To the Graduate Council:

I am submitting herewith a thesis written by Graham M. Barrowman entitled "A CAD/CAM Interface for Geographic Databases: A Transportation Database Application." I have examined the final copy of this thesis for form and content and recommend that it is accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Geography.

Bruce A. Ralston, Major Professor

We have read this thesis
and recommend its
acceptance:

James R. Carter

Thomas L Bell

Accepted for the Council:

Vice Provost
and Dean of the Graduate School

A CAD/CAM INTERFACE FOR GEOGRAPHIC DATABASES: A TRANSPORTATION DATABASE APPLICATION

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Graham M. Barrowman

August 1988

ACKNOWLEDGEMENTS

I would like to thank Dr. Thomas L. Bell and Dr. James R. Carter for their helpful comments and assistance during the completion of this research. In particular, I would like to thank Dr. Bruce A. Ralston who inspired this research, and who dedicated a great deal of time and effort to make it possible.

Dedicated to my parents,

Clifford and Doreen Barrowman,

and my brother, Clifford.

ABSTRACT

Transportation corridors are defined by a beginning and end point, and a corridor radius. These corridors contain all or part of the transportation network within this area. In this paper a method of extracting and displaying transportation corridors from a geographic database is outlined.

In particular, the extraction of corridors containing both transportation network elements, nodes and links, and the calculation of the corridor limits are discussed. Similarly, problems of displaying multi-modal transportation networks and the methods used to overcome these problems, by interfacing the geographic database with a Computer-Aided Design (CAD) package, are outlined.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	GEOGRAPHIC DATABASES	1
	Introduction to Geographic Databases	3
	Models for Storing and Processing Geographic Data	5
	Map Processing Models	9
	Variance in the Content of Geographic Databases	12
	Transportation Network Databases	12
II.	THE AFRICAN TRANSPORTATION NETWORK DATABASE AND RESEARCH ENVIRONMENT	17
	The African Transportation Network Database	18
	The Research Environment	23
III.	THE DEFINITION OF TRANSPORTATION CORRIDORS	25
	Sample Problem	26
	Methodology: Determining the Distance Between Points and Lines	28
	Coordinate Distortion and Adjustments	30
	The Calculation and Comparison of Distance	33
	The Problem of Infinite Length	35
	Definition of the Transportation Corridor Buffer	38
	Calculation of the Straight Line Segments of Buffer	40
	Calculation of the Semi-Circular Ends of the Buffer	49
	Program Design	49

CHAPTER		PAGE
IV.	LINK TRANSPORTATION CORRIDORS AND CORRIDOR ATTRIBUTES	51
	Nodes, Links, and Transshipment Points	52
	Extraction Arrays and Counters	53
	Transshipment Point Corridors	54
	Transport Link Corridors	55
	The Extraction of Corridor Attributes	59
	Network Attributes and Map Symbols	60
	Transportation Corridor Display	62
	Graphics Subroutine Libraries	62
V.	A CAD/CAM INTERFACE FOR GEOGRAPHIC DATABASES	66
	Prodesign II Drawing Files	68
	Prodesign II and the Relational Map Model .	72
	On-Screen Editing Capabilities	76
	Creating a Hard-copy Using Prodesign II ...	77
	Plotting More Than One Link In Prodesign II	77
	Program Diskette	81
VI.	CONCLUSION	82
	BIBLIOGRAPHY	86
	APPENDICES	89
	APPENDIX ONE The African Transportation Network Database: Database Structure	90
	APPENDIX TWO Prodesign II Drawing Files..	96
	VITA	98

TABLE OF FIGURES

FIGURE		PAGE
3.1	Area of Sample Problem and Link Being Analyzed	27
3.2	The Link AB, Node P, and Point Q	29
3.3	Cases 3 and 4	34
3.4	Re-assigning the Point Q	37
3.5	Transportation Corridors With and Without Buffer	39
3.6	Straight Line Segments of the Buffer Cases 1 and 2	41
3.7	Straight Line Segment of Buffer Case 3	45
3.8	Straight Line Segment of Buffer Case 4	48
4.1	Transport Link Corridors Cases 1 and 2	56
5.1	Transportation Corridor	75
5.2	Transportation Corridor with Title, Border, and Legend	78
5.3	Connecting Transportation Corridors	80

CHAPTER I

GEOGRAPHIC DATABASES

A Geographic Information System (G.I.S.) is a collection of tools for the collection, storage, analysis, and display of digital geographic information. The core element of a G.I.S. is the digital geographic database. One of the main fields of geographic research at the present time is the development of a set of tools for the processing and display of the contents of these databases. Similar to traditional geographic databases, such as topographic map sheets, the new digital databases are rarely used in their entirety. Questions are more commonly asked of small parts of the database, and the analytical and graphic capabilities of a G.I.S. are used on sub-sets of the database.

Generally speaking, two types of questions can be asked of geographic databases; those which simply require the extraction of data from the database, and those which require the creation and analysis of new data values. The former type of question requires the comparison of the data contained in the database with a value defined by the user, and the extraction of data values which meet a certain user-defined criteria. The latter type of question is more complex, and requires the creation of new

data values using the analytical capabilities of the geographic information system. An example of an analytical query of a database would be: given a line segment, such as a highway, and a series of nodes, list those nodes which are within a user-defined distance of the highway. To answer this question the distance between the link and each individual node has to be calculated. This distance is then compared to the user-defined distance or buffer to determine which nodes are to be extracted from the database.

The objective of this thesis is, similar to the second type of question posed above, to extract information from a geographic database on the basis of newly-created data values. The geographic database to be used is an African transportation network database developed by the Matrix Group, a Washington, D.C. consulting firm. The problem is to extract transportation corridors from this existing database, and, in addition, to display the output from these database queries in an effective way. All of this work has been performed using the computer hardware and software available within the Department of Geography at The University of Tennessee, Knoxville.

Before addressing these questions, it is necessary to discuss the nature and content of geographic databases, and, more specifically, the nature and content of

transportation databases.

1.1 Introduction to Geographic Databases.

Geographic databases represent real world geographic entities as digital features composed of locational information and a set of attribute information. The locational data define the position of geographic features on the ground, and are handled either as a vector database or a raster database. Vector databases define geographic features as a series of straight line segments, whereas raster databases define features as matrices of dots or pixels. For example, a highway in a vector database would be represented by a series of straight line segments connecting the two endpoints of the road. In a raster database the highway would be represented by "turning on" the appropriate pixels in the matrix to form a straight line. Vector databases effectively display a transportation network consisting of a series of nodes connected by straight line links. Raster databases and raster display devices excel at displaying continuous mathematic surface as shades and tones.

The attribute data are the section of the database which describe the features defined by the locational data. Names of regions, names of features, feature types, and characteristic vegetation are all examples of attribute data. This information is related to the

locational data by the distinguishing feature of geographic databases. By linking attributes to the locational data, the data are referenced to a coordinate system. Consider, for example, the representation of a highway in a geographic database. The locational data describes the geography of each link, that is, where it starts and where it ends. The attribute data describe the non-geographic characteristics of each set of locational data, e.g., the length of the link, mode, directionality of the link (one way or two way), and the number of traffic lanes.

Database queries can differ with respect to which section of the database they examine. Questions can be asked solely of the locational data or the attribute data, or, alternatively, questions can be asked which link together these two data types. The most common queries of geographic databases are those which link together the locational and attribute data types. An example would be a query to list all of the highways, of four traffic lanes or more, which lie within a given region. In answering this query of the database correctly, only those highways of four traffic lanes or more which lie partially or entirely within the region should be listed. The link between the locational and attribute data lies in the fact that, to achieve the output of a list of highway names, attribute data had to be searched to find if each highway

segment had four lanes or more, and locational data had to be analyzed to determine whether or not the highway was within the user-specified region.

1.2 Models For Storing and Processing Geographic Data.

Locational data in a geographic database are handled in either the raster or vector fashion; geographic features can be defined by giving the location of the endpoints of a series of straight line segments, or by creating a matrix of pixels. Several methods or models exist for the handling of attribute data. Aronson developed a typology of five models for handling attribute data: the simple tabular model, the hierarchical model, the network model, the relational model, and the object oriented model.[1] Aronson's typology will be used in the following discussion.

The simple tabular model stores data as collections of independent tables consisting of rows and columns. Rows are also known as records, and columns as fields.[2] This model allows the association of attribute data with geographic features. For example, in a transportation database the location and type of several highways could

---

[1]Peter Aronson, "Attribute Handling For Geographic Information Systems," Proceedings Eighth International Symposium On Computer-Assisted Cartography, Baltimore, Maryland, March 29 - April 3, 1987, p.347.

[2]Ibid p.347.

be stored in one data table, and the name of the highways in another. However, each table is independent and cannot be combined when using the simple tabular model. The only way of combining these two tables is to overlay two maps showing the contents of each data table.

It is possible when using the simple tabular data model that independent tables contain the same data, and, as a result, identical variables stored in separate tables can differ. This may result in confusion over which one is correct. The simple tabular model suffers, therefore, in terms of storage efficiency and data integrity. However, this data structure is easy to program and to convert from system to system.[3]

The hierarchical model arranges data in a tree structure. The tree structure is a hierarchy of elements in which every element, with the exception of the uppermost or root element, is related to one other element at a higher level in the tree. This limitation on the number of parents which each data element can have is not always found in geographic data. For example, a node in a transportation network which is created by the intersection of two links will have two parents. As a result, database management systems using the hierarchical model are not widely used in geographic information

---

[3]Ibid. p.347.

systems.[4]

The network model is closely related to the hierarchial model. The data are organized in a plex structure which, like the tree structure, can be described in terms of children and parents. However, in a plex structure an element may have more than one parent, a feature which is more suited to representing some of the relationships found in geographic data. In the example presented to highlight the weakness of the hierarchical model in geographic applications, if the network model was used it would be possible for a node to have more than one parent. However, both network data models and hierarchical data models suffer from the fact that use of these databases requires knowledge of the database structure, and, despite being able to handle large volumes of data, they are confined to a rigid framework of relations.[5]

The relational model arranges the data in tables consisting of rows and columns. The set of values from which the actual values in a column are drawn is called the domain, and each row, or tuple, is a set of permanently associated values. The difference between the

---

[4]Ibid. p.347.

[5]Duane F. Marble and Lalita Sen, "The Development of Standardized Benchmarks For Spatial Database Systems," Proceedings, Second Annual Symposium On Spatial Data Handling, Seattle, Washington, July 5-10, 1986, p. 488.

relational model and the simple tabular model is the fact that the data tables are not independent. Tables can be joined, or related, to one another by columns sharing a common domain. For example, two tables, one containing highway names and mode, and another containing highway names and length, can be joined by their common domain, highway name, to form one table. This table would contain highway names, modes, and length.

Waugh and Healey believe that the relational model is well positioned to become the major storage mechanism for collections of geographic data. Three reasons support this belief. First, the model structure is simplistic; all of the data are stored in simple tables. Secondly, the model has great flexibility. Any set of tables with common domains can be joined together. Thirdly, the model is efficient in its storage of data. Proper design of the tables can eliminate redundancy, or the repetition of information.[6]

The newest and least used of the five data models is the object oriented model. The basic unit of this model is the object, or a collection of data elements which together are considered to be a single geographic entity. Objects are typed, or classified, and the format and

---

[6]T. C. Waugh and R. C. Healey, "The GEOVIEW Design: a relational database approach to geographical data handling," Proceedings, Second International Symposium On Spatial Data Handling, Seattle, Washington, July 5-10, 1986, p.194

operations performed on a certain object are the same as those outlined for an object prototype. A highway object, for example, could be defined by two nodes. In addition, the object would be described by a list of attribute data; e.g., road number, road surface, number of lanes, and length. The methods used to symbolize the feature, such as color and line type, would also be included. Unlike the simple tabular model the object oriented approach does not store each geographic element as an individual entity. Objects are created by the collection of geographic elements. This approach to the handling of a geographic database has the advantage that objects can be bundled together with attributes at the user's discretion. This feature allows the querying of the database to be very natural. For example, what is the length of the highway object?

1.3 Map Processing Models.

The nature and content of geographic databases, locational data and attribute data, lend themselves to the output being displayed in the form of a map. Aronson has identified three processing models which are commonly used for mapping. These are the simple map processing model; the composite map processing model; and, the relational

map processing model.[7]

The simple map processing model assumes that the data set represents a single independent map sheet which cannot be split into multiple maps by subject; there is a maximum of one attribute data value for each geographic element. Similarly, maps generated by this model cannot be combined, and, as a result, the simple map processing model is used in conjunction with the simple tabular data model. An example of a map generated using this model is a map of a transportation network displaying a series of roads and the road names. This map could be generated from a simple table containing the location and the name of each road.

