisfactory as presently developed.

6.2 Myoglobin Background Information

Myoglobin was chosen because its structure has been well established, and X-ray diffraction and electron density data were available. Its crystal's unit cell structure is relatively small, and is comprised of two asymmetric units related by a simple symmetry operation ($P2_1$ -- two-fold screw axis). Myoglobin possesses a readily identifiable, large, dense, non-amino acid moiety: the planar heme group, for use as a starting point.

The molecule is slightly planar and has the approximate dimensions forty-five by thirty-five by twenty-five Angstroms. The molecule consists of 1,260 atoms, excluding hydrogens, arranged as a linear polypeptide of 153 amino acid residues and an iron-

containing heme group. The molecular weight of this protein is 17,500. "It is a somewhat atypical protein in two respects: possession of a relatively high content of alpha-helix (77 per cent), and a complete lack of disulfide bridges or free sulfhydryl groups."[15]

Two molecules pack into a unit cell of the crystal. The edges of the unit cell have the dimensions:

a = 64.5 A, b = 30.86 A, c = 34.7 A, beta = 106 degrees.

Beta is the angle between the a and c axes. The other two angles are both right angles.

The electron density function of the unit cell was Fourier synthesized from intensity and phase information out to four Angstroms. Values of this function were evaluated at points in a sixty by thirty by thirty matrix in the unit cell. It should be mentioned that, for simplicity in the development of the user interaction routines, we assumed:

a = 60 A, b = 30 A, c = 30 A, beta = 90 degrees.

In other words, the unit cell was considered to be rectangular, and the distance between two data points in the electron density matrix was considered to be one Angstrom. The net result was merely a minor distortion of the shapes of the unit cell and molecule. If beta had been larger than 106 degrees, we would have added at this time a two-way coordinate system transformation to relate the crystallographic axes to a rectangular system. Now that the interaction routines are functioning smoothly, this transformation will soon be included in the computer procedure.

6.3 Evaluation of Results

6.3.1 Results

Approximately eighteen man-hours of labor and forty-five minutes of computer time, spread over six separate sessions of two to four hours duration, were used in an attempt to apply the methods developed in this thesis to a four Angstrom map of myoglobin, evaluated at one Angstrom intervals. The programs performed smoothly, and user interaction provisions, except for picture flicker, were quite smooth. The overall results were not as successful as anticipated, but during the trial there were many individual moments of encouragement. At the end of this brief trial, these results could be verified from the established structure:

1) The planar heme group could be easily identified, and its orientation determined. A bridge between the center of the heme (the iron atom -- by far the densest point on the map) and an apparently connected, hooked, tubular structure could easily be seen. It also appeared that two of the corners of the heme were closely associated with tubular regions which passed by. One corner was particularly isolated.

2) The connectedness of the iron atom at the center of the heme to a neighboring section of peptide chain could be established.

3) The neighboring section of peptide chain was verified to be the F helix.

4) The B helix was also found.

5) Portions of the C, G, and H helices were found.

6.3.2. Difficulties

It proved to be extremely difficult to follow the polypeptide

chain away from the heme and the F helix towards either the E or G helices. Continuity between helices was impossible to determine unambiguously. No corners between helices could be determined with certainty. Quite probably no neighboring molecule was intruded, but several adjacent unit cells were, as anticipated.

The assumption developed in Chapter V appeared to break down rather frequently. Display space limitations were annoying, and picture flicker caused fatigue.

Side groups were quite probably observed at times, but did not aid in determining the pathway of the backbone, and no attempt was made to identify them.

6.3.3 Individual Situations

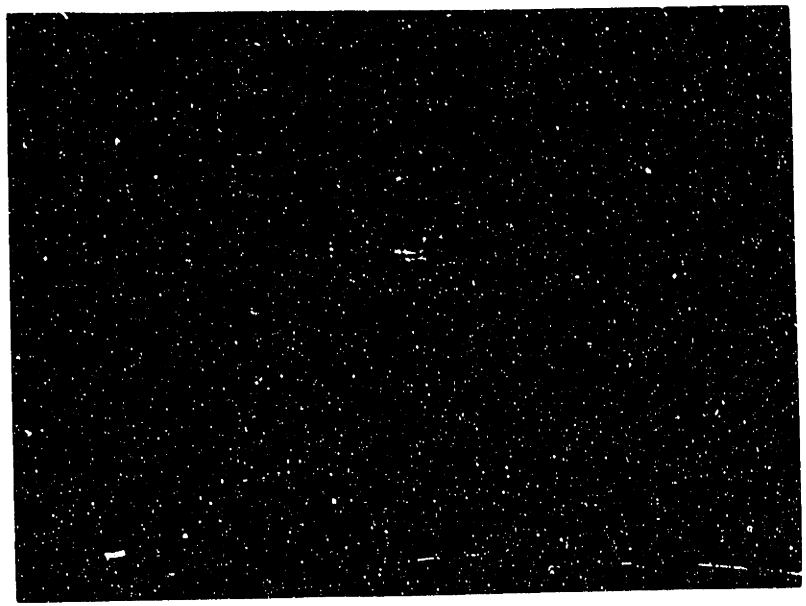
Several situations encountered should be elaborated. They are important as "learning experiences" for the future application of these procedures, as indications of the power of the methods developed with respect to the assumption developed in the previous chapter, and as indications of necessary improvements.

