COMPUTERS AND BIOMEDICAL RESEARCH 8, 244–253 (1975)

Transformation of Horsley-Clarke Coordinate for Electrode Implantation Experiments in Stereotaxic Surgery*

CHACKO C. NEROTH, MELVIN SIMMONS, AND ELWIN MARG

School of Optometry and Department of Electrical Engineering and Computer Science, University of California, Berkeley, California 94720

Received April 29, 1974

The speed and versatility of Horsley–Clarke stereotaxic procedures in the drug-free monkey preparation was extended by computerization. Computer programs are described for transformation of the coordinates of the stereotaxic frame and that of the grid plate mounted on the skull. This provided for calculating the electrode entry points and depth of the electrode penetration for areas of interest described in the brain atlas.

1. INTRODUCTION

The development of the Horsley–Clarke stereotaxic instrument was a major step in neurophysiological investigations of brain function. Chronic experiments using stereotaxically implanted electrodes provide basic data from drug-free animals which can be obtained in no other way.

The stereotaxic procedure consists of mounting a rigid metal frame on the head of the subject in a precise manner, using certain bony marks of the skull as reference points and then implanting the electrodes with three-dimensional control into the interior points of the brain. Since chronic experiments often require that the animal be awake and unanesthetized (*1*), the repeated placement of the animal in a stereotaxic frame would be practicable. A grid plate is mounted on the skull, and the implantation is controlled with respect to this plate. If direct readings from the brain atlas are to be used effectively, the grid plate has to be precisely positioned on the atlas reference coordinates. However, it is often desirable to have the grid plate positioned at some convenient position other than the atlas reference coordinates.

In order to establish a means by which the experimenter could work as freely in the transformed system of coordinates (of the grid plate) as in the standard stereotaxic system, a number of computer programs described here have been developed. The programs do the transformations between the two sets of coordinates, establish an atlas file on the disk storage which describes the areas of interest page by

* Research for this paper was supported through grant number GB33338, from the National Science Foundation.

page from the brain atlas, and calculate the electrode entry and exit points and the depth of the electrode for the areas described by the atlas file for a given electrode implantation. All input data is entered using the standard brain atlas coordinates. Answers for the implantations are given in the grid plate coordinates, but the entry

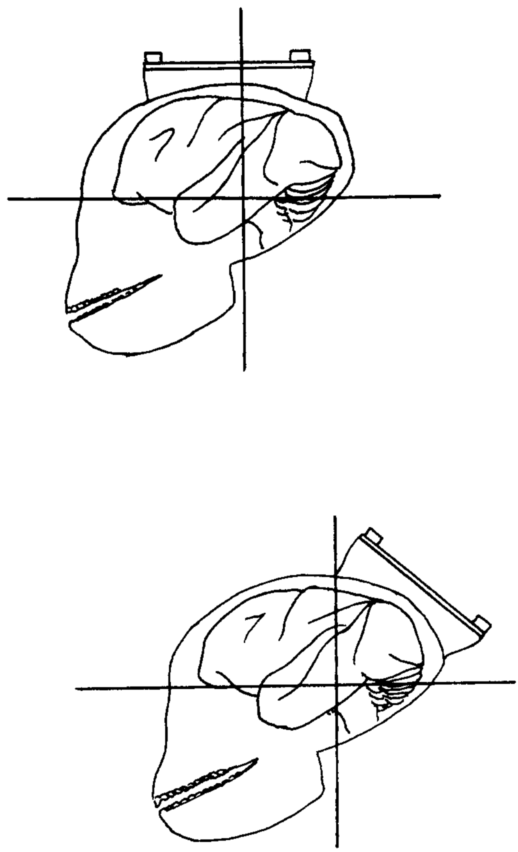

FIG. 1.

and exit points are given in the standard atlas coordinates. With these programs it makes it very easy, for example, in visual cortex experiments, to position the grid plate far enough back so that the visual cortex could be reached easily with an electrode perpendicular to the plate as shown in Fig. 1.

2. Program Description

The programs are divided into three sections: the transformation program, the atlas data file program, and the entry/exit point program. All programs are written in FORTRAN. This makes it very easy for adapting them for systems other than the one described in Section 3.