The composite map processing model assumes that a data set is a combined set of map sheets. Composite maps are created by overlaying attribute data sets with common geographic references (i.e. locational data). Once again the simple tabular model is applied. The map described above associated with the simple map processing model could be transformed into a composite map by overlaying a second simple map. If a map showing the location and the class of each link, whether it is an interstate, highway, or single lane road, was overlain on the map of roads and

---

[7]Peter Aronson, "Attribute Handling For Geographic Information Systems," Proceedings, Eighth International Symposium On Computer-Assisted Cartography, Baltimore, Maryland, March 29 - April 3, 1987, p.350.

road names a composite map showing the roads, road names, and each road type would be created.

The relational map processing model treats a geographic data set as a set of overlapping, independent map sheets and associated attribute tables. These map sheets may be combined, but are not permanently joined. Also, unlike the two previous models, the data set is not restricted to one map sheet. Any available data which are in the correct format may be included in an operation with any other data, assuming they have a common location or share a common thematic domain. The obvious data model for this map processing model is the relational model, as both the map processing model and the data model use data sets which can be joined on common spatial or attribute domains.[8] The relational database structure is responsible for the layered graphic output now characteristic of geographic information systems. In the example given above, the name, the class, and the location of the endpoints of each road could be stored in three separate data sets, and would be combined only when a map containing the location, name and class of each road was desired. This is not the same as the composite model which overlays common locational data and different attribute values to form a composite map.

---

[8]Ibid. p.352.

1.4 Variance in the Content of Geographic Databases.

Most geographic information systems have been designed to perform a very specific class of duties. Consequently, the databases which form the integral part of these systems are very specific to the objective of the system. For example, a G.I.S. used for transportation planning would employ a database containing information important to the development of new links. e.g. current network elements, traffic flows, traffic projections, and land ownership.

Most geographic databases are characterized by the content of their attribute data. The locational data represent the same geometric features, points, lines, and polygons, and it is only the different attribute data which distinguish databases. However, certain geographic databases exist which differ from typical databases in the content of their locational and attribute data. Transportation network databases are an example of this distinct type of database.

1.5 Transportation Network Databases.

Transportation network databases consist of nodes and links which are geographically referenced to the earth's surface. Polygon data are not normally considered when discussing transportation databases, although some networks are decomposed on a regional basis, and the

regions, which act as sinks and sources of movement on the network, are defined as polygons.[9]

Two types of transportation databases exist: (1) a topological transportation network, and, (2) a geographic transportation network. The topological network treats the network as a series of nodes connected by straight lines. Alternatively, the geographic network treats the links between nodes with greater detail. Links are not represented by straight lines but are defined by a series of nodes which, when connected, display the route of the link.

Transportation nodes are the intersections of transportation links or arcs, and represent places in the network where it is possible to access the links in the network or change mode of transport. Nodes at which it is possible to change the mode of transport are referred to as transfer or transshipment points. Nodes can also represent terminals and urban areas, which, in turn, represent sub-networks which may be too detailed to appear in the database, but which represent transportation costs in terms of time and energy use.[10] Network nodes are generally represented by their name, coordinates, and a series of attribute values.

---

[9]Electric Power Research Institute, Transportation Network Models for Energy Supply Analysis, Vol.3, (Palo Alto, Electric Power Research Institute, 1982) p.2-1.

[10]Ibid, p.2-4.

Transportation links are defined by an ANODE (start point), BNODE (end-point), and a list of attribute values. Common attribute data attached to links include the transportation mode, the length of the link, the cost per unit of distance for transporting goods along the link, the capacity of the link, and whether or not the link is one-way or two-way.

The movement of goods is not free, and knowledge of the cost of transportation on a particular link is important. As a result, attribute data found in a transportation database are commonly associated with the mode of transport, the amount being transported, and the cost of transporting a unit weight of a certain commodity. Consider, for example, the FIELDS (Food For Peace Inland Emergency Logistics Distribution Software) package, designed by researchers at The University of Tennessee, Knoxville for the United States Agency for International Development (USAID) to improve the distribution of food in famine stricken areas of Africa. Its transportation database consists of node and link files with attribute data which describe the demand for food, cost and capacity of storage, the number and capacity of trucks at nodes in the network, and the capacity, length, and cost of transportation for the transportation links. Similarly, a database created by the Electric Power Research Institute (EPRI) for the analysis of the movement

of coal between supply and demand points in the United States contains attribute values which describe the cost of movement (in terms of both time and money), capacity, and the energy requirements for transporting coal.

Some transportation databases handle their attribute data in a class-like fashion. For example, the road system of the United States can be split into road classes. Nodes and links with the same characteristics are grouped together in classes, and disutility functions, the unit cost of using a transportation facility, are calculated for that particular class rather than for each particular network element.

Transportation network databases can be created as a single entity, or alternatively they can be a part of a greater scheme. The EPRI database and the database created for the FIELDS system are "stand alone" databases, whereas a transportation database created for an urban information system will probably constitute only a few layers of the many layers which should be constructed for a multi-purpose G.I.S.

In summary, transportation network databases are geographic databases which represent a network as a connected set of nodes and links. Nodes represent transport terminals, mode switching facilities, or points of access to the network. Links represent the actual movement corridors of the network, and are the features

upon which the movements on the network take place. Each node and link in the database is assigned a series of characteristic attribute data values, either individually or as a member of a network element class.

CHAPTER II

THE AFRICAN TRANSPORTATION NETWORK DATABASE AND RESEARCH ENVIRONMENT

During the implementation of a geographic information system, the tasks to be performed by the system must be clearly defined so that the appropriate hardware and software can be purchased, or developed, and the geographic databases can be created in a suitable format or structure. The advantage of a well designed data structure is that the data and the system hardware and software are compatible, and the need for interchange programs, which transform data into a format suitable for certain pieces of computer hardware or software, is minimized.

In many cases, however, a digital database may already exist and only the hardware and software is required to complete the system. The opposite situation also could be true. That is, the hardware and software may exist and a suitable database may be needed. In each of these cases the appropriate course of action is to purchase hardware and software, or to set up the database in a fashion, that would complement the elements of the system already in place. A third possible case is that

the database and the system hardware and software exist independently. In this case it is possible that the database and the hardware and software are not compatible, and the database has to be modified or re-structured.

The starting point of this thesis was not one that would be considered ideal for trying to integrate a digital geographic database and a geographic information system. Nonetheless, it is a situation which can be found in many organizations. As in the third case above, the computer hardware and software and the digital database were in place at the beginning of the research. It is important to note that the hardware, software, and the database were not implemented together, nor were the hardware and software and the database purchased and developed together, and, as a result they are not directly compatible.

The digital database of concern is a transportation network database of the African continent, and the hardware and software consist of the facilities available in the Department of Geography at The University of Tennessee, Knoxville.

2.1 The African Transportation Network Database.

The African transportation database consists of two main topological transportation networks: a detailed network and an aggregate network. The detailed network

consists of 21213 links, 17522 nodes, and 1498 transshipment pairs. The aggregate network consists of 5275 links, 4846 nodes, and 765 transshipment pairs. All of the techniques discussed in this thesis will be developed using the aggregate transportation network. However, these techniques will also work on the detailed network, assuming the computer program array sizes have been increased to accommodate the larger amounts of data to be processed.

The data in both the detailed and the aggregate network is delivered in tabular files with each country having three separate files: a node file, a link file, and a transshipment file. The countries represented in the database, and the format and contents of each of the data files are listed in Appendix One.

The network node files are the only ones which contain values of latitude and longitude. The link and transshipment files define nodes (the start and end-point of links) by name only. This feature of the database means that the relational data model is used, and modification of data extraction techniques is necessary when dealing with link and transshipment files.

The three data files have common attribute data types; country code, node name, and transportation mode are found in all three files. However, apart from these three attributes, the content of the files differ. Node

files differentiate between nodes which are border crossings and those which are not, and nodes are split into cost and delay classes dependent upon their mode. Link files contain information on the length of the link, and split the links into cost and delay classes based upon highway surface material, type of railroads (single or double track), and the direction of barge movements. Classes also exist for the capacity of rail links with rail gauge, train length, days and hours of operation, and whether or not the railway is electrified. Transshipment files are the smallest of the three files and contain only the three attributes common to all files (i.e., node-names, country code, and mode of transport).

Attribute data in the database are structured using the relational data model. The attribute values common to all three data files can be used to relate the three files. For example, coordinate values can be added to the link files by matching common node names and inserting the appropriate values of latitude and longitude. Files can also be concatenated to create a database for more than one country.

The most suitable map processing model for the database is the relational map processing model. This model is created by relating data files for each country to create a map containing information form more then one file. For example, it is not possible to plot

transportation links using node names. Coordinate values are required, therefore, the link file and the node file, which contains the locational data, have to be related, in the fashion described above, before a map can be produced.

The current structure of the database, as displayed in Appendix One, was created with an operations research function in mind. As a result, the structure is not the most suitable for a geographic information system.

Problems of data repetition, a characteristic of the simple tabular data model, are compounded by the required operations research format of the database. The database was created by the Matrix Group, a Washington D.C. consulting firm, for the operation of the University of Pennsylvania's freight network equilibrium models.[1] As a result, every transshipment point is required to be listed twice in each node file, as well as being included in the transshipment file. Likewise, if two nodes are joined by more than one link of differing transportation modes each link has to be listed in the link file. This results in the node and link files having data redundancy.[2]

---

[1]Terry L. Friesz and Patrick J. Harker, "Freight Network Equilibrium: A Review of the State of the Art." Analytical Studies in Transport Economics, Cambridge University Press, 1985, p. 161.

[2]Data redundancy is, in part, a result of the history of the database in addition to being a function of the database design criteria. Changes in the database through time have resulted in a poor database structure. This structure has not been corrected because it would require changes to be made in the database software.

Problems of data repetition are exacerbated by the fact that not all of the data contained in each file are useful for plotting certain attributes of the network, and are, therefore, merely taking up memory space. This problem can slow down operations on the database.

Several methods can be used to improve the structure of the database. One specific method is first to reduce the problem of data redundancy by making the transshipment file obsolete and removing it from the database, and, secondly, to ensure that each node is listed in the node file only once. This can be achieved by creating a new attribute value in the node file. A 13 integer array, corresponding to the thirteen different modes, was created to store the transshipment data. Depending on whether or not a particular transport mode was available, the corresponding value in the array would have a value of one or zero. For example, if barge(mode 2), highway(mode 3), and pipeline(mode 13) modes were available at a particular node the array representing that node would have the following values 0110000000001. This change in the database structure does not incur any losses of information, ensures that node data are not listed several times in both the node and transshipment files, and results in savings in computer storage space. For the country of Benin, the unaltered node file uses 2160 bytes

of memory. The altered file for the same country, which contains both node and transshipment data, and renders the transshipment file obsolete, requires only 968 bytes of memory.

2.2 The Research Environment.

All of the work completed on the African transportation database took place on an IBM Personal Computer XT with an enhanced graphics adapter (EGA). The remainder of the system hardware consists of a 6-pen IBM Color Plotter and an Okidata microline printer. These output devices were used for the creation of hard-copy maps from the database management techniques.

The system software includes a Microsoft FORTRAN Optimizing Compiler, which is used to compile FORTRAN programs on a variety of database management and graphics functions, the PLOT 88 graphics subroutine library, and the Prodesign II Computer Aided Design system.

All of the software packages listed above are completely independent of one another, and, as a result, they do not represent an integrated geographic information system. The database used and all of the software available were not completely compatible, a characteristic which would not be found in a well designed G.I.S.. The fulfillment of the objectives of this thesis, therefore, required the use of interchange programs to link the

database to software packages, and, thus, allowed the query, analysis, and display of the database.

CHAPTER III

THE DEFINITION OF TRANSPORTATION CORRIDORS

Transportation corridors are used to determine which elements of a geographic database are within a user-specified distance of a transportation link. The definition of a transportation corridor is an eminently geographic question. By defining a corridor, the database user is determining which network features are within a given distance of a point or feature of interest. This information, particularly in areas with a poor transportation infrastructure, can be of great importance. For example, certain areas of Africa are plagued by railroad systems which employ a variety of different rail gauges, and also by political instability. When transporting food-aid between two points by rail, it would be beneficial to know of alternative routes in case the intended route was blocked. A transportation corridor containing rail links of an appropriate gauge would provide appropriate information for re-routing the shipment.

Transportation corridors contain all or part of the transportation network which lies within a user-defined area, and are extracted by defining a link and a corridor radius. The key to working with corridors is defining

their limits and testing which elements of the database are within this area. This procedure requires the calculation of coordinates and distance values.

This chapter will deal with the extraction of transportation corridors from geographic databases. Examples will be presented from the African transportation network database.

3.1 Sample Problem.

To illustrate the issues and the problems of extracting transportation corridors it is useful to start with a sample problem. Consider an area ranging from 0.00 to 5.00N in latitude and from 0.00 to 5.00E in longitude.[1] The sample problem is to extract a corridor, with a radius of 100km, around a link running from the node (2.00, 2.00) to (4.00, 4.00). The area of the sample problem and the link being analyzed are illustrated in Figure 3.1.