The first situation is illustrated in Fig. 16. A pointed protrusion on a surface was considered to be a good candidate for the peptide chain pathway. The SUROND option in STICKS was used to obtain the first picture. Note the two <u>conical</u> surfaces pointing at each other. The gap is actually on the order of four Angstroms, but appears smaller due to the angle of observation. CONTIG was initiated with the light pen, resulting in the display in the next photograph.

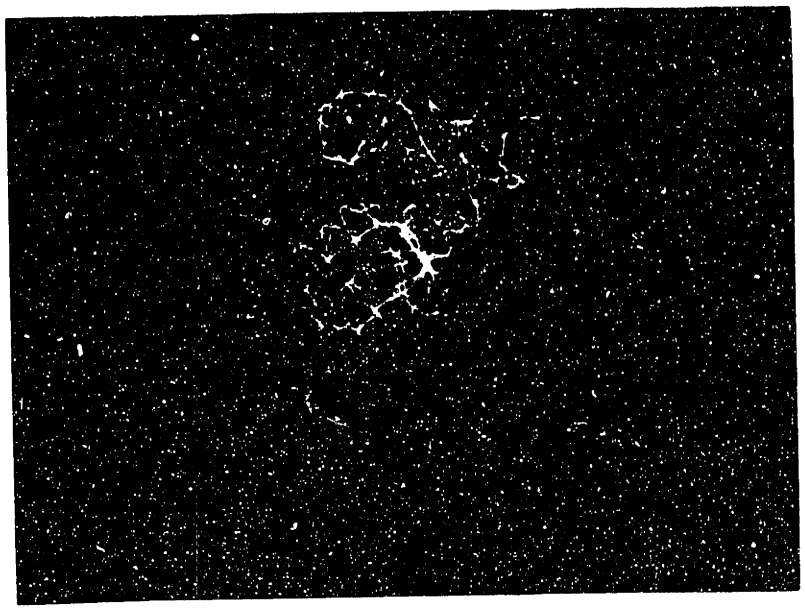
Figure 14. Assorted Pictures of Myoglobin Trial

All surface contours approximate .16 el/A^3 above mean electron density. Approximately one Angstrom between contours.

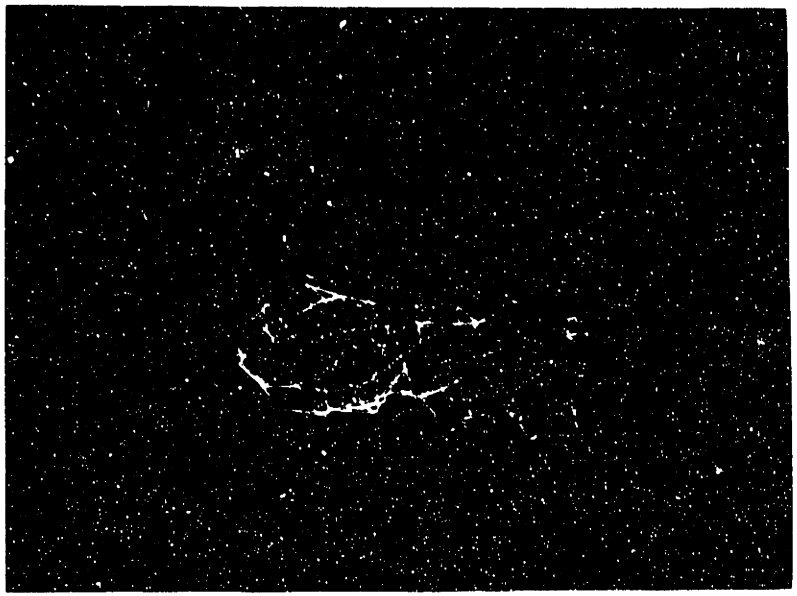
a) x-y projection of plane of heme (heme contours not completed). Attachment to F helix, and portion of F helix above and behind the heme.