2.1 *The Transformation Program*

2.1.1. *Method of computation.* To do the orthogonal transformation between the two coordinate systems, three reference points of one system in terms of the other

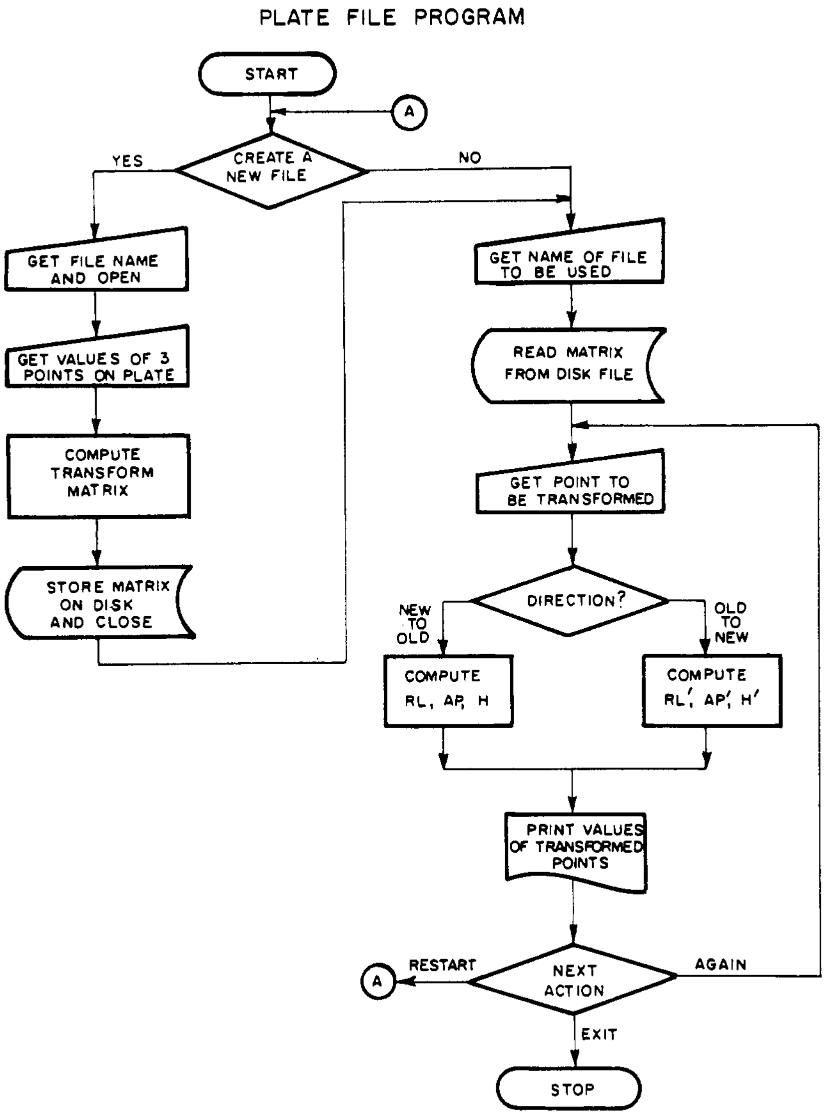

FIG. 2.

system must be known. Let X, Y, Z be the three orthogonal axes of the stereotaxic frame, and X', Y', Z' that of the grid plate. P: (x, y, z), denotes a point in the former system and P' (x', y', z') denotes a point in the latter system.

The three reference points for transformation are measured on the grid plate, after it has been afixed to the skull, in terms of the stereotaxic frame. The reference points chosen are $P0$: $(x0, y0, z0)$ the center of the grid plate, $P1$: $(x1, y1, z1)$ a point on the X' axis 10 mm away from the center and $P2$: $(x2, y2, z2)$, a point on the Y' axis 10 mm away from the center.