---

[1]In the African transportation network database values of latitude north of the equator and values of longitude west of the prime meridian have a negative value. All of the equations and calculations used in this research assume that values of latitude north of the equator and values of longitude west of the prime meridian have a positive value. As a result, all values of latitude are multiplied by negative one before being used in any calculations.

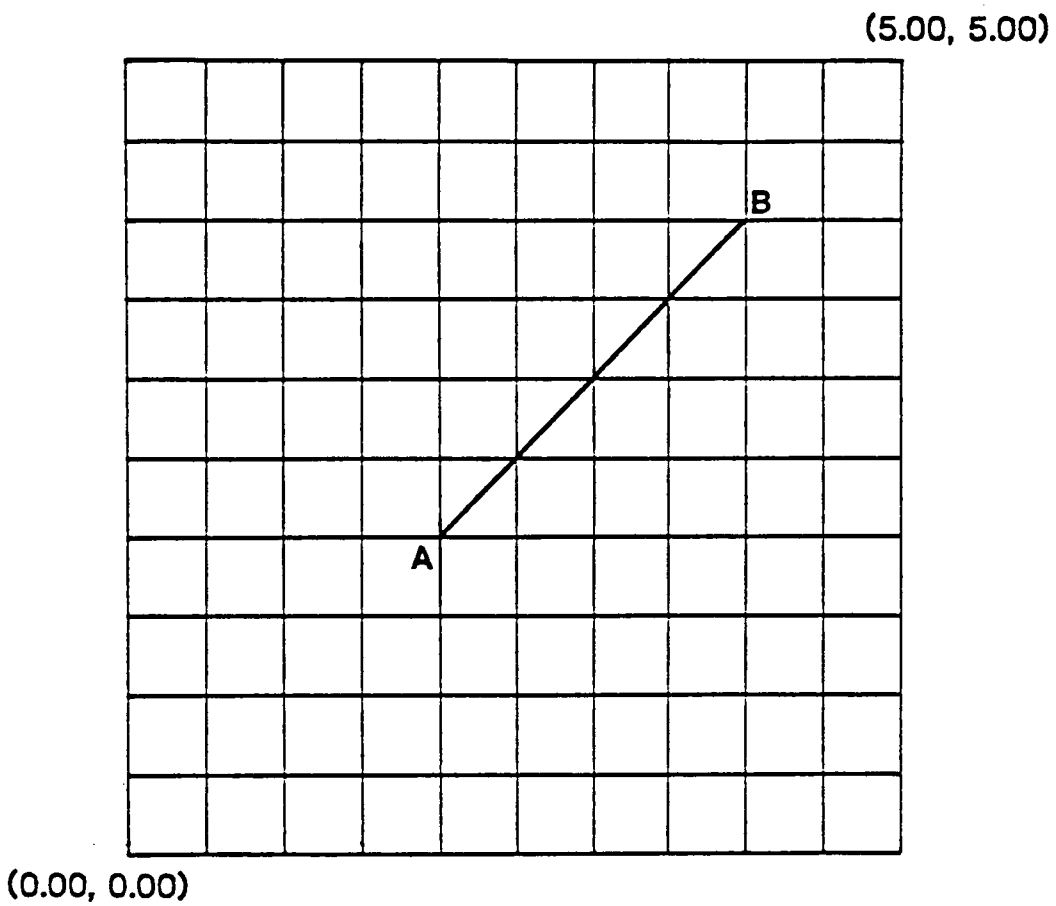


FIGURE 3.1

Area of Sample Problem and Link being Analyzed

3.2 Methodology: Determining the Distance Between Points and Lines.

To determine whether or not a point is within the user-defined distance of a transportation corridor, the distance between the node and the link has to be calculated and compared to the radius of the buffer zone. The first task in this procedure is to find the point on the transportation link which, when joined to the node being analyzed, will form a line perpendicular to the link. This is best visualized using a rectangular coordinate system. Figure 3.2 illustrates the transportation link AB, the node P, and the point being calculated Q.

To calculate the point Q, the coordinates of the end-points of the link and the node being tested, which are in degrees of latitude and longitude, are converted to rectangular coordinates. The cylindrical equidistant map projection was used to convert the end-points and nodes into rectangular coordinates because the meridians and parallels are equidistant straight lines which intersect at right angles.[2]

A transportation corridor can be one of four cases. The link around which the corridor is defined can have a positive or a negative slope, a zero slope (horizontal),

---

[2]John P. Snyder, Map Projections Used By The U.S. Geological Survey, 2nd Edition. U.S. Government Printing Office, Washington: 1982. p. 89.

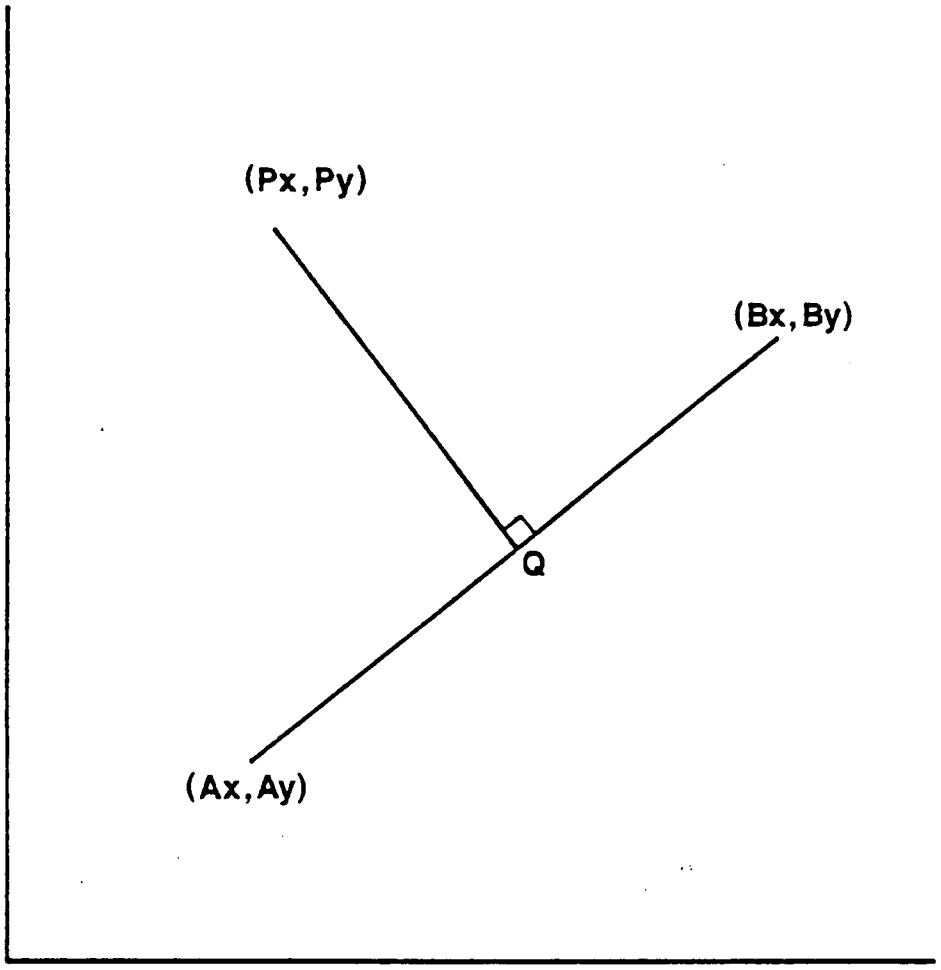


FIGURE 3.2

The Link AB, Node P, and Point Q.

or an infinite slope (vertical). The slope of the link is important because it determines the method used to calculate the coordinates of the point Q. Before discussing the methodology of each case, a discussion of the distortion of coordinates introduced by the map projection is appropriate.

3.3 Coordinate Distortion and Adjustments.

Only at the equator, the standard parallel of the cylindrical equidistant projection, are degrees of latitude and longitude approximately equal in length. For example, at 60 degrees north or south of the equator one degree of longitude is half the length of one degree of latitude. As a result, adjustments have to be made to the projected values of longitude which do not lie on the standard parallel of latitude. Calculations in this research control the influence of this distortion by adjusting values of longitude by the cosine of the average value of latitude along a link. Use of the average value of latitude does not correctly adjust the entire link; the only point which is correctly adjusted is the mid-point of the link and the adjustment factor is, therefore, an approximation. However, if the corridor being extracted is sufficiently short in length use of this approximation does not severely impair mathematical calculations, nor the visual display of the resultant corridor.

The distortion control outlined above is not the only method of adjusting values of longitude, however, it is used consistently throughout this research. The following discussions outline the extraction of transportation corridors of all four cases, and the use of the distortion control.

Cases One and Two: The positive and Negative Cases. The point Q, shown in Figure 3.2, is calculated using the equation of a straight line and the fact that the product of the slopes of two perpendicular lines is equal to negative one. Equation 3.1 is the equation of a straight line.

$$Y = (\text{slope of line} * X) + \text{Constant} \qquad (3.1)$$

Using the notation in Figure 3.2, the slope of the line is equal to the difference between By and Ay divided by the difference between Bx and Ax. To ensure the coordinates of the point Q are calculated correctly, the values of Bx and Ax are corrected using the distortion control defined above. The denominator of the slope calculation therefore becomes cosine((Ay+By)/2)*(Bx-Ax). The constant value in the equation is the point where the line crosses the Y axis.

In Euclidean space, the point Q is unique. It is the only point that lies on both the link AB and the perpendicular line PQ. The coordinates of Q can be calculated by finding the intersection of these two lines. The X value of Q is calculated as follows:

$$\text{Equation of AB:} \quad Y = (m * X) + C1$$
$$\text{Equation of PQ:} \quad Y = (-1/m * X) + C2 \qquad (3.2)$$

The above equations have two unknowns, the X and Y values of Q, but can be re-written as follows:

$$(m * X) + C1 = (-1/m * X) + C2 \qquad (3.3)$$

The X value can be calculated from this equation, and can be put into either the equation of the link AB, or the equation of the line PQ to calculate the Y value. This completes the calculation of the rectangular coordinates of the point Q.

Case 3: The Horizontal Case (Zero Slope). The calculation of the coordinates of the point Q is less complex in the horizontal case because the angle between the link AB and the X axis is zero when using rectangular coordinates. As a result, the use of the slope of a straight line and the intersection of perpendicular lines is not required.

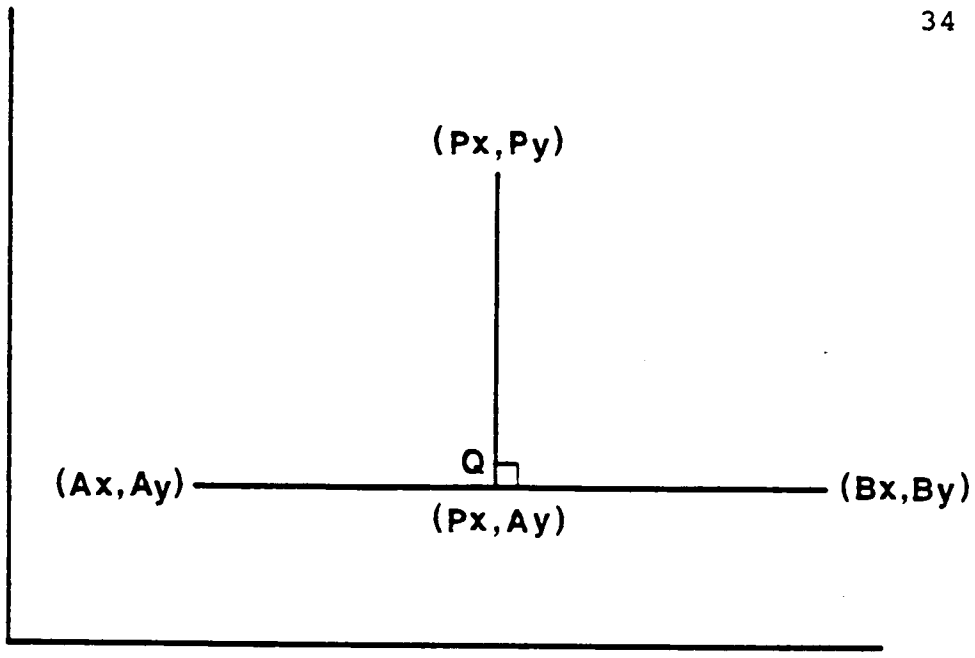
In Euclidean space, the coordinates of Q can be derived from both the endpoints of the link and the node P. Using the notation in Figure 3.3a, the X coordinate of Q will be the same as the X coordinate of the point P, and the Y coordinate will be the same as the Y coordinate of either point A or point B. The coordinates of Q will, therefore, be (Px, Ay).