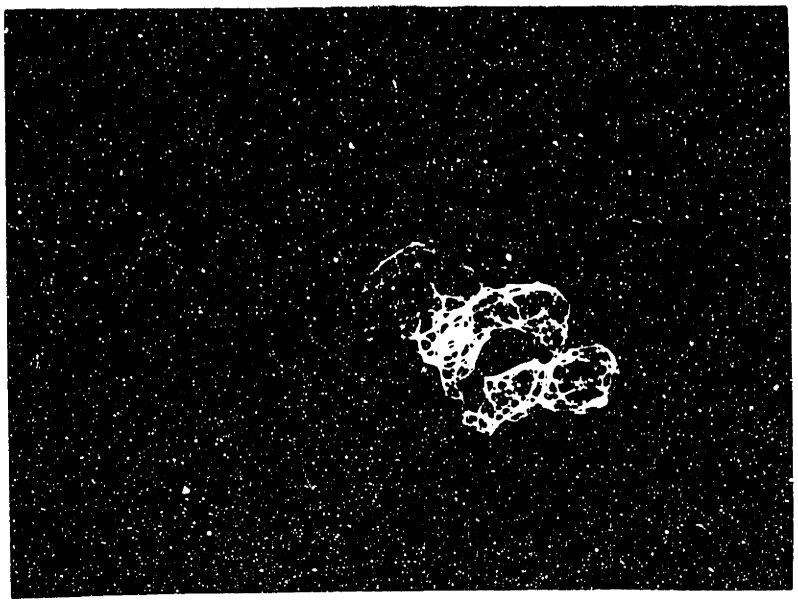
b) Another view of the heme, with (above) the connection to the F helix extending behind, and slightly up. Note upper corner particularly isolated.



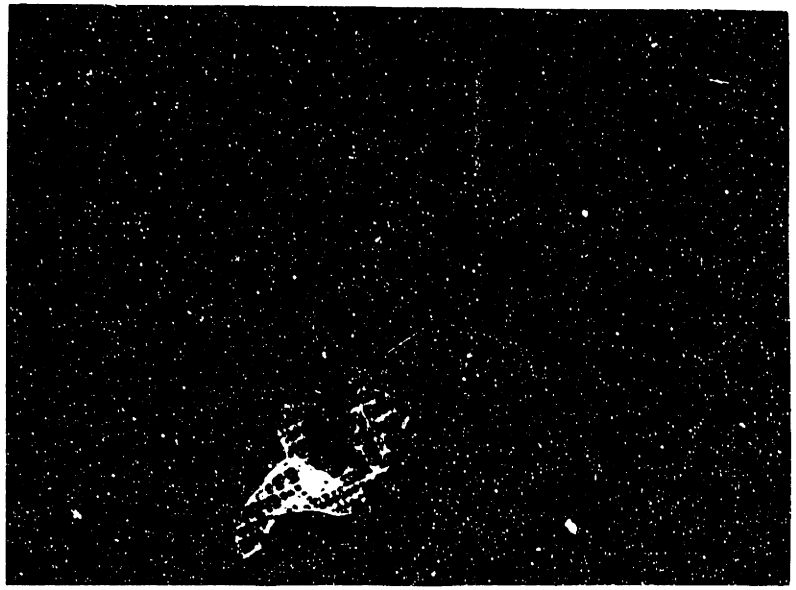
c) Portion of the F helix.



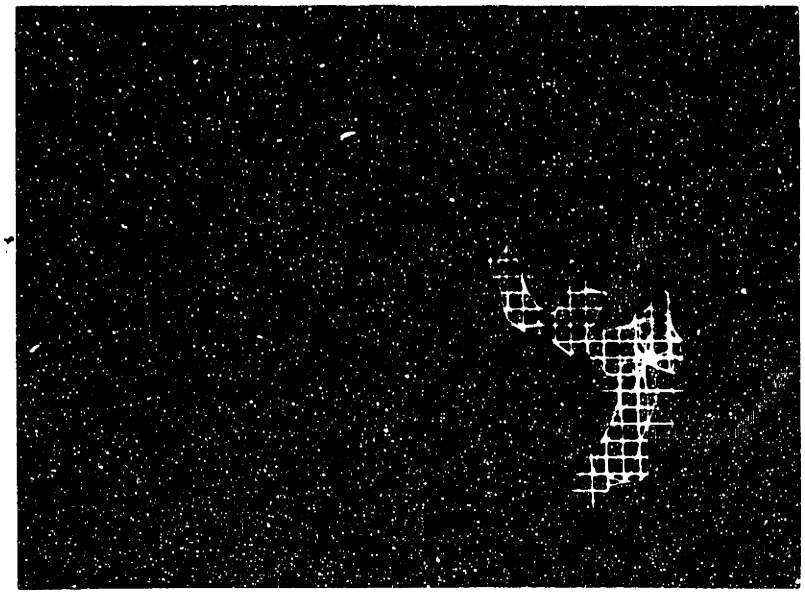
d) Troublesome view of the connection from the iron to the F helix. Large empty loop at top is a result of exhausting display space.



e) Encouraging example of coiled, tubular region extending from a portion near the corner of the heme.

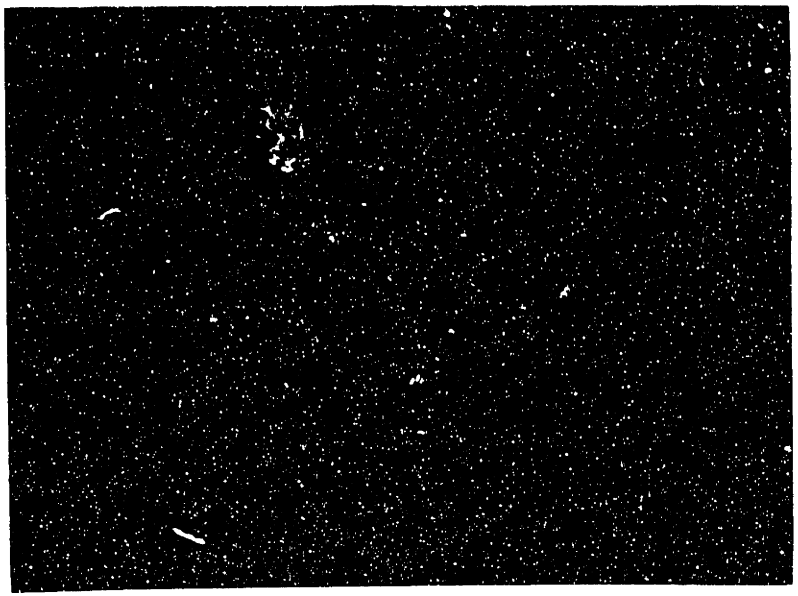


f) Tubular region extending from other corner of heme. Picture includes line segments drawn to model continuous regions already found.



g) Example of the use of SUROND option.

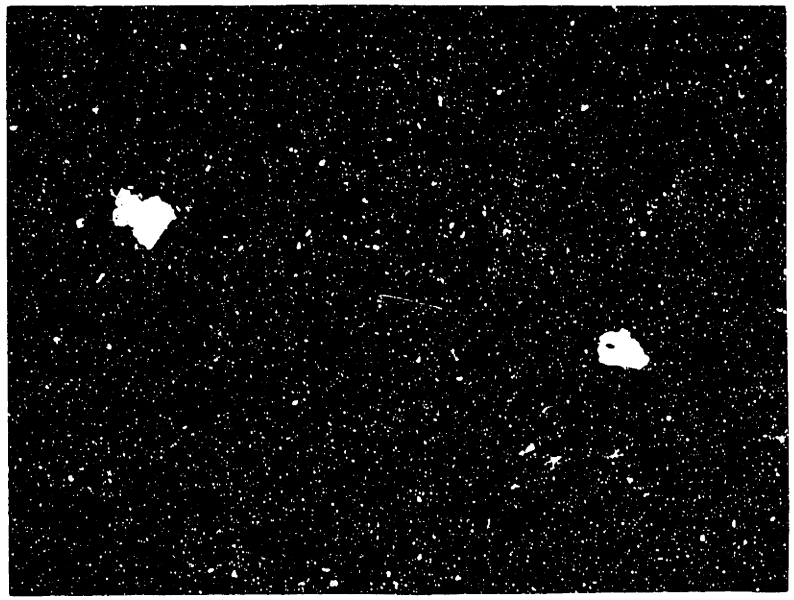
h) Result of applying CONTIG to (g).



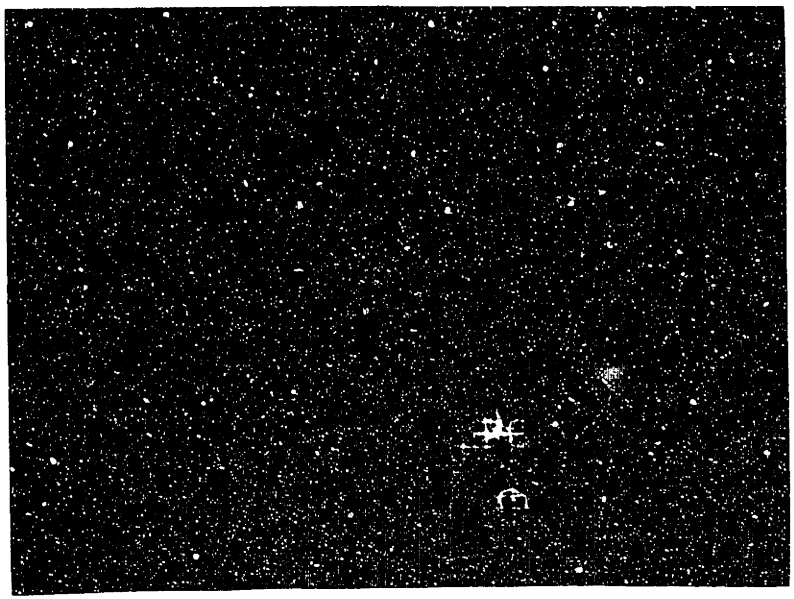
i) (h) from better perspective. Another case of display space exhaustion.

j) Model at this stage of investigation.

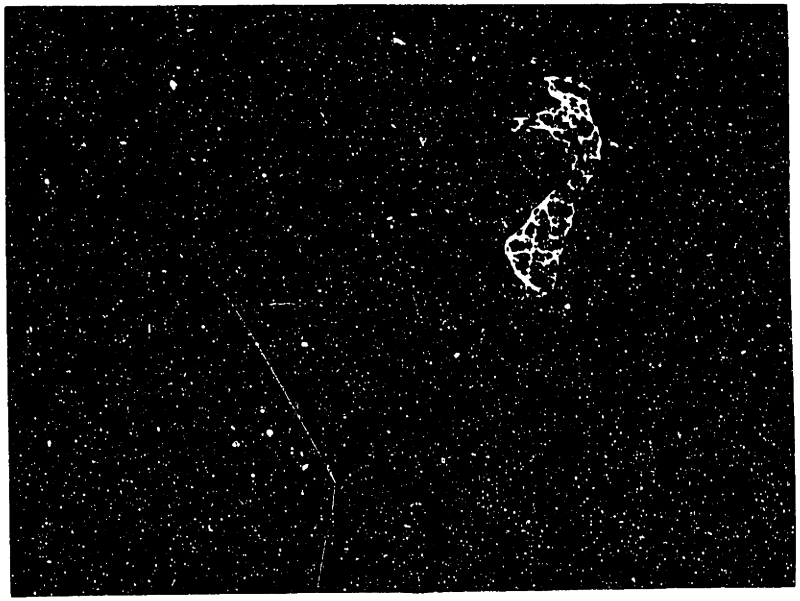
k) Excellent example of application of SUROND option to continue a high density region. Note coiled formation of surface.



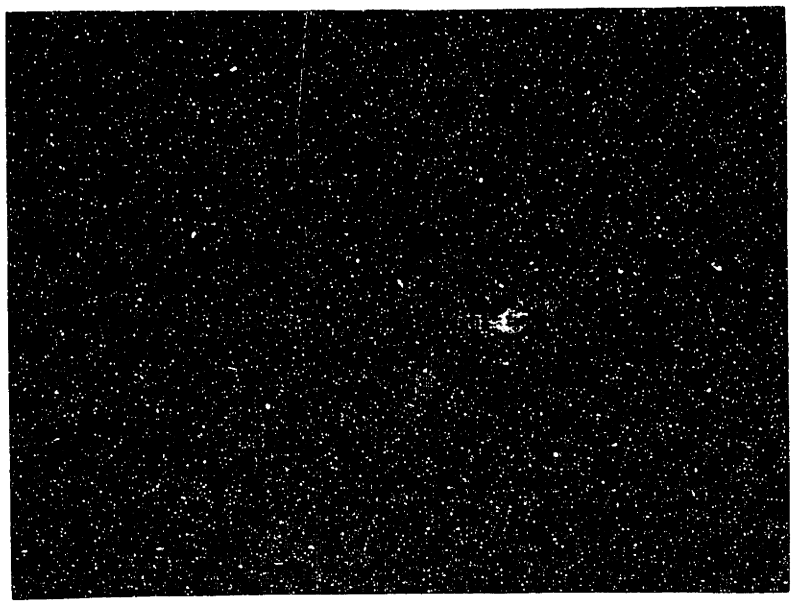
l) Example of a less encouraging application of the SUROND option. Note the several apparently isolated regions.



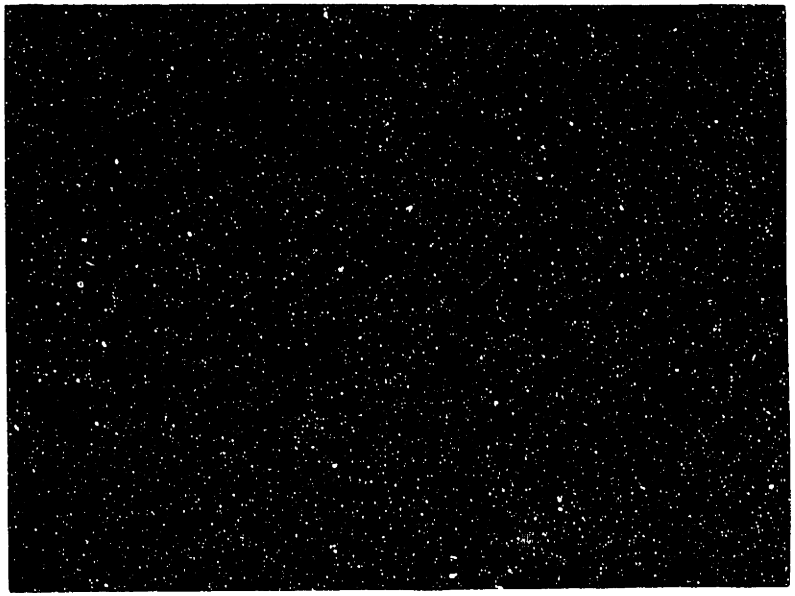
m) Tubular region extending from second corner of the heme.



n) Attempt to establish continuity.



o) Another attempt to establish continuity.



p) Complexity of the model towards the end. Note large number of disconnected segments.

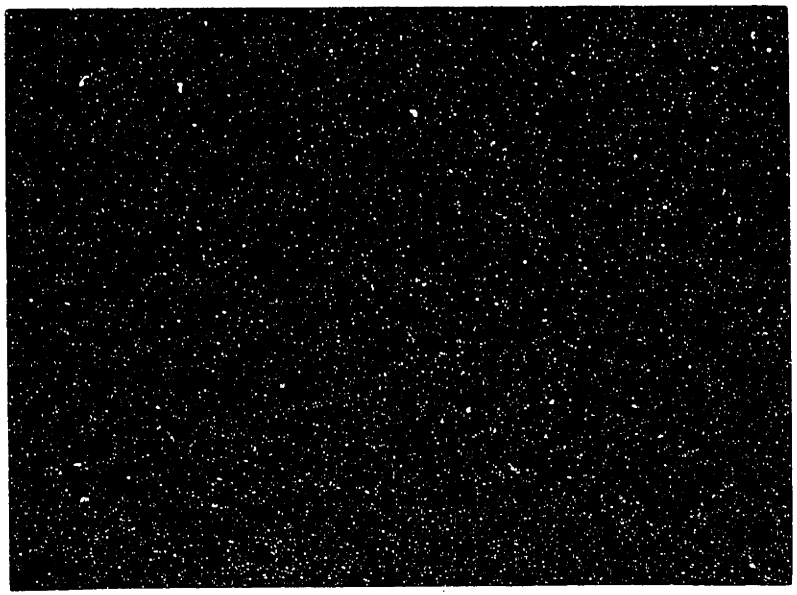
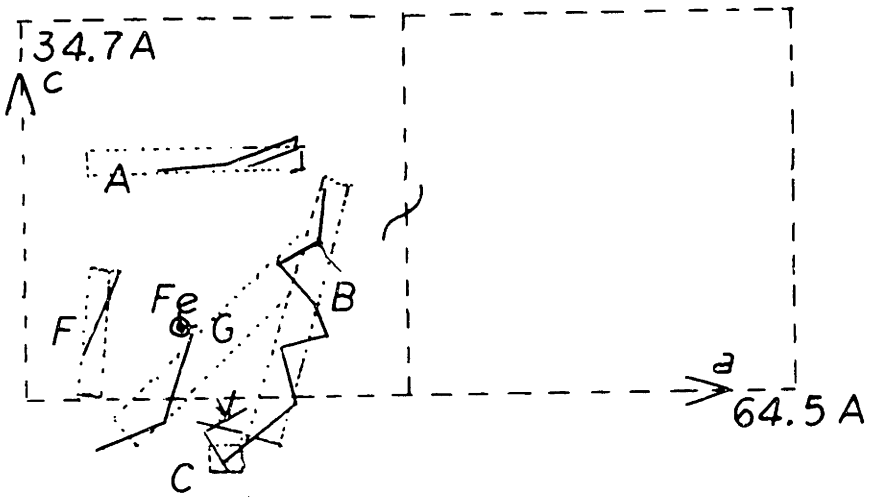


Figure 15. Verified Results

a) Model at end of trial: z-x projection, corresponding to a-c projection of helices below in (b).



b) Superposition of (a) and verified position of helices. Heme not shown but iron location, as well as unit cell and asymmetric unit boundaries indicated. Note coincidence of F, B, and A helices in constructed and established models.

Such conical surfaces pointed at each other were encountered several times, and proved to be extremely good initiating points for CONTIG. The length, pattern, and consistent thickness of many of the regions determined by CONTIG were encouraging. With higher resolution data, the effectiveness of this routine can only increase.

Several apparent branch points were encountered when CONTIG was applied, as indicated in Fig. 17. Hopefully, this situation will also be alleviated by higher resolution data. Furthermore, on several occasions, the CONTIG routine bounded off in several directions without completing the "arms" it began due to display space limitations.

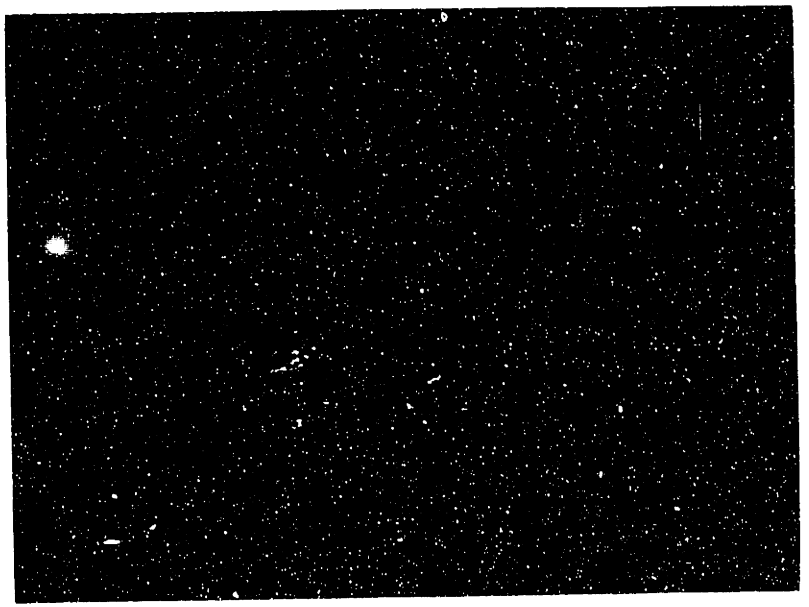
6.3.4 Improvements Indicated

The necessity for several improvements was indicated by this trial. The intensity of the "local effect" of the CONTIG routine (i.e., its tendency to complete a surface in a confined region before determining one or two contours of an arm extending significantly from that region) could be increased through one of two simple methods.

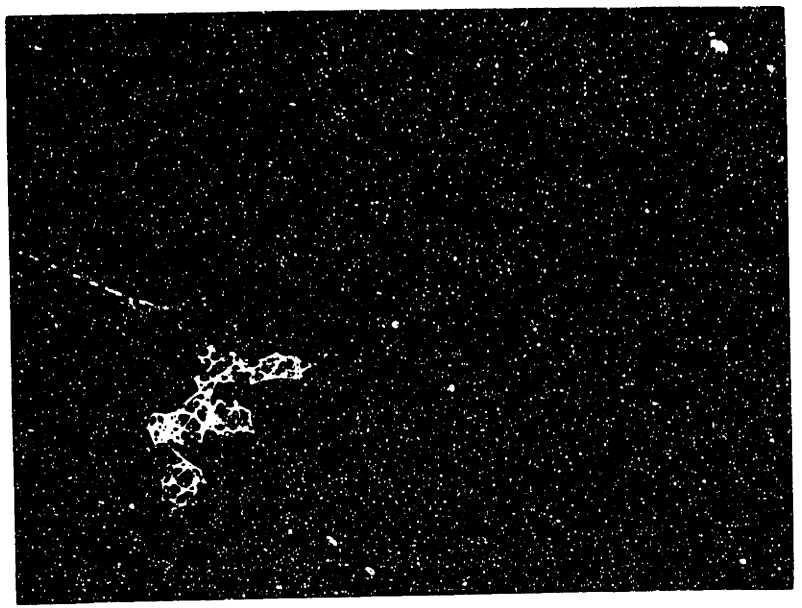
A necessity for higher resolution data was obvious from this trial. This topic will be returned to later.

A display system capable of plotting more than 2,000 straight line segments (free of flicker) would permit the procedure to be much more effective. Similarly, an increase in core memory space, although not absolutely necessary, would make use of higher resolution data more convenient.

Figure 16.



a) SUROND option yields display of two conical surfaces pointed at each other over gap of a few Angstroms. Gap appears smaller because of angle of observation. It is known that cone on right belongs to a long, continuous region of high density passing near the heme.



b) Result of applying CONTIG to surface on left in (a). Good indication that the continuity of the chain does indeed cross the gap.

Figure 17.

Example of the confusion caused by an apparent branch in the high density pathway. Note the three large regions meeting at a common central point.

VII. CONCLUSIONS AND SUGGESTIONS

Considering the results of the myoglobin trial, I think that this procedure, with the present display space limitations, is applicable only to higher resolution electron density functions. The method is not very satisfactory for directly giving an overview of the molecule, or an overview of structural features larger than about fifteen Angstroms in extent. However, with higher resolution data, its success at extracting and displaying small, continuous regions of high density should improve. It is necessary though, that the data have sufficient resolution to increase the validity of the assumption developed in Chapter V. For example, our method appears to be better suited to finding and displaying an alpha helix as a coiled region of high density, than as a thick rod. The stick model constructed by the user from visualizing small sections can be satisfactory in giving an overview of the structure of the molecule.

The relationship between the width of the frequency spectrum of the data supplied to the Fourier synthesis program, the sampling frequency of the electron density array, and the degree of success of the procedure developed in this thesis should be carefully investigated. Core space limitations are not insurmountable if it is found desirable to evaluate the electron density function every two-thirds Angstrom to take advantage of higher resolution data.

Any increase in display space and reduction of flicker would be extremely helpful. Hopefully, these improvements will come with the next generation of display equipment.

Integration of this procedure with the Molecular Model-Building Package developed by the Biology Group of Project MAC is feasible and would aid in three-dimensional structure determination from the electron density function.

User interaction was quite smooth except for two situations. Flicker was distracting, disturbing, and fatiguing. Also, many sections of disconnected line segments, representative of high density regions found, but whose continuity could not be established with certainty, could be confusing and detract from the user's ability to mentally establish a three-dimensional frame of reference.

I think that the problem of user adaptation to the type of display presented (i.e., adaptation to the type of transformation the procedure performs on this continuous, three-dimensional scalar function to relate it to a more familiar visual realm) is not a very severe one. However, execution of the various options of the procedure in the most advantageous manner is a skill which can only be gained with experience. The ability to recognize specific structures is a function of resolution, as well as a function of experience.

In summary, then, I believe that the type of approach presented here for enabling a molecular biologist to comprehend and model the information contained in a continuous three-dimensional electron density function is a promising approach. I do not believe the procedure is optimum or satisfactory as developed. However, the procedure may be satisfactory with higher resolution data. This question can only be answered by investigating several other molecules at various levels of resolution.

FOOTNOTES

[1]W. R. Brody, Computer Display of Three-Dimensional Scalar Functions, Master's Thesis for Massachusetts Institute of Technology, August, 1966.

[2]Cyrus Levinthal, "Molecular Model Building by Computer," Scientific American, June, 1966, p. 52.

[3]H. R. Mahler and E. H. Cordes, Biological Chemistry, (New York: Harper and Row, 1966), p. 101.

[4]W. R. Brody.

[5]Ibid., p. 50.

[6]T. H. Gossling, "Two Methods of Presentation of Electron-Density Maps Using a Small-Store Computer," Acta Crystallographa, 22:465-68, p. 465.

[7]Robert H. Stotz and John E. Ward, "Operating Manual for the ESL Display Console," Project MAC Internal Memorandum, MAC-M-217, p. 3.

[8]Cyrus Levinthal, "Molecular Model Building by Computer," Scientific American, June, 1966, p. 50.

[9]W. R. Brody, p. 49.

[10]William Siebert, "Circuit Signals and Systems," Massachusetts Insitute of Technology notes for Course 6.05, chapter IV, p. 51.

[11]Cyrus Levinthal, p. 47.

[12]J. C. Kendrew, R. E. Dickerson, B. E. Strandberg, R. G. Hart, and D. R. Davies, "Structure of Myoglobin," Nature, 185:422-27, February 13, 1960, p. 422.

[13]J. C. Kendrew, G. Bodo, H. M. Dintzis, R. G. Parrish, and H. Wyckoff, "A Three-Dimensional Model of the Myoglobin Molecule Obtained by X-Ray Analysis," Nature, 181, March 8, 1958, p. 667.

[14]J. C. Kendrew, R. E. Dickerson, et. al., p. 423.

[15]H. R. Mahler and E. H. Cordes, Biological Chemistry, (New York: Harper and Row, 1966), p. 103.

BIBLIOGRAPHY

Arden, B., Galler, B., and Graham, R., The Michigan Algorithm Decoder Manual.

Bodo, G., Dintzis, H. M., Kendrew, J. C., and Wyckoff, H.W., "A Low-resolution Three-Dimensional Fourier Synthesis of Sperm-whale Myoglobin Crystals," Proceeds Royal Society, 253:70-102, 1959.

Brody, W.R., Computer Display of Three-dimensional Scalar Functions, Master's Thesis for Massachusetts Institute of Technology, August, 1966.

Crisman, P. A., ed., The Compatible Time-sharing System, A Programmer's Guide, 2d ed., (M.I.T. Press: Cambridge, Massachusetts, 1965).

Fano, R. M. and Corbato, F. J., "Time-sharing on Computers," Scientific American, 215:128-43, September, 1966.

Fridborg, K., Kannan, K. K., Tiljas, A., Lundin, J., Strandberg, B., Strandberg, R., Tilander, B., and Wiren, G., "Crystal Structure of Human Erythrocyte Carbonic Anhydrase C," Journal of Molecular Biology, 25:505-16, 1967.

Gossling, T. H., "Two Methods of Presentation of Electron-density Maps Using a Small-store Computer," Acta Crystallographa, 22:465-68, 1967.

Kendrew, John C., "The Three-dimensional Structure of a Protein Molecule," Scientific American, December, 1961.

Kendrew, J. C., Bodo, G., Dintzis, H. M., Parrish, R. G., and Wyckoff, H., "A Three-dimensional Model of the Myoglobin Molecule Obtained by X-ray Analysis," Nature, 181:662-66, March 8, 1958.

Kendrew, J. C., Dickerson, R. E., Strandberg, B. E., Hart, R. G., and Davies, D. R., "Structure of Myoglobin," Nature, 185:422-27, February 13, 1960.

Levinthal, Cyrus, "Molecular Model Building by Computer," Scientific American, June, 1966.

Mahler, H. R. and Cordes, E. H., Biological Chemistry, (Harper and Row: New York, 1966).

Matthews, B. W., Sigler, P. B., Henderson, R., and Blow, D. M., "The Three-dimensional Structure of Tosyl-alpha-chymotrypsin," unpublished paper.

Oettinger, Anthony G., "The Uses of Computers in Science," Scientific American, 215:160-75, September, 1966.

Ross, D. T., Stotz, R. H., Thornhill, D. E., Lang, C.A., "The Design and Programming of a Display Interface System Integrating Multi-access and Satellite Computers," paper presented at the ACM/SHARE Fourth Annual Design Automation Workshop, Los Angeles, California, June 26-28, 1967.

Stotz, Robert H., and Ward, John E., "Operating Manual for the ESL Display Console," Project MAC Internal Memorandum, MAC-M-217.

Watson, J. D., Molecular Biology of the Gene, (W. A. Benjamin, Inc.: New York, 1965).

Wilson, H. R., Diffraction of X-rays by Proteins, Nucleic Acids and Viruses, (Edward Arnold, Ltd.: London, 1966).