The unit vectors A: $(a1, a2, a3)$, B: $(b1, b2, b3)$ and C: $(c1, c2, c3)$ of the grid plate system in terms of the other system are given by:

$$A\colon (a1, a2, a3) = P1 - P0\colon \left(\frac{x1 - x0}{d1}, \frac{y1 - y0}{d1}, \frac{z1 - z0}{d1}\right),$$

$$B\colon (b1, b2, b3) = P2 - P0\colon \left(\frac{x2 - x0}{d2}, \frac{y2 - y0}{d2}, \frac{z2 - z0}{d2}\right),$$

$$C\colon (c1, c2, c3) = A \times B\colon (a2^* b3 - a3^* b2, a3^* b1 - a1^* b3, a1^* b2 - a2^* b1),$$

where

$$d1 = | P1 - P0 | = ((x1 - x0)^2 + (y1 - y0)^2 + (z1 - z0)^2)^{\frac{1}{2}},$$
$$d2 = | P2 - P0 | = ((x2 - x0)^2 + (y2 - y0)^2 + (z2 - z0)^2)^{\frac{1}{2}}.$$

Given the unit vectors A, B, C, a point P: (x, y, z) in one system and the same point P' (x', y', z') in the other system are related by the equations.

$$\begin{vmatrix} x \\ y \\ z \end{vmatrix} = \begin{vmatrix} a1 & b1 & c1 \\ a2 & b2 & c2 \\ a3 & b3 & c3 \end{vmatrix} \begin{vmatrix} x' \\ y' \\ z' \end{vmatrix} + \begin{vmatrix} x0 \\ y0 \\ z0 \end{vmatrix}$$

and

$$\begin{vmatrix} x' \\ y' \\ z' \end{vmatrix} = \begin{vmatrix} a1 & b1 & c1 \\ a2 & b2 & c2 \\ a3 & b3 & c3 \end{vmatrix} \begin{vmatrix} x - x0 \\ y - y0 \\ z - z0 \end{vmatrix} .$$

The first part of the program takes in the reference points, computes the transformation matrix and stores it on the disk under the described name. The inverse matrix for this orthogonal transformation is the transpose of this and hence is not stored separately. If a current file name is requested during this phase, then the program assumes that a transformation of coordinate is desired and uses the disk file for the stored transformation matrix.

2.1.2. *Sample run.* In the output AP stands for the anterior–posterior (Y) axis and RL for the right–left (X) axis.

```
ENTER 0 TO USE EXISTING FILE
ENTER 1 TO ESTABLISH NEW FILE
1
NAME OF OUTPUT FILE = ISARUI

IN TERMS OF ATLAS COORDINATES

ENTER A POINT ON THE POSITIVE AP AXIS
AP = -3.3   RL = 3.2   H = 45.4

ENTER A POINT ON THE POSITIVE RL AXIS
AP = -12.6   RL = 12.9   H = 41.5

ENTER THE ZERO POINT
AP = -12.9   RL = 3.0   H = 42.6

ENTER DECISION
1 = AGAIN, 2 = RESTART, 3 = EXIT
```

At this point 2 restarts the program just as above; 1 assumes a transformation of coordinate with the current file.

```
1
ENTER POINT TO BE TRANSFORMED
AP = 0.   RL = 0.   H = 0.

ENTER TRANSFORMATION DIRECTION
0 = FROM OLD TO NEW
1 = FROM NEW TO OLD
0
TRANSFORMED POINT
AP = 0.40   RL = 2.11   H = -44.55
ENTER DECISION
1 = AGAIN,   2 = RESTART,   3 = EXIT
```

Next section shows a reverse transformation

```
1
ENTER POINT TO BE TRANSFORMED

AP = 0.   RL = 0.   H = 0

ENTER TRANSFORMATION DIRECTION
0 = FROM OLD TO NEW
1 = FROM NEW TO OLD
1
TRANSFORMED POINT

AP = -12.90   RL = 3.00   H = 42.60

ENTER DECISION
1 = AGAIN,   2 = RESTART,   3 = EXIT
3
```

2.2. *The atlas data file program*

2.2.1. *Program description.* This program forms a disk file containing the description of various areas of interest of the brain atlas. A six-letter code is assigned to