Case Four: The Vertical Line (Infinite Slope). The angle between the horizontal axis and a vertical link is equal to ninety degrees. However, the coordinates of the point Q can be calculated in a similar manner as the horizontal case. Figure 3.3b illustrates that, in Euclidean space, the X coordinate of Q is equal to the X coordinate of either point A or point B, and the Y coordinate is equal to the Y coordinate of P. The coordinates of Q will, therefore, be (Ax, Py).

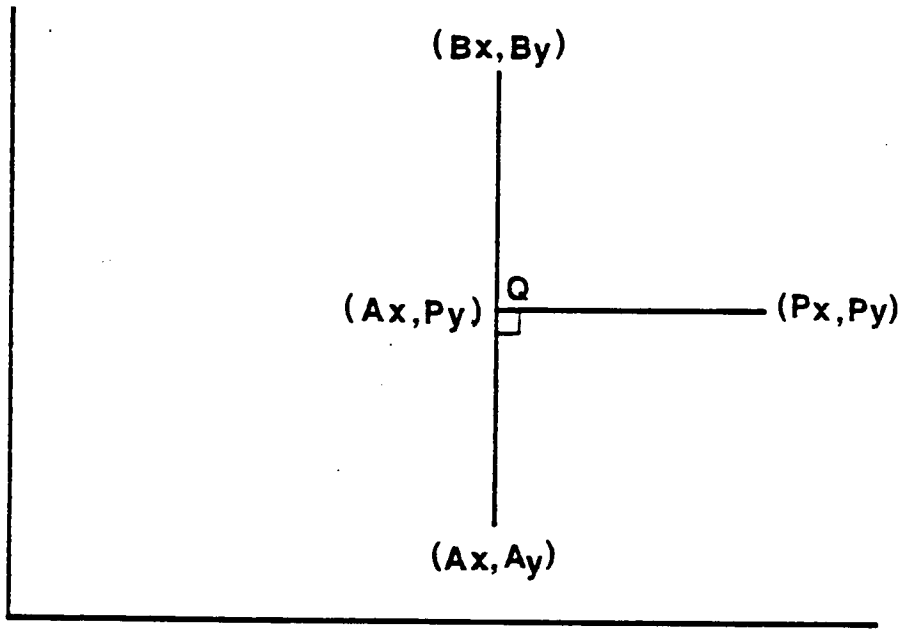
3.4 The Calculation and Comparison of Distance.

Once the X and Y values of the point Q have been determined the squared distance between the node and the link can be calculated using equation 3.4. The projected values of longitude (Px and Qx) are adjusted by the cosine of the average latitude of PQ to control distortion.

$$\text{Dist. } PQ^2 = (Py-Qy)^2+((Px-Qx)^2*(\cos(Py+Qy)/2)^2) \quad (3.4)$$



a. Horizontal Link



b. Vertical Link

FIGURE 3.3

Cases 3 and 4

The square root of the value calculated by equation 3.4 is compared to the user-specified radius of the corridor to determine whether or not the node lies within the corridor. If the length of PQ is less than or equal to the radius then the point is within the corridor.

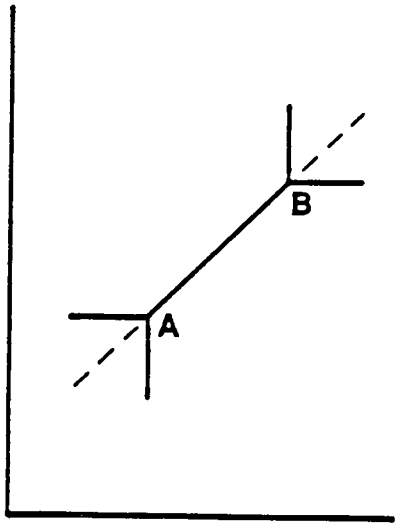
3.5 The Problem of Infinite Length.

One problem remains unsolved. The transportation link in the above calculations, because it is represented by the equation of a straight line, has infinite length. As a result, the transportation corridor defined will be as long as the coordinates analyzed will allow. For example, in the sample problem the present method would extract all of the points within the user-defined radius between the points (0.00, 0.00) and (5.00, 5.00). These two points, and all of the points on a straight line in between, are on the line with the equation of the transportation link between (2.00, 2.00) and (4.00, 4.00), and, as a result of its infinite length, would be included.

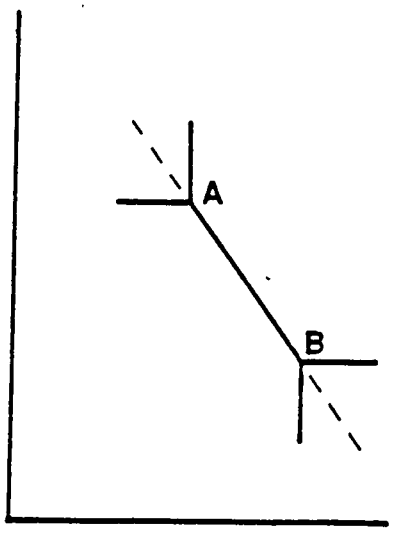
To solve this problem, a test is required of each of the points Q calculated to lie on the line defined by the equation of the transportation corridor. It can be assumed, without loss of generality, that, except for the vertical case in which the A node is furthest to the south, the A node of the link is always furthest to the west. In the positive slope case, if the point Q has an X

value and a Y value less than the A node it will assume the coordinates of the A node. Figure 3.4a illustrates that the point Q lies to the south-west of the A node, outside the range of the link, and, therefore has to be re-assigned for the distance calculation between the link and the node to be valid. Likewise, if the point Q has an X value and a Y value greater than the B node it will assume the coordinates of the B node. Figure 3.4a shows that Q lies outside the range of AB, in this case to the north east, and has to be re-assigned. In the negative slope case, illustrated by Figure 3.4b, if the point Q has an X value of less than the A node and a Y value of greater than the A node then it will assume the coordinates of the A node. Similarly, if the point Q has an X value greater than the B node and a Y value less than the B node it will assume the coordinates of the B node.

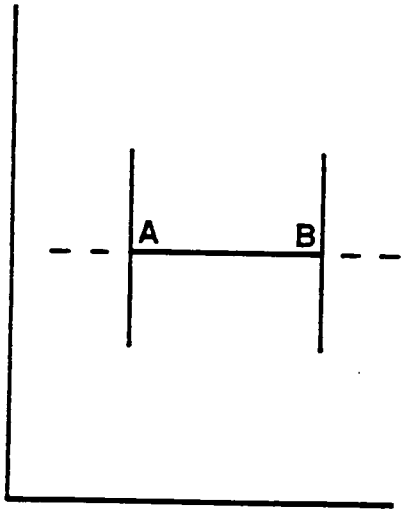
Unlike the positive and negative cases, the horizontal and vertical cases do not need to test both the X and Y values of the point Q against the end points of the link. In the horizontal case, only the X value of Q needs to be tested. Figure 3.4c shows that if the X value of the point Q is less than the X value of the A node it will assume the coordinates of the A node. If the X value of the point Q is greater than the X value of the B node then it will assume the coordinates of the B node. In the vertical case, illustrated by figure 3.4d, only the Y



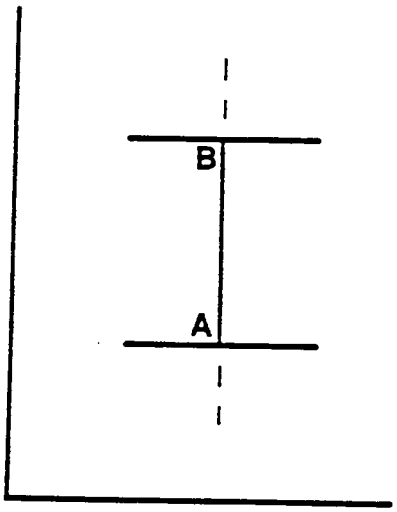
a. Positive Slope



b. Negative Slope



c. Zero Slope



d. Infinite Slope

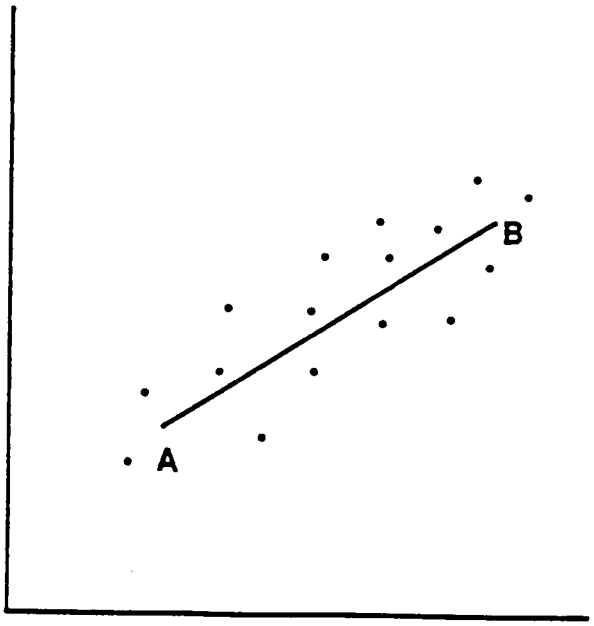
FIGURE 3.4

Re-assigning the Point Q

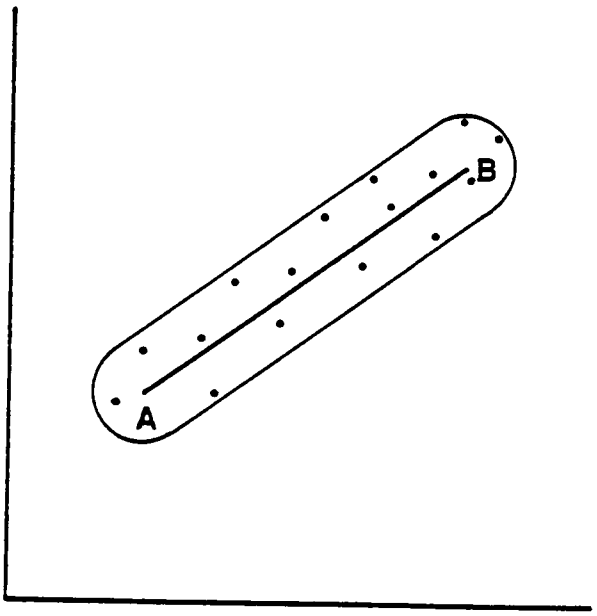
value of the point Q has to be tested. If the Y value of the point Q is less than the Y value of the A node then it will assume the coordinates of the A node. Alternatively, if the Y value of the point Q is greater than the Y value of the B node it will assume the coordinates of the B node.

3.6 Definition of the Transportation Corridor Buffer.

The calculation of the points which lie within a user-defined area is the basic problem when extracting transportation corridors from a geographic database. However, if some form of output other than a listing of the nodes within the corridor is desired, other elements of the corridor have to be calculated. For example, if a visual display is desired, either as a hard-copy from a printer or plotter, or on a computer screen, the display would be enhanced if the limits of the corridor were included. This would allow the user to see how far or how close each node is from the limit of the corridor. Alternatively, the user may wish to overlay the buffer on the entire network. Figure 3.5a shows an example of a corridor without the limits of the buffer displayed, and Figure 3.5b shows a corridor with the limits of the buffer limits.



a. Corridor Without Buffer



b. Corridor and Buffer

FIGURE 3.5

Transportation Corridor With and Without Buffer

The limit to a transportation corridor is composed of two straight lines on either side of the transportation link, and two semi-circular ends. The semi-circular ends have the user-defined radius and are drawn with the endpoints of the link as their center.

Similar to the node extraction procedure, the methodology for defining the straight line segments of a corridor buffer is dependent upon the slope of the link. Again, four cases exist: the positive and negative slope cases, the horizontal case, and the vertical case.

3.7 Calculation of the Straight Line Segments of the Buffer.

Cases One and Two: Positive and Negative Slope.

Figure 3.6 illustrates the four points, C, D, E, and F, which define the corridor buffer. It is important to note that the line AB and line CD and EF are not parallel. This is because all three lines have a different average value of latitude and, as a result, their adjusted slopes will differ. Only in true rectangular coordinates, in which one degree of latitude is equal to one degree of longitude, would the lines be parallel.

To determine the coordinates of the points C, D, E, and F, their location in relation to the end points of the link has to be calculated. For each point the Y value has to be calculated first so that the X value can be adjusted

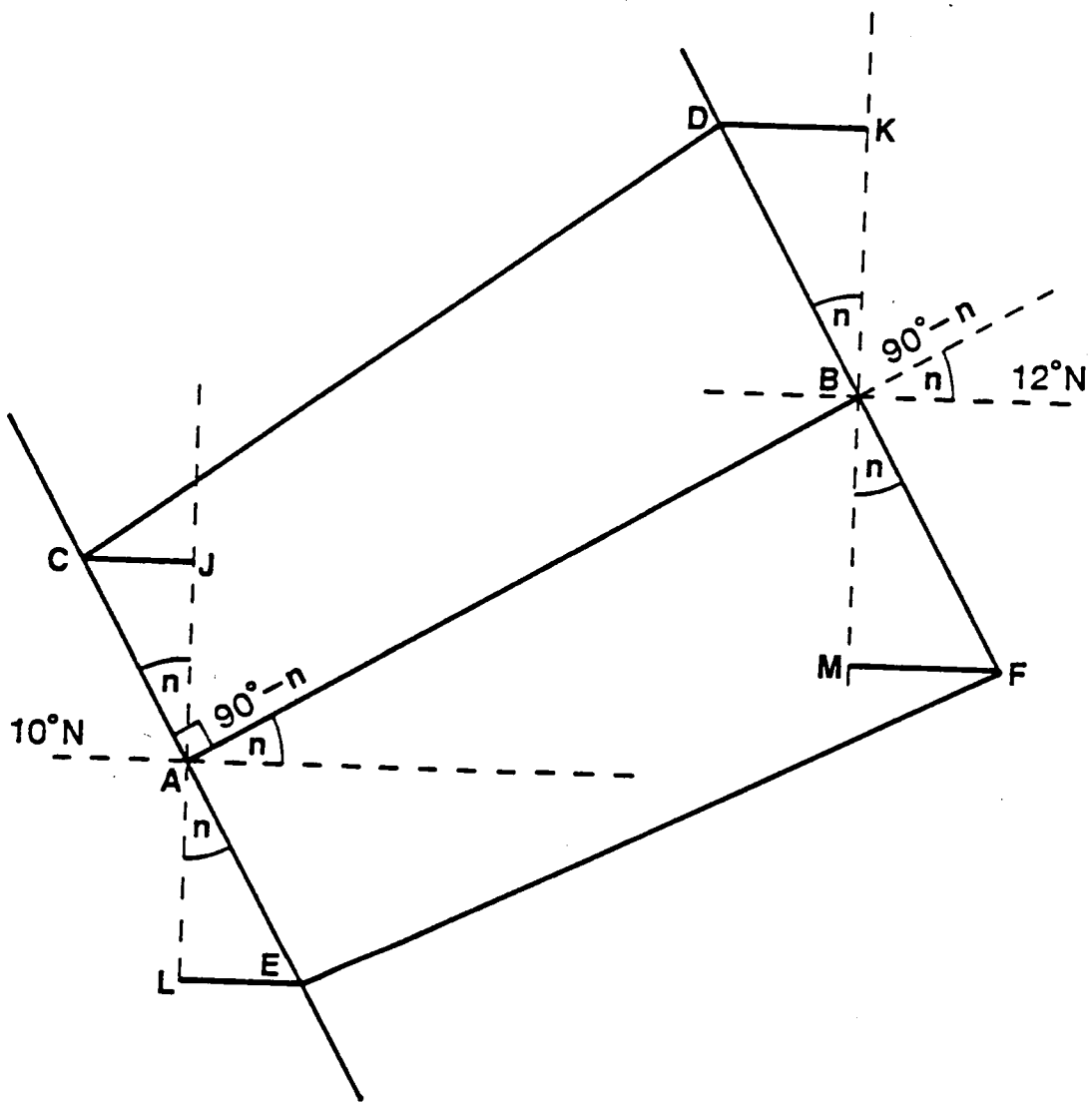


FIGURE 3.6

Straight Line Segemnts of the Buffer Cases 1 and 2

correctly.

Figure 3.6 shows that the point C is above and behind the A node of the link. Using the notation in this figure the Y value of C can be calculated using equation 3.5a. The cosine of the angle n is equal to the radius of the buffer[3] (line AC) divided by the length AJ. Thus, the length AJ is equal to AC * cos(n). The addition of the length of AJ to the Y value of the A node gives the Y value of the point C. The X value of C is calculated by subtracting the length CJ from the X value of the point C. CJ is equal to the Radius(AC) * sin(n). To ensure a close approximation of the location of the point C, the length of CJ has to be adjusted by the cosine of the average latitude of the points A and C. The full equation for the calculation of the X value of C is given by equation 3.5b.

$$Cy = Ay + (Radius * \cos(n)) \qquad (3.5a)$$

$$Cx = Ax - (Radius * \sin(n))/(\cos(Ya+Yc)/2) \qquad (3.5b)$$

---

[3]During the initial node extraction procedure, the coordinates of each node were projected and scaled so that one unit of projected space is equal to one meter. This allows the use of user-defined distances, measured in meters, in calculations with the projected end points of a transportation link.

The point E lies below and ahead of the A node. It's coordinates are calculated by subtracting the length of the line AL from the Y value of the A node, and by adding the adjusted length of the line EL to the X value of the A node. For the point E, the appropriate adjustment is the cosine of the average value of latitude for the points E and A. The equations used to calculate the coordinates of the point E are as follows:

$$Ey = Ay - (Radius * \cos(n)) \quad (3.6a)$$

$$Ex = Ax + (Radius * \sin(n))/(\cos(Ay + Ey)/2) \quad (3.6b)$$

The location of the points D and F are calculated in relation to the B node of the link. Figure 3.6 also contains the notation for calculation of the points D and F. Point D lies above and behind the B node of the link. Therefore, the Y value of D is equal to the Y value of the B node plus the length of the line BK, and the X value is equal to the X value of B minus the adjusted length of the line DK. The appropriate adjustment is the cosine of the average latitude of the line BD. The equations for the calculation of the coordinates of the point D are as follows:

$$Dy = By + (Radius * \cos(n)) \quad (3.7a)$$

$$Dx = Bx - (Radius * \sin(n))/(\cos(By + Dy)/2) \quad (3.7b)$$

Point F lies below and ahead of the B node. The Y value of F is equal to the Y value of B minus the length of the line BM, and the X value is equal to the X value of B plus the adjusted length of the line FM. For the point F, the appropriate adjustment is the cosine of the average latitude of line BF. Equation 3.8a and 3.8b represent the calculations for the coordinates of the point F.

$$Fy = By - (Radius * \cos(n)) \quad (3.8a)$$

$$Fx = Bx + (Radius * \sin(n))/(\cos(By + Fy)/2) \quad (3.8b)$$

Case Three: The Horizontal Line. The four points C, D, E, and F, illustrated in Figure 3.7, are first calculated on the assumption that a true rectangular coordinate system is being used. These points are calculated by adding or subtracting the corridor radius from the Y values of the link end points. The following equations represent the calculation of the straight line segments of the corridor buffer in true rectangular coordinates.

$$\begin{aligned} Cx &= Ax & Cy &= Ay + Radius \\ Dx &= Bx & Dy &= By + Radius \\ Ex &= Ax & Ay &= Ay - Radius \\ Fx &= Bx & Fy &= Fy - Radius \end{aligned} \quad (3.9)$$

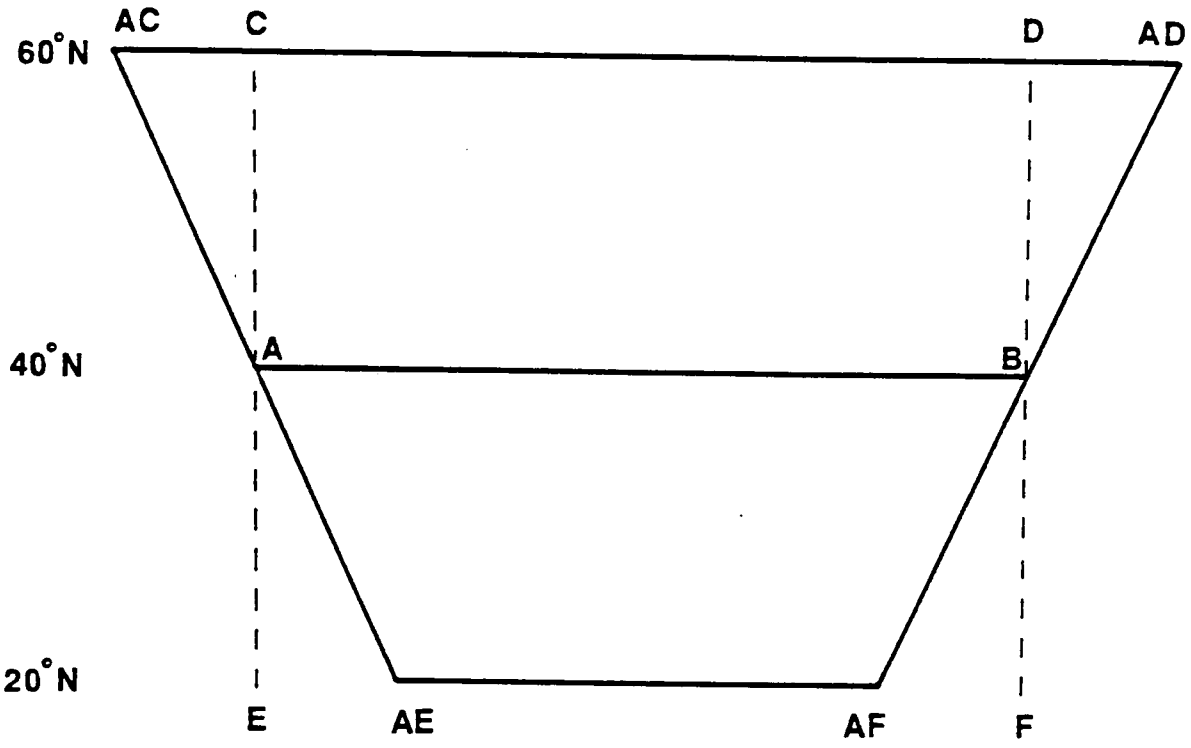


FIGURE 3.7

Straight Line Segment of Buffer Case 3

A horizontal transportation corridor will not have a regular buffer defined by the points C, D, E, and F. In the northern hemisphere the buffer will appear as defined by the adjusted points AC, AD, AE, and AF in Figure 3.7. To determine the appropriate end points a scaling factor is used to ensure the lines CD, AB, and EF are the same length at the three different parallels of latitude. Above the link AB, the line CD has to be stretched, and below the link AB the line EF has to be compressed. As a result, a different scaling factor is used to scale the line CD than is used to scale the line EF. Equations 3.10a and 3.10b represent the scaling factors used. Eu is the scaling factor above the link and El is the scaling factor below the link.

$$Eu = 0.5*((\cos(Ay)/\cos(Cy))-1.0)*(Bx-Ax) \qquad (3.10a)$$

$$El = 0.5*(1.0-(\cos(Ay)/\cos(Ey))*(Bx-Ax) \qquad (3.10b)$$

The X coordinates of each point AC, AD, AE, and AF are calculated by adjusting the X coordinates of the points C, D, E, and F. The X coordinate of AC is equal to Cx - Eu, and the X coordinate of AD is equal to Dx + Eu. Below the link, the X coordinate of AE is equal to Ex + El, and the X coordinate of AF is equal to Fx - El.

Case Four: The Vertical Line. The X coordinate of the points C and E, shown in Figure 3.8, can be calculated by subtracting the adjusted length of the corridor radius and adding the adjusted length of the corridor radius to the X value of the A node. The Y value of both C and E is equal to the Y value of the A node. The equations for the calculation of the coordinates of these points are as follows:

$$Cx = Ax - (1/\cos(Ay)*Radius) \qquad Cy = Ay$$
$$Ex = Ax + (1/\cos(Ay)*Radius) \qquad Ey = Ay \qquad (3.11)$$

Similar to the horizontal case, the X value of the point D (Figure 3.8) is calculated by subtracting the corridor radius from the X value of the B node, and the X value of the point F (Figure 3.8) is calculated by adding the corridor radius to the X value of the B node. These values are then adjusted by the scaling factor (Eu) defined by equations 3.12. The Y value of each point is the same as the Y value of the B node.

$$Eu = ((\cos(Ay)/\cos(By))-1.0)*(Ex-Ax)$$
$$Dx = Bx - (Radius + Eu) \qquad Dy = By$$
$$Fx = Bx + (Radius + Eu) \qquad Fy = By \qquad (3.12)$$

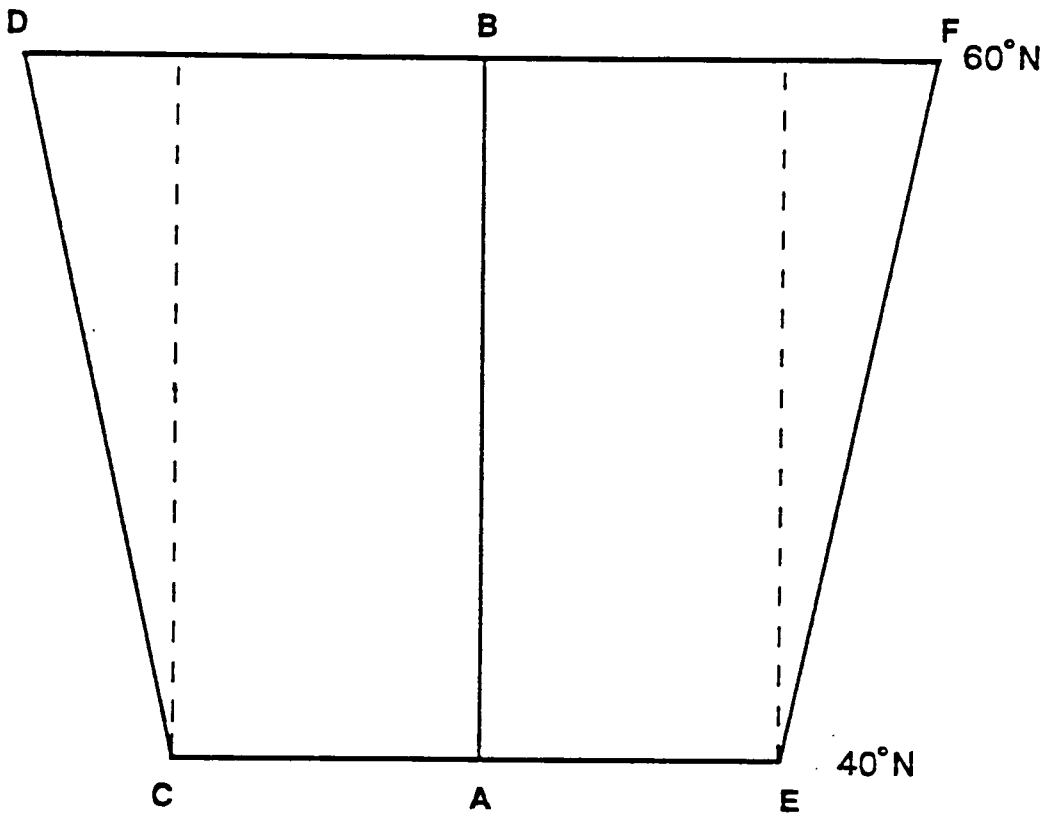


FIGURE 3.8

Straight Line Segment of Buffer Case 4

3.8 Calculation of the Semi-Circular Ends of the Buffer.

A semi-circle can be defined by it's center and the beginning and end points of it's arc. Given the coordinates of the end points of the link being analyzed and the end points of the corridor buffer, enough information is available to define the semi-circular ends of the corridor buffer. At each end of the link, the point A and the point B are the center of the semi-circle and the end points of the buffer are the beginning and the end of the arc. For example, in Figure 3.7, at the A node end of the link the semi-circle would be drawn by connecting the point C to the point E with a counter-clockwise arc centered on the Point A.

The semi-circular ends of the corridor and the straight line sections of the corridor can be displayed using a suitable mapping package or system to connect the appropriate points. The method used to display the transportation corridor and it's limits is discussed in Chapter Five.

3.9 Program Design.

All of the calculations outlined in this chapter were included in a FORTRAN program which read a test file created for the sample problem, extracted user-defined corridors of all four cases, and calculated and adjusted the coordinates of the limit. This program was easily

converted for use on the node files from the African transportation network database. The format statement used to read in the locational data was changed to suit the database node file, and long decimal values, which allow up 17 significant digits, were used to ensure the accuracy of calculations.

CHAPTER IV

LINK TRANSPORTATION CORRIDORS AND CORRIDOR ATTRIBUTES

The method developed to extract nodes within a transportation corridor is limited in two ways. First, the only types of data which can be extracted from the African transportation network database are node data. The database contains link and transshipment data in addition to node data, and transportation corridors containing this information may be desired by a database user. Secondly, geographic databases, discussed in Chapter One, contain both locational and attribute data. However, only locational data are processed during the node extraction procedure. Each of the network elements contained in the database, nodes, links, and transshipment points, has a series of attribute values or flags attached. These data have not been considered in the procedure outlined in the previous chapter. During a corridor definition, nodes are extracted on the basis of whether they were within a user-specified distance of a transportation link. However, some database queries may not rely solely on the measurement and comparison of distance. For example, a user may wish to list all of the nodes within 100km of a given highway that are border nodes. In answering this query correctly, attribute data

have to be analyzed in addition to the locational data.

This chapter will deal with (1) the extraction of transportation corridors containing transportation links and transshipment points, and (2) the extraction of corridors containing network elements with certain attribute values. These two tasks are linked by the fact that they require the processing of both attribute as well as locational data. In addition, a short discussion focused on the display of the output from transportation corridor queries will be presented.

4.1 Nodes, Links, and Transshipment Points.

Node files are the only ones in the African transportation network database which contain locational data (values of latitude and longitude). As a result, the extraction of nodes is fundamental to all further extraction procedures, although certain modifications or extensions of the method detailed in the previous chapter are required for the extraction of links and transshipment points.

The output from the extraction procedure outlined in Chapter Three is a listing, or a series of arrays, containing the coordinates of nodes within a user-defined corridor, and four values which represent the coordinates of the corridor limit. Links are defined by two nodes, a start-point and an end-point, and transshipment points are

defined by one node. As a result, the arrays of node coordinates produced by the node extraction procedure contain the types of information which describe the links and transshipment points within the corridor.

Nodes can be defined in two ways: as a coordinate pair, and, by their unique node name. Three attribute values, node name, mode, and country code, exist in all three database files. Thus, if a node name array is created along with coordinate arrays by the initial node extraction, the node names in the link file and the transshipment file can be compared to the contents of the node name extraction array to determine which additional network elements are also within the corridor. Before discussing the extraction of transshipment and link transport corridors, a discussion of arrays and counters used in this research is appropriate.

4.2 Extraction Arrays and Counters.

The method developed to extract transportation corridors containing links and transshipment points assumes that the initial node extraction procedure has been carried out, and a series of arrays describing nodes within a user-defined corridor have been created. In total, three arrays are used: (1) a character array containing the node names of nodes extracted, (2) a real array containing the X coordinate of each node extracted,

and (3) a real array containing the Y coordinate of each node extracted.

An array is a set of numbers or character strings, which can be stored by a computer, the elements of which are ordered and numbered. The ordering and numbering of the elements of the array provides the means to uniquely identify a particular element. The three arrays described above were created in such a way that the first element in both the node name array has the coordinates of the first elements in the X coordinate and the Y coordinate arrays. Likewise, the second element of the node name array has the coordinates of the second elements in the X and Y coordinate arrays and so on. Therefore, if a match is found between a node name of the endpoint of a link and the node name of an extracted node, and a counter records the position of the match, the coordinates of the node can be determined by reading the corresponding position in the other two arrays. This use of counters and arrays is useful if the coordinates of the link endpoints are required for plotting the link. A simple example of the use of arrays and counters is the extraction of corridors containing transshipment points.

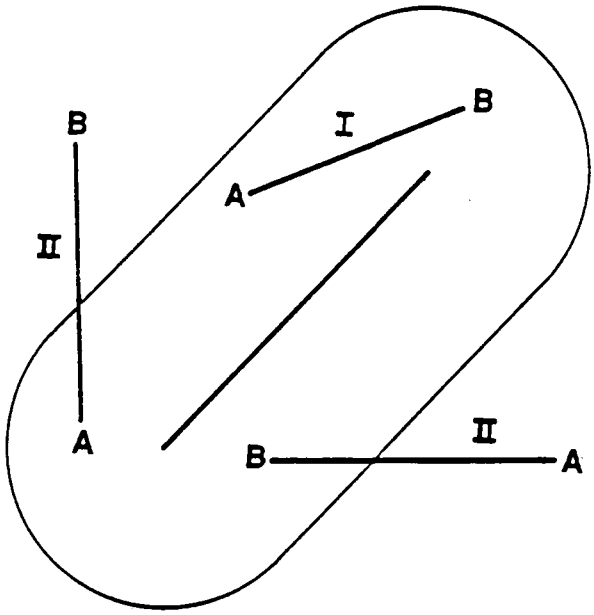
4.3 Transshipment Point Corridors.

The extraction of transshipment corridors requires the reading of the node name in the transshipment file,

followed by the comparison of this name to each node name in an array called NODEIN. For each element in NODEIN 'READ', a counter which starts with a value of zero, is incremented by one. Once a match has been found, the value of the counter is stored, and is used to read the appropriate coordinate values from the X and Y coordinate arrays. If a match is not made between the node name and the NODEIN array, the entire array will be read, and the counter will be incremented to the number of elements in the array before being reset to zero for the search for the next transshipment point.

4.4 Transport Link Corridors.

Two types of transportation link corridors can be defined: (1) a corridor which contains links which lie entirely within the limits of the corridor, and (2) a corridor which contains links which lie entirely or partially within the limits of the corridor. Figure 4.1 illustrates the difference between a link which is entirely within a transportation corridor and one which is partially within the corridor. A slightly different method of extraction is used in each case, particularly if the corridor is to be displayed. As a result, each case and the appropriate method will be discussed individually.



I: Links Entirely Within the Corridor

II: Links Partially Within the Corridor

FIGURE 4.1

Transport Link Corridors Cases 1 and 2

Case 1: Links Contained Entirely Within The Corridor. To extract links which lie entirely within the corridor a match has to be made between the node name of both the A node and the B node of the link, and an element of the NODEIN array (see case I in Figure 4.1). For example, the node name of both the A node and the B node is read from the link file. The A node name is then compared to every node name in NODEIN until a match is found. If a match is not found for the A node the link does not lie entirely within the corridor. In this case, the B node is not compared to the NODEIN array, and the next link in the link file is analyzed. Alternatively, if a match is found for the A node, the B node name from the link file is compared to all of the node names in NODEIN. If a match is found between the B node name and the NODEIN array, the link is determined to be wholly within the corridor. Otherwise, the link is said to lie partially within the corridor.

Case 2: Links Partially Within The Corridor. A link which lies partially within a transportation corridor has only one of its two endpoints within the user-defined corridor (see cases II and III in Figure 4.1). The method used to extract these links is dependent upon the output desired by the user. If a simple listing of the links is desired, the method outlined above is suitable. If one

endpoint of the link is determined to be within the corridor, by the comparison of the link file and the NODEIN array, the node names of the link endpoints can be listed from the link file. However, if the links are to be plotted, the method outlined above is unsuitable. The comparison of the node names of the endpoints of a link against an array of nodes within the corridor will only produce the coordinates of the endpoint within the corridor. This means that the entire link cannot be plotted.

To solve this problem, the assumed starting point for the extraction of links, the three arrays containing the node names and coordinates of the nodes within the corridors, has to be modified. In addition to the nodes within the corridor, arrays have to be created which contain the coordinates and the node names of the nodes which lie outside the corridor. The node name of the A node is then compared to the NODEIN array. Similar to the method presented in case one, if a match is found, the B node is also compared to NODEIN. However, if a match is not found, the program does not analyze the next link in the link file. Rather, the node name of the B node is compared to the node-names in NODEIN. Only if a match is not found between the B node and NODEIN and between the A node and NODEIN, does the program analyze the next link. If a match is found between only one of the endpoints and

the NODEIN array, the endpoint of the link which lies outside the corridor is compared to the array which contains the node names of the nodes outside the corridor. This array is called NODEOUT. Once a match has been made between the link file node name and NODEOUT, the coordinates of the point outside the corridor can be determined, using an array counter, and the entire link can be plotted.

4.5 The Extraction of Corridor Attributes.

Transportation corridor extraction methods developed to date extract elements of the African transportation network database without any regard to transportation attributes such as mode, border crossings, or railway gauge. The extraction of nodes within 100km of a link will result in all nodes within this defined area being listed. This operation may extract more information than is actually required by the user. For example, if the user is only interested in highway nodes, it is more appropriate to ask for a listing of nodes with a highway mode attribute within 100km of the link.

To perform this type of data extraction, the mode of a transportation node should be read along with its associated coordinate values. If the mode is the same as that desired by the database user, then the coordinates should be passed through the extraction process to

determine if the link is within the defined corridor. If the mode is not the same as that desired by the user, then the next node should be analyzed. By using this method, only highway nodes will be extracted from the database.

An alternative to extracting nodes of a particular transport mode is to read the attribute values of the nodes being analyzed and create more attribute extraction arrays. These arrays, when coupled with the four arrays defined above, NODEIN, NODEOUT, and the coordinate arrays, will provide locational and attribute values suitable for mapping.

4.6 Network Attributes and Map Symbols.

Computer mapping packages and subroutine libraries are capable of producing maps which consist of many different colors and line-types. Likewise, it is possible to produce a series of map symbols such as circles, squares, and stars.

Arrays containing locational data and the mode of a network element, or some other attribute value, lend themselves to being displayed using different colors or symbols for different attribute values. However, the production of computer maps of the network, and the use of different colors for different modes, means that the attribute data become more important to the user. For example, if the database user wishes to display transport

nodes which lie within a corridor by mode, with one of five different colors assigned to each of the five basic modes in the network (highway, rail, deep draft, barge, and pipeline), then the following procedure has to be performed.

Start with the four extraction arrays. These are the three arrays which are required for the extraction of links and a new array containing the mode of the nodes extracted. The array containing the mode of the transportation nodes is read and the appropriate map symbol call is made from the graphics library, or mapping package, to display the contents of the coordinate arrays.

The node name extraction array is only used for the extraction of links and transshipment points. To display links by mode the procedure is changed slightly. Instead of using the mode extraction array, the mode of the link is read along with the node names of the endpoints used to determine whether or not the link is within the corridor. This value is then used to determine which map symbol or color should be used to display the link.

To display attribute values of transportation nodes, attribute arrays have to be created during the initial node extraction procedure. Alternatively, the display of link attributes requires that only the node name and the coordinate arrays (NODEIN and NODEOUT) are created by the initial extraction process. All of the attribute values

required for the display of link attributes can be read when the node names of the link endpoints are read from the link file.

4.7 Transportation Corridor Display.

The locational and attribute extraction arrays created by the node extraction procedure are placed in the computer memory by a FORTRAN program. To display these arrays and to prevent the creation of external data files, it is most convenient and efficient for the same program to generate the graphic output. One method of doing this is to use graphic subroutine calls from a graphic subroutine library.

4.8 Graphics Subroutine Libraries.

Graphics subroutine libraries consist of a series of graphics calls, written in a suitable programming language, which generate line and point symbols. Early attempts at displaying the African transportation network database used the PLOT88 subroutine library. This library is written in FORTRAN and contains a series of graphic calls which can be incorporated into FORTRAN programs. PLOT88 can effectively display point and line data, but certain problems exist with the use of this display method.

The editing of the graphic output and the creation of a suitable finished map is difficult and time consuming using PLOT88. Every call made to PLOT88 has to be included in a FORTRAN program. As a result, modifications made to a map may result in the re-compilation and re-linking of the source program. This problem is exaggerated by the fact that the placement of a map title and the creation of a map border requires a great deal of trial and error. These features are placed on the output using FORTRAN calls which contain the numeric location of these features, and only after the program is re-compiled and re-linked is the final product drawn to the screen. Consequently, the positioning of the title and other map features often takes several attempts before the desired output is created.

Graphic output generated on the screen by PLOT88 is an example of a map created using the simple map processing model. The output is generated by a series of locational data and associated attribute values. The use of this model creates a problem when displaying the African transportation network database. Certain nodes and links are repeated in the database because they have two or more different modes. For example, if a graphic output of links within 100km of a transportation link is desired with highways displayed in brown, railroads in pink, deep draft in blue, barge links in green, and

pipelines in white, problems will occur if one or more links has more than one associated mode. A link will have more than one mode if it is possible to travel between two points in the network by more than one mode. One link will be superimposed on the other when the network is plotted, and, as a result, the color representing one or more links will be omitted.

One solution to this problem is to use the relational map processing model. This creates a data structure which can store attribute data in different layers. If the map of links was created with a layered structure, and each mode and its associated locational data was stored in a separate layer, the problem of overwriting some links would not occur. The complete map could be generated with all layers turned on. Layers can be turned off, or made inactive, to remove some links from the picture, and, thus, determine which links have more than one mode.

To solve the problems of the PLOT88 subroutine library described above, a method of displaying the database had to be found which allowed the editing of output while it was still visible on the screen, and the creation of a layered graphics file. The use of a computer-aided design (CAD) system solved both of these problems, but required the interface of the database with such a system. The following chapter outlines the method

of interfacing the database output with the Prodesign II CAD system, and highlights the advantages of using this display method.

CHAPTER V

A CAD/CAM INTERFACE FOR GEOGRAPHIC DATABASES

A computer-aided design (CAD) system is a powerful drafting package which allows the creation and editing of drawings, by entering drawing commands using a computer keyboard or mouse, and command menu. CAD drawings are defined using a coordinate system, and can be created using a series of different colors and line types. Geographic databases contain locational data (coordinates) and different types of attribute data which can be displayed on a map using a variety of map symbols or colors. As a result, CAD packages can be used to generate maps either by digitizing an existing map from a digitizing tablet, or by interfacing a series of coordinate pairs with the package.

Two CAD packages were available for use at the start of this research. These were the AutoCAD system and the Prodesign II system. The capabilities and effectiveness of both of these systems were examined and it was decided that, despite being the least powerful and least expensive of the two systems, the Prodesign II system was best suited to fulfilling the objectives of this research.

Several reasons exist for choosing to use Prodesign II rather than AutoCAD. Prodesign is capable of

performing all of the tasks required to display the output from the database extraction techniques developed. Also, it supports the same output devices as AutoCAD, with the exception of a high resolution graphics monitor. This is not regarded as a drawback as the resolution and the color graphics of the IBM color monitor, with an enhanced graphics adaptor, is sufficient for the editing of drawings and the presentation of output. Prodesign II drawing files which are used to interface data from external programs are less complex than the interchange files used by the AutoCAD system. For example, an interchange file which draws a straight line on the screen requires 5 lines in Prodesign II and 17 lines in AutoCAD. All commands used by Prodesign can be input from the keyboard, normally with a single keystroke, or from a menu which appears on the screen. Likewise, the cursor, or cross-hairs, which are used to edit drawings and to position new drawing features can be controlled by the keyboard. This capability removes the need for a mouse or digitizing tablet for cursor control. The AutoCad system does not require the use of a mouse or digitizing tablet, however, it is recommended and the user manual is oriented towards this additional hardware.

Two specific features of the Prodesign II system were required for the effective display of the transportation corridors extracted. These were the

capability of placing different parts of the drawing in different data layers, and the on-screen editing capabilities for the addition of text and map borders to the output from the database extraction techniques. The use of different drawing layers created a composite map and removed the problem of over-writing multi-modal links. Before discussing these features in detail, it is necessary to outline the method of interfacing the African transportation network database and the Prodesign II system.

5.1 Prodesign II Drawing Files.

External programs interface the Prodesign II Cad system using drawing files. Prodesign II drawing files are named using a valid DOS filename and a .PD1 extension. Once created, these files can be retrieved by Prodesign and the drawing elements represented in the file are generated on the screen. An example of a Prodesign II drawing file is listed in Appendix Two.

Prodesign II drawing files are ASCII files with the data present in character format. Each line in the file is a record in a text file, and can be one of four types. These line types are: a header line, command line, point line, and a string line. The string line was not used in any programs written to display the African transportation network database because no text was passed to the CAD

system. All of the text on the output created by Prodesign II was added to the drawing using the editing capabilities of the system.

The header line appears only once at the top of the drawing file. This line contains five numbers which define the limits of the drawing. These numbers are the minimum X coordinate of the file being created, the minimum Y coordinate of the file being created, the horizontal length of the drawing (the difference between the minimum X value and the maximum X value), and the vertical height of the drawing (the difference between the minimum Y value and the maximum Y value of the drawing). The fifth number in the header line is a zero. This section of the header line is no longer used and only remains for compatibility with earlier versions of Prodesign.

The command line instructs Prodesign II on what it is to draw, in which layer drawing elements should be placed, and in what color drawing elements should be plotted. The command line appears in the drawing file each time a new drawing element (e.g., line, circle, or box) is to be drawn. The command line contains five numbers. The first number contains the type of command which the user wants the system to perform. For example, if the first number is one then the system is instructed to draw a vector with the endpoints contained in the point

line. Similarly, if the first number is sixteen the system is instructed to draw a circle or circular arc. The second number in the command line lists the number of points which are included in the point line. The point line, which is discussed below, follows the command line. For example, if a vector was created by joining ten points with straight lines the first two numbers in the command line would be one and ten. Ten coordinate pairs would be listed in the following point line.

Entries three to five in the command line control the line features (e.g., line type and line width), and the color of drawing elements. The third entry in the command line specifies the line pattern scale. An entry of one creates a normal line pattern. Hatch patterns are also available. The fourth number specifies the line type to be used. Different line types are distinguished by their width which is measured in drawing units. An entry of 1 will create a line of one drawing unit in width, and an entry of two will create a line of two drawing units in width and so on. All of the drawing files created as part of this research use a drawing width of one. The fifth number in the command line specifies the line type and the color. Eight line types and sixteen colors are available from the Prodesign II system. The continuous line type is used throughout this research, but changes in color are used to display different modes.

The command line, in addition to specifying the drawing parameters described above, is also used to create drawing layers and locate sections of the drawing in particular layers. If the first number in the command line is twenty-one the system is instructed to locate the following drawing elements to the layer specified by the second number in the command line. In total, twenty drawing layers are available in the Prodesign II system. The line pattern, line width, line type and color indicators are unchanged in a command line which specifies a new layer.

The point lines follow the command line. Each individual line contains two numbers: the X coordinate and the Y coordinate of the point on the screen. During the retrieval of a drawing file and the generation of graphics on the screen, the system user is asked to define four points, using the cursor, within which the drawing should be displayed. If these four points are placed at the four corners of the screen, assuming the dimensions in the header file are correct, the screen coordinates of the CAD system will be equal to the coordinates input from the drawing file. If this process is not used the drawing can be drawn outside the limits of the screen, and the zoom command has to be used to generate the drawing on the screen. This can result in problems in trying to center the drawing on the screen.

5.2 Prodesign II and the Relational Map Model.

The relational map model, which was discussed in chapter one, is created by relating data from individual data files or tables and overlaying each piece of information to produce a map. Relational maps may have a layered structure in which different types of information are stored in a different map layer. Use of the data layer capabilities of the Prodesign II CAD package allows the creation of a map using the relational map model.

The relational map model is useful when the user wishes to extract a transportation corridor containing links and nodes, with each of the five basic modes in the database (i.e. highway, railway, deep draft, barge, and pipeline) displayed in a different color or layer. In this case the nodes and links within the corridor must be extracted from the database and stored along with their mode in extraction arrays. The contents of these arrays must then be analyzed. For example, if highway nodes and links are to be plotted as brown circles and lines in layer one, the appropriate command lines must precede the coordinates representing the highway nodes. This means that the algorithm being used must read the mode of a transportation node from the mode array, and write the appropriate command lines for the layer and map symbol required. A similar process has to occur for each node and link for each individual mode.

The above process was incorporated in menu-driven FORTRAN programs which define a transportation corridor and give the user a choice of displaying transportation nodes by mode; links entirely within the corridor by mode; links partially within the corridor by mode; and, railways by gauge. In addition, it is possible to plot nodes and one set of links simultaneously. For example, a map could be plotted which displayed transportation nodes by mode and links entirely within the corridor by mode.

All of the menu options described above write each mode, either node or link, to a different layer in the CAD system, and plot each mode in a different color. Highways are written to layer one and are plotted in brown, railways are written to layer two and are plotted in pink, deep draft are written to layer three and are plotted in blue, barge are written to layer four and are plotted in green, and pipelines are written to layer five and are plotted in white. The coordinates of the corridor buffer are also generated using the drawing file. This section of the drawing is written to layer six so that layers one through five can be manipulated (i.e., turned off and on) while the buffer is always present on the screen.

Layers are turned off and on within Prodesign II using the LAYER command. This command lists all twenty layers available and their current status, either on or off. If the status of a layer is represented by a value

of one, the layer is on and its contents will be visible on the screen. Alternatively, a value of zero indicates that a layer is turned off. The status of a layer is easily changed by using the LAYER command to change the status value from one to zero, or from zero to one.

Two advantages result from being able to write different modes to individual data layers. First it removes the problem of overwriting multi-modal links, and representing a link with more than one mode as a single mode link. Layers can be turned off to determine whether or not a particular link has more than one mode. Second, this approach removes the need for the user to extract nodes or links on the basis of their mode. If all nodes and links within a corridor are extracted, and a drawing file is written to Prodesign II which locates different modes in a separate layers. If the user only wishes to view the nodes and links which have a railway mode, then all layers containing node and link data except the railway layer can be turned off. In this way only railway nodes and railway links will be displayed. Figure 5.1 is an example of a transportation corridor with a radius of 400km running from (0.00n, 16.00E) to (0.00N, 24.00E). In this example the layer containing the barge nodes and links, and the layer containing the corridor buffer and link are turned on. The contents of all other layers are not displayed.

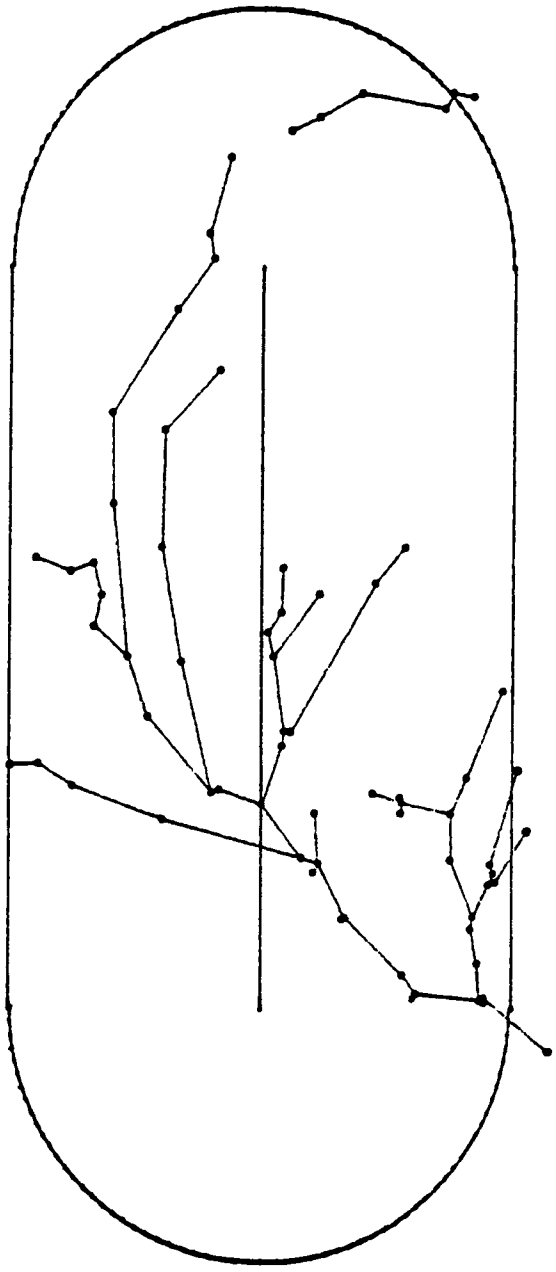


FIGURE 5.1

Transportation Corridor

5.3 On-Screen Editing Capabilities.

The output generated by the menu-driven program creates a drawing of the limits of a user-defined transportation corridor, and a series of nodes and links layered and colored by mode. Certain features have to be added before this map should be plotted. A map title stating the location of the transportation corridor, a map legend defining each color symbol, and a map border would improve the quality of the map.

All of these features can be added using either a series of draw commands (e.g., TEXT, LINE, and BOX), or, alternatively, the menu commands from the Prodesign II draw menu. Text can be added by using the cursor to define the location of the text on the screen, and then entering the desired text from the keyboard. Similar to the display of different modes, drawing elements added by the user can be located in different layers and in different colors.

The editing of drawings always takes place with the drawing being modified being visible on the screen. As a result, all changes can be seen as they occur.

5.4 Creating a Hard-copy Using Prodesign II.

Once the drawing has been edited to a suitable standard, a hard-copy can be produced on either the IBM color plotter or a dot matrix printer using the PRINT or PLOT commands. Figure 5.2 is the same corridor from Figure 5.1 with a title, legend, and border added.

5.5 Plotting More than One Link in Prodesign II.

The calculation of a transportation corridor may require that a corridor has to be calculated for more than one link. For example, two points of interest may be connected by a series of links, and, as a result, the corridor needs to be broken down into individual links.

This problem can be solved using Prodesign II and a computer program which allows the user to analyze more than one link with each program execution. Following the analysis of a link the user must be given the choice of either terminating the program or examining another link. If another link is to be examined, the same procedures are followed for the extraction and display of nodes and links as those outlined in this and earlier chapters with one small exception. The contents of the header line, the first line of the drawing file which contains the minimum X and Y values and the difference in the minimum and maximum X and Y values of the nodes extracted, will change with the examination of each new link. To ensure that the

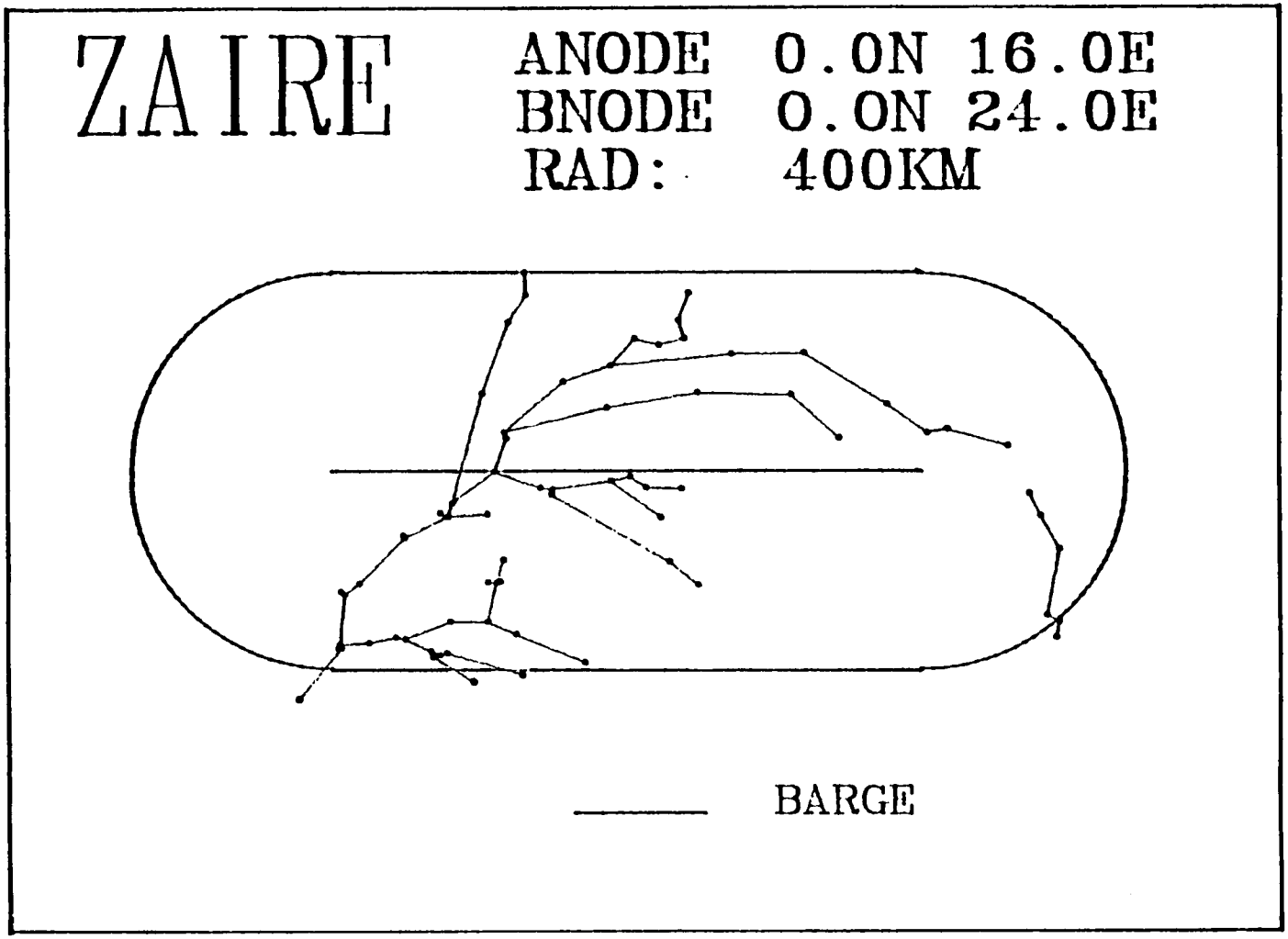


FIGURE 5.2

Transportation Corridor with Title, Border, and Legend

contents of the header line are correct all the nodes within each corridor must be extracted before it is written to the drawing file. This poses the problem of having to write the first line of the drawing file last.

To solve this problem the command lines and the points lines of the drawing file are written to a temporary file. Once all of the links required to define the corridor have been examined, and the buffer of the corridor has been calculated, the correct values for the header line can be written to the drawing file. The temporary file containing the remainder of the drawing file can then be concatenated to this file to create the complete file. Once the contents of the drawing file have been generated on the screen, the editing capabilities of Prodesign II can be used to connect the buffer endpoints, and, thus, create a complete corridor.

Figure 5.3 is an example of a corridor created by the analysis of more than one link. In this example two corridors, with radii of 100km, have been defined. One from (7.00N, 2.00W) to (8.00, 2.50W), and another from (8.00N, 2.5W) to (8.5N, 4.00W). For simplicity, all layers with the exception of those containing the highway nodes, links, and the corridor buffer have been turned off.

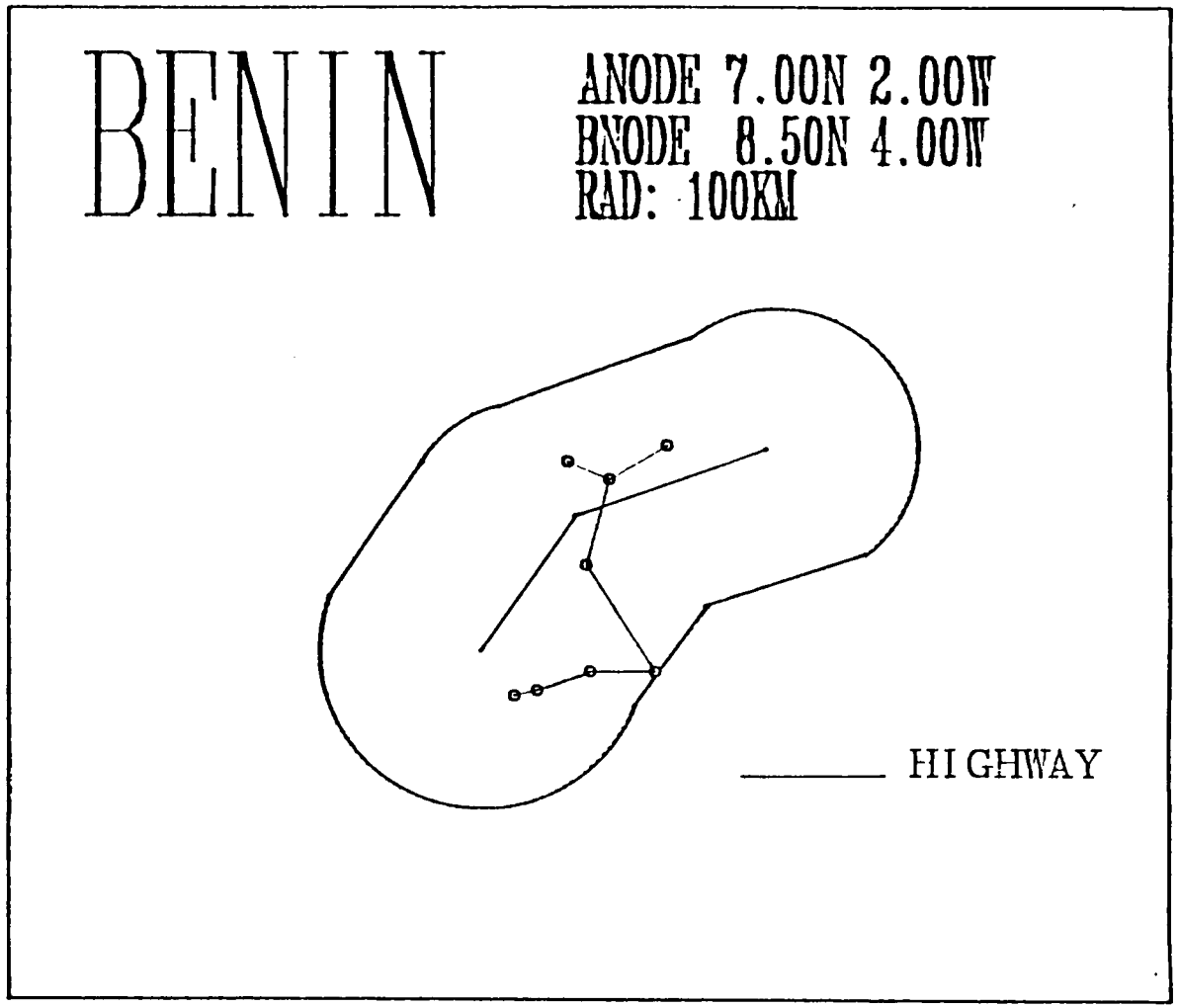


FIGURE 5.3

Connecting Transportation Corridors

5.6 Program Diskette.

The diskette accompanying this document contains two FORTRAN programs, MENU.FOR and CORR.FOR, and a data declaration file, MENU.INC. These files when compiled and linked create the executable file MENU.EXE. This file, when executed, drives a menu-driven program which prompts the user to define a transportation corridor, extracts the corridor, and creates an output file which is in a format compatible with the Prodesign II CAD package. This file can be retrieved by Prodesign II to display and edit the extracted corridor. The file HLP.TXT, also on the diskette, contains full instructions on running the MENU program.

CHAPTER VI

CONCLUSION

The extraction of transportation corridors from the African transportation network database required the transformation of locational data into rectangular coordinates, and the calculation of new coordinate values and distance measurements within this rectangular coordinate scheme. Similarly, both the locational and attribute data were transformed into a format which enabled the output from the extraction procedures to be displayed and edited using a Computer-Aided Design (CAD) package.

The methodology used to complete this research is not the only method of extracting and displaying transportation corridors. Several alternative methods were examