NBS TECHNICAL NOTE 829

U.S. DEPARTMENT OF COMMERCE / National Bureau of Standards

Multicommodity Network Plotting via Program NETPLT

NATIONAL BUREAU OF STANDARDS

The National Bureau of Standards[1] was established by an act of Congress March 3, 1901. The Bureau's overall goal is to strengthen and advance the Nation's science and technology and facilitate their effective application for public benefit. To this end, the Bureau conducts research and provides: (1) a basis for the Nation's physical measurement system, (2) scientific and technological services for industry and government, (3) a technical basis for equity in trade, and (4) technical services to promote public safety. The Bureau consists of the Institute for Basic Standards, the Institute for Materials Research, the Institute for Applied Technology, the Institute for Computer Sciences and Technology, and the Office for Information Programs.

THE INSTITUTE FOR BASIC STANDARDS provides the central basis within the United States of a complete and consistent system of physical measurement; coordinates that system with measurement systems of other nations; and furnishes essential services leading to accurate and uniform physical measurements throughout the Nation's scientific community, industry, and commerce. The Institute consists of a Center for Radiation Research, an Office of Measurement Services and the following divisions:

Applied Mathematics — Electricity — Mechanics — Heat — Optical Physics — Nuclear Sciences[2] — Applied Radiation[2] — Quantum Electronics[3] — Electromagnetics[3] — Time and Frequency[3] — Laboratory Astrophysics[3] — Cryogenics[3].

THE INSTITUTE FOR MATERIALS RESEARCH conducts materials research leading to improved methods of measurement, standards, and data on the properties of well-characterized materials needed by industry, commerce, educational institutions, and Government; provides advisory and research services to other Government agencies; and develops, produces, and distributes standard reference materials. The Institute consists of the Office of Standard Reference Materials and the following divisions:

Analytical Chemistry — Polymers — Metallurgy — Inorganic Materials — Reactor Radiation — Physical Chemistry.

THE INSTITUTE FOR APPLIED TECHNOLOGY provides technical services to promote the use of available technology and to facilitate technological innovation in industry and Government; cooperates with public and private organizations leading to the development of technological standards (including mandatory safety standards), codes and methods of test; and provides technical advice and services to Government agencies upon request. The Institute consists of a Center for Building Technology and the following divisions and offices:

Engineering and Product Standards — Weights and Measures — Invention and Innovation — Product Evaluation Technology — Electronic Technology — Technical Analysis — Measurement Engineering — Structures, Materials, and Life Safety[4] — Building Environment[4] — Technical Evaluation and Application[4] — Fire Technology.

THE INSTITUTE FOR COMPUTER SCIENCES AND TECHNOLOGY conducts research and provides technical services designed to aid Government agencies in improving cost effectiveness in the conduct of their programs through the selection, acquisition, and effective utilization of automatic data processing equipment; and serves as the principal focus within the executive branch for the development of Federal standards for automatic data processing equipment, techniques, and computer languages. The Institute consists of the following divisions:

Computer Services — Systems and Software — Computer Systems Engineering — Information Technology.

THE OFFICE FOR INFORMATION PROGRAMS promotes optimum dissemination and accessibility of scientific information generated within NBS and other agencies of the Federal Government; promotes the development of the National Standard Reference Data System and a system of information analysis centers dealing with the broader aspects of the National Measurement System; provides appropriate services to ensure that the NBS staff has optimum accessibility to the scientific information of the world. The Office consists of the following organizational units:

Office of Standard Reference Data — Office of Information Activities — Office of Technical Publications — Library — Office of International Relations.

---

[1] Headquarters and Laboratories at Gaithersburg, Maryland, unless otherwise noted; mailing address Washington, D.C. 20234.
[2] Part of the Center for Radiation Research.
[3] Located at Boulder, Colorado 80302.
[4] Part of the Center for Building Technology.

Multicommodity Network Plotting via Program NETPLT

Zella G. Ruthberg, Gloria R. Bolotsky,

and

William Slater, Jr.

Systems and Software Division
Institute for Computer Sciences and Technology
U.S. National Bureau of Standards
Washington, D.C. 20234

Sponsored by the
Defense Communications Agency
Director, DCEC, Attention R-150
1860 Wiehle Avenue
Reston, Virginia 22090

U.S. DEPARTMENT OF COMMERCE, Frederick B. Dent, *Secretary*

NATIONAL BUREAU OF STANDARDS, Richard W. Roberts, *Director*

Issued June 1974

Library of Congress Catalog Card Number: 74-6000 74

National Bureau of Standards Technical Note 829
Nat. Bur. Stand. (U.S.), Tech. Note 829, 67 pages (June 1974)
CODEN: NBTNAE

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1974

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402
(Order by SD Catalog No. C13.46:829). Price $1.10

PREFACE

This report is being written to satisfy the documentation requirements for transmitting to the Defense Communications Agency a plotting software package developed by NBS for their particular needs. It includes a User's Manual in Part I and a Programmer's Manual in Part II. It was felt that the applicability of this program to plotting any network makes it suitable for more general distribution.

TABLE OF CONTENTS

LIST OF FIGURES

MULTICOMMODITY NETWORK PLOTTING
VIA PROGRAM NETPLT

Zella G. Ruthberg, Gloria R. Bolotsky, William Slater, Jr.**

ABSTRACT

In the design and operation of complex networks, it is often an advantage to obtain a visual representation that readily allows for a quick appraisal of the network's current configuration or of its changed appearance due to variations of its nodes and links. The program NETPLT enables a user to plot the two leading characteristics of any multicommodity network: 1) nodes and links and 2) the multiple source-sink structure (multicommodity property). The unique feature of NETPLT is its unambiguous planar representation of links. NETPLT uses an arc of a circle, instead of the usual straight line, to represent a connector between a node pair (link).

Key Words: Communication network; multicommodity network; network; network display; plotting algorithm; plotting program.

PART I USER'S MANUAL

1.0 GENERAL DESCRIPTION

1.1 Purpose of the User's Manual

The objective of this User's Manual is to provide the user with the information necessary to effectively use the NETPLT program.

1.2 Project Background

The Defense Communications Agency is concerned with various national and world-wide communications networks consisting of rate centers or switches (nodes) and their communication lines (links). In the design and operation of such complex networks it is often an advantage to obtain a visual representation that readily allows for a quick appraisal of the network's current configuration or of its changed appearance due to variations of its nodes and links. Such a need arose in a survivability study* being done for this agency at the National Bureau of Standards. The program NETPLT enables a user to plot the two leading

---

* The results of this study will be published by the National Bureau of Standards in a future Technical Note.

**Formerly summer student employee. Now with the Peace Corps.

characteristics of any multicommodity* network: 1) its nodes and links, and 2) the multiple source-sink structure imposed by the multiple command data** existing in the network. Within the resolution of the plot size used, the resulting network drawing is unambiguous. The manner of illustrating the multiple source-sink information on the network plot is also unambiguous but loses its definition if this data is too dense on any one plot. One must adjust the size of the plot to assure that clarity is not lost. (See Part II, Program Maintenance Manual for algorithms used.)

2.0 SYSTEM SUMMARY

2.1 System Factors

The program NETPLT contains 417 FORTRAN IV statements and has been run on the UNIVAC 1108 at NBS as well as compiled on an IBM 370/155 at the Defense Communication Agency's DCEC facility, Derey Building, Reston, Virginia. It utilizes approximately 6800 words for the program code and the common blocks. It has been designed to accommodate a network of up to 900 nodes and an unrestricted number of links. If the number of nodes is to exceed 900, the five node-related arrays (see PART II, 2.2.1, variables 2, 3, 31, 47, 48) must have their dimensions correspondingly increased. The program expects the system to contain the CalComp software package and utilizes its subroutines PLOT, SYMBOL, NUMBER, and CIRCLE (See Part II, 2.1).

In any one run, the computer used generates a plot-tape that is compatible with its CalComp plotter. This tape drives the plotter in an off-line mode. The plotter at NBS is the CalComp 763, a machine having a continuous roll of 30" width paper. It can accommodate narrower paper widths, if desired. This plotter provides the operator with the option of manually selecting the pen color for any plot. The most commonly used colors are black, red, blue, and green, but others are available.

The program NETPLT can be used on other size CalComp plotter models such as the larger 718 with a 72" x 54" flatbed since the plotter dimensions are parameters of the program. If another model CalComp is used, its software calls should be checked to assure that the computer installation has not varied their parameter meanings.

---

* A multicommodity network has two sets of node sets, $S=s_1, \ldots s_q$ and $T=t_1, \ldots t_q$, such that the node set pair $[s_j, t_j]$ is associated with the transfer in the network of a distinct commodity j from sources s_j to sinks t_j. See [1], pg. 2, for more precise definitions in the subject area.

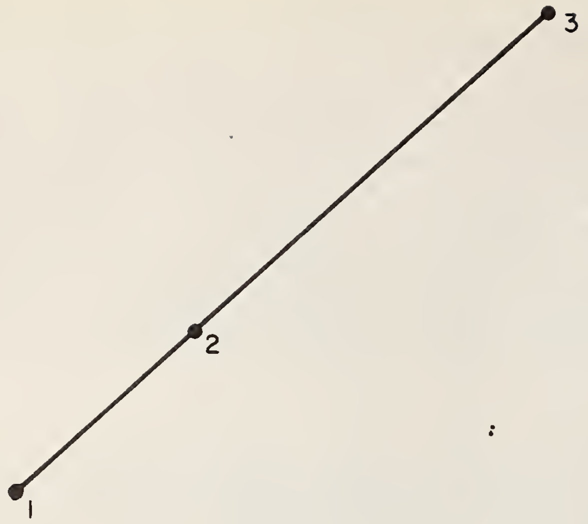
** In a military communications network each commander has a distinct commodity, his command's messages, and a distinct set of subordinates. The network connection nodes (homings) of each commander and his subordinates determine the sets of source and sink nodes for that command (commodity).

2.2 Rationale of NETPLT

2.2.1 Plotting the Network: The network is specified by its nodes (rate centers or switches) and its links (connected node pairs). The information needed by the program to uniquely locate a node on a plot is an identification code and a pair of x, y coordinates on some kind of rectangular grid. Link locations are then uniquely specified by a node code pair. However, the planar representation of such a link can be ambiguous when straight lines are used for such node pair connectors (see Figure 1). The program NETPLT eliminates this problem of ambiguity by using an arc of a circle as the planar representation of a connector between a node pair.

The uniqueness of this representation stems from the following considerations. The straight line drawn between a node pair can be considered as the chord of a circle subtending an angle δ at the circle's center (see Figure 2). For each node pair and angle δ there are two and only two circles possible. To reduce this number of possible arcs from two to one, the program was designed to draw all arcs with either a 'downward' or an 'upward' curvature. An arc has a 'downward' curvature when the radius to its center forms an angle with the horizontal ranging from greater than 0° through 180°. It has an 'upward' curvature when that angle ranges from greater than 180° through 360° (see Figure 3a). Figure 3b illustrates the range of arcs in a downward and an upward arc plot in relation to the angle α. Note that the angle α of the arc's chord with the horizontal covers the first and fourth quadrants of the plane for downward arcs and the second and third quadrants for upward arcs. The overlap at the boundary between these two regions in the plane was resolved by assigning $\alpha = +90°$ to the downward arc plots and $\alpha = -90°$ to the upward arc plots. A downward arc plot always draws the arc a in Figure 2 while an upward arc plot draws arc b. Figure 5 shows a simple network drawn first with downward arcs and then with upward arcs. Additional links between a given node pair are drawn by increasing the angle δ and thus changing the circle used. The limiting arc is the semicircle whose diameter is the chord drawn between the two nodes in question (see Figure 4). For reasons of resolution, a fixed interval between multiple links is specified in the program. This limits the number of multiple links that can be drawn. The network plot begins to lose some of its clarity if many widely separated node pairs are highly interconnected. (See PART II, 1.2 for the description of the geometry used.)

2.2.2 Plotting the Multiple Command Data: Since a network drawing can be expected to be a somewhat complex figure in itself, it was decided to indicate the multiple command information by means of points located alongside the border and connected via straight lines to the relevant network nodes. That is, each commander and each of his subordinates is represented by a point outside the network plot border. This point is then connected via straight lines to the network nodes that are its homings. Note that the program allows for up to five connection nodes



Is this a picture of:

a) Link 1-3 and 1-2

b) or Link 1-2 and 2-3

c) or Link 2-3 and 1-3

d) or Link 1-2 and 2-3 and 1-3

?

Figure 1. Ambiguity of Straight Lines for Links

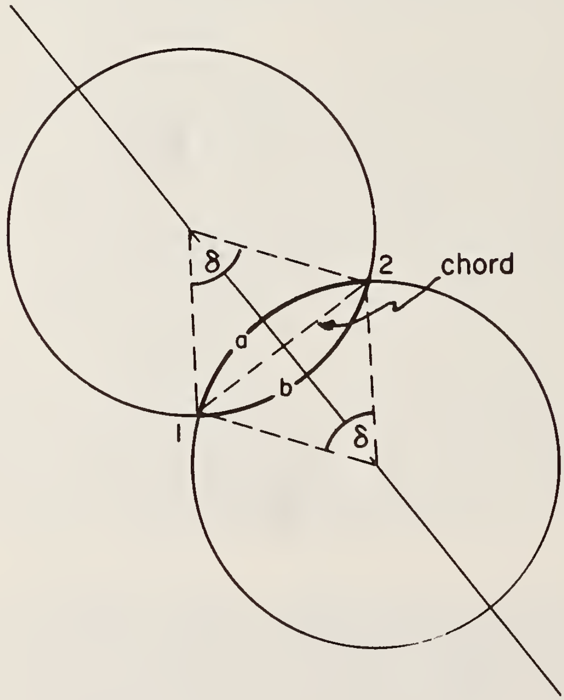


Figure 2. The Two Circles for Node Pair 1-2 that Subtend Angle δ.

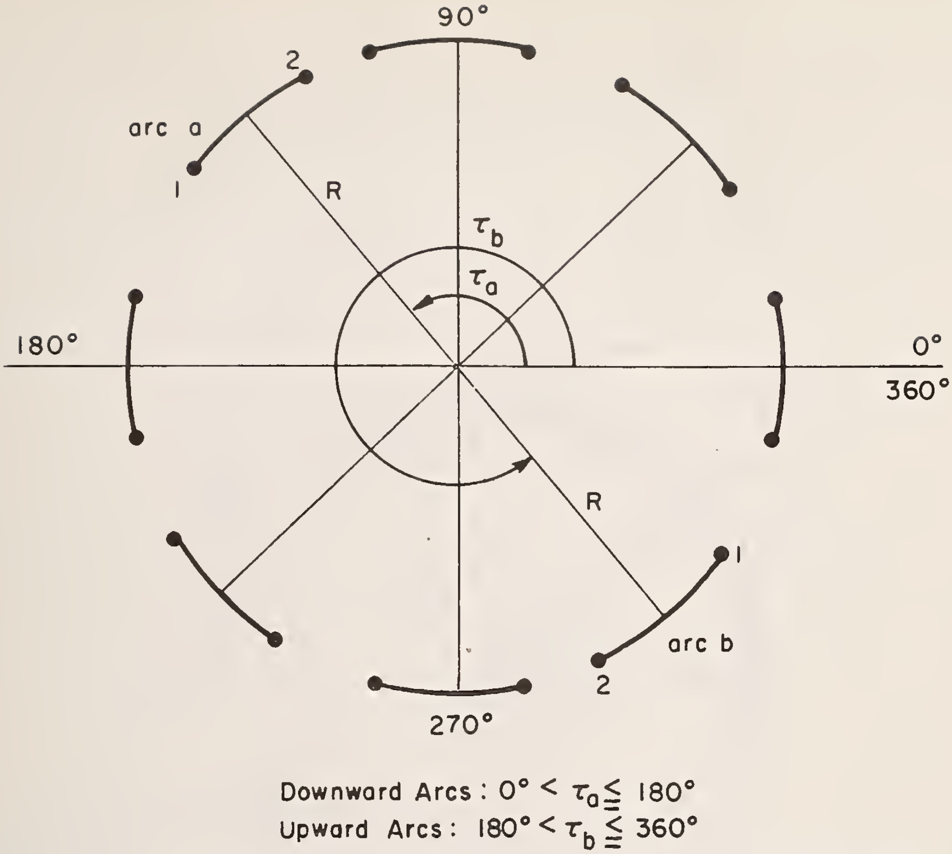


Figure 3a. Possible Orientations For an Arc Having a Radius R. Radius Points Down for "Downward Arcs. Radius Point Up for "Upward" Arcs.

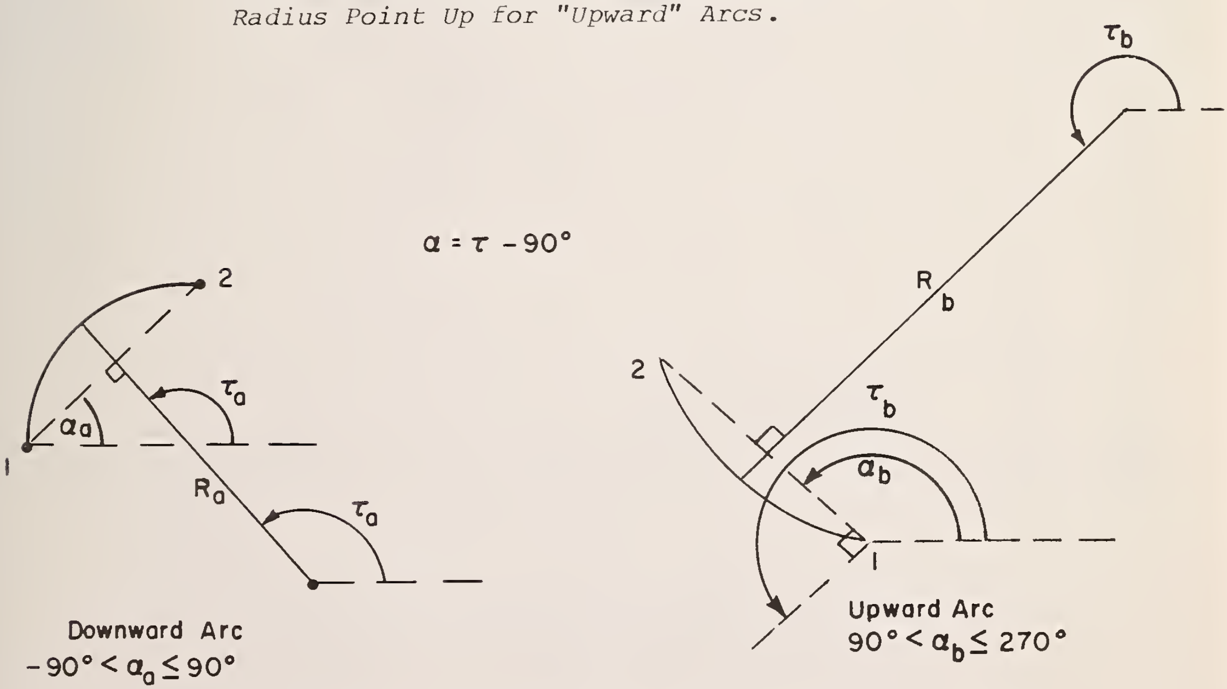


Figure 3b. Relation of Downward and Upward Arcs to Angle α.

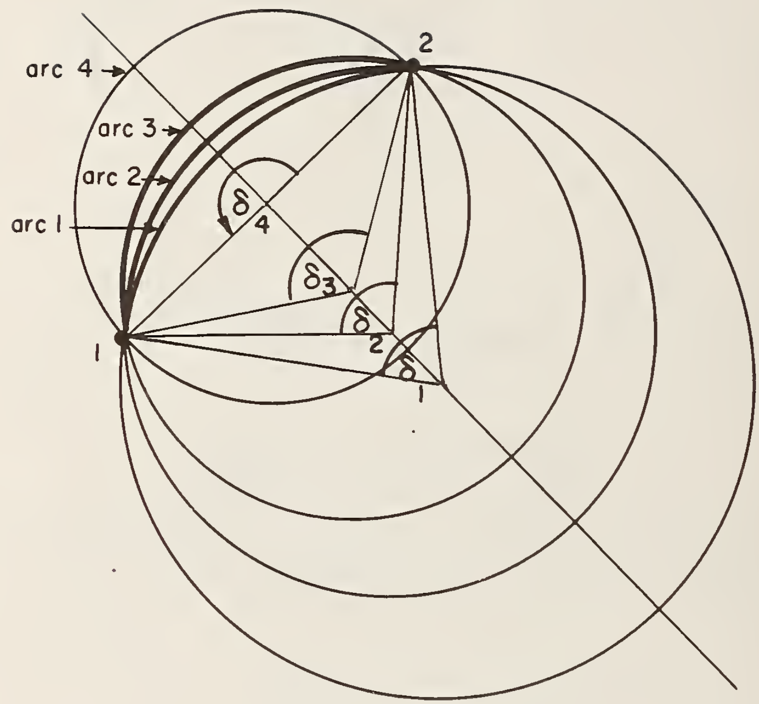


Figure 4. Primary and Secondary Arcs Between Nodes 1 and 2 -- Downward Arc Plot

TEST NETWORK--DOWNWARD ARCS

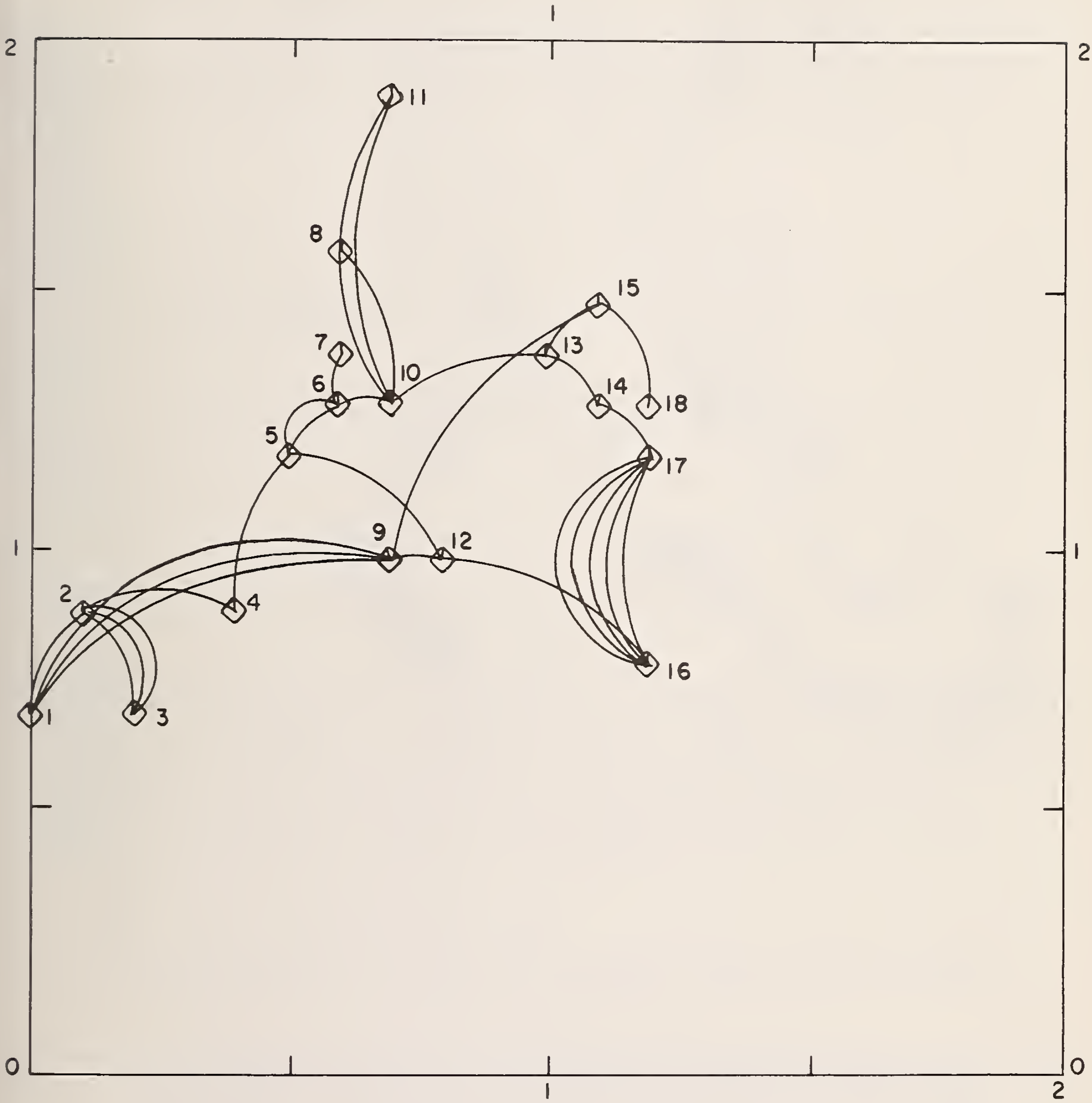


Figure 5a. Downward Arc Plot of Network in Figure 5b. (Type 2 Plot)

TEST NETWORK--UPWARD ARCS

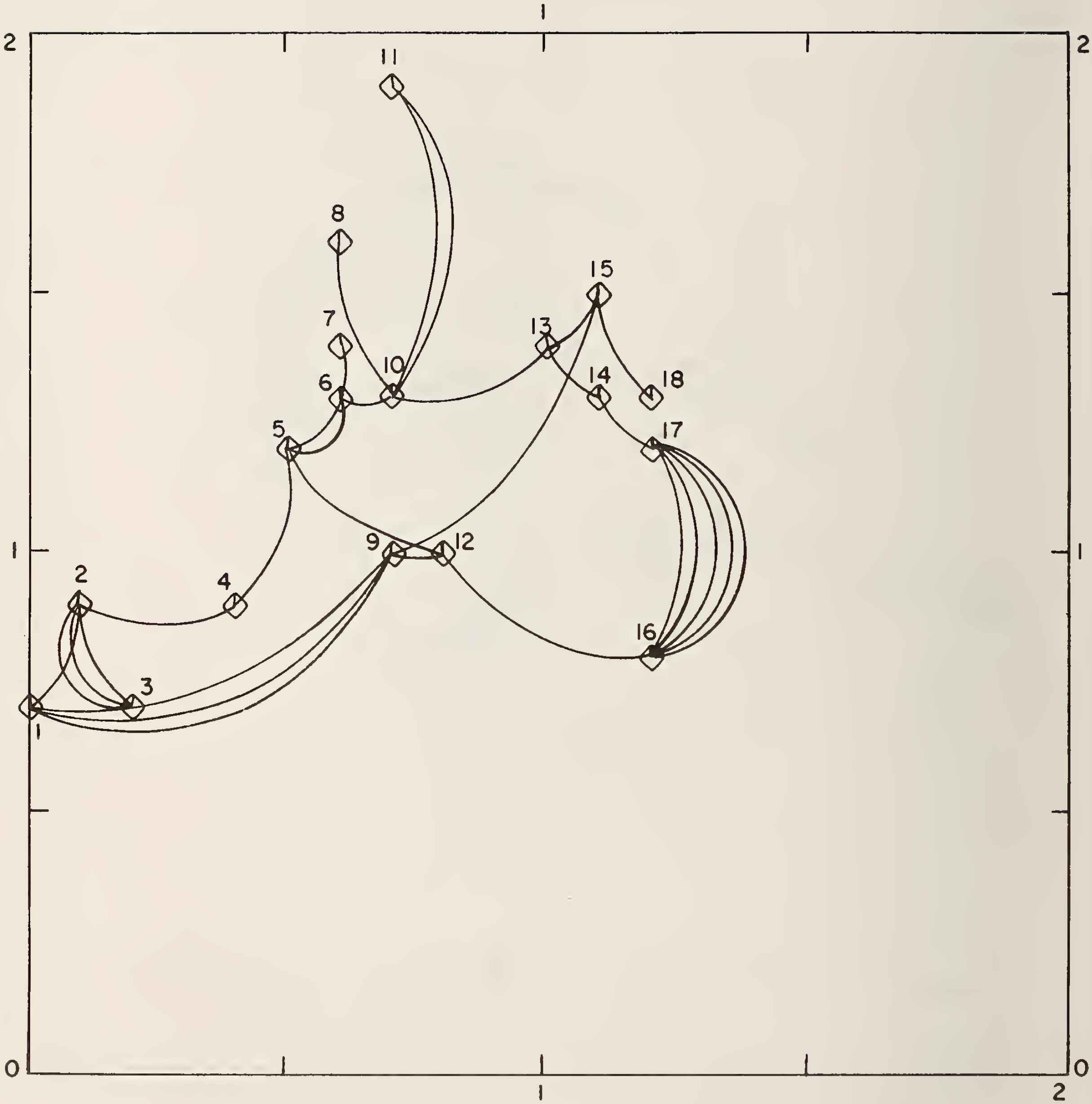


Figure 5b. Upward Arc Plot of Network in Figure 5a. (Type 2 Plot)

per subscriber (commander or subordinate). A commander code number and a subordinate code number are used as a label near these exterior border points to inform the user as to which subscriber is involved (see Figure 6).

In order to minimize the lengths of these lines from the border into the network, a sort was evolved which locates these border points as closely as possible to their corresponding network nodes. This sort also distributes the subscriber points evenly around the border using this minimum distance concept. (See Appendix D for listing.)

2.3 Options of NETPLT

This program allows the user two basic plot options:

1) Draw the network and its subscriber (commander/subordinate) data on one sheet

or 2) Draw the network on one sheet and its subscriber information on one or more additional transparent overlays.

This second option is itself split into two options where either

a) All subscriber information is plotted on a single overlay

or b) Subscriber information for five commanders at a time is plotted on a single overlay.

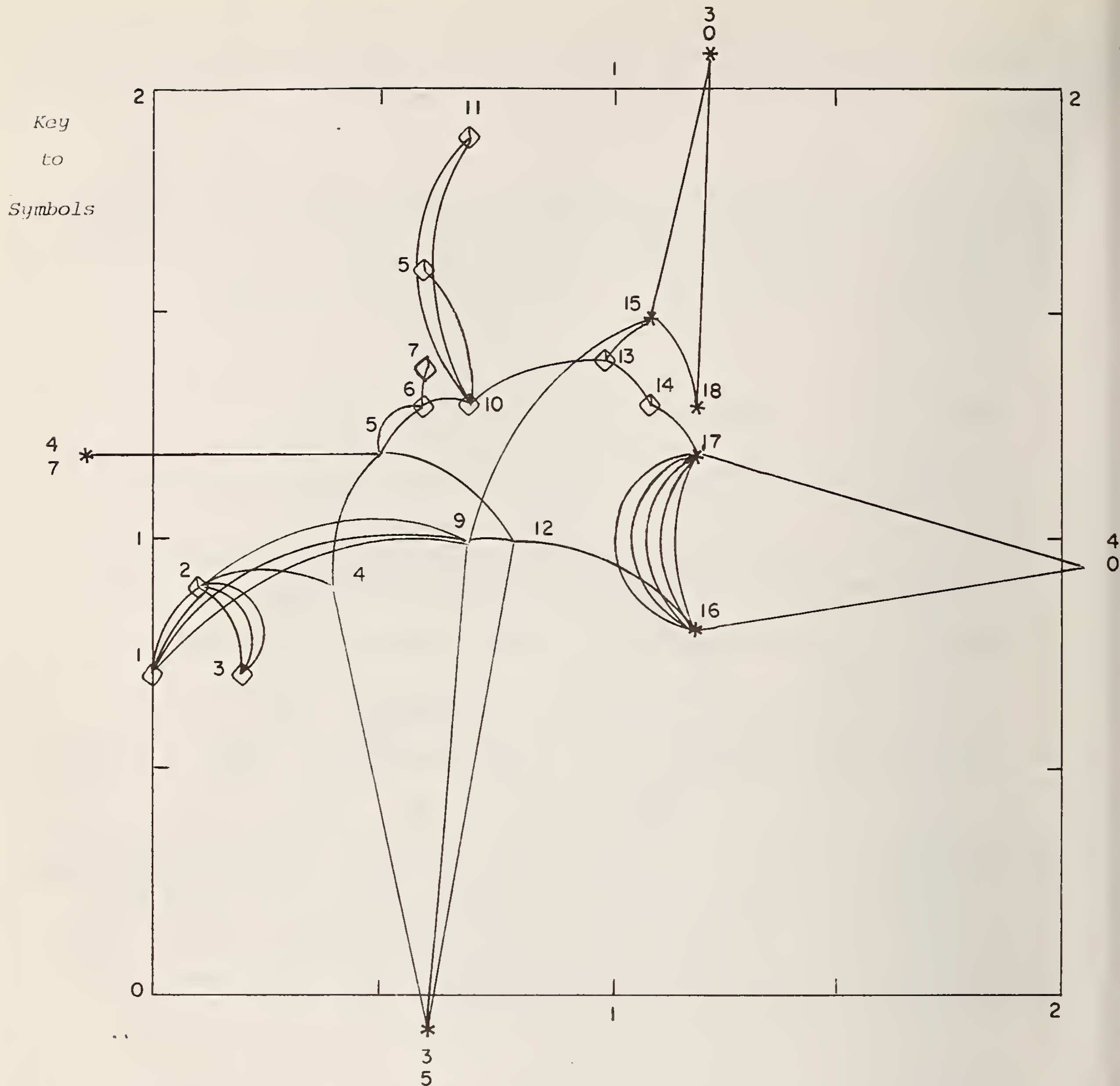
By properly setting up the data deck, the 2a option can be used to plot only commander homings (connection nodes) and the 2b option to plot only subordinate homings (connection nodes). Note that for any one set of plots, all subscriber information is sorted at one time around the border so that each subscriber is assigned a unique location. (See Appendix D for the listing of this pre-sort.) This allows overlays to be viewed together as well as separately.

Some further general options are:

3) The plotter dimensions are parameters that can be changed by the user.

4) The dimensions of the total plot are parameters that can be varied by the user within the constraints of the plotter dimensions. Note that all plots in a single run have the same size, however.

5) The content of the heading on each plot must be composed by the user. (See PART I, 3.1.2, card 4.)



1) Network has 18 nodes, 30 links.

2) Command 3 enters at nodes 15 and 18. His Subordinate 5 exits at nodes 4, 9, and 12.

3) Command 4 enters at nodes 16 and 17. His subordinate 7 exits at node 5.

Figure 6. Plot Type 1 - Network Plus Subscribers

The program also contains the following four network options:

6) A downward or upward arc plot must be specified by the user.

7) One of the two symbols* provided by the program must be specified by the user to mark each network node's position.

8) A single arc or multiple arcs must be specified for each network link.

9) The node x, y coordinates may be in any units providing they are all in the same units. The program converts these to inches on the plotting paper.

2.4 Performance Experience

Following numerous test plots, the program has thus far been applied to two large networks: 1) a fictitious, sparsely** connected, 844 node, 1059 link network having 23 commanders with a total of 57 entries and 634 exits (comparable to national networks used by DCA) and 2) an 86 node, 761 link version of the world-wide Autovon network. The model 763 CalComp Plotter was used to generate plots on 60" x 30" sheets (paper or transparent matte triacetate).

2.4.1 Sparsely Connected Network: The fictitious sparsely connected national network was plotted on one sheet of matte triacetate, the commander structure on a second sheet of the same, and the subordinate structure on five subsequent sheets of that same material. The network was drawn in black ink, the commander plot in red ink, and the subordinate plot in blue ink since this plotter allows the operator a manual choice of pen colors before executing a plot. The nodes of the network were pre-sorted (see PART I, 2.6) monotonically increasing on x and alternately increasing and decreasing on y (for speedier plotting) and then given a sequential integer identification code (1 to 844). Ordinary nodes used one symbol while homing nodes used a second one. The links were pre-sorted so that the pen traced continuous paths through the network when drawing the links. Whenever a path terminated, the pen backtracked in a raised position to the last node with an untraced branch. This node was then chosen as the new starting point. (See Appendix D for listing of the sort.)

The input units for node data were on a 12 x 9 grid while the actual plot was on a 48" x 27" grid on the paper. Also the commanders

---

* The two symbols used by the program are * and ◇.

** The number of links in a network of n nodes can vary from a minimum of (n-1) to a maximum of $\frac{n(n-1)}{2}$. A measure of the sparseness of a network is the ratio # of links/# of nodes. If this number is in the vicinity of 1 to 2 the network is considered sparsely connected.

and their subordinates were identified by an integer code. The entire set of plots required several hours for execution on the CalComp plotter. The computer run that generated the plot tape used about two minutes cpu time on the UNIVAC 1108 (EXEC II). (See Figure B1 for the network and some of its overlays.)

2.4.2 Densely Connected Network: The denser Autovon network required a few modifications to accommodate its idiosyncracies. Since there were three types of nodes, provision was made to increase the number of node symbols available to the program from two to three. Also, since the node identification code was alphabetic, this information was read and printed alphanumerically rather than in integer mode. Finally, since the links in this highly connected network were all of a multiple type, coding was added that printed this multiplicity number at the center of each arc. The data for this application was pre-processed so as to check for a connected network and for faulty or duplicate link entries. The network nodes and links were then pre-sorted in the same manner as the sparse network of 2.4.1 for speedier plotting. In this case, however, the input node data was converted from a latitude, longitude grid to a 60" x 27" plot on the paper. (See Figure B2 for this network.)

2.5 The Data Base

The bulk of the data for a run consists of the node set, the link set, and the subscriber set. The node set, which contains one card per node, has on that card the node identification code, the x, y coordinates of that node, and the type. (See PART I, 3.1.2, card 5.) The link set, which also has one card per link, contains the link's two incident node identification codes and its multiplicity.* (See PART I, 3.1.2, card 6) The subscriber set has two cards per subscriber (commander or subordinate) with the first card containing the number of homings and the second card containing the node identification codes of these homings. (See PART I, 3.1.2, cards 10 and 11.)

2.6 Preprocessing

The program plots the node set, the link set, and the subscriber set in the order in which they are read in. Hence, for speedier plotting, each of these sets of data is preprocessed via an appropriate sort. The nodes are sorted in narrow bands of monotonically increasing x values with the y values alternately increasing and decreasing in successive bands. The links are sorted into continuous chains through the network. Whenever a chain reaches an end point, the next chain starts with the last traced node with an untraced branch. The subscriber data for a run is sorted to minimize the distance from the border points to its homings. These are then spaced evenly around the

---

* The multiplicity of a link is the number of distinct connectors between its two incident nodes.

border with a minimum distance required between subscriber locations for best resolution. Since the natural distribution of the points around the border is in general uneven, manual adjustments are made before the subscribers are given their final coordinate values. (See Appendix D for the sort listings.)

2.7 Inputs and Outputs, General

2.7.1 Submitting a Run: The FORTRAN IV program must be submitted with the proper system control cards for the computer being used and with the appropriate data deck. Figure 7 shows the run deck and Figure 8 the data deck. The output of such a run is a printout of the network data (See Appendix A) and a plot-tape for the CalComp plotter. The program can readily be put on tape or disc if this offers advantages in its use.

2.7.2 Requesting a Plot: The format for requesting the off-line plot by the CalComp plotter varies from installation to installation. However, the most pertinent information that may be needed is the plot-tape identification code, the size and type paper needed, the color pen desired for each plot, the number of plots on the plot-tape, and an estimate of the time required for the plot.

3.0 INPUTS AND OUTPUTS, SPECIFIC

3.1 The Data Deck

The data deck consists of two basic sets of cards with the second set repeated for each plot in the run. Figure 8 shows a schematic drawing of a data deck and the following describes it in detail. The formats of the individual card types are shown in Figure 9 and their contents are described below.

3.1.1 Set One: This set contains information common to the four types of plots.

Card 1: Contains the number of nodes, the number of links, and the kind of curvature (up or down) desired. (See ICURV in PART II, 2.2.1.) If a run contains only subscriber plots, one can set the number of links and the curvature parameter to 0.

Card 2: Contains the number of plots, the length (in inches) of the x and y axes, the length (in input units) of the x and y axes, the length (in inches) of the x and y plotter paper dimensions.*

** The program allows for the use of any input units derived from a rectangular coordinate system.*

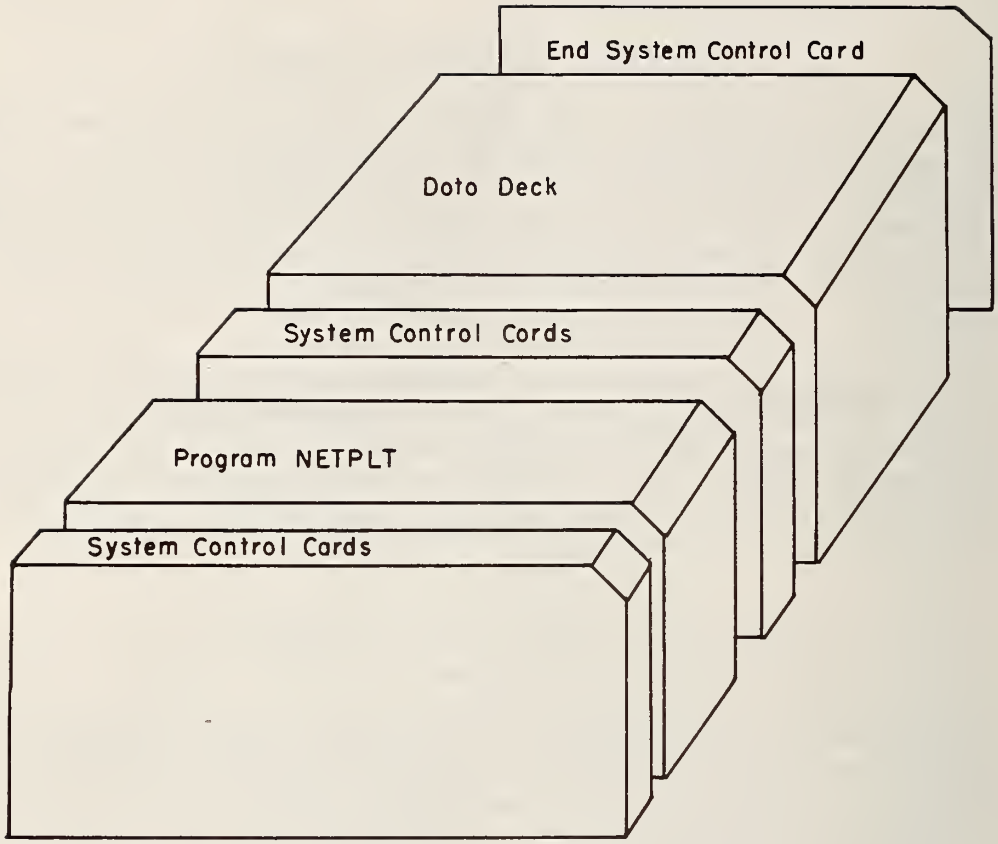
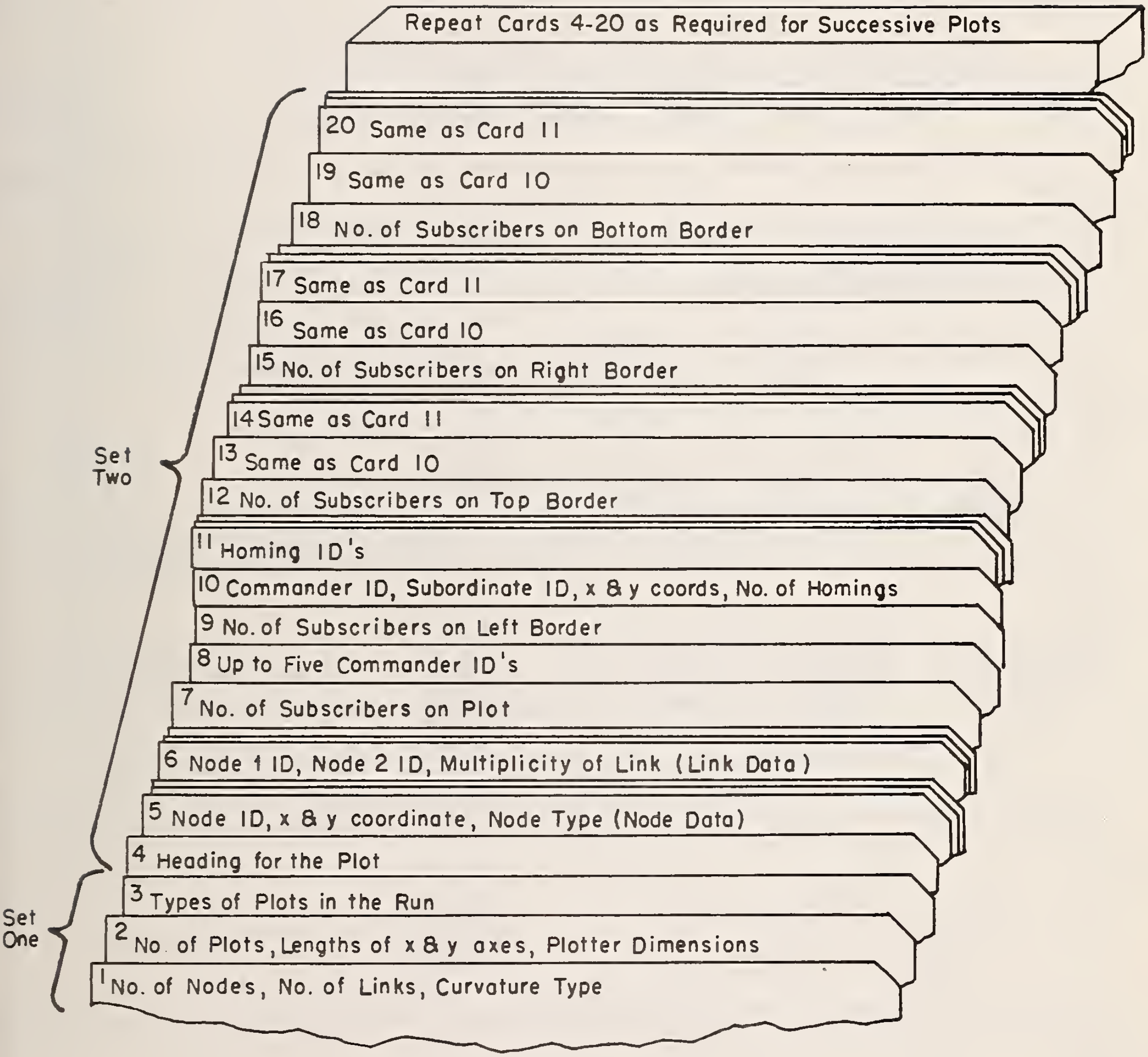


Figure 7. Deck for a Run of NETPLT.



Note: Cards 5, 6, 7, 9-20 are generated by the preprocessing routines.

Figure 8. The Data Deck.

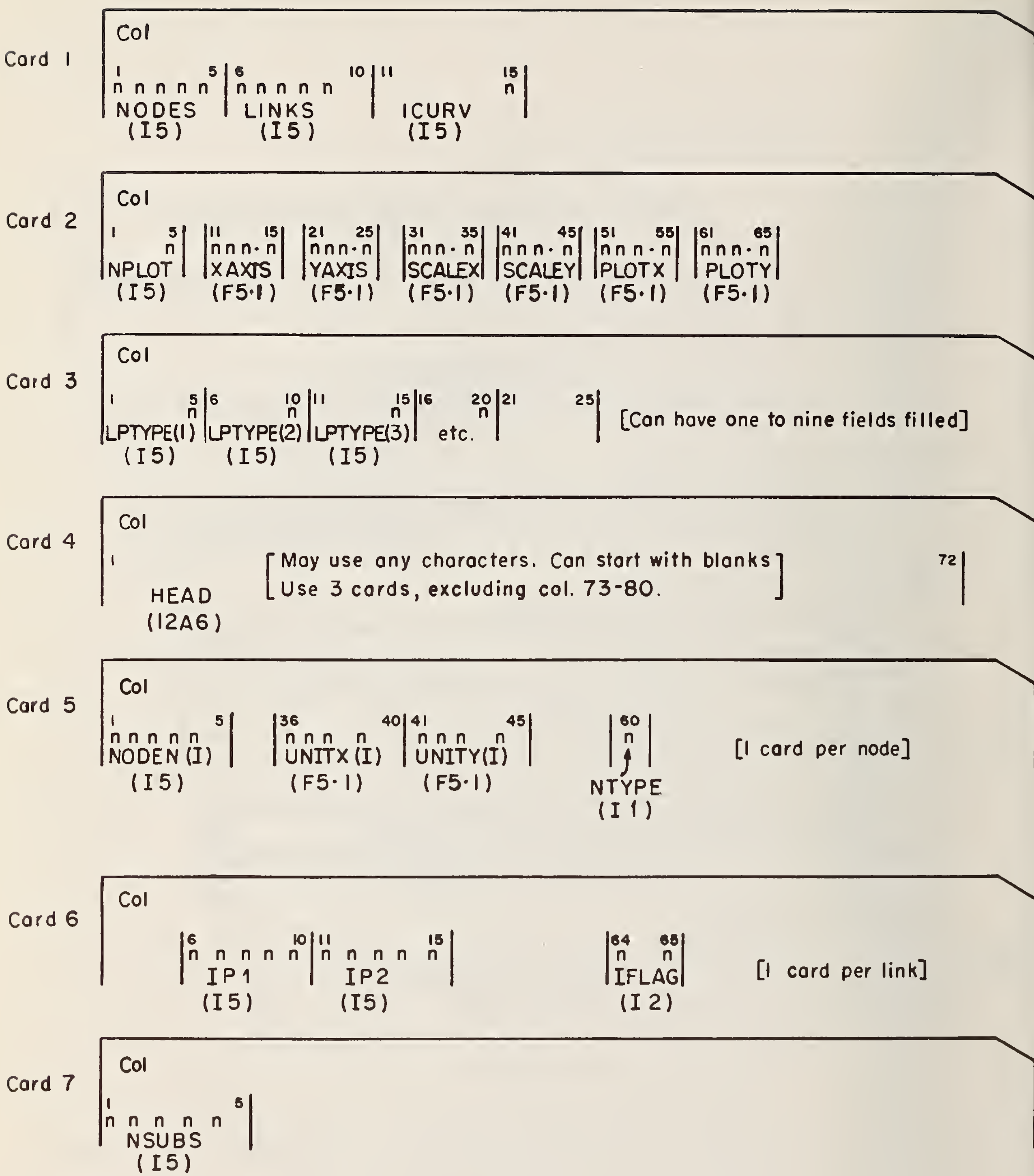


Figure 9a. The Data Deck Card Content

Card 8

Col	1–5	6–10	11–15	16–20	21–25
	nnnnn	nnnnn	nnnnn	nnnnn	nnnnn
	NCD(1)	NCD(2)	NCD(3)	NCD(4)	NCD(5)
	(I5)	(I5)	(I5)	(I5)	(I5)

[Can have one to five fields filled]

Card 9

Col	1–5
	nnnnn
	K
	(I5)

Card 10

Col	1–5	6–10	36–45	46–55	56–60
	nnnnn	nnnnn	nnnn.nnnnn	nnnn.nnnnn	n
	NCMD	NSUB	XUNIT	YUNIT	NHOME
	(I5)	(I5)	(F10.5)	(F10.5)	(I5)

Card 11

Col	1–5	6–10	11–15	16–20	21–25
	nnnnn	nnnnn	nnnnn	nnnnn	nnnnn
	NHM(1)	NHM(2)	NHM(3)	NHM(4)	NHM(5)
	(I5)	(I5)	(I5)	(I5)	(I5)

[Can have one to five fields filled]

Cards 12, 13, 14; 15, 16, 17; and 18, 19, 20 repeat cards 9, 10, 11.

Note: Each field on the card contains its column range, its integer content (denoted by n's), its variable name, and its format. All numbers are right justified. The values possible for any n are blank, 0, 1, 2, ... 9. Decimal numbers have the decimal point shown.

Figure 9b. The Data Deck Card Contents (cont'd.)

Card 3: Contains the types of plots in the run. (See LPTYPE in Part II, 2.2.1.)

3.1.2 Set Two: This set consists of the specific information needed for each plot. Not all card types appear for each type plot (LPTYPE). Therefore, each card now described has a note indicating the plot type.

Card 4: (For all plot types.) Contains the heading for the plot. It can be in alphanumeric form and consists of a maximum of 216 characters on 3 cards, 72 characters per card. These 216 characters are centered at the top of the plot.

Card 5: (For the first plot in the run only.) Contains the node data, one card per node. Each card carries the node identification code, the x coordinate (in input units), the y coordinate (in input units), and the node type. (See NTYPE in PART II, 2.2.1.)

Card 6: (For plot types 1 or 2 only.) Contains the link data, one card per link. Each card consists of the identification code of node 1, the identification code of node 2, and the multiplicity of the link.

Card 7: (For plot types 1, 3, or 4 only.) Contains the number of subscribers for the current plot.

Card 8: (For plot type 4 only.) Contains the identification code numbers of up to five commanders for a plot.

Card 9: (For plot types 1, 3, or 4 only.) Contains the number of subscriber points along the left border, starting at the lower left corner and ending at the upper left corner.

Cards 10 and 11: (For plot types 1, 3, or 4.) Contain the subscriber data, one pair of cards 10 and 11 per subscriber on left border.

Card 10: Contains the commander identification code, the subordinate identification code, the x and y coordinates (in inches) of the subscriber points, and the number of homings.

Card 11: Contains the identification codes of the homings for the subscriber in Card 10.

Cards 12, 13, and 14: Repeat cards 9, 10, and 11 for subscribers along the top border, from left to right.

Cards 15, 16, and 17: Repeat cards 9, 10, and 11 for subscribers along the right border, from top to bottom.

Cards 18, 19, and 20: Repeat cards 9, 10, and 11 for subscribers along the bottom border, from right to left.

The data for the first plot in a run contains the appropriate card types selected from 1 to 20 and always includes node data (card 5). The data for successive plots in a run contain the appropriate card types from 4 to 20, omitting the node and link data (cards 5 and 6).

3.2 NETPLT Outputs

The outputs of the NETPLT program consist of the computer printout, the plot-tape for the CalComp plotter, and the plots drawn by the CalComp plotter.

3.2.1 The Computer Printout: The computer printout always contains a network heading and a node table. For plot types 1 or 2, the link table appears next. Finally, for plot types 1, 3 or 4, the subscriber table appears. Note that each subscriber plot in a run has its own heading and this heading precedes the subscriber table in the printout. See Appendix A for sample.

The node table contains a line of printout for each node. On this line, from left to right, are the node identification code number, its x and y coordinates in input units, its x and y coordinates in inches, and its type (NTYPE). Type 0 is an ordinary node while type 1 is a homing node. (See Appendix A.1 for sample printout.)

The link table contains a line of printout for each link. On this line, from left to right, are the link identification code number, its node 1 code number, its node 2 code number, the multiplicity of that link, and the number of links plotted. Whenever a secondary link is plotted between two nodes, an additional line of printout appears, preceding the primary link line in the table. It lists the link counter, starting at one, and its angles PHI, T1, and T2. (See PART II, 2.2.3 for definitions.) This secondary link printout is included for checking purposes since these angles can be hand calculated. (See Appendix A.2 for sample printout.)

Each subscriber table contains a heading (unless the plot is of type 1) and a line of information per subscriber. On this line is the commander code number, the subordinate code number, the x and y coordinates (in input units) of the border node representing it, these same coordinates in inches, the number of homings, and the code numbers of these homings (one to five of them allowed). (See Appendix A.3 for sample printout.)

3.3 CalComp Output

The plot tape contains the instructions for the CalComp plotter and must be submitted for plotting according to the installation's *procedures. (See PART I, 2.7.2.) Each plot proceeds with the placement of the heading at the top and the drawing of a border with the specified dimensions. Tic marks are drawn at half input unit intervals with their integer numerical values drawn at the whole input unit intervals. The specific plot, which can be one of four types, is then drawn. Type 1 plots the network and all subscribers on one sheet, type 2 plots only the network on the sheet, type 3 plots only subscribers (or commanders) on one sheet, and type 4 plots subscribers or subordinates, up to five commands at a time. Finally, each plot has an appropriate key to the node symbols on that plot in the upper left corner of the paper. For examples of the plots, see Figures 5 and 6 and Appendix B. For further details about the key, see Part II, 2.3.1.*

PART II PROGRAM MAINTENANCE MANUAL

1.0 GENERAL DESCRIPTION

1.1 Purpose

The objective of this manual is to provide maintenance programmer personnel with the information necessary to effectively maintain or modify the program NETPLT.

1.2 The Basic Algorithms

Peripheral calculations such as the conversion factors in the x and y directions from node coordinates to inches, the drawing of the border, and the annotation for the border and key to symbols, are straightforward and require little explanation. However, the heart of this program is based on some circle geometry which is more involved and will now be discussed. As mentioned in PART I, 2.2.1, every link can be represented by the arc of one of two circles. Figure 10 shows the geometry applicable to both the downward and upward arc cases. Note that x_1 and x_2 are chosen so that $x_1 \geq x_2$ for all downward arcs and $x_1 \leq x_2$ for all upward arcs.

The range for angle δ found most suitable in tests was 30° to 60°, but the value finally used in the two large networks was 60°. The information needed by the CIRCLE routine in CalComp software is β_1 (the angle with the horizontal of node 1's circle radius), β_2 (the angle with the horizontal of node 2's circle radius), and R (the radius of the circle whose arc is being drawn). These quantities are calculated as follows:

$$\gamma = \frac{\delta}{2} \qquad (1.2.1)$$

$$\omega = 90° - \gamma \qquad (1.2.2)$$

$$L = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \qquad (1.2.3)$$

$$\alpha = \arc \sin \frac{y_1 - y_2}{L} \text{ (downward arcs)} \qquad (1.2.4.1)$$

$$\alpha = \arc \sin \frac{y_2 - y_1}{L} + 180° \text{ (upward arcs)} \qquad (1.2.4.2)$$

$$R = \frac{L}{2 \cos \omega} \qquad (1.2.5)$$

$$\beta_1 = \alpha + \omega \qquad (1.2.6)$$

$$\beta_2 = \beta_1 + \delta \qquad (1.2.7)$$

Note that the case $x_1 = x_2$ is made unique by arbitrarily assigning $\alpha = +90°$ to downward arc plots and $\alpha = -90°$ to the upward arc plots.

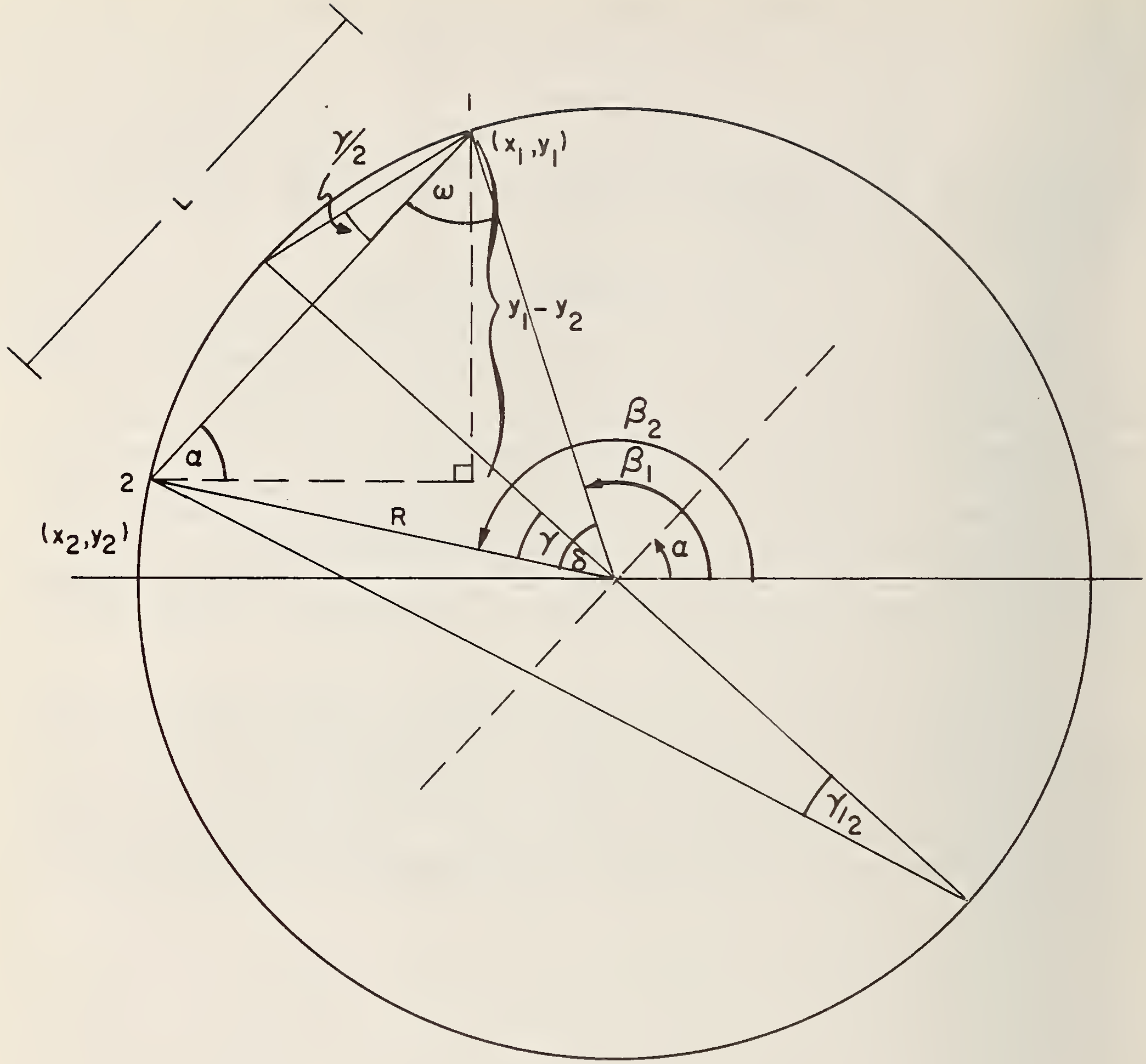


Figure 10a. Geometry for Primary Arcs (Downward)

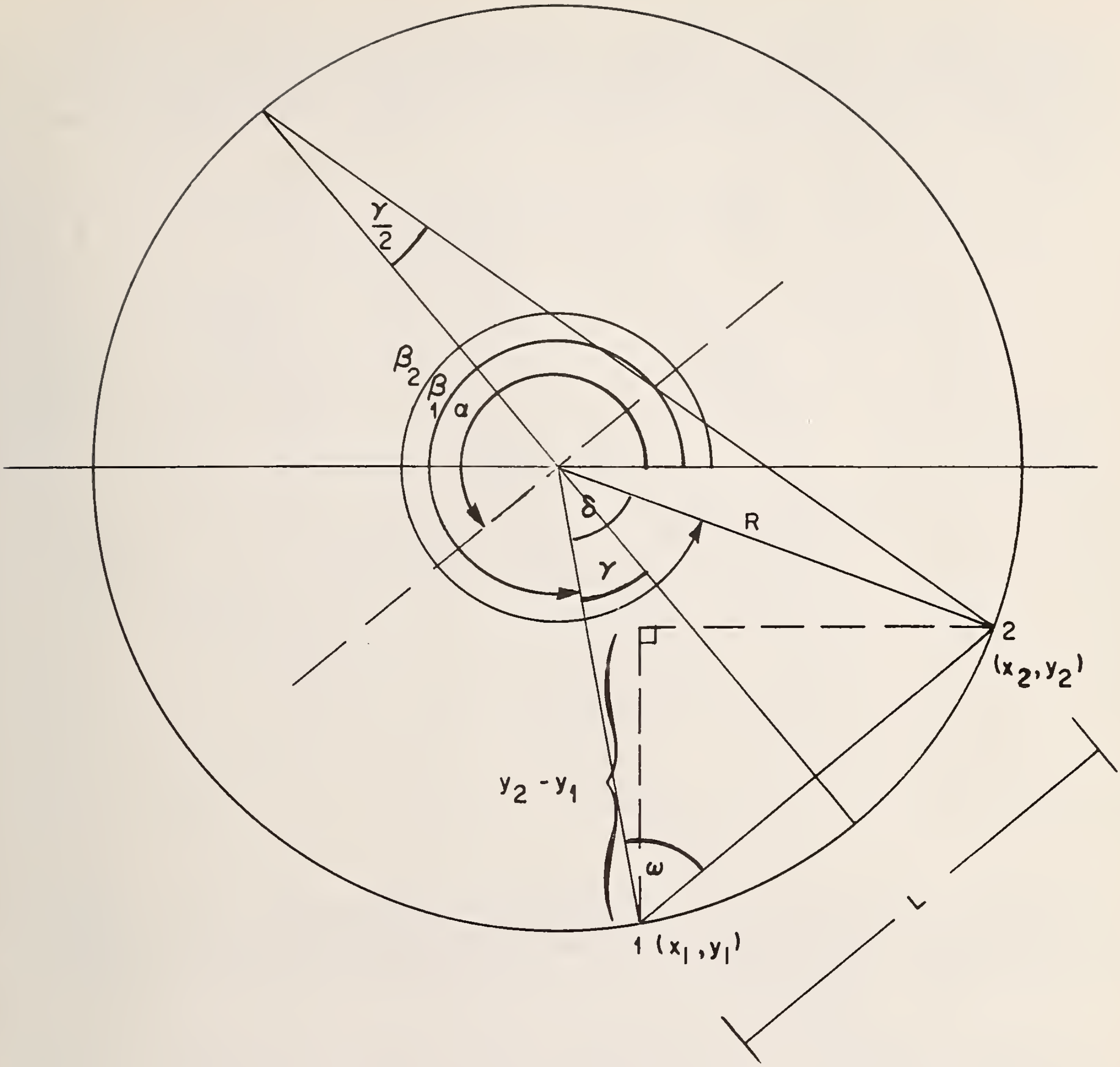


Figure 10b. Geometry for Primary Arcs (Upward).

Secondary arcs are drawn using the geometry of Figure 11. The value chosen for D, the distance between arcs, was set at .1 inches for reasonable resolution. The quantities needed by the CalComp CIRCLE routine for the secondary arc are Θ_1, Θ_2, and R . The equations used are:

$$m = \frac{L}{2} * \tan \frac{\gamma}{2} \qquad (1.2.8)$$

$$n = D + m \qquad (1.2.9)$$

$$*R_2 = \frac{n^2 + (\frac{L}{2})^2}{2 \cdot n} \qquad (1.2.10)$$

$$\phi = \arcsin \frac{L}{2 \cdot R_2} \qquad (1.2.11)$$

$$\Theta_1 = \beta_1 + \gamma - \phi \qquad (1.2.12)$$

$$\Theta_2 = \Theta_1 + 2\phi \qquad (1.2.13)$$

It should be noted that all trigonometric function subroutines using these angles require them in radians while the CalComp Routine calls use degrees. The program does the proper conversion where necessary.

A flag (IDIR) is introduced to indicate the direction, clockwise or counter clockwise, for drawing an arc. Thus, the plotter pen will draw a series of arcs -- multiple arcs or arcs in a path -- with a minimum amount of movement. (See Figure 12.)

2.0 SYSTEM DESCRIPTION

2.1 CalComp Subroutines Used

The program NETPLT uses four subroutines, PLOT, SYMBOL, NUMBER, and CIRCLE, from the CalComp software package. A brief description of these routines will be given here, but more detailed information can be found in references [2] and [3].

---

* *Using Figure 11, we can see that* $R_2^2 = (\frac{L}{2})^2 + (R_2 - n)^2$. *Solving for* R_2 *yields equations 1.2.10.*

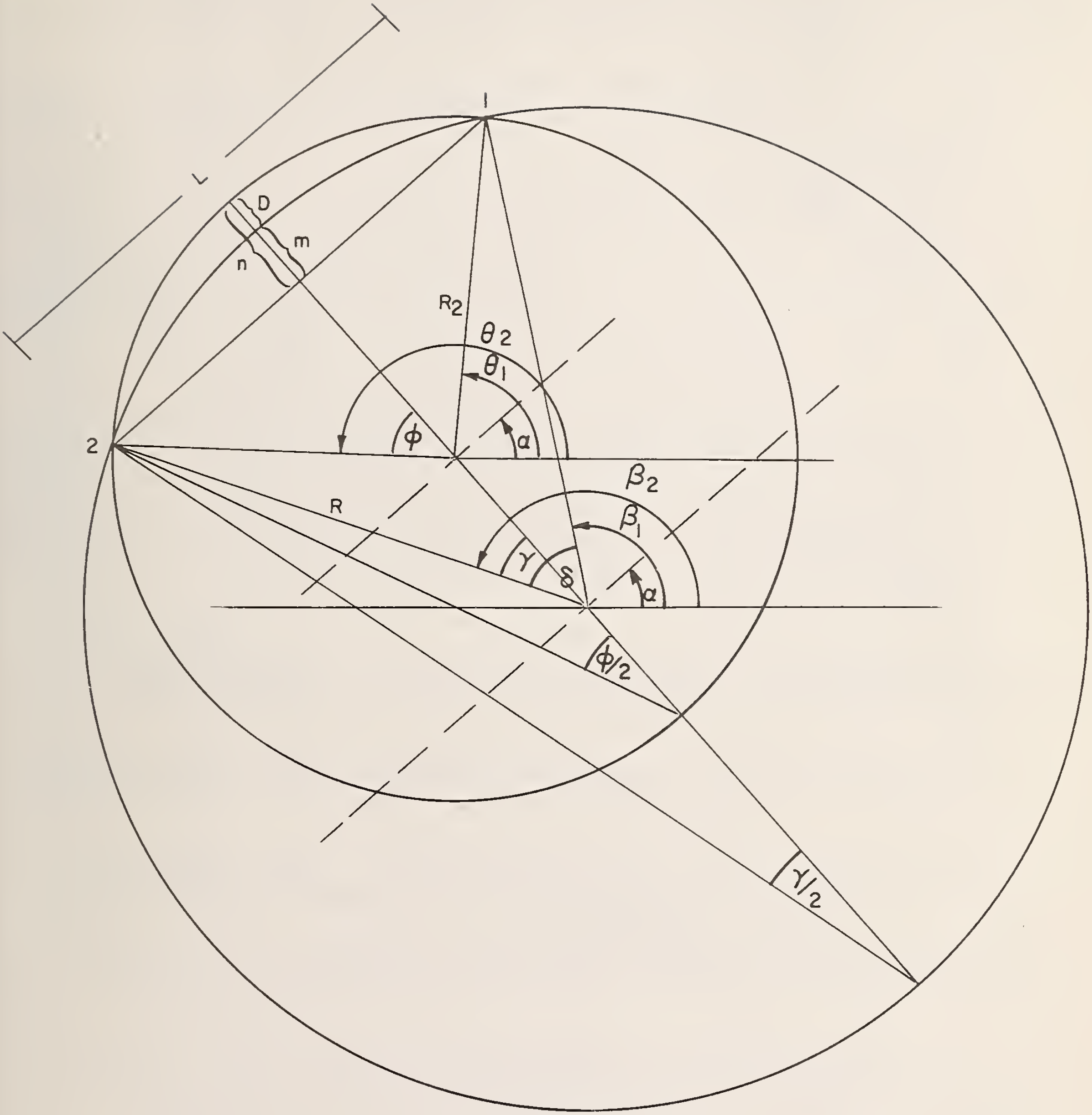


Figure 11. Geometry for Secondary Arcs (Downward) (Note: For upward arcs reverse points 1 and 2 as in Figure 10b.)

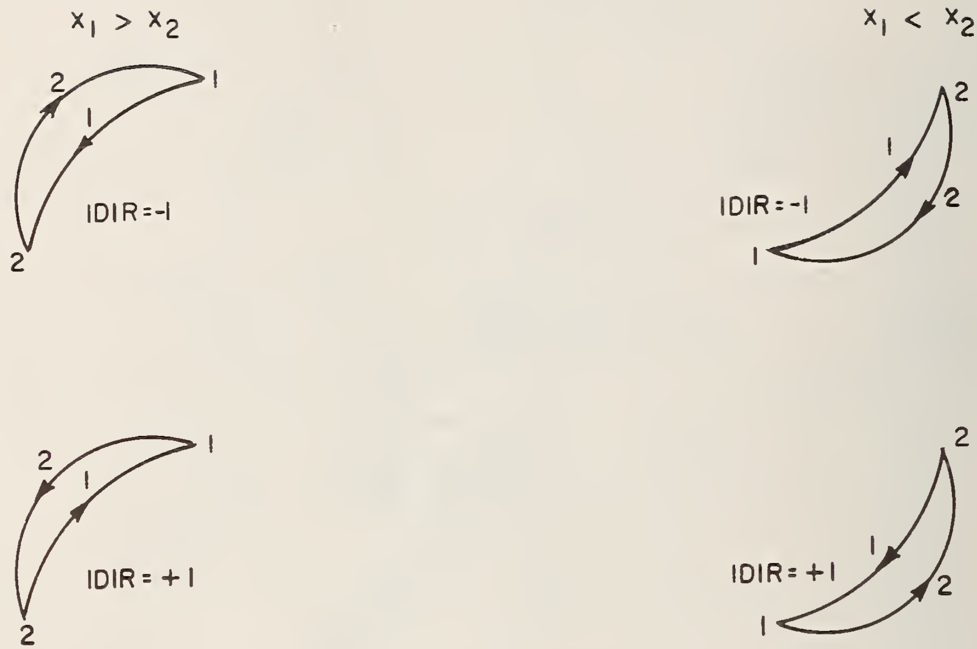


Figure 12. The Four Possible Ways to Start Drawing Multiple Arcs. Draw Arc 1 and Then Arc 2.

2.1.1 PLOT: This routine converts all plotting data to hardware codes for a pen plotter and handles all output to the tape or controller. It buffers and controls the output without user intervention.

2.1.2 SYMBOL: This routine writes alphanumeric text and special symbols anywhere and at any angle desired on the plot.

2.1.3 NUMBER: This routine writes computer values on the plot in a format (integer or decimal) chosen by the user.

2.1.4 CIRCLE: This routine draws a circle, arc, or spiral starting at a given point on the plot.*

2.2 Variables and Arrays in NETPLT

The following is an alphabetical listing by program segment of all the significant variables that appear in each segment. Note that the standard word type convention has been followed where all words are real except those beginning with letters I to N. Those variables and arrays which store data from the data deck are followed by the word Data. Also, array names are always followed by their dimensions.

2.2.1 MAIN:

(1) BUFF (1100) - an array used as a buffer by the CalComp routine PLOT to block the information going out on the plot tape.

(2) COORDX (900) - an array used to store the x coordinate, in inches, of the nodes in the network.

(3) COORDY (900) - an array used to store the y coordinate, in inches, of the nodes in the network.

(4) CX - the x conversion factor for transforming a node's input x coordinate into inches on the plot paper. In common block BRDR.

(5) CY - the y conversion factor for transforming a node's input y coordinate into inches on the plot paper. In common block BRDR.

(6) DEG - the angle δ subtended by the arc drawn between two network nodes. Presently set to 60°.

(7) DI - a parameter used by the CIRCLE subroutine to indicate that a solid arc (DI=0.0) or a dashed arc (DI=0.5) is to be drawn. In common block ARC.

* A modified version of CIRCLE which takes two thirds of the time has been written. See reference [2], pg. 7.

(8) FP - a floating point version of the commander identification number. This form is needed by subroutine NUMBER.

(9) FPN - a floating point representation of an integer needed by subroutine NUMBER. (Same purpose as FP.)

(10) FRN - same as FPN.

(11) HEAD (36) - the array that stores the heading for a plot. In common block BRDR. Data.

(12) HT - the height of the input unit integers around the border, the commander and subordinate code numbers around the border, and the node code numbers in the network plot. Presently set at .1. In common block BRDR.

(13) ICURV - the flag for indicating a downward arc (ICURV=1) or an upward arc (ICURV=2) network plot. In common block ARC. Data.

(14) IFLAG - stores the number of multiple links between a network node pair. In common block ARC. Data.

(15) IFLG - a flag used in the program to draw multiple homing links (links from a subscriber to his network nodes) with a minimum amount of lifting of the pen.

(16) IP1 - the node code number for a link's first node. Data.

(17) IP2 - the node code number for a link's second node or the code number of a homing node. Data.

(18) IREAD - the logical unit number for the computer input read device. Presently set to 5.

(19) ITYPE - the type of the plot currently being done. Same meaning as contents of LPTYPE. Can have values 1, 2, 3, or 4.

(20) IWRITE - the logical unit number for the computer system's output print device. Presently set to 6. In common block ARC.

(21) K - the number of subscriber points along one edge of the border. Obtained from pre-sorting the data. Data.

(22) L - the symbol code for a network node or a subscriber point being drawn on the plot. The plotter has twelve similarly drawn symbols available if others are preferred.

(23) LINKS - the number of links in the network plot. Data.

(24) LL(5) - an array containing the symbol code for a subscriber node along the border of an ITYPE=4 plot. Presently set to 0, 1, 2, 6, 10. This symbol code may change with the installation.

(25) LPTYPE(9) - an array containing the type of each plot in the run. A run is limited to nine plots. Data.

LPTYPE=1 <=> plot network plus subscribers on the border.

LPTYPE=2 <=> plot network only.

LPTYPE=3 <=> plot an overlay with all subscribers or commanders only on border. All commands use the same symbol.

LPTYPE=4 <=> plot an overlay with at most five commanders and/or subordinates on the border. Each command uses a distinct symbol.

(26) N - the number of the edge along which subscribers are being plotted.

(27) NCD(5) - an array storing the five commander code numbers for plot of ITYPE=4. Data.

(28) NCMD - the code number of the command for the current subscriber. Data.

(29) NHM(5) - an array storing the network node code numbers of the homings for a subscriber. Data.

(30) NHOME - the number of homings for a subscriber. Data.

(31) NODEN(900) - the sequential code numbers of the network nodes. Data.

(32) NODES - the number of nodes in the network plot. Data.

(33) NOPLT - the number of multiple links plotted between a network node pair. In common block ARC.

(34) NPL - the number of plots to be drawn in the run. Same as NPLOT.

(35) NPLOT - the number of plots to be drawn in the run. Data.

(36) NSUB - the code number of the current subordinate being plotted. If NSUB=0, the border point refers to a commander entry. Data.

(37) NSUBS - the total number of subscribers in the current run. Data.

(38) NSUM - a word accumulating the number of subscribers plotted. After all plots in the run are drawn, NSUM should equal NSUBS.

(39) NTYPE - the type of a network node. Data.

NTYPE=0<=> an ordinary node.

NTYPE=1<=> a subscriber homing node.

(40) PLOTX - the x dimension of the plotter or the size paper used (in inches). Data.

(41) PLOTY - the y dimension of the plotter or the size paper used (in inches). Data.

(42) POINT - the x coordinate of the starting point of the plot's heading.

(43) PY - the y coordinate of the starting point of the plot's heading.

(44) SCALEX - the length of the plot's x axis in input units. Data.

(45) SCALEY - the length of the plot's y axis in input units. Data.

(46) TEMP - a temporary storage location.

(47) TIC - the height of the tic marks along the border. Presently set to .5 inches. In common block BRDR.

(48) UNITX(900) - the x coordinate of the network nodes in input units. Data.

(49) UNITY(900) - the y coordinate of the network nodes in input units. Data.

(50) XAXIS - the length of the plot's desired x-axis, in inches. In common block BRDR.

(51) XCOORD - a word used to store a node's x coordinate for use by subroutine SYMBOL.

(52) XII - the x coordinate, in inches, of the lower left corner of the border for a plot.

(53) XIP1 - the x coordinate, in inches, of node 1 for a link. Used for network links or subscriber links. In common block ARC.

(54) XIP2 - the x coordinate, in inches, of node 2 for a link. Used for network links or subscriber links. In common block ARC.

(55) XL - the length assigned to a heading (3 cards of 216 characters).

(56) XUNIT - the x coordinate, in input units, of a subscriber point along the border. Data.

(57) XX - the x coordinate, in inches, of the commander and subordinate code number. Used for plotting purposes.

(58) YAXIS - length, in inches, of the plot's desired y-axis. In common block BRDR.

(59) YCOORD - a word used to store a node's y coordinate for use by subroutine SYMBOL.

(60) YII - the y coordinate, in inches, of the lower left corner of the border for a plot.

(61) YIP1 - the y coordinate, in inches, of node 1 for a link. Used for network links or subscriber links. In common block ARC.

(62) YIP2 - the y coordinate, in inches, of node 2 for a link. Used for network links or subscriber links. In common block ARC.

(63) YJ - the y coordinate, in inches, for the code number of a subordinate. Used by subroutine NUMBER.

(64) YUNIT - the y coordinate, in input units, of a subscriber point along the border. Data.

(65) YY - either the y coordinate, in inches, of the code number of a commander, or the y coordinate of the words in the plot's key to the symbols. Used by subroutine SYMBOL.

2.2.2 BORDER

(1) CX - same as in 2.2.1.

(2) CY - same as in 2.2.1.

(3) DIF - the difference, in inches, between the specified XAXIS or YAXIS and the nearest integer number of input data units that fit into XAXIS or YAXIS.

(4) FPN - a floating point representation of the tic mark integer along the border. Needed by subroutine NUMBER.

(5) HEAD(36) - same as in 2.2.1.

(6) HT - same as in 2.2.1.

(7) K - the nearest integer number of input data units in XAXIS or YAXIS.

(8) S - the number of inches along the x or y axis that contain the integer number of input data units along that axis.

(9) TIC - same as 2.2.1.

(10) X - used most often as the x coordinate, in inches, for the drawing of the border.

(11) XAXIS - same as 2.2.1.

(12) XX - used most often as the x coordinate, in inches, of the tic mark integer to be drawn.

(13) Y - used most often as the Y coordinate, in inches, for the drawing of the border.

(14) YAXIS - same as 2.2.1.

(15) YY - used most often as the Y coordinate, in inches, of the tic mark integer to be drawn.

2.2.3 PLTARC: (See Figures 10 and 11.)

(1) A - the angle α made by a link's chord with the horizontal. In degrees.

(2) AM - the maximum distance m, along a radius, between a primary link's chord and its arc. (See Figure 11.)

(3) AN - the maximum distance n, along a radius, between a secondary link's chord and its arc. (See Figure 11.)

(4) B1 - the angle β_1, formed by the primary circle's radius to node 1 and the horizontal.

(5) B2 - the angle β_2, formed by the primary circle's radius to node 2 and the horizontal.

(6) D - the distance between a primary arc and its secondary arc.

(7) DEG - the angle δ subtended by the arc drawn between nodes 1 and 2.

(8) DI - same as in 2.2.1.

(9) DIST - the length L of the chord between nodes 1 and 2.

(10) DIST2 - one-half of the length L.

(11) GAMA - half of the angle δ, in degrees. Denoted by γ in Figure 10.

(12) GAMAR - the angle γ in radians.

(13) ICURV - same as 2.2.1.

(14) IDIR - a flag for drawing secondary arcs more efficiently.

IDIR=1<=> draw arc clockwise from node 2 to node 1.

IDIR= -1<=> draw arc counter clockwise from node 1 to node 2.

(15) IFLAG - same as in 2.2.1.

(16) IND - a counter for the number of secondary links drawn.

(17) IWRITE - same as in 2.2.1.

(18) NOPLT - same as in 2.2.1.

(19) OMEG - the angle ω which, when added to angle α, yields angle β_1. In degrees.

(20) OMEGR - the angle ω in radians.

(21) PHI - the angle ϕ, which is half the subtended angle for secondary arc. Analogous to γ.

(22) R - the radius of the circle upon which a primary arc is drawn.

(23) R2 - the radius of the circle upon which a secondary arc is being drawn.

(24) TEMP - a temporary storage location.

(25) T1 - the angle Θ_1, formed by the secondary circle's radius to node 1 and the horizontal. Analogous to β_1.

(26) T2 - the angle Θ_2, formed by the secondary circle's radius to node 2 and the horizontal. Analogous to β_2.

(27) XIP1 - same as in 2.2.1.

(28) XIP2 - same as in 2.2.1.

2.3 Components of NETPLT

2.3.1 MAIN: The two initial read statements establish the size of the network (number of nodes and links), the type of arcs chosen (upward or downward), the number of plots for the run, the length of the x and y axis in inches and input data units, and the x and y dimensions of the plotter paper. The program then sets up a scale factor for each dimension and initializes the location of the plotter pen in the lower left corner of the paper. Next, the types of the plots are read into an array LPTYPE. The DO 100 loop then draws each of the plots, branching within itself for each type.

For all four type plots, the heading is read and written on the run's printout. Then subroutine BORDER is called in to draw the border. (See 2.3.2.) In the DO 10 loop, the node data, consisting of the node code number, its x and y coordinates, and its node type is read. The program then plots the node symbols with their identification codes above them, for types 1 and 2, and, for all type plots, writes the node table in the printout. Plot types 1 and 2 then go into the link plotting phase. The DO 20 loop reads in the link node pairs and their multiplicity number, calls on PLTARC to plot the arcs, and writes the link table in the printout.

At this point, plot types 1, 3, and 4 go on to draw subscribers. For the subscriber plotting, the individual plot's number of subscribers is read, and then, for type 4, commander code numbers are read in. Then, for each of the four edges of the border, the number of subscribers along that border edge is read from one card; the commander code number, subordinate code number, subscriber node x and y coordinates, and the number of homings are read from a second card; the code number for each commander/subordinate node is drawn (using the convention that a subordinate code number of zero means the node is a commander); the subscriber homing nodes are read from a third card; and the straight line connectors between subscriber points on the border and homings are drawn. The second and third cards are repeated for each subscriber along the border edge. During this part of the program the subscriber data is also written into a table for the printout.

For all type plots the heading is drawn across the top and a key to the symbols is drawn in the upper left corner of the paper. For type 1 and 2 plots the symbols for an ordinary network node and a homing node are shown in the key; for types 1 and 3 the explanation of the border node symbol is given in the key lower down on the paper; and finally, for type 4 the explanation of the border point symbols and their correlation to commanders is given still further down on the paper. The entries in the key are spread down the page so that they can be read when several transparent plots are viewed together. (See Appendix B figures for samples of this key.)

2.3.2 BORDER: This subroutine draws the border for a plot, starting in the lower left corner and proceeding continuously along the lower, the right, the top, and the left edges. Tic marks are drawn at half integers and numbers added at the whole integers. The input Data units are used for the scale.

2.3.3 PLTARC: This routine does all the network arc plotting. The crucial information that it needs is the flag for upward or downward arcs (ICURV), and the coordinates of nodes 1 and 2. For downward arcs, XIP1 must be greater than XIP2 while for upward arcs XIP1 must be less than XIP2. The program examines the coordinates and may interchange them to assure that this is the case. The tie for the equality case is broken by arbitrarily assigning $\alpha = +90°$ to downward arc plots and $\alpha = -90°$ to upward arc plots. The flag IDIR is always initially set so that counterclockwise arcs are drawn from node 1 to node 2 and clockwise arcs from node 2 to node 1. The length of the link's chord (DIST), and the quantities GAMA, OMEG, OMEGR, DIST2, and R are then calculated. The proper values of A, B1, and B2 for upward or downward arcs is then obtained and the clockwise or counterclockwise arc drawn.

If secondary arcs are to be drawn, the counter NOPLT is initiated to 1, the quantity AM is calculated, and the proper R2, PHI, T1 and T2 obtained. These values appear in the link table in the printout and then the clockwise or counterclockwise secondary arc is drawn. IDIR is alternated between 1 and -1 so that secondary arcs are drawn without lifting the pen. The drawing of secondary arcs ceases when all multiple arcs are drawn or the last arc is as close as possible to the semicircle.

3.0 INPUT/OUTPUT DESCRIPTION

The inputs, outputs, and data deck have been described in full in PART I and will not be described here. Figure 9 shows the data deck card formats while Figures 7 and 8 show the run deck and data deck, respectively.

REFERENCES

[1] *Ruthberg, Z. G., Bellmore, M., User's Manual for Program DFENSE, NBS Report 10821, January 1972.*

[2] *Ruthberg, Z. G., Bolotsky, G. R., Programmer's Manual for the ONC Map Plotting Program SIPLOT, NBS Report 10619, October 1971.*

[3] *CalComp Software Reference Manual, California Computer Products, Inc., Anaheim, California 92803, February 1968.*

APPENDIX A. SAMPLE PRINTOUT FROM NETPLT

A.1 Node Table

INITIAL NETWORK

	COORDINATES OF NODE				
	INITIAL		INCHES		
NODEN	UNITX(I)	UNITY(I)	COORDX(I)	COORDY(I)	TYPE
1	.0	.7	.0	2.1	1
2	.1	.9	.4	2.7	0
3	.2	.7	.8	2.1	0
4	.4	.9	1.6	2.7	0
5	.5	1.2	2.0	3.6	0
9	.6	1.6	2.4	4.8	0
8	.6	1.4	2.4	4.2	0
7	.6	1.3	2.4	3.9	0
6	.6	.8	2.4	2.4	0
10	.7	.8	2.8	2.4	0
11	.7	1.0	2.8	3.0	0
12	.7	1.3	2.8	3.9	0
13	.7	1.9	2.8	5.7	1
15	.8	1.0	3.2	3.0	0
14	.8	.8	3.2	2.4	0
16	.9	1.0	3.6	3.0	0
18	1.0	1.4	4.0	4.2	0
17	1.0	1.2	4.0	3.6	0
19	1.1	1.3	4.4	3.9	0
20	1.1	1.5	4.4	4.5	0
21	1.1	1.6	4.4	4.8	1
25	1.2	1.3	4.8	3.9	0
24	1.2	1.2	4.8	3.6	0
23	1.2	.8	4.8	2.4	1
22	1.2	.7	4.8	2.1	0
26	1.3	.7	5.2	2.1	0
27	1.4	.7	5.6	2.1	0
28	1.5	.7	6.0	2.1	0
29	1.5	1.1	6.0	3.3	1
30	1.5	1.2	6.0	3.6	0
31	1.6	3.1	6.4	9.3	1
32	1.7	.6	6.8	1.8	0
33	1.7	.9	6.8	2.7	1
34	1.7	1.5	6.8	4.5	1
35	1.7	2.4	6.8	7.2	1
36	1.7	3.2	6.8	9.6	1
44	1.8	3.1	7.2	9.3	1
43	1.8	2.9	7.2	8.7	1
42	1.8	2.5	7.2	7.5	1
41	1.8	2.2	7.2	6.6	0
40	1.8	1.6	7.2	4.8	1
39	1.8	.7	7.2	2.1	0
38	1.8	.4	7.2	1.2	1
37	1.8	.1	7.2	.3	1
45	1.9	.8	7.6	2.4	0
46	1.9	.9	7.6	2.7	0
47	1.9	2.0	7.6	6.0	1
48	1.9	3.0	7.6	9.0	1
49	1.9	3.3	7.6	9.9	0
50	1.9	3.5	7.6	10.5	0
51	2.0	2.5	8.0	7.5	0
52	2.1	.1	8.4	.3	0
53	2.1	2.5	8.4	7.5	0
54	2.1	2.7	8.4	8.1	0
55	2.1	2.8	8.4	8.4	1
56	2.1	3.3	8.4	9.9	0
60	2.2	6.2	8.8	18.6	0
59	2.2	3.3	8.8	9.9	1
58	2.2	1.2	8.8	3.6	0
57	2.2	.8	8.8	2.4	1
61	2.3	1.3	9.2	3.9	1
62	2.3	3.4	9.2	10.2	0
63	2.3	5.1	9.2	15.3	1
64	2.3	6.2	9.2	18.6	0
65	2.3	6.3	9.2	18.9	1
66	2.4	3.0	9.6	9.0	1
67	2.5	3.0	10.0	9.0	1
68	2.5	3.6	10.0	10.8	0
69	2.5	3.7	10.0	11.1	1
70	2.5	4.2	10.0	12.6	1
75	2.6	4.4	10.4	13.2	1
74	2.6	4.3	10.4	12.9	1
73	2.6	4.2	10.4	12.6	1
72	2.6	3.7	10.4	11.1	0
71	2.6	3.6	10.4	10.8	0

INDEX	LINKS NODE 1	NODE 2	NO OF LINKS IFLAG	NO OF LINK PLOTTED NOPLT
1	0	0	0	1
2	2	3	1	1
3	2	4	1	1
4	4	5	1	1
5	5	8	1	1
6	8	7	1	1
7	8	12	1	1
8	12	6	1	1
9	12	13	1	1
10	12	17	1	1
11	17	19	1	1
12	19	23	1	1
13	23	24	1	1
14	24	14	1	1
15	14	5	1	1
16	14	11	1	1
17	11	1	1	1
18	11	20	1	1
19	20	17	1	1
20	20	22	1	1
21	22	19	1	1
22	22	21	1	1
23	22	34	1	1
24	34	33	1	1
	1	67.4832	59.3878	194.3541
25	34	41	2	2
26	41	40	1	1
27	40	47	1	1
28	40	53	1	1
29	53	54	1	1
30	54	55	1	1
31	55	39	1	1
32	39	35	1	1
33	39	51	1	1
34	55	38	1	1
35	38	62	1	1
36	62	49	1	1
37	49	56	1	1
38	56	48	1	1
39	48	36	1	1
40	48	37	1	1
41	37	31	1	1
42	56	50	1	1
43	56	58	1	1
44	58	62	1	1
45	62	66	1	1
46	66	75	1	1
47	75	68	1	1
48	68	69	1	1
49	69	75	1	1
50	75	80	1	1
51	80	69	1	1
52	69	79	1	1
53	79	88	1	1
54	88	89	1	1
55	89	74	1	1
56	74	80	1	1
57	80	102	1	1
58	102	94	1	1
59	94	84	1	1
60	84	82	1	1
61	82	73	1	1
62	73	70	1	1
63	70	72	1	1
64	72	73	1	1
65	73	81	1	1
66	81	72	1	1
67	70	71	1	1
68	71	83	1	1
69	82	86	1	1
70	86	62	1	1
71	86	85	1	1
72	85	88	1	1
73	88	87	1	1
74	87	79	1	1
75	88	104	1	1
76	104	126	1	1

A.3 Command Table

INITIAL NETWORK COMMANDS

CMDSUB		UNITX(I)	UNITY(I)	COORDX(I)	COORDY(I)	NHOME	HOMINGS				
20	0	-.1250	5.2308	-.5000	15.6924	2	63	123			
19	0	3.2967	9.1667	13.1868	27.5000	2	144	122			
23	0	3.4690	9.1667	13.8758	27.5000	2	153	147			
8	0	3.6412	9.1667	14.5649	27.5000	2	163	161			
5	0	4.3303	9.1667	17.3210	27.5000	5	316	181	296	168	116
17	0	4.6748	9.1667	18.6991	27.5000	2	243	252			
6	0	6.0529	9.1667	24.2114	27.5000	2	411	412			
3	0	6.7419	9.1667	26.9676	27.5000	4	339	442	560	452	
13	0	6.9141	9.1667	27.6566	27.5000	2	445	481			
4	0	7.0003	9.1667	28.0011	27.5000	2	450	449			
2	0	7.5171	9.1667	30.0682	27.5000	5	605	586	585	584	571
11	0	8.8951	9.1667	35.5805	27.5000	5	605	586	585	584	571
7	0	9.7564	9.1667	39.0257	27.5000	2	712	710			
18	0	12.1250	6.4679	48.5000	19.4038	2	764	765			
22	0	12.1250	4.4008	48.5000	13.2025	2	838	824			
1	0	12.1250	1.4150	48.5000	4.2450	2	781	772			
15	0	9.4354	-.1667	37.7416	-.5000	2	679	668			
9	0	9.2631	-.1667	37.0526	-.5000	2	679	668			
12	0	6.4269	-.1667	25.6834	-.5000	2	441	440			
21	0	6.3347	-.1667	25.3389	-.5000	2	426	419			
16	0	4.5260	-.1667	18.1040	-.5000	2	231	250			
10	0	3.4924	-.1667	13.9698	-.5000	2	171	170			
14	0	3.2341	-.1667	12.9362	-.5000	2	127	130			

INITIAL NETWORK

SUBORDINATES

CMDSUB	UNITX(1)	UNITY(1)	COORDX(1)	COORDY(1)	NHOME	HOMINGS				
3 1	-.1250	1.0966	-.5000	3.2897	2	33	29			
3 2	-.1250	2.5895	-.5000	7.7685	2	76	90			
5 25	-.1250	2.8192	-.5000	8.4575	3	1	70	47		
3 35	-.1250	3.1637	-.5000	9.4910	2	39	35			
3 3	-.1250	3.8527	-.5000	11.5582	2	127	130			
5 31	-.1250	4.0824	-.5000	12.2472	2	38	48			
4 1	-.1250	4.3121	-.5000	12.9362	2	38	48			
2 1	-.1250	4.4269	-.5000	13.2808	2	38	48			
3 4	-.1250	4.6566	-.5000	13.9698	2	66	67			
5 24	-.1250	5.9198	-.5000	17.7595	3	1	70	194		
5 22	-.1250	6.0347	-.5000	18.1040	3	1	70	194		
5 23	-.1250	6.1495	-.5000	18.4486	3	1	70	194		
5 2	-.1250	7.6424	-.5000	22.9273	2	112	122			
5 21	-.1250	8.1018	-.5000	24.3054	2	65	97			
3 5	.0238	9.1667	.0951	27.5000	2	98	132			
5 20	.2822	9.1667	1.1286	27.5000	2	124	125			
5 32	1.3157	9.1667	5.2629	27.5000	2	147	153			
2 2	1.4880	9.1667	5.9519	27.5000	2	147	153			
4 2	1.5741	9.1667	6.2964	27.5000	2	147	153			
5 12	1.8325	9.1667	7.3300	27.5000	2	157	138			
5 16	2.2631	9.1667	9.0526	27.5000	2	116	194			
5 17	2.3493	9.1667	9.3971	27.5000	2	116	194			
5 1	2.5215	9.1667	10.0861	27.5000	2	163	161			
5 3	2.6938	9.1667	10.7752	27.5000	2	194	119			
3 7	3.0383	9.1667	12.1532	27.5000	2	178	179			
5 13	3.9857	9.1667	15.9430	27.5000	2	218	198			
3 6	4.5025	9.1667	18.0101	27.5000	2	222	223			
4 3	4.5886	9.1667	18.3546	27.5000	2	241	242			
2 3	4.7609	9.1667	19.0436	27.5000	2	241	242			
5 33	5.0193	9.1667	20.0772	27.5000	2	241	242			
3 11	5.3638	9.1667	21.4552	27.5000	2	264	278			
5 19	5.7945	9.1667	23.1778	27.5000	2	292	314			
3 12	6.3974	9.1667	25.5895	27.5000	2	393	394			
3 14	6.5696	9.1667	26.2785	27.5000	2	413	392			
3 15	6.6558	9.1667	26.6230	27.5000	2	415	407			
3 13	7.3448	9.1667	29.3792	27.5000	2	411	412			
5 10	7.4309	9.1667	29.7237	27.5000	2	411	412			
5 8	7.9477	9.1667	31.7908	27.5000	1	442				
5 35	8.2061	9.1667	32.8244	27.5000	2	450	449			
2 5	8.2922	9.1667	33.1689	27.5000	2	450	449			
3 24	8.6367	9.1667	34.5470	27.5000	2	461	462			
3 22	9.2397	9.1667	36.9586	27.5000	2	460	504			
3 23	9.4980	9.1667	37.9922	27.5000	2	480	539			
3 25	9.5842	9.1667	38.3367	27.5000	2	503	525			
5 9	9.9287	9.1667	39.7148	27.5000	1	560				
5 4	10.2732	9.1667	41.0928	27.5000	5	605	586	585	5[illegible]4	571
2 6	10.4455	9.1667	41.7819	27.5000	2	605	582			
4 5	10.6177	9.1667	42.4709	27.5000	2	605	582			
5 36	10.7039	9.1667	42.8154	27.5000	2	605	582			
3 26	11.0484	9.1667	44.1935	27.5000	2	675	711			
3 34	11.5652	9.1667	46.2606	27.5000	2	764	765			
5 11	11.7374	9.1667	46.9497	27.5000	2	764	765			
3 33	12.1250	8.5351	48.5000	25.6051	2	825	790			
3 32	12.1250	8.0757	48.5000	24.2271	2	808	807			
5 28	12.1250	7.3867	48.5000	22.1600	3	839	825	826		
5 26	12.1250	7.2718	48.5000	21.8154	3	839	825	826		
5 27	12.1250	7.1570	48.5000	21.4709	3	839	825	826		
5 30	12.1250	6.8124	48.5000	20.4373	2	825	839			
3 30	12.1250	5.7789	48.5000	17.3367	2	823	824			
3 31	12.1250	5.6641	48.5000	16.9922	2	823	824			
5 37	12.1250	5.2047	48.5000	15.6141	2	839	837			
2 7	12.1250	5.0899	48.5000	15.2696	2	839	837			
4 6	12.1250	4.9750	48.5000	14.9251	2	839	837			
3 29	12.1250	4.7453	48.5000	14.2360	2	822	811			
5 29	12.1250	4.0563	48.5000	12.1689	2	825	744			
1 14	12.1250	3.9415	48.5000	11.8244	2	820	827			
1 2	12.1250	3.8266	48.5000	11.4799	2	820	827			
1 8	12.1250	3.7118	48.5000	11.1353	2	820	827			
1 20	12.1250	3.5969	48.5000	10.7908	2	820	827			
1 19	12.1250	3.4821	48.5000	10.4463	2	721	733			
1 7	12.1250	3.3673	48.5000	10.1018	2	835	829			
1 21	12.1250	3.2524	48.5000	9.7573	2	746	737			
1 4	12.1250	3.1376	48.5000	9.4127	2	832	818			
1 18	12.1250	3.0227	48.5000	9.0682	2	802	803			
1 1	12.1250	2.9079	48.5000	8.7237	2	746	736			
3 28	12.1250	2.7931	48.5000	8.3792	2	736	746			
1 9	12.1250	2.4485	48.5000	7.3456	2	736	744			

APPENDIX B SAMPLE PLOTS FROM NETPLT

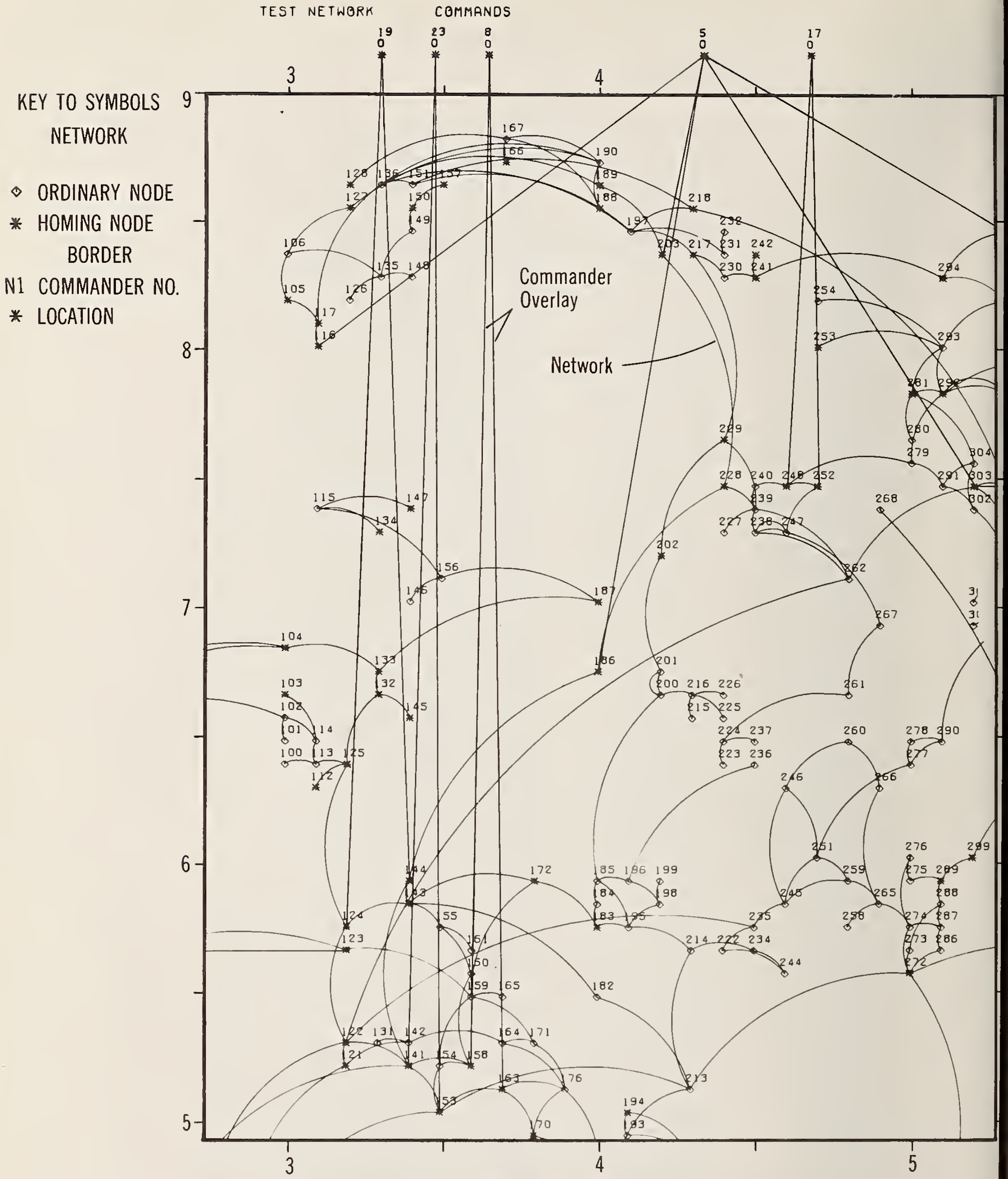


Fig. B1 Sparsely Connected Network (DCA) with Commander Overlay

(Section of full drawing)

KEY TO SYMBOLS
NETWORK

ORDINARY NODE
HOMING NODE
UNKNOWN

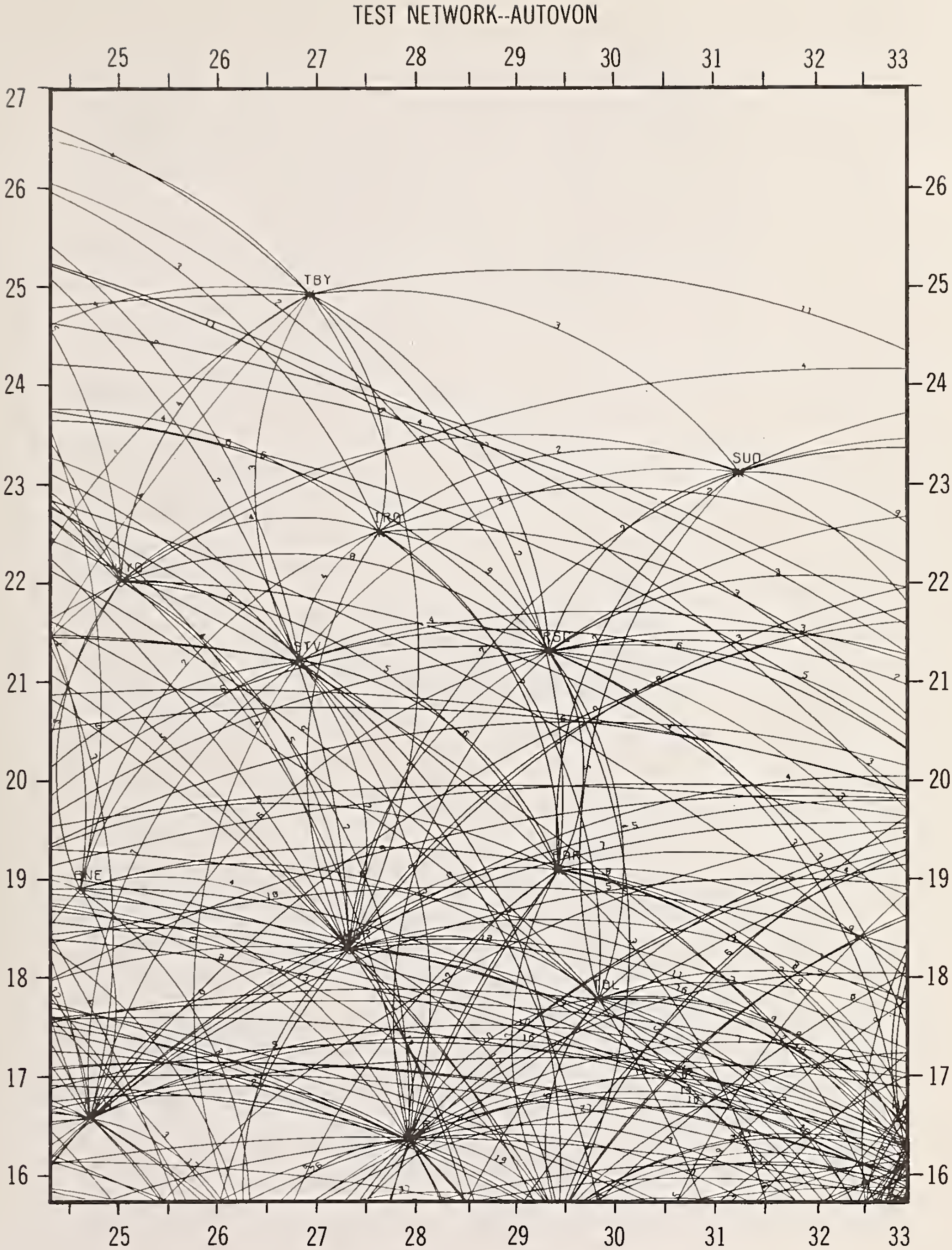


Fig. B2 Densely Connected Network (Autovon)

(Section of full drawing)

```
      OIMENSION UNITX(9DO),UNITY(900),NODEN(900),HEAO(36),COORDX(9DD),
     1CODROY(900),RUFF(1100),NHM(5),LPTYPE(9),NCO(5),LL(5)
      COMMON/BRDR/XAXIS,YAXIS,TIC,HT,CX,CY,HEAO
      COMMDN/ARC/  ICURV,XIP1,XIP2,YIP1,YIP2,IFLAG,DEG,NOPLT,IWRITE,DI
      IREAO = 5
      IWRITE = 6
C     NOOES >NUMBER OF NODES FOR NETWORK.
C     LINKS >NUMBER OF LINKS FOR NETWORK.
C     ICURV SETS LINK TYPE.  I MEANS DOWNWARD ARCS. 2 MEANS UPWARD ARCS.
      READ(IREAO,1000) NOOES ,LINKS ,ICURV
 1000 FORMAT(415)
C     VARIABLES IN NEXT REAO STATEMENT = PLOT PARAMETERS.
C     NPLDT>ND. DF PLOTS FDR NETWORK ANO ITS SUBSCRIBERS [I OR MORE].
C     XAXIS=LENGTH OF X AXIS IN INCHES.
C     YAXIS=LENGTH OF Y AXIS IN INCHES.
C     SCALEX=LENGTH OF X AXIS IN OATA UNITS.
C     SCALEY=LENGTH OF Y AXIS IN OATA UNITS.
C     PLOTX=X DIMENSION DF PLOTTER.
C     PLOTY=Y DIMENSION OF PLOTTER.
      REAO(IREAO,1001) NPLDT         ,XAXIS,YAXIS,SCALEX,SCALEY,PLOTX,PLO
     1TY
 1001 FORMAT( I5,6(5X,F5.1))
C     CALCULATE CONVERSION FACTORS CX, CY FOR AXES.
      CX = XAXIS/SCALEX
      CY = YAXIS/SCALEY
      CALL PLOTS(BUFF,1000,7)
      IF(NPLOT ) 5030,5030,18
 5030 WRITE(IWRITE,5006)
 5006 FORMAT[16H ERROR IN NPLOT.]
      STOP
   18 NPL = NPLOT
C     LPTYPE=1 MEANS PLOT NETWORK PLUS SUBSCRIBERS ON BOROER.
C     LPTYPE=2 MEANS PLOT NETWORK ONLY.
C     LPTYPE=3 MEANS PLOT AN OVERLAY WITH SUBSCRIBERS OR COMMANDERS ON
C     BOROER.
C     LPTYPE=4 MEANS PLOT AN OVERLAY WITH 4 OR 5 COMMANOERS ANO/OR
C     THEIR SUBORDINATES ON THE BORDER.
      REAO(IREAO,1011)(LPTYPE(I),I=1,NPL)
 1011 FDRMAT(10I5)
      XII = (PLOTX-XAXIS)/2
      YII= (PLOTY-.8-YAXIS)/2.
      TIC = .05
      HT = .1
      DD 100 LJ=1,NPL
      XI = XII
      YI = YII
      CALL PLOT(XI,YI,-3)
      REAO (IREAO,2001) HEAO
 2001 FDRMAT ((12A6))
      WRITE(IWRITE,8)
    8 FORMAT(1H1)
      WRITE(IWRITE,2001) HEAD
      ITYPE = LPTYPE(LJ)
      CALL BORDER
C     REAO IN NOOE NO. ANO COORDINATES. CONVERT TO INCHES ON PLOTTER.
      IF(LJ-I) 33,33,34
   33 IF ( NOOES  ) 5000,5000,3002
 5000 WRITE(IWRITE,5002)
 5002 FORMAT[41H OATA CAROS CONTAINING NOOES IS IN ERROR.]
      STOP
 3002 WRITE(IWRITE,1)
    1 FDRMAT(20X,19HCOOROINATES OF NODE/16X,8H INITIAL,14X,6HINCHES/2X,5
     1HNDOEN,4X,8HUNITX(I),2X,8HUNITY(I),2X,9HCOOROX(I),2X,9HCOOROY(I),2
     2X,5H TYPE)
      DO 10 I=1,NOOES
C     NTYPE=0 MEANS NODE IS OROINARY.  NTYPE=1 MEANS NODE IS A SUBSCRIB
C     ER HOMING. [ENTRY OR EXIT]
      READ(IREAD,1002) NDDEN(I),UNITX(I),UNITY(I),NTYPE
 1002 FORMAT (I5,30X,2F5.1,14X,I1)
      COORDX(I) = UNITX(I)*CX
      CDORDY(I) = UNITY(I)*CY
      XCDORD = CODRDX(I)
      YCDDRO = CODRDY(I)
      IF(ITYPE-2) 14,14,10
   14 IF(NTYPE) 11,12,11
   11 L = 11
      GO TD 13
   12 L = 5
   13 FPN = NOOEN(I)
      CALL SYMBOL(XCOORO,YCOORO,HT,L,0.,-1)
      CALL NUMBER(XCOORO-HT/2.,YCDORD+HT,HT,FPN,0.,-1)
```

```
      WRITE(IWRITE,2) NOOEN(I),UNITX(I),UNITY(I),COOROX(I),COOROY(I),NTY
     1PE
    2 FORMAT(3X,I4,5X,F5.1,3X,F5.1,8X,F5.1,6X,F5.1,6X,I1)
   10 CONTINUE
   34 IF(ITYPE-2) 35,35,36
   35 OEG = 60.
      IF(LINK5 )3030,3030,3020
 3030 WRITE(IWRITE,5003)
 5003 FORMAT[27H VALUES FOR LINK5 IN ERROR.]
      5TOP
 3020 WRITE(IWRITE,3)
    3 FORMAT(1H1,26X,5HNO OF,2X,11HNO OF LINK5/11X,13H    LINK5    ,3X,5
     1HLINK5,2X,11H  PLOTTEO  /2X,5HINOEX,3X,6HNOOE 1,2X,6HNOOE 2,3X,6HI
     2FLAG ,3X,7HNOPLT  )
      OI = 0.0
      OO 20 I=1,LINK5
      REAO (IREAO,1003) IP1,IP2,IFLAG
 1003 FORMAT (5X,2I5,48X,I2)
      XIP1 = COORBX(IP1)
      XIP2 = COOROX(IP2)
      YIP1 = COORBY(IP1)
      YIP2 = COORDY(IP2)
      CALL PLTARC
      WRITE(IWRITE,4) I,IP1,IP2,IFLAG,NOPLT
    4 FORMAT(3X,I4,5X,I4,2X,I4,6X,I3,7X,I3)
   20 CONTINUE
   36 IF(ITYPE-2) 3004,334,3004
C     N5U85 > NUMBER OF 5U85CRIBER5 IN NETWORK.
 3004 REAO(IREAO,1008) N5U85
 1008 FORMAT(2I5)
      IF(N5U85 ) 2030,2030,3040
 2030 WRITE(IWRITE,5004)
 5004 FORMAT[29H VALUE FOR N5U85 I5 IN ERROR.]
      5TOP
 3040 N5UM = 0
      IFLAG = 1
      WRITE(IWRITE,5)
    5 FORMAT(2X,6HCM05U8,3X,8HUNITX(I),2X,8HUNITY(I),2X,9HCOOROX(I),2X,9
     1HCOOROY(I),2X,5HNHOME,2X,7HHOMING5)
      IF(ITYPE-3) 62,62,63
   63 REAO(IREAO,1007)(NCO(II),II=1,5)
 1007 FORMAT(5I5)
      LL(1)=0
      LL(2)=1
      LL(3)=2
      LL(4)=6
      LL(5)=10
   62 L=11
      OI = 0.5
      OO 30 II=1,4
      N = II
      REAO(IREAO,1004) K
 1004 FORMAT(I5)
      N5UM = N5UM+K
      OO 40 J =1,K
      REAO(IREAO,1005) NCMO,N5UB,XUNIT,YUNIT,NHOME
 1005 FORMAT(2I5, 25X, 2F10.5, I5)
      XCOORO = XUNIT*CX
      YCOORO = YUNIT*CY
      FPN = NCMO
      FRN= N5U8
      IF(ITYPE-3) 65,64,65
   65 OO 60 IJ=1,5
      NC=NCO(IJ)
      IF(NC-NCMO) 60,61,60
   61 L=LL(IJ)
      GO TO 64
   60 CONTINUE
      WRITE(IWRITE,5007)
 5007 FORMAT(42H A COMMANOER-5U8OROINATE CARO I5 IN ERROR.)
   64 GO TO (41,42,43,44),N
   41 XX = XCOORO-3*HT
      YY = YCOORO
      YJ = YY-HT-.05
      GO TO 45
   42 XX = XCOORO-HT/2.
      YY = YCOORO+HT*2+.05
      YJ = YY-HT-.05
      GO TO 45
   43 XX = XCOORO+HT
      YY = YCOORO
      YJ = YY-HT-.05
      GO TO 45
   44 XX = XCOORO-HT/2.
      YY = YCOORO-2*HT
      YJ = YY-HT-.05
   45 CALL 5YM8OL(XCOORO,YCOORO,HT,L,0.,-1)
```

```
      CALL NUMBER(XX,YY,HT,FPN,0.,-1)
      CALL NUMBER(XX,YJ,HT,FRN,0.,-1)
      IFLG = 1
      REAO(IREAO,1006) (NHM(I),I=1,NHOME)
 1006 FORMAT(5I5)
      OO 50 IJ = 1,NHOME
      XIP1 = XCOORO
      YIP1 = YCOORO
      IP2 = NHM(IJ)
      XIP2 = COOROX(IP2)
      YIP2 = COOROY(IP2)
      IFLG = -1*IFLG
      IF(IFLG-1) 54,55,55
   55 TEMP = XIP1
      XIP1 = XIP2
      XIP2 = TEMP
      TEMP = YIP1
      YIP1 = YIP2
      YIP2 = TEMP
   54 CALL PLOT(XIP1,YIP1,3)
      CALL PLOT(XIP2,YIP2,2)
   50 CONTINUE
      WRITE(IWRITE,6) NCMO,NSUB,XUNIT,YUNIT,XCOORO,YCOORO,NHOME,(NHM(I),
     1I=1,NHOME)
    6 FORMAT(2X,2I3,4F10.4,4X,I3,2X,5I5)
   40 CONTINUE
   30 CONTINUE
      IF(NSUBS-NSUM) 5040,334,5040
 5040 WRITE(IWRITE,5005)
 5005 FORMAT[69H THERE IS AN ERROR BETWEEN NSUBS ANO NUMBER OF SUBSCRIBE
     1R OATA CAROS.]
  334 CONTINUE
C     HEAOING
      XL = 216*.125
      POINT = (XAXIS-XL)/2
      PY = YAXIS+1.
      CALL SYMBOL(POINT,PY,.125,HEAO,0.,216)
C     KEY TO SYMBOLS
      CALL SYMBOL(-3.0,YAXIS,.125,14HKEY TO SYMBOLS,0.,14)
      GO TO (80,80,82,83),ITYPE
   80 YY = YAXIS-.25
      CALL SYMBOL(-2.625,YY,HT,7HNETWORK,0.,7)
      YY = YY-.2
      CALL SYMBOL(-2.95,YY+HT/2,HT,5,0.,-1)
      CALL SYMBOL(-2.75,YY,HT,13HOROINARY NOOE,0.,13)
      YY = YY-.2
      CALL SYMBOL(-2.95,YY+HT/2,HT,11,0.,-1)
      CALL SYMBOL(-2.75,YY,HT,11HHOMING NOOE,0.,11)
      IF(ITYPE-2)82,335,82
   82 YY = YAXIS-.85
      CALL SYMBOL(-2.5,YY,HT,6HBOROER,0.,6)
      YY = YY-.2
      CALL SYMBOL(-3.0,YY,HT,2HN1,0.,2)
      CALL SYMBOL(-2.75,YY,HT,13HCOMMANOER NO.,0.,13)
      YY = YY-.15*2
      CALL SYMBOL(-2.95,YY+HT/2,HT,11,0.,-1)
      CALL SYMBOL(-2.75,YY,HT,8HLOCATION,0.,8)
      GO TO 335
   83 YY = YAXIS-1.60
      CALL SYMBOL(-2.5,YY,HT,6HBOROER,0.,6)
      OO 89 IK=1,5
      YY = YY-.2
      NC = NCO(IK)
      FP = NC
      IF(NC) 90,89,90
   90 CALL NUMBER(-3.0,YY,HT,FP,0.,-1)
      CALL SYMBOL(-2.75,YY,HT,13HCOMMANDER NO.,0.,13)
      YY = YY-.15
      CALL SYMBOL(-3.0,YY,HT,2HN2,0.,2)
      CALL SYMBOL(-2.75,YY,HT,15HSUBORDINATE NO.,0.,15)
      YY = YY-.15
      L = LL(IK)
      CALL SYMBOL(-2.95,YY+HT/2,HT,L,0.,-1)
      CALL SYMBOL(-2.75,YY,HT,8HLOCATION,0.,8)
   89 CONTINUE
  335 XI = XAXIS
      YI = -YI1
      CALL PLOT(XI,YI,-3)
      CALL PLOT (0.0,0.0,999)
  100 CONTINUE
      ENO


      SUBROUTINE BOROER
      COMMON/BPOR/XAXIS,YAXIS,TIC,HT,CX,CY,HEAO
C     BOTTOM X AXIS
      X = 0.0
```

```
      K = XAXIS/CX
      S = K*CX
      DIF = XAXIS-S
      DO 10 I = 1,K
      DO 20 J = 1,2
      X = X+CX/2.
      CALL PLOT(X,0.,2)
      CALL PLOT(X,-TIC,2)
      CALL PLOT(X,0.0,3)
   20 CONTINUE
      XX = X-HT/2.
      FPN = X/CX
      YY = -2*TIC-HT
      CALL NUMBER(XX, YY,HT,FPN,0.,-1)
   10 CALL PLOT(X,0.0,3)
      IF(DIF .LE. 0.) GO TO 31
      CALL PLOT(XAXIS,0.0,2)
   31 X = XAXIS
C     RIGHT Y AXIS
      Y = 0.0
      K = YAXIS/CY
      S = K*CY
      DIF = YAXIS-S
      DO 30 I = 1,K
      DO 40 J = 1,2
      Y = Y+CY/2.
      CALL PLOT(X,Y,2)
      XX = X+TIC
      CALL PLOT( XX,Y,2)
      CALL PLOT(X,Y,3)
   40 CONTINUE
      XX = X+2*TIC
      FPN = Y/CY
      CALL NUMBER(XX,Y,HT,FPN,0.,-1)
   30 CALL PLOT(X,Y,3)
      IF(DIF .LE. 0.) GO TO 51
      Y = YAXIS
      CALL PLOT(X,Y,2)
C     TOP X AXIS
   51 K = XAXIS/CX
      Y = YAXIS
      S = K*CX
      DIF = XAXIS-S
      IF(DIF .LE. 0.) GO TO 61
      X = XAXIS-DIF
      CALL PLOT(X,Y,2)
      YY = Y+TIC
      CALL PLOT (X, YY,2)
      CALL PLOT (X,Y,3)
   61 DO 50 I = 1,K
      DO 60 J = 1,2
      X = X-CX/2
      CALL PLOT(X,Y,2)
      YY = Y+TIC
      CALL PLOT(X, YY,2)
      CALL PLOT(X,Y,3)
   60 CONTINUE
      XX = X-HT/2
      YY = Y+2*TIC
      FPN = X/CX
      CALL NUMBER(XX,YY,HT,FPN,0.,-1)
   50 CALL PLOT(X,Y,3)
      X = 0.0
C     LEFT Y AXIS
      K = YAXIS/CY
      S = K*CY
      DIF = YAXIS-S
      IF(DIF .LE. 0.) GO TO 81
      Y = YAXIS-DIF
      CALL PLOT(0.0,Y,2)
      CALL PLOT(-TIC,Y,2)
      CALL PLOT (0.0,Y,3)
   81 DO 70 I = 1,K
      DO 80 J = 1,2
      Y = Y-CY/2
      CALL PLOT(0.0,Y,2)
      CALL PLOT(-TIC,Y,2)
      CALL PLOT(0.0,Y,3)
   80 CONTINUE
      FPN = Y/CY
      XX = -2*TIC-HT
      YY = Y-HT/2
      CALL NUMBER(XX,YY,HT,FPN,0.,-1)
   70 CALL PLOT(0.,Y,3)
      RETURN
      END
```

```
      SUBROUTINE PLTARC
      COMMON/ARC/  ICURV,XIP1,XIP2,YIP1,YIP2,IFLAG,DEG,NDPLT,IWRITE,DI
C     FLAG IDIR=-1 MEANS LINK IS DRAWN COUNTERCLOCKWISE FROM NODE 1 TO
C     NODE 2.
C     FLAG IDIR=1 MEANS LINK IS DRAWN CLOCKWISE FROM NODE 2 TO NODE 1.
      IF(ICURV-1) 250,250,450
  250 IF(      XIP1 -XIP2         ) 265,270,275
  265  TEMP = XIP1
      XIP1 = XIP2
      XIP2 =  TEMP
       TEMP = YIP1
      YIP1 = YIP2
      YIP2 =  TEMP
      IDIR = 1
      GO TO 280
  450 IF(      XIP1-XIP2          ) 475,470,465
  465  TEMP = XIP1
      XIP1 = XIP2
      XIP2 =  TEMP
       TEMP = YIP1
      YIP1 = YIP2
      YIP2 =  TEMP
      IDIR = 1
      GO TO 280
  470 IF(      YIP1-YIP2          ) 475,475,465
  475 IDIR = -1
      GO TO 280
  270 IF(      YIP1-YIP2          ) 265,275,275
  275 IDIR = -1
  280 DIST = SQRT((XIP1-XIP2)**2+(YIP1-YIP2)**2)
      IF(ICURV-1) 290,290,291
  291 A = (ASIN((YIP2-YIP1)/DIST))*180./3.1415
      GO TO 292
  290 A = (ASIN((YIP1-YIP2)/DIST))*180./3.1415
  292 GAMA = DEG/2.
      DMEG = 90.-GAMA
      DMEGR = DMEG*3.1415/180.
      DIST2 = DIST/2.
      R = DIST2/COS(DMEGR)
      IF(ICURV-1) 260,260,460
  460 B1 = A+DMEG+180.
      GO TO 261
  260 B1 = A+DMEG
  261 B2 = B1+DEG
      IF ( IDIR-1 )278,276,276
  276 CALL CIRCLE(XIP2,YIP2,B2,B1,R,R,DI)
      GO TO 279
  278 CALL CIRCLE(XIP1,YIP1,B1,B2,R,R,DI)
  279 D = .10
      IND = IFLAG
      NDPLT = 1
      IF(IND-1)200,200,201
  201 IND = IND-1
      GAMAR = GAMA*3.1415/180.
       AM = DIST2*TAN(GAMAR/2.)
       DO 295 JJ = 1,IND
       IDIR = -1*IDIR
       AN = D+AM
       IF(AN-DIST2) 283,283,281
  283  R2 = (AN**2 + DIST2**2)/(2.*AN)
       IF(R2-DIST2) 281,282,282
  282  PHI = (ASIN(DIST2/R2))*180./3.1415
       T1 = B1+GAMA-PHI
       T2 = T1+2.*PHI
       WRITE(IWRITE,6002)   JJ,PHI,T1,T2
 6002  FORMAT(9X, I4,3F10.4)
       GO TO 285
  281  NDPLT = JJ
       GO TO 200
  285  IF ( IDIR-1) 287,286,286
  286  CALL CIRCLE(XIP2,YIP2,T2,T1,R2,R2,DI)
       GO TO 288
  287  CALL CIRCLE(XIP1,YIP1,T1,T2,R2,R2,DI)
  288  D = D+.10
  295  CONTINUE
       NDPLT = IND+1
  200  RETURN
       END
```

```
      COMMON /ND/ XCOORD(1000), YCOORD(1000), NUM (1000),
     1 ISPEC(1000), NODES
      COMMON /LK/ ITRANS(1100), IREC(1100),             IDUP(1100), IND
      COMMON /CMDR/ ICOM(500),ISUB(500),IHOME1(500),IHOME2(500), NS
      COMMON /ECON/ IPSUB(500),AVX(500),AVY(500),IZON2(500),IZON4(500),
     1 IZON6(500),IZON8(500),
     2              IHOM(5), YLEFT(200), XTOP(200), YRITE(200),
     3 XBOTOM(200), IPT, NCOM, NSCRIB, IBLANK, IOUT,L
      DIMENSION ICONN(1000), ISTAT(1100), ISTACK(800), IPNT(1100),
     1              NULNKS(1100)
      IOUT = 6
      REWIND 8
      READ(8), NODES,(YCOORD(I), XCOORD(I), IBUFF,NUM(I), I=1,NODES)
      READ(3), IND,( IBUFF, IBUFF, ITRANS(I), IBUFF, IBUFF, IREC(I),
     1 IBUFF, IDUP(I), NULNKS(I), I=1,IND)
      PRINT 15, (I, NUM(I),XCOORD(I),YCOORD(I),I=1,NODES)
   15 FORMAT(1H1, 23X, 'ORIGINAL NODE TABLE'//
     1 2X, 'NO.', 4X, 'NODE NO.', 5X, 'X COORD.', 3X, 'Y COORD.',//
     2 (1X, I4, 7X, I4, 7X, F6.1, 2X, F6.1))
      PRINT 5
    5 FORMAT(1H1, 23X, 'ORIGINAL LINK TABLE'//)
      PRINT 201
      PRINT 202, (I, NULNKS(I), ITRANS(I), IREC(I), IDUP(I), I=1,IND)
C     SORT NODES ON MINOR SORT Y ALTERNATING THE ORDER FOR SUCCESSIVE X'S.
C     NODES ARE ASSUMED TO BE IN ASCENDING SORT ORDER ON X AND Y.
      J = 1
  580 TEMP = XCOORD(J)
      DO 510 I=J,NODES
      IBEG = I
      IF(XCOORD(I) .NE. TEMP) GO TO 520
  510 CONTINUE
      GO TO 530
  520 TEMP = XCOORD(IBEG)
      DO 540 I=IBEG,NODES
      K = I
      IF(XCOORD(I) .NE. TEMP) GO TO 550
  540 CONTINUE
      J = K+1
      GO TO 551
  550 J = K
  551 IEND = J-1
      IF(IEND .EQ. IBEG) GO TO 570
C     INVERT RECORDS.
      L = IBEG
  554 M=IEND
      TEMP1 = XCOORD(IEND)
      TEMP2 = YCOORD(IEND)
      ITEMP = NUM(IEND)
  553 XCOORD(M) = XCOORD(M-1)
      YCOORD(M) = YCOORD(M-1)
      NUM(M) = NUM(M-1)
      M = M-1
      IF(M .LE. L) GO TO 552
      GO TO 553

  552 XCOORD(L) = TEMP1
      YCOORD(L) = TEMP2
      NUM(L) = ITEMP
      L = L+1
      IF(L .LT. IEND) GO TO 554
C     ADJUST NODES IN LINK TABLE.
      ITERM = IBEG + IEND
  556 DO 650 I=1,IND
      IF(ITRANS(I).GE.IBEG.AND.ITRANS(I).LE.IEND) ITRANS(I) = ITERM -
     1 ITRANS(I)
      IF(IREC(I).GE. IBEG.AND.IREC(I).LE.IEND) IREC(I)=ITERM-
     1IREC(I)
  650 CONTINUE
  570 IF(J .GE. NODES) GO TO 530
      GO TO 580
C     TRACE NETWORK FOR REORDERING LINKS.
C     FIND CONNECTIVITY OF NODES.
  530 DO 30 I=1,IND
   30 ISTAT(I) = 'OUT'
      DO 31 I=1,NODES
   31 ICONN(I) = 0
```

```
      DO 32 I=1,IND
      J = ITRANS(I)
      IF(J .GT. NODES) GO TO 1000
      ICONN(J) = ICONN(J) + 1
      J = IREC(I)
      IF(J .GT. NODES) GO TO 1000
   32 ICONN(J) = ICONN(J) + 1
C     COUNT NO. OF NODES WITH 0 CONNECTIVITY.
      IJUMP = 0
      DO 34 I = 1,NODES
      IF(ICONN(I) .EQ. 0) IJUMP = IJUMP + 1
   34 CONTINUE
      PRINT 105, IJUMP
  105 FORMAT(2X, I5, 2X, 'NODES HAVE ZERO CONNECTIVITY.'/)
      ICNT = 0
      IP = 0
      IL = 0
   40 IFLAG = 0
      ICNT = ICNT + 1
C     LOOK FOR A STARTING NODE - CONNECTIVITY GREATER THAN 0.
      DO 41 I= 1,NODES
      IF(ICONN(I) .EQ.0) GO TO 41
      J = I
      GO TO 42
   41 CONTINUE
      PRINT 106
  106 FORMAT(2X, 'ERROR - NO TERMINAL NODE FOUND. LINK PRESENT THAT HAS
     1 NOT BEEN TRACED.')
      PRINT 500, (I, ICONN(I), I=1,NODES)
  500 FORMAT(7(5X, I5, 2X, I5))
      STOP
C     START TRACING TREE.
   42 IP = IP + 1
      ISTACK(IP) = J
   46 DO 43 I=1,IND
      IF(ISTAT(I) .NE. 'OUT') GO TO 43
      IF(ITRANS(I) .EQ. J) GO TO 44
      IF(IREC(I) .NE. J) GO TO 43
      K = ITRANS(I)
      ITRANS(I) = IREC(I)
      IREC(I) = K
      GO TO 45
   44 K = IREC(I)
   45 ISTAT(I) = 'IN'
      ICONN(J) = ICONN(J) - 1
      ICONN(K) = ICONN(K) - 1
      J = K
      IL = IL + 1
      IPNT(IL) = I
      GO TO 42
   43 CONTINUE
C      END OF A PATH REACHED.
C     TRACE BRANCH BACKWARDS TO FIRST BRANCHING NODE.
   47 J = ISTACK(IP)
      IF(ICONN(J) .GT. 0) GO TO 46
      IP = IP - 1
      IF(IP .LE. 0) GO TO 50
      GO TO 47
C     NETWORK IS COMPLETELY TRACED. SEE IF THERE IS A DISCONNECTED SUBNETWORK.
   50 CONTINUE
C  50 PRINT 51, ICNT
C  51 FORMAT(1H1, 2X, 'CONNECTED SUBNETWORK ', I3// 15X, 'LINK NO.', 5X,
C    1 'TRANSMITTER', 5X, 'RECEIVER'//)
      NO = 0
      DO 52 I=1,IND
      IF(ISTAT(I) .EQ. 'OUT') IFLAG = 1
      IF(ISTAT(I) .NE. 'IN') GO TO 52
      ISTAT(I) = 'USED'
      NO = NO + 1
   52 CONTINUE
      IF(IFLAG .EQ. 1) GO TO 40
   51 FORMAT(2X, 'THERE ARE ',I3, ' SUBNETWORKS.'//)
      PRINT 51, ICNT
C     SORT LINKS TO BE PLOTTED IN MAIN TABLE.
  910 DO 920 I=1,IL
      ITEMPT = ITRANS(I)
      ITEMPR = IREC(I)
      ITEMPD = IDUP(I)
      ITEMPN = NULNKS(I)
      J = IPNT(I)
      ITRANS(I) = ITRANS(J)
      IREC(I) = IREC(J)
      IDUP(I) = IDUP(J)
      NULNKS(I) = NULNKS(J)
```

```
      ITRANS(J) = ITEMPT
      IREC(J) = ITEMPR
      IDUP(J) = ITEMPD
      NULNKS(J) = ITEMPN
      L = I+1
      IF(L .GT. IL) GO TO 920
      DO 930 K=L,IL
      IF(IPNT(K) .EQ. I) IPNT(K) = J
  930 CONTINUE
  920 CONTINUE
      IND = IL
      CALL SUBSRB
C     PRINT FINAL NODE AND LINK TABLE
      PRINT 300, (I, NUM(I), XCOORD(I), YCOORD(I),ISPEC(I),I=1,NODES)
  300 FORMAT(1H1,16X,'FINAL NODE TABLE'// 2X, 'NO.',
     1 4X, 'NODE NO.', 5X, 'X COORD.', 3X,'Y COORD.',13X,'HOMING NODE'//
     2 (1X,I4,7X,I4,7X,F6.1,2X,F6.1,18X,I1))
      PRINT 400
  400 FORMAT(1H1, 43X, 'FINAL LINK TABLE'//)
      PRINT 201
  201 FORMAT(15X,'LINK NO', 12X, 'TRANSMITTING NODE', 23X,
     1 'RECEIVING NODE', 24X, 'NO. OF LINKS'//)
      PRINT 202, (I,NULNKS(I),ITRANS(I),IREC(I),IDUP(I),I=1,IND)
  202 FORMAT((2X, I4, 12X, I4, 19X, I4, 36X, I4, 35X, I4))
C     WRITE(3, 960) (NUM(I), XCOORD(I), YCOORD(I),ISPEC(I),I,I=1,NODES)
  960 FORMAT(I5,30X, 2F5.1, 14X, I1, 15X, I5)
      WRITE(3, 970) (NULNKS(I), ITRANS(I), IREC(I), IDUP(I), I, I=1,IND)
  970 FORMAT(3I5, 48X, I2, 5X, I5)
      CALL SUB2
      STOP
 1000 PRINT 1001, J
 1001 FORMAT(2X, 'NODE ', I5, ' NOT FOUND IN NODE TABLE.')
      STOP
      END
```

```
      SUBROUTINE SUBSRB
      COMMON /ND/ XCOORD(1000), YCOORD(1000), NUM (1000),
     1 ISPEC(1000), NODES
      COMMON /CMDR/ ICOM(500),ISUB(500),IHOME1(500),IHOME2(500), NS
      COMMON /ECON/ IPSUB(500),AVX(500),AVY(500),IZON2(500),IZON4(500),
     1 IZON6(500),IZON8(500),
     2             IHOM(5), YLEFT(200), XTOP(200), YRITE(200),
     3 XBOTOM(200), IPT, NCOM, NSCRIB, IBLANK, IOUT,L
      DIMENSION ITEMP2(25), ITEMP4(25), ITEMP6(25), ITEMP8(25)
      DIMENSION NAMECD(25)
      DATA/IBLANK/ 6H
      READ (8), NS, (ICOM(I),ISUB(I),BUFF,BUFF,IHOME1(I),BUFF,BUFF,
     1 IHOME2(I), BUFF, I=1,NS)
C     TRANSLATE HOMINGS TO REORDERED SEQUENCE NO. FOR NODES.
C     MARK NODES THAT ARE HOMINGS
      DO 11 I=1,NODES
   11 ISPEC(I) = 0
      DO 1 I=1,NS
      J=IHOME1(I)
      DO 2 K=1,NODES
      IF(J .NE. NUM(K)) GO TO 2
      IHOME1(I) = K
      ISPEC(K) = 1
      GO TO 3
    2 CONTINUE
      WRITE(IOUT,1000) J
 1000 FORMAT(2X,32HCANNOT FIND TRANSLATION FOR NODE, I4)
      STOP
    3 J=IHOME2(I)
      IF(J .EQ. -1) GO TO 1
      DO 4 K=1,NODES
      IF(J .NE. NUM(K)) GO TO 4
      IHOME2(I) = K
      ISPEC(K) = 1
       GO TO 1
    4 CONTINUE
      WRITE(IOUT, 1000) J
      STOP
    1 CONTINUE
      RETURN
      ENTRY SUB2
C     ***********************************************************************
C     CONSTANTS FOR PLOT
```

```
      XUNITS = 12.0
      YUNITS = 9.0
      XIN = 48.0
      YIN = 27.0
      BORDIN = 0.5
C     *********************************************************************
      WRITE(IOUT,340)
  340 FORMAT(1H1, 36X, 'INITIAL SUBSCRIBER TABLE'/)
      WRITE(IOUT,201)
  201 FORMAT(7X,'COMMANDER',3X,'SUBSCRIBER',12X,'HOMEING 1',24X,
     1 'HOMEING 2'//)
      WRITE(IOUT,202) (I, ICOM(I), ISUB(I), IHOME1(I), IHOME2(I),I=1,NS)
  202 FORMAT((2X, I3, 4X, A3, 10X, A3, 17X, I4, 28X, I4))
      NAMEPT = 0
      NPS = 0
      NCOM = IBLANK
      NSCRIB = IBLANK
C     SET UP SUBSCRIBER TABLE WITH AV. X AND AV. Y VALUES OF HOMINGS AS THE
C     REPRESENTATIVE POINTS.
C     SET UP A TABLE WITH ALPHA NAMES OF COMMANDERS.
      DO 10 I=1,NS
      L = IHOME1(I)
      X1 = XCOORD(L)
      Y1 = YCOORD(L)
      IF(ICOM(I) .EQ. NCOM) GO TO 5
C     NEW SUBSCRIBER
      NCOM = ICOM(I)
      NAMEPT = NAMEPT + 1
      NAMECD(NAMEPT) = NCOM
      GO TO 6
    5 IF(ISUB(I) .EQ. NSCRIB) GO TO 7
    6 NSCRIB = ISUB(I)
      IF(NPS .EQ. 0) GO TO 8
      AVX (NPS) = TX/CNT
      AVY(NPS) = TY/CNT
    8 NPS = NPS + 1
      TX = 0.0
      TY = 0.0
      CNT = 0.0
      IPSUB(NPS) = I
    7 TX = TX + X1
      TY = TY + Y1
      CNT = CNT + 1.0
      IF(IHOME2(I) .EQ. -1) GO TO 10
      L = IHOME2(I)
      X1 = XCOORD(L)
      Y1 = YCOORD(L)
      TX = TX + X1
      TY = TY + Y1
      CNT = CNT + 1.0
   10 CONTINUE
      AVX(NPS) = TX/CNT
      AVY(NPS) = TY / CNT
C     SORT SUBSCRIBER TABLE ON AV. X AND AV. Y VALUES
      J = NPS - 1
      DO 83 I=1,J
      LPT = IPSUB(I)
      TAVX = AVX(I)
      TAVY = AVY(I)
      K = I+1
      DO 84 L = K,NPS
      IF(AVX(L) - TAVX) 85, 86, 84
   86 IF(AVY(L) - TAVY) 85, 84, 84
   85 TAVX2 = TAVX
      TAVY2 = TAVY
      LPT2 = LPT
      TAVX = AVX(L)
      TAVY = AVY(L)
      LPT = IPSUB(L)
      AVX(L) = TAVX2
      AVY(L) = TAVY2
      IPSUB(L) = LPT2
   84 CONTINUE
      IPSUB(I) = LPT
      AVX(I) = TAVX
      AVY(I) = TAVY
   83 CONTINUE
      PRINT 2000, (I,IPSUB(I), AVX(I), AVY(I), I=1,NPS)
 2000 FORMAT(1H1, 2X, 'POINTER TO SUBSCRIBERS', 2X, 'AVER. X', 2X,
     1 'AVER. Y'/ (3X, I3,6X,I3,13X,F5.2, 3X, F5.2))
C     COLLECT SUBSCRIBERS IN APPROPRIATE ZONES.
C     THE ZONE IS DETERMINED BY FINDING THE SHORTEST DISTANCE FROM THE
C     SUBSCRIBER TO ITS AVERAGE HOMING POINT, ASSUMING THE SUBSCRIBER LIES
C     ON THE HORIZONTAL OR VERTICAL LINE WITH THE AVERAGE HOMING POINT.
C     EQUATIONS FOR THE DIVIDING LINES IN THE CORNERS ARE AS FOLLOWS -
C            BETWEEN ZONE 2 AND ZONE 8 -
```

```
C                X = Y                          OR    Y = X
C            BETWEEN ZONE 2 AND ZONE 4 -
C                X = 27 - Y                     OR    Y = -X + 27
C            BETWEEN ZONE 4 AND ZONE 6 -
C                48 - X = 27 - Y                OR    Y = X - 21
C            BETWEEN ZONE 6 AND ZONE 8 -
C              48 - X = Y          .            OR    Y = -X + 48
C       NOTE - THE VARIABLES ARE MEASURED IN INCHES.
        DELTA = XIN - YIN
        INPXUN = XIN / XUNITS
        INPYUN = YIN / YUNITS
        IZ2 = 0
        IZ4 = 0
        IZ6 = 0
        IZ8 = 0
C       COMPUTE BOUNDARY VALUES FOR X ALONG SIDE BORDERS
C       THE MAX. VALUE = ONE HALF THE Y DIMENSION IN INCHES.
        XMAX = (YIN / 2) *(XUNITS / XIN)
        XMIN = XUNITS - XMAX
        DO 12 I=1,NPS
        IF(AVX(I) - XMAX) 13, 13, 14
     13 IF(AVY(I) - YUNITS/2) 15, 16, 16
C       POINTS FALL IN ZONE 2 OR ZONE 8.
C       DETERMINE THE LOCATION RELATIVE TO THE DIAGONAL.
     15 Y = AVX(I) * INPXUN
        IF(AVY(I) * INPYUN - Y) 17, 18, 18
     17 IZ8 = IZ8 + 1
        IZON8(IZ8) = I
        GO TO 12
     18 IZ2 = IZ2 + 1
        IZON2(IZ2) = I
        GO TO 12
C       POINTS FALL IN ZONE 2 OR ZONE 4.
C       DETERMINE THE LOCATION RELATIVE TO THE DIAGONAL.
     16 Y = YIN - AVX(I) * INPXUN
        IF(AVY(I) * INPYUN - Y) 18, 18, 20
     20 IZ4 = IZ4 + 1
        IZON4(IZ4) = I
        GO TO 12
     14 IF(AVX(I) - XMIN) 21, 28, 28
     21 IF(AVY(I) - YUNITS / 2) 17, 20, 20
     28 IF(AVY(I) - YUNITS/2) 33, 36, 36
C       POINTS FALL IN ZONE 6 OR ZONE 8.
C       DETERMINE THE LOCATION RELATIVE TO THE DIAGONAL.
     33 Y = XIN - AVX(I) * INPXUN
        IF(AVY(I) * INPYUN - Y) 17, 35, 35
     35 IZ6 = IZ6 + 1
        IZON6(IZ6) = I
        GO TO 12
C       POINTS FALL IN ZONE 4 OR ZONE 6.
C       DETERMINE THE LOCATION RELATIVE TO THE DIAGONAL.
     36 Y = AVX(I) * INPXUN - DELTA
        IF(AVY(I) * INPYUN - Y) 35, 35, 20
     12 CONTINUE
    410 FORMAT((9(4X, I3, 2X, I3)))
C       COMPUTE NORMAL DISTRIBUTION AROUND BORDER..
        TOTDIS = 2 * (YIN + XIN + 4*BORDIN)
        DINT = TOTDIS / NPS
        CY = YUNITS / YIN
        CX = XUNITS / XIN
        DINTX = DINT * CX
        IYL = 0
        IXT = 0
        IYR = 0
        DINTY = DINT * CY
        IXB = 0
        XLEFT = (-BORDIN) * CX
        YTOP =(YIN + BORDIN) * CY
        XRITE =(XIN + BORDIN) * CX
        YBOTOM =(-BORDIN) * CY
C       COMPUTE COORDINATES FOR LEFT BORDER.
        CNSTNT = YBOTOM
        CNT = 0.0
        COORD = CNSTNT
     37 IYL = IYL + 1
        YLEFT(IYL) = COORD
        CNT = CNT + 1.0
        COORD = CNSTNT + CNT * DINTY
        IF(COORD - YTOP) 37, 38, 38
C       COMPUTE COORDINATES FOR TOP BORDER.
     38 DIFF =(COORD - YTOP) * CX/CY
        CNSTNT = XLEFT + DIFF
        CNT = 0.0
        COORD = CNSTNT
     39 IXT = IXT + 1
        XTOP(IXT) = COORD
```

```
      CNT = CNT + 1.0
      COORD = CNSTNT + CNT * DINTX
      IF(COORD - XRITE) 39, 40, 40
C     COMPUTE COORDINATES FOR RIGHT BORDER.
   40 DIFF =(COORD - XRITE) * CY/CX
      CNSTNT = YTOP - DIFF
      CNT = 0.0
      COORD = CNSTNT
   41 IYR = IYR + 1
      YRITE(IYR) = COORD
      CNT = CNT + 1.0
      COORD = CNSTNT - CNT * DINTY
      IF(COORD - YBOTOM) 42, 42, 41
C     COMPUTE COORDINATES FOR BOTTOM BORDER.
   42 DIFF =(YBOTOM - COORD) * CX/CY
      CNSTNT = XRITE - DIFF
      CNT = 0.0
      COORD = CNSTNT
   43 IXB = IXB + 1
      XBOTOM(IXB) = COORD
      CNT = CNT + 1.0
      COORD = CNSTNT - CNT * DINTX
      IF(COORD - XLEFT) 44, 44, 43
C     END OF SUBSCRIBER COORDINATES ASSIGNMENT.
   44 IF(NPS .NE. (IYL+IXT+IYR+IXB)) IXB = IXB - 1
      IF(NPS .NE. (IYL+IXT+IYR+IXB)) STOP
C     FIND DEVIATIONS FROM NORM FOR EACH SIDE.
      IDEV2 = IZ2 - IYL
      IDEV4 = IZ4 - IXT
      IDEV6 = IZ6 - IYR
      IDEV8 = IZ8 - IXB
      PRINT 52, IYL, IXT, IYR, IXB
   52 FORMAT(2X, 'NORMAL DISTRIBUTION IS  ', 4I5)
      PRINT 53, IDEV2,IDEV4,IDEV6,IDEV8
   53 FORMAT(2X, 'DEVIATIONS FROM NORM ARE  ', 4I5)
      PRINT 55 , IZ2, IZ4, IZ6, IZ8
   55 FORMAT(2X, 'FIXED VALUES IN ZONES ARE ', 4I5)
C     PUNCH CARDS FOR COMMANDERS
C     ASSIGN COMMANDER LOCATIONS CLOSEST TO THE AVERAGE HOMINGS
      CALL SORT(IZON2, IZ2, 1)
      CALL SORT(IZON4, IZ4, 2)
      CALL SORT(IZON6, IZ6, 3)
      CALL SORT(IZON8, IZ8, 4)
 3000 FORMAT (2X,I3,2X,2A6,5X,2F10.2)
      PRINT 3001
 3001 FORMAT('0', 20X, 'ZONE 2 - BEFORE ADJUSTING'/)
      DO 3002 I = 1,IZ2
      CALL PRECRD (I, IZON2)
      PRINT 3000, I, NCOM, NSCRIB, AVX(L), AVY(L)
 3002 CONTINUE
      PRINT 3003
 3003 FORMAT('0', 20X, 'ZONE 4 - BEFORE ADJUSTING'/)
      DO 3004 I = 1,IZ4
      CALL PRECRD (I, IZON4)
      PRINT 3000, I, NCOM, NSCRIB, AVX(L), AVY(L)
 3004 CONTINUE
      PRINT 3005
 3005 FORMAT('0', 20X, 'ZONE 6 - BEFORE ADJUSTING'/)
      DO 3006 I = 1,IZ6
      CALL PRECRD (I, IZON6)
      PRINT 3000, I, NCOM, NSCRIB, AVX(L), AVY(L)
 3006 CONTINUE
      PRINT 3007
 3007 FORMAT('0', 20X, 'ZONE 8 - BEFORE ADJUSTING'/)
      DO 3008 I = 1,IZ8
      CALL PRECRD (I, IZON8)
      PRINT 3000, I, NCOM, NSCRIB, AVX(L), AVY(L)
 3008 CONTINUE
      CALL PULCOM(IZON2, ITEMP2, IZ2, IPT2)
      CALL PULCOM(IZON4, ITEMP4, IZ4, IPT4)
      CALL PULCOM(IZON6, ITEMP6, IZ6, IPT6)
      CALL PULCOM(IZON8, ITEMP8, IZ8, IPT8)
C     LEFT BORDER
      WRITE(IOUT, 400)
      WRITE(IOUT, 401)
C     WRITE(3, 318) IPT2
      LEAD = 1
  422 ICNT = 0
      L = ITEMP2(LEAD)
      TAVY = AVY(L)
C     LOOK FOR THE CLOSEST ASSIGNMENT TO COMMANDER
      DO 413 I=1,IYL
      J = I
      IF(YLEFT(I) - TAVY) 413, 415, 416
  416 IF(J-1) 415, 415, 417
  417 IF((YLEFT(I) - TAVY) - (TAVY - YLEFT(I-1)))   415, 415, 418
```

```
  418 J = I-1
      GO TO 415
  413 CONTINUE
  415 IF(J.EQ.1 .OR. J.EQ.IYL) GO TO 419
      IF(LEAD .GE. IPT2) GO TO 419
C     LOOK FOR A COMMANDER WITH A DUPLICATE AVERAGE Y COORDINATE.
      K = ITEMP2(LEAD + 1)
      IF(TAVY - AVY(K)) 419, 420, 419
  420 ICNT = 1
      J = J-1
  419 CALL PRECRD (LEAD, ITEMP2)
C     WRITE(3,223) NCOM, NSCRIB, XLEFT, YLEFT(J), IPT
C     WRITE(3,318) (IHOM(I), I=1,IPT)
      WRITE(IOUT,402) LEAD,NCOM,NSCRIB,XLEFT,YLEFT(J),AVX(L),AVY(L),
     1 (IHOM(I), I=1,IPT)
      YLEFT(J) = 0
      LEAD = LEAD + 1
      IF(ICNT .EQ. 0) GO TO 421
      ICNT = 0
      J = J+2
      GO TO 419
  421 CALL PUSHUP(YLEFT,IYL)
      IF(LEAD .LE. IPT2) GO TO 422
C     TOP BORDER
      WRITE(IOUT, 403)
      WRITE(IOUT, 401)
C     WRITE(3, 318) IPT4
      LEAD = 1
  432 ICNT = 0
      L = ITEMP4(LEAD)
      TAVX = AVX(L)
C     LOOK FOR THE CLOSEST ASSIGNMENT TO COMMANDER
      DO 423 I=1,IXT
      J = I
      IF(XTOP(I) - TAVX) 423, 425, 426
  426 IF(J-1) 425, 425, 427
  427 IF((XTOP(I) - TAVX) - (TAVX - XTOP(I-1))) 425, 425, 428
  428 J = J-1
      GO TO 425
  423 CONTINUE
  425 IF(J.EQ.1 .OR. J.EQ.IXT) GO TO 429
      IF(LEAD .GE. IPT4) GO TO 429
C     LOOK FOR A COMMANDER WITH A DUPLICATE AVERAGE X COORDINATE.
      K = ITEMP4 (LEAD+1)
      IF(TAVX - AVX(K)) 429, 430, 429
  430 ICNT = 1
      J = J-1
  429 CALL PRECRD (LEAD, ITEMP4)
C     WRITE(3,223) NCOM,NSCRIB,XTOP(J),YTOP,IPT
C     WRITE(3,318) (IHOM(I), I=1,IPT)
      WRITE(IOUT,402) LEAD,NCOM,NSCRIB,XTOP(J),YTOP,AVX(L),AVY(L),
     1 (IHOM(I), I=1,IPT)
      XTOP(J) = 0
      LEAD = LEAD + 1
      IF(ICNT .EQ. 0) GO TO 431
      ICNT = 0
      J = J+2
      GO TO 429
  431 CALL PUSHUP (XTOP, IXT)
      IF(LEAD .LE. IPT4) GO TO 432
C     RIGHT BORDER
      WRITE(IOUT, 404)
      WRITE(IOUT, 401)
C     WRITE(3, 318) IPT6
      LEAD = 1
  442 ICNT = 0
      L = ITEMP6 (LEAD)
      TAVY = AVY(L)
C     LOOK FOR THE CLOSEST ASSIGNMENT TO COMMANDER.
      DO 433 I=1,IYR
      J = I
      IF(YRITE(I) - TAVY) 436, 435, 433
  436 IF(J-1) 435, 435, 437
  437 IF((YRITE(I-1) - TAVY) - (TAVY-YRITE(I))) 438,435,435
  438 J = J-1
      GO TO 435
  433 CONTINUE
  435 IF(J.EQ.1 .OR. J.EQ.IYR) GO TO 439
      IF(LEAD .GE. IPT6) GO TO 439
C     LOOK FOR A COMMANDER WITH A DUPLICATE AVERAGE Y COORDINATE.
      K = ITEMP6(LEAD + 1)
      IF(TAVY - AVY(K)) 439, 440, 439
  440 ICNT = 1
      J = J-1
  439 CALL PRECRD(LEAD,ITEMP6)
C     WRITE(3,223) NCOM,NSCRIB,XRITE,YRITE(J),IPT
```

```
C     WRITE(3,318) (IHOM(I), I=1,IPT)
      WRITE(IOUT,402) LEAD,NCOM,NSCRIB,XRITE,YRITE(J),AVX(L),AVY(L),
     1 (IHOM(I), I=1,IPT)
      YRITE(J) = 0
      LEAD = LEAD + 1
      IF(ICNT .EQ. 0) GO TO 441
      ICNT = 0
      J = J+2
      GO TO 439
  441 CALL PUSHUP(YRITE,IYR)
      IF(LEAD .LE. IPT6) GO TO 442
C     BOTTOM BORDER
      WRITE(IOUT, 405)
      WRITE(IOUT, 401)
C     WRITE (3, 318) IPT8
      LEAD = 1
  452 ICNT = 0
      L = ITEMP8 (LEAD)
      TAVX = AVX(L)
C     LOOK FOR THE CLOSEST ASSIGNMENT TO COMMANDER
      DO 443 I=1,IXB
      J = I
      IF(XBOTOM(I) - TAVX) 446, 445, 443
  446 IF(J-1) 445, 445, 447
  447 IF((XBOTOM(I-1) - TAVX) - (TAVX-XBOTOM(I))) 448, 445, 445
  448 J=J-1
      GO TO 445
  443 CONTINUE
  445 IF(J.EQ.1 .OR. J.EQ.IXB) GO TO 449
      IF(LEAD .GE. IPT8) GO TO 449
C     LOOK FOR A COMMANDER WITH A DUPLICATE AVERAGE X COORDINATE.
      K = ITEMP8(LEAD+1)
      IF(TAVX - AVX(K)) 449, 450, 449
  450 ICNT = 1
      J = J-1
  449 CALL PRECRD (LEAD, ITEMP8)
C     WRITE(3, 223) NCOM,NSCRIB,XBOTOM(J),YBOTOM,IPT
C     WRITE(3, 318) (IHOM(I), I=1,IPT)
      WRITE(IOUT,402) LEAD,NCOM,NSCRIB,XBOTOM(J),YBOTOM,AVX(L),AVY(L),
     1 (IHOM(I), I=1,IPT)
      XBOTOM(J) = 0
      LEAD = LEAD + 1
      IF(ICNT .EQ. 0) GO TO 451
      ICNT = 0
      J = J+2
      GO TO 449
  451 CALL PUSHUP (XBOTOM, IXB)
      IF(LEAD .LE. IPT8) GO TO 452
      IF(IDEV2 .EQ. 0 .AND. IDEV4 .EQ. 0 .AND. IDEV6 .EQ. 0 .AND. IDEV8
     1 .EQ. 0) GO TO 200
C     ROTATE AROUND CORNERS FOR NORMALIZING DISTRIBUTION OF SUBSCRIBERS.
      PRINT 250, IDEV2, IDEV4, IDEV6, IDEV8
  250 FORMAT(2X, 'START ROTATING CLOCKWISE'/ (4(2X, I4)))
C     ***********************************************************************
C     THIS SECTION IS PRESENTLY INTERACTIVE  CAN BE PROGRAMMED TO BE AUTOMATIC.
      J = 34
      CALL ROTATE(IZON8,IZ8,IZON2,IZ2,J,4)
      IDEV8 = IDEV8 - J
      IDEV2 = IDEV2 + J
      J = 14
      CALL ROTATE(IZON4,IZ4,IZON2,IZ2,J,4)
      IDEV4 = IDEV4 - J
      IDEV2 = IDEV2 + J
      J = 14
      CALL ROTATE(IZON8,IZ8,IZON6,IZ6,J,2)
      IDEV8 = IDEV8 - J
      IDEV6 = IDEV6 + J
      J = 8
      CALL ROTATE(IZON6,IZ6,IZON4,IZ4,J,1)
      IDEV6 = IDEV6 - J
      IDEV4 = IDEV4 + J
      GO TO 205
C     ***********************************************************************
      IF(IDEV2 .LE. 0 .OR. IDEV4 .GE. 0) GO TO 101
      J = IDEV4 * (-1)
      IF(IDEV2 .LT. J) J = IDEV2
      CALL ROTATE(IZON2, IZ2, IZON4, IZ4, J, 1)
      IDEV2 = IDEV2 - J
      IDEV4 = IDEV4 + J
  101 IF(IDEV4 .LE. 0 .OR. IDEV6 .GE. 0) GO TO 102
      J = IDEV6 * (-1)
      IF(IDEV4 .LT. J) J=IDEV4
      CALL ROTATE (IZON4, IZ4, IZON6, IZ6, J, 2)
      IDEV4 = IDEV4 - J
      IDEV6 = IDEV6 + J
  102 IF(IDEV6 .LE. 0 .OR. IDEV8 .GE. 0) GO TO 103
```

```
      J = IDEV8 * (-1)
      IF(IDEV6 .LT. J) J=IDEV6
      CALL ROTATE(IZON6, IZ6, IZON8, IZ8, J, 3)
      IDEV6 = IDEV6 - J
      IDEV8 = IDEV8 + J
  103 IF(IDEV8 .LE. 0 .OR. IDEV2 .GE. 0) GO TO 104
      J = IDEV2 * (-1)
      IF(IDEV8 .LT. J) J=IDEV8
      CALL ROTATE (IZON8, IZ8, IZON2, IZ2, J, 4)
      IDEV8 = IDEV8 - J
      IDEV2 = IDEV2 + J
  104 IF(IDEV2 .EQ. 0 .AND. IDEV4 .EQ. 0 .AND. IDEV6 .EQ. 0 .AND. IDEV8
     1 .EQ. 0) GO TO 200
      PRINT 116, IDEV2, IDEV4, IDEV6, IDEV8
  116 FORMAT(2X, 'ROTATE SUBSCRIBERS COUNTERCLOCKWISE'/(4(2X,I4)))
C     ROTATE SUBSCRIBERS COUNTERCLOCKWISE.
      IF(IDEV2 .LE. 0 .OR. IDEV8 .GE. 0) GO TO 105
      J = IDEV8 * (-1)
      IF(IDEV2 .LT. J) J=IDEV2
      CALL ROTATE(IZON2, IZ2, IZON8, IZ8, J, 3)
      IDEV2 = IDEV2 - J
      IDEV8 = IDEV8 + J
  105 IF(IDEV8 .LE. 0 .OR. IDEV6 .GE. 0) GO TO 106
      J = IDEV6 * (-1)
      IF(IDEV8 .LT. J) J=IDEV8
      CALL ROTATE(IZON8, IZ8, IZON6, IZ6, J, 2)
      IDEV8 = IDEV8 - J
      IDEV6 = IDEV6 +J
  106 IF(IDEV6 .LE. 0 .OR. IDEV4 .GE. 0) GO TO 107
      J = IDEV4 * (-1)
      IF(IDEV6 .LT. J) J=IDEV6
      CALL ROTATE(IZON6, IZ6, IZON4, IZ4, J, 1)
      IDEV6 = IDEV6 - J
      IDEV4 = IDEV4 + J
  107 IF(IDEV4 .LE. 0 .OR. IDEV2 .GE. 0) GO TO 108
      J = IDEV2 *(-1)
      IF(IDEV4 .LT. J) J=IDEV4
      CALL ROTATE(IZON4, IZ4, IZON2, IZ2, J, 4)
      IDEV4 = IDEV4 - J
      IDEV2 = IDEV2 + J
  108 IF(IDEV2.EQ.0.AND.IDEV4.EQ.0.AND.IDEV6.EQ.0.AND.IDEV8.EQ.0) GO TO
     1 200
C     LAST ADJUSTMENT BEGINS
      PRINT 109, IDEV2, IDEV4, IDEV6, IDEV8
  109 FORMAT(2X, 'LAST ROTATION BEGINS'/(4(2X, I4)))
      IF(IDEV2 .GT. 0 .AND. IDEV6 .LT. 0) GO TO 110
      IF(IDEV4 .GT. 0 .AND. IDEV8 .LT. 0) GO TO 111
      IF(IDEV6 .GT. 0 .AND. IDEV2 .LT. 0) GO TO 112
      IF(IDEV8 .GT. 0 .AND. IDEV4 .LT. 0) GO TO 113
  205 IF(IDEV2 .EQ. 0 .AND. IDEV4 .EQ. 0 .AND. IDEV6 .EQ. 0 .AND.
     1 IDEV8 .EQ. 0) GO TO 200
  218 PRINT 117, IZ2, IYL, IZ4, IXT, IZ6, IYR, IZ8, IXB, NPS
  117 FORMAT(2X, 'ERROR - ALL DEVIATIONS SHOULD = 0', 9I6)
      STOP
  110 IF((IDEV2 + IDEV6) .NE. 0) GO TO 218
      J = IDEV2 / 2
      IF(J .EQ. 0) GO TO 120
      CALL ROTATE(IZON2, IZ2, IZON4, IZ4, J, 1)
      IDEV2 = IDEV2 - J
      IDEV4 = IDEV4 + J
      CALL ROTATE(IZON4, IZ4, IZON6, IZ6, J, 2)
      IDEV4 = IDEV4 - J
      IDEV6 = IDEV6 + J
  120 J = IDEV2
      CALL ROTATE(IZON2, IZ2, IZON8, IZ8, J, 3)
      IDEV2 = IDEV2 -J
      IDEV8 = IDEV8 +J
      CALL ROTATE(IZON8, IZ8, IZON6, IZ6, J, 2)
      IDEV8 = IDEV8 - J
      IDEV6 = IDEV6 + J
      GO TO 205
  111 IF((IDEV4 + IDEV8) .NE. 0) GO TO 218
      J = IDEV4 / 2
      IF(J .EQ. 0) GO TO 121
      CALL ROTATE(IZON4, IZ4, IZON6, IZ6, J, 2)
      IDEV4 = IDEV4 - J
      IDEV6 = IDEV6 + J
      CALL ROTATE(IZON6, IZ6, IZON8, IZ8, J, 3)
      IDEV6 = IDEV6 - J
      IDEV8 = IDEV8 + J
  121 J = IDEV4
      CALL ROTATE(IZON4, IZ4, IZON2, IZ2, J, 4)
      IDEV4 = IDEV4 - J
      IDEV2 = IDEV2 + J
      CALL ROTATE(IZON2, IZ2, IZON8, IZ8, J, 3)
      IDEV2 = IDEV2 - J
```

```
      IDEV8 = IDEV8 + J
      GO TO 205
  112 IF((IDEV6 + IDEV2) .NE. 0) GO TO 218
      J = IDEV6 / 2
      IF(J .EQ. 0) GO TO 122
      CALL ROTATE(IZON6, IZ6, IZON8, IZ8, J, 3)
      IDEV6 = IDEV6 -J
      IDEV8 = IDEV8 + J
      CALL ROTATE(IZON8, IZ8, IZON2, IZ2, J, 4)
      IDEV8 = IDEV8 - J
      IDEV2 = IDEV2 + J
  122 J = IDEV6
      CALL ROTATE(IZON6, IZ6, IZON4, IZ4, J, 1)
      IDEV6 = IDEV6 - J
      IDEV4 = IDEV4 + J
      CALL ROTATE(IZON4, IZ4, IZON2, IZ2, J, 4)
      IDEV4 = IDEV4 - J
      IDEV2 = IDEV2 + J
      GO TO 205
  113 IF((IDEV8 + IDEV4) .NE. 0) GO TO 218
      J = IDEV8 / 2
      IF(J .EQ. 0) GO TO 123
      CALL ROTATE(IZON8, IZ8, IZON2, IZ2, J, 4)
      IDEV8 = IDEV8 - J
      IDEV2 = IDEV2 + J
      CALL ROTATE(IZON2, IZ2, IZON4, IZ4, J, 1)
      IDEV2 = IDEV2 - J
      IDEV4 = IDEV4 + J
  123 J = IDEV8
      CALL ROTATE(IZON8, IZ8, IZON6, IZ6, J, 2)
      IDEV8 = IDEV8 - J
      IDEV6 = IDEV6 + J
      CALL ROTATE(IZON6, IZ6, IZON4, IZ4, J, 1)
      IDEV6 = IDEV6 - J
      IDEV4 = IDEV4 + J
      GO TO 205
C      SORT ALL ZONES FOR PRINTOUT
  200 CALL SORT(IZON2, IZ2, 1)
      CALL SORT(IZON4, IZ4, 2)
      CALL SORT(IZON6, IZ6, 3)
      CALL SORT(IZON8, IZ8, 4)
C     FIND THE COMMANDERS TO BE PLOTTED ON THIS CHART.
C     THERE WILL BE AT MOST 5 COMMANDERS AND THEIR SUBORDINATES ON A CHART.
      NOSPER = 5
      NOCMDR = NAMEPT
      IEND = 0
      ICHART = 0
C     FIND THE NO. OF CHARTS.
      NOCHRT = NOCMDR / NOSPER
      IREMDR = NOCMDR - NOSPER * NOCHRT
      IF(IREMDR .NE. 0) NOCHRT = NOCHRT + 1
C      FIND THE NO. OF SUBSCRIBERS PER CHART
      NOSPER = NOCMDR / NOCHRT
      IREMDR = NOCMDR - NOSPER * NOCHRT
  460 ICHART = ICHART + 1
      IF(ICHART .GT. NOCHRT) RETURN
      IBEGIN = IEND + 1
      IEND = IEND + NOSPER
      IF(ICHART .LE. IREMDR) IEND = IEND + 1
      IF(IEND .LE. NAMEPT) GO TO 457
      PRINT 456, NAMEPT, IEND
  456 FORMAT(2X, 'ERROR - NAMEPT = ',I3,2X,'IEND = ',I3)
      STOP
C     PUNCH CARDS FOR LEFT BORDER
  457 ICNT = 0
      WRITE(IOUT, 400)
  400 FORMAT(1H1, 35X, 11HLEFT BORDER//)
      WRITE (IOUT, 401)
  401 FORMAT(7X, 9HCOMMANDER, 2X, 11HSUBORDINATE, 8X, 8HX COORD., 7X,
     1 8HY COORD.,2X,'AV. X',1X,'AV. Y', 8X, 7HHOMINGS/)
  318 FORMAT(5I5)
      DO 220 I=1,IZ2
      CALL PRECRD (I, IZON2)
      DO 458 K = IBEGIN, IEND
      IF(NCOM .EQ. NAMECD (K)) GO TO 455
  458 CONTINUE
      GO TO 220
  455 CONTINUE
      ICNT = ICNT + 1
C     WRITE(3, 223) NCOM, NSCRIB, YLEFT, YLEFT(I), IPT
  223 FORMAT(2X, A3, 2X, A3, 25X, 2F10.5, 2X, I3)
C     WRITE(3, 318) (IHOM(J), J=1,IPT)
      WRITE(IOUT,402) ICNT,NCOM,NSCRIB,XLEFT,YLEFT(I),AVX(L),AVY(L),
     1 (IHOM(J), J=1,IPT)
  402 FORMAT(2X,I3,4X,A3,9X,A3,10X,F10.5,6X,F10.5,
     1 2X,F5.2,1X,F5.2,5X,5(I4,2X))
  220 CONTINUE
```

```
C     WRITE (3, 318) ICNT
C     PUNCH CARDS FOR TOP BORDER
      ICNT = 0
      WRITE (IOUT,403)
  403 FORMAT(1H1, 36X, 10HTOP BORDER//)
      WRITE (IOUT, 401)
      DO 224 I=1, IZ4
      CALL PRECRD(I, IZON4)
      DO 501 K = IBEGIN, IEND
      IF(NCOM .EQ. NAMECD (K)) GO TO 502
  501 CONTINUE
      GO TO 224
  502 CONTINUE
      ICNT = ICNT + 1
C     WRITE(3, 223) NCOM, NSCRIB, XTOP(I), YTOP, IPT
C     WRITE(3, 318) (IHOM(J), J=1,IPT)
      WRITE(IOUT,402) ICNT,NCOM, NSCRIB, XTOP(I), YTOP,AVX(L),AVY(L),
     1 (IHOM(J), J=1,IPT)
  224 CONTINUE
C     WRITE (3, 318) ICNT
C     PUNCH CARDS FOR RIGHT BORDER
      ICNT = 0
      WRITE (IOUT, 404)
  404 FORMAT(1H1, 35X, 12HRIGHT BORDER//)
      WRITE (IOUT, 401)
      DO 225 I=1,IZ6
      CALL PRECRD(I, IZON6)
      DO 503 K = IBEGIN, IEND
      IF(NCOM .EQ. NAMECD (K)) GO TO 504
  503 CONTINUE
      GO TO 225
  504 CONTINUE
      ICNT = ICNT + 1
C     WRITE(3, 223) NCOM, NSCRIB, XRITE, YRITE(I), IPT
C     WRITE(3, 318) (IHOM(J), J=1,IPT)
      WRITE(IOUT,402) ICNT, NCOM, NSCRIB, XRITE, YRITE(I),AVX(L),AVY(L),
     1 (IHOM(J), J=1,IPT)
  225 CONTINUE
C     WRITE (3, 318) ICNT
C     PUNCH CARDS FOR BOTTOM BORDER
      ICNT = 0
  405 FORMAT(1H1, 34X, 13HBOTTOM BORDER//)
      WRITE (IOUT, 405)
      WRITE (IOUT, 401)
      DO 226 I=1, IZ8
      CALL PRECRD(I, IZON8)
      DO 505 K = IBEGIN, IEND
      IF(NCOM .EQ. NAMECD (K)) GO TO 506
  505 CONTINUE
      GO TO 226
  506 CONTINUE
      ICNT = ICNT + 1
C     WRITE(3, 223) NCOM, NSCRIB, XBOTOM(I), YBOTOM, IPT
C     WRITE(3, 318) (IHOM(J), J=1,IPT)
      WRITE(IOUT,402) ICNT,NCOM,NSCRIB,XBOTOM(I),YBOTOM,AVX(L),AVY(L),
     1 (IHOM(J), J=1,IPT)
  226 CONTINUE
C     WRITE (3, 318) ICNT
      GO TO 460
      END
```

```
      SUBROUTINE PRECRD(I, IZON)
      COMMON /CMDR/ ICOM(500),ISUB(500),IHOME1(500),IHOME2(500), NS
      COMMON /ECON/ IPSUB(500),AVX(500),AVY(500),IZON2(500),IZON4(500),
     1 IZON6(500),IZON8(500),
     2                IHOM(5), YLEFT(200), XTOP(200), YRITE(200),
     3 XBOTOM(200), IPT, NCOM, NSCRIB, IBLANK, IOUT,L
      DIMENSION IZON(200)
      L = IZON(I)
      LL = IPSUB(L)
      NCOM = ICOM(LL)
      NSCRIB = ISUB(LL)
      IPT = 0
  121 IPT = IPT + 1
      IHOM(IPT) = IHOME1(LL)
      IF(IHOME2(LL) .EQ. -1) RETURN
      IPT = IPT + 1
      IHOM(IPT) = IHOME2(LL)
      LL = LL + 1
      IF(ICOM(LL) .EQ. NCOM .AND. ISUB(LL) .EQ. NSCRIB) GO TO 121
      RETURN
      END
```

```
      SUBROUTINE ROTATE(IFROM, IPFR, ITO, IPTO, NO, IFLAG)
      DIMENSION IFROM(10), ITO(10)
      CALL SORT(IFROM, IPFR, IFLAG)
      DO 1 I=1,NO
      IPTO = IPTO + 1
      ITO(IPTO) = IFROM(IPFR)
    1 IPFR = IPFR - 1
      RETURN
      END
```

```
      SUBROUTINE SORT(IARRAY, IPTA, IFLAG)
      COMMON /FCON/ IPSUB(500),AVX(500),AVY(500)
      DIMENSION IARRAY(IPTA)
C     SORT IS DEPENDENT ON IFLAG
C     IF IFLAG = 1 - ASCENDING Y, ASCENDING X
C     IF IFLAG = 2 - ASCENDING X, ASCENDING Y
C     IF IFLAG = 3 - DESCENDING Y, DESCENDING Y
C     IF IFLAG = 4 - DESCENDING X, DESCENDING Y
      J = IPTA - 1
      IF(J .LE. 0) RETURN
      DO 1 I=1,J
      IP = IARRAY(I)
      TAVX = AVX(IP)
      TAVY = AVY(IP)
      K = I+1
      DO 2 L=K,IPTA
      IPP = IARRAY(L)
      IF(IFLAG .NE. 1) GO TO 10
      IF(AVY(IPP) - TAVY) 3, 4, 2
    4 IF(AVX(IPP) - TAVX) 3, 2, 2
   10 IF(IFLAG .NE. 2) GO TO 20
      IF(AVX(IPP) - TAVX) 3, 5, 2
    5 IF(AVY(IPP) - TAVY) 3, 2, 2
   20 IF(IFLAG .NE. 3) GO TO 30
      IF(AVY(IPP) - TAVY) 2, 6, 3
    6 IF(AVX(IPP) - TAVX) 2, 2, 3
   30 IF(IFLAG .EQ. 4) GO TO 31
      PRINT 32
   32 FORMAT(2X, 'ERROR IN FLAG TO SORT ROUTINE. VALUE MUST BE FROM 1-4.
     1 '/)
      STOP
   31 IF(AVX(IPP) - TAVX) 2, 7, 3
    7 IF(AVY(IPP) - TAVY) 2, 2, 3
    3 TAVX = AVX(IPP)
      TAVY = AVY(IPP)
      IARRAY(L) = IP
      IP = IPP
    2 CONTINUE
      IARRAY(I) = IP
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PULCOM (IZON, ITEMP, IPZ, ITT)
      COMMON /CMDR/ ICOM(500),ISUB(500),IHOME1(500),IHOME2(500), NS
      COMMON /FCON/ IPSUB(500),AVX(500),AVY(500),IZON2(500),IZON4(500),
     1 IZON6(500),IZON8(500),
     2             IHOM(5), YLEFT(200), XTOP(200), YRITE(200),
     3 XBOTOM(200), IPT, NCOM, NSCRIB, IBLANK, IOUT,L
      DIMENSION IZON(10), ITEMP(10)
      DATA/IBLANK/ 6H
      ITT = 0
      DO 1 I=1,IPZ
      L = IZON(I)
      LL = IPSUB(L)
      IF(ISUB(LL) .NE.IBLANK) GO TO 1
      ITT = ITT + 1
      ITEMP(ITT) = L
      IZON(I) = 0
    1 CONTINUE
      CALL PUSHUP (IZON, IPZ)
      RETURN
      END
```

```
      SUBROUTINE PUSHUP (IZON, IPZ)
      DIMENSION IZON(10)
      ICNT = 0
      DO 1 I=1,IPZ
      IF(IZON(I)) 3, 2, 3
    2 ICNT = ICNT + 1
      GO TO 1
    3 IF(ICNT) 1, 1, 4
    4 IZON (I - ICNT) = IZON(I)
    1 CONTINUE
      IPZ = IPZ - ICNT
      RETURN
      END
```

4A (REV. 7-73)

.S. DEPT. OF COMM. ILIOGRAPHIC DATA SHEET	1. PUBLICATION OR REPORT NO. NBS TN-829	2. Gov't Accession No.	3. Recipient's Accession No.

LE AND SUBTITLE

Multicommodity Network Plotting via Program NETPLT

5. Publication Date: June 1974

6. Performing Organization Code: *640.01*

THOR(S)

Zella G. Ruthberg, Gloria R. Bolotsky, William Slater,Jr.

8. Performing Organ. Report No.

RFORMING ORGANIZATION NAME AND ADDRESS

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

10. Project/Task/Work Unit No. *640.2456*

11. Contract/Grant No. *RO 72-78*

onsoring Organization Name and Complete Address *(Street, City, State, ZIP)*

Defense Communications Agency

13. Type of Report & Period Covered *Final*

14. Sponsoring Agency Code

PPLEMENTARY NOTES

Library of Congress Catalog Card Number: 74-600074

STRACT *(A 200-word or less factual summary of most significant information. If document includes a significant liography or literature survey, mention it here.)*

In the design and operation of complex networks, it is often an advantage to obtain a visual representation that readily allows for a quick appraisal of the network's current configuration or of its changed appearance due to variations of its nodes and links. The program NETPLT enables a user to plot the two leading characteristics of any multicommodity network: (1) nodes and links and (2) the multiple source-sink structure (multicommodity property). The unique feature of NETPLT is its unambiguous planar representation of links. NETPLT uses an arc of a circle instead of the usual straight line, to represent a connector between a node pair (link).

Y WORDS *(six to twelve entries; alphabetical order; capitalize only the first letter of the first key word unless a proper ne; separated by semicolons)*

Communication network; multicommodity network; network; network display; plotting algorithm; plotting program.

AILABILITY	19. SECURITY CLASS (THIS REPORT)	21. NO. OF PAGES
[X] Unlimited For Official Distribution. Do Not Release to NTIS	UNCLASSIFIED	67
Order From Sup. of Doc., U.S. Government Printing Office Washington, D.C. 20402, SD Cat. No. C13 .46:829 Order From National Technical Information Service (NTIS) Springfield, Virginia 22151	20. SECURITY CLASS (THIS PAGE) UNCLASSIFIED	22. Price $1.10

USCOMM-DC 29042-P74

GOVERNMENT PRINTING OFFICE: 1974— 543-048:41

NBS TECHNICAL PUBLICATIONS

PERIODICALS

JOURNAL OF RESEARCH reports National Bureau of Standards research and development in physics, mathematics, and chemistry. Comprehensive scientific papers give complete details of the work, including laboratory data, experimental procedures, and theoretical and mathematical analyses. Illustrated with photographs, drawings, and charts. Includes listings of other NBS papers as issued.

Published in two sections, available separately:

- **Physics and Chemistry (Section A)**

Papers of interest primarily to scientists working in these fields. This section covers a broad range of physical and chemical research, with major emphasis on standards of physical measurement, fundamental constants, and properties of matter. Issued six times a year. Annual subscription: Domestic, $17.00; Foreign, $21.25.

- **Mathematical Sciences (Section B)**

Studies and compilations designed mainly for the mathematician and theoretical physicist. Topics in mathematical statistics, theory of experiment design, numerical analysis, theoretical physics and chemistry, logical design and programming of computers and computer systems. Short numerical tables. Issued quarterly. Annual subscription: Domestic, $9.00; Foreign, $11.25.

DIMENSIONS, NBS

The best single source of information concerning the Bureau's measurement, research, developmental, cooperative, and publication activities, this monthly publication is designed for the layman and also for the industry-oriented individual whose daily work involves intimate contact with science and technology —*for engineers, chemists, physicists, research managers, product-development managers, and company executives.* Annual subscription: Domestic, $6.50; Foreign, $8.25.

NONPERIODICALS

Applied Mathematics Series. Mathematical tables, manuals, and studies.

Building Science Series. Research results, test methods, and performance criteria of building materials, components, systems, and structures.

Handbooks. Recommended codes of engineering and industrial practice (including safety codes) developed in cooperation with interested industries, professional organizations, and regulatory bodies.

Special Publications. Proceedings of NBS conferences, bibliographies, annual reports, wall charts, pamphlets, etc.

Monographs. Major contributions to the technical literature on various subjects related to the Bureau's scientific and technical activities.

National Standard Reference Data Series. NSRDS provides quantitative data on the physical and chemical properties of materials, compiled from the world's literature and critically evaluated.

Product Standards. Provide requirements for sizes, types, quality, and methods for testing various industrial products. These standards are developed cooperatively with interested Government and industry groups and provide the basis for common understanding of product characteristics for both buyers and sellers. Their use is voluntary.

Technical Notes. This series consists of communications and reports (covering both other-agency and NBS-sponsored work) of limited or transitory interest.

Federal Information Processing Standards Publications. This series is the official publication within the Federal Government for information on standards adopted and promulgated under the Public Law 89–306, and Bureau of the Budget Circular A–86 entitled, Standardization of Data Elements and Codes in Data Systems.

Consumer Information Series. Practical information, based on NBS research and experience, covering areas of interest to the consumer. Easily understandable language and illustrations provide useful background knowledge for shopping in today's technological marketplace.

BIBLIOGRAPHIC SUBSCRIPTION SERVICES

The following current-awareness and literature-survey bibliographies are issued periodically by the Bureau:

Cryogenic Data Center Current Awareness Service (Publications and Reports of Interest in Cryogenics). A literature survey issued weekly. Annual subscription: Domestic, $20.00; foreign, $25.00.

Liquefied Natural Gas. A literature survey issued quarterly. Annual subscription: $20.00.

Superconducting Devices and Materials. A literature survey issued quarterly. Annual subscription: $20.00. Send subscription orders and remittances for the preceding bibliographic services to the U.S. Department of Commerce, National Technical Information Service, Springfield, Va. 22151.

Electromagnetic Metrology Current Awareness Service (Abstracts of Selected Articles on Measurement Techniques and Standards of Electromagnetic Quantities from D-C to Millimeter-Wave Frequencies). Issued monthly. Annual subscription: $100.00 (Special rates for multi-subscriptions). Send subscription order and remittance to the Electromagnetic Metrology Information Center, Electromagnetics Division, National Bureau of Standards, Boulder, Colo. 80302.

Order NBS publications (except Bibliographic Subscription Services) from: Superintendent of Documents, Government Printing Office, Washington, D.C. 20402.

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
Washington, D.C. 20234

OFFICIAL BUSINESS

Penalty for Private Use, $300

POSTAGE AND FEES PAID
U.S. DEPARTMENT OF COMMERCE
COM-215

U.S.MAIL

AMERICAN REVOLUTION BICENTENNIAL
1776-1976