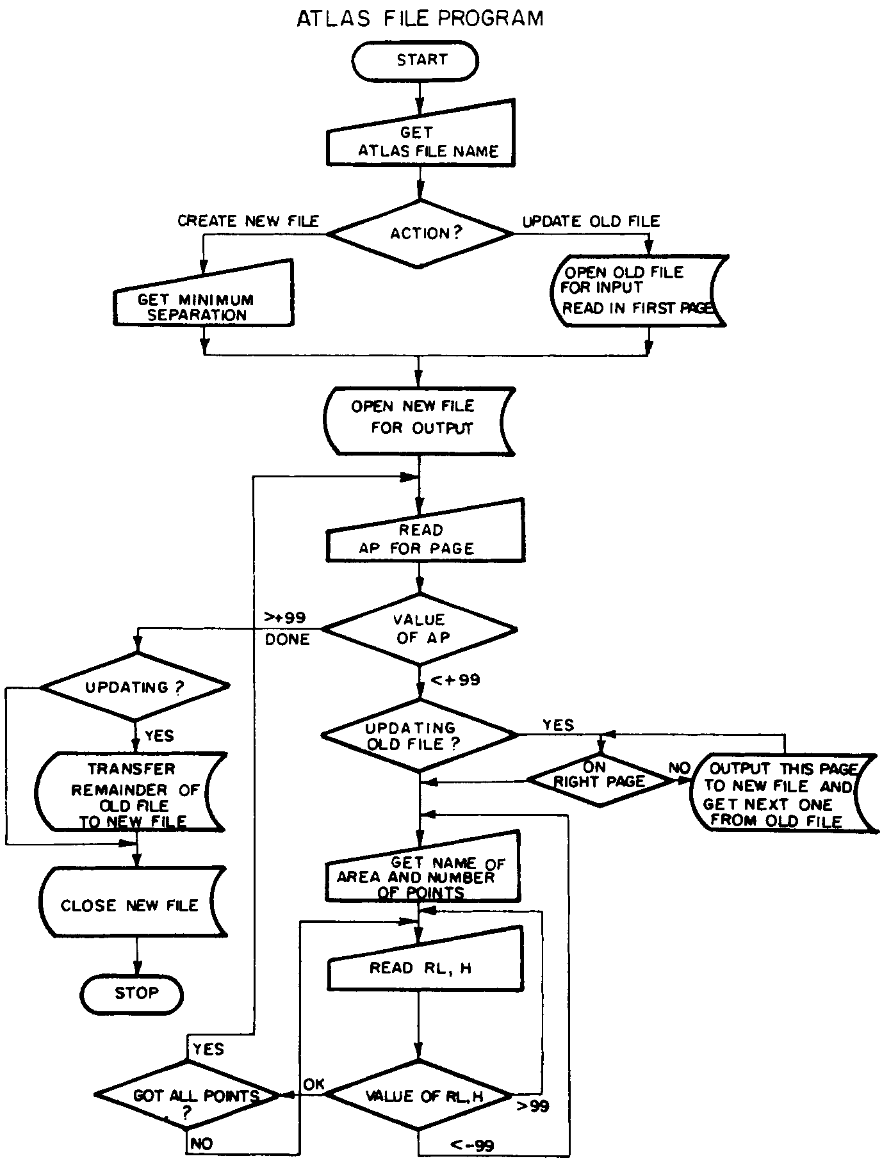

FIG. 3.

each area for identification. The entry/exit program described in Section 2.3 gets its data from this file. Because of the symmetry, only the left or the right half need be described.

The data file is organized by the standard atlas page description, and is ordered in ascending AP values. Each page may have up to 30 areas, each described by a maximum of nine points consecutively around the perimeter of the area. An area may extend to any number of pages.

The absolute coordinate values should be less than 99.0. Other values are used for editing purposes. A page coordinate (AP) of > +99.0 signals the end of a file creation or modification; RL or H > +99.0 signals the repetition of that point; RL or H < −99.0 signals the program to accept all points of the current area over again. The file is sequential, and changes to the file require that the entire file be read and written back. Since the atlas does not change, the only changes made to the data file should be for correcting errors or for adding new area descriptions.

2.2.2. *Using this program.*

1. The program starts by asking the code name for the file. Respond with the six alphanumeric character code name.
2. The program then asks if you wish to establish a new file or just change an existing file. If you establish a new file, the program asks for the distance between pages, otherwise it assumes the existing values.
3. The program prints out a note on the values used to control the input of data.
4. The program asks for the page on which you wish to describe an area.
5. The program asks for the number of points needed to describe that area.
6. The program then accepts the point coordinates by point number. The points must describe the area either clockwise or counterclockwise around the perimeter of that area.
7. Steps 4, 5, and 6 are repeated until an AP value greater than +99.0 is given.

Example:

Suppose the name of the atlas file is RHESUS and we wish to describe the LGB for AP = 2.0. The separation distance between pages for the rhesus atlas is .5 mm.

```
NAME OF ATLAS FILE = RHESUS
ENTER 0 TO UPDATE EXISTING FILE, OTHERWISE 1
1
ESTABLISHING A NEW FILE
MINIMUM AP SEPARATION = .5

ENTER PAGES IN ASCENDING AP VALUES

AP VALUES GREATER THAN +99.0 = END OF INPUT
RL, H VALUES GREATER THAN +99.0 = RE-ENTER POINT VALUES
RL, H VALUES LESS THAN -99.0 = RE-ENTER ALL POINTS FOR THE AREA

POINTS MUST BE CONSECUTIVE AROUND THE PERIMETER
```

```
FOR THIS PAGE,  AP = 2.0
NAME OF AREA = LGB
NO. OF POINTS FOR LGB = 5
POINT NO. 1   RL = 9.75    H = 15.3
POINT NO. 2   RL = 10.20   H = 14.4
POINT NO. 3   RL = 10.     H = 13.90
POINT NO. 4   RL = 9.7     H = 14.1
POINT NO. 5   RL = 9.3     H = 15.3
FOR THIS PAGE,  AP =
```

Now suppose we wish to delete the above area.

```
FOR THIS PAGE,  AP = 2.0
NAME OF AREA = LGB
NO. OF POINTS FOR LGB = 0
FOR THIS PAGE,  AP =
```

Now suppose we make a mistake when entering the above.

```
FOR THIS PAGE,  AP = 2.0
NAME OF AREA = LGB
NO. OF POINTS FOR LGB = 5
POINT NO. 1   RL = 9.95   H = 100      (repeat last point)
POINT NO. 1   RL = 9.75   H = 15.3
POINT NO. 2   RL = 10.2   H = −100.    (repeat all points)
NO. OF POINTS FOR LGB = 0
FOR THIS PAGE,  AP = 99.          (end of input)
```

2.3 *Entry/Exit Computation*

This program determines the areas described in the atlas through which the electrode passes and the corresponding points of entry and exit. It uses both the atlas data file and the transformation file developed in the previous sections.

2.3.1. *Method of Computation.* The computation is done for two cases. The first one is when the electrode passes through an area lying in one page. The second case is for the electrode to penetrate (or exit) a body between two consecutive pages. A brief description is given here; for details, please see Ref. (*2*).

For the first case, the point of intersection of the electrode and the atlas page is computed. To determine whether this point lies within an area, a line is drawn from the point along the RL axis in the positive direction. If the line crosses an area boundary an odd number of times, then the point is within the area. All such points on the various pages are determined including those for the second case, and the points in the largest and smallest pages give the entry and exit points.

For the second case, it is assumed that a body (area) does not change shape for half the distance between pages. Now the plane formed by a line connecting two

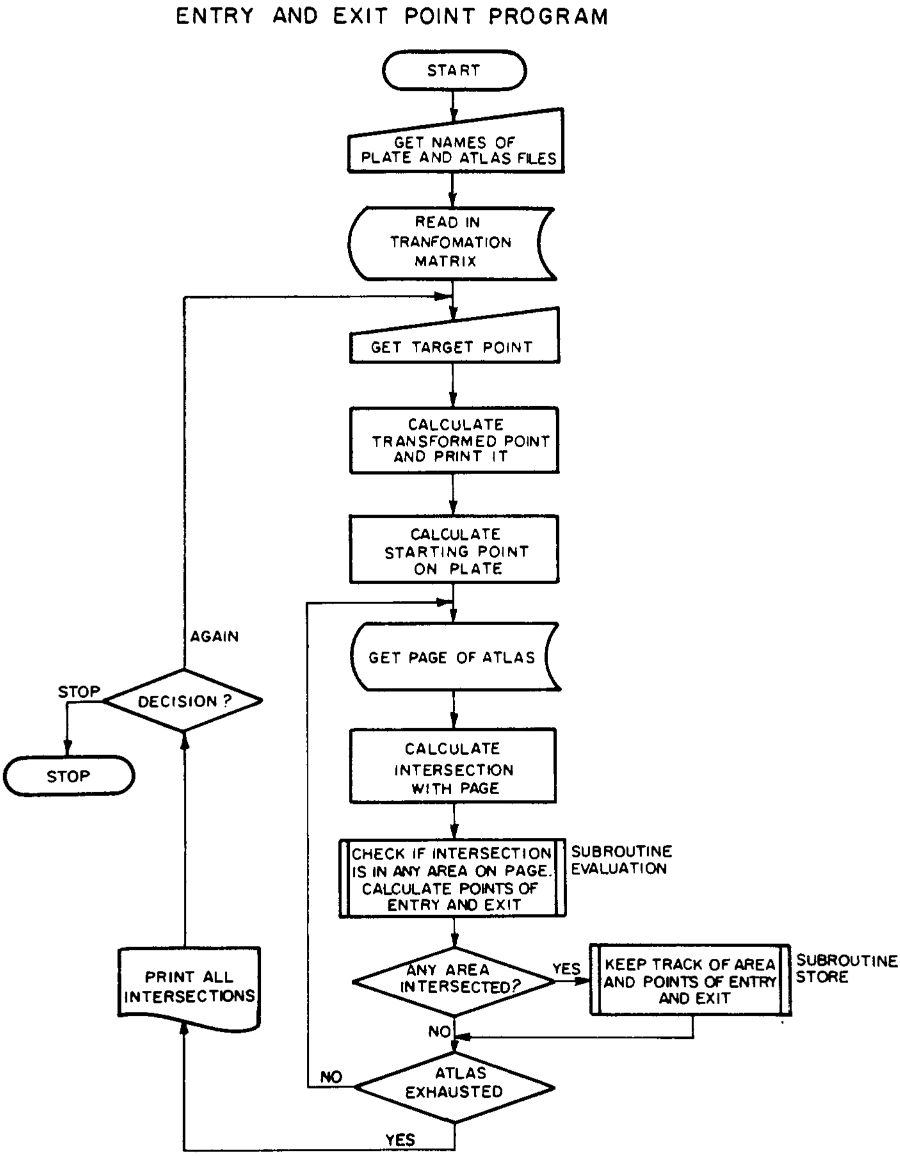

FIG. 4.

consecutive points of the perimeter along the AP axis is determined. On this plane, a region bounded by the current AP values plus and minus half the distance between pages and of length between the two consecutive points in the perimeter is computed. If the point of intersection of the electrode and this plane lies in this region it is added to the points determined in the first case.

2.3.2. *Sample run.*

1. Enter the name of the atlas file and the plate file.
2. Enter the target point in terms of atlas coordinates.
3. The first response by the program is to provide the target point transformed into the coordinates of the plate. This provides information for drilling of the hole in the plate. This hole is the starting point for the electrode.
4. A list of all areas through which the electrode passes will be given, listing the code names of the area, the depth of the electrode at the point of entry and at the point of exit, and the atlas coordinates for these two points.

Suppose the name of the atlas file is RHESUS and the name of the plate file is SHISAI. Also suppose the only area stored on the disk is the LGB and the target point is AP = 7.0, RL = 8.5 and H = 12.0 then the following is what might take place.

```
NAME OF ATLAS FILE = RHESUS

NAME OF PLATE FILE = SHISAI

POINT OF INTEREST IS AP = 7.  RL = 8.5  H = 12.

TRANSFORMED POINT
AP = 13.22  RL = 9.40  H = 33.57

BODY  ENTRY  ATLAS  COORDINATES  EXIT   ATLAS  COORDINATES
NAME  DEPTH  AP       RL   H     DEPTH  AP        RL   H
LGB   30.90  5.28   9.26   14.94 36.46  8.63     8.25 9.62

ENTER DECISION
1 = AGAIN,  2 = STOP
2
```

3. Hardware Configuration

The programs are written for a Digital Equipment Corporation PDP/8 with 8K of core, a disk, and a teletype for input/output. The programs are written in FORTRAN, and operates under the PS 8 disk monitor system for the above configuration of devices.

4. Mode of Availability of Programs

A copy of the FORTRAN listings of the programs can be obtained from the authors.

REFERENCES

1. Wurtz, R. H. Visual receptive fields of striate cortical neurons in awake monkeys. *J. Neurophysiol.* **32**, 727–747 (1969).
2. Simmons, M. B. Computerized transformation of Horsley–Clarke coordinates and electrode penetration of solid objects for stereotaxic surgery. M.S. Project Report, Department of Engineering and Computer Sciences, University of California, Berkeley, CA, 1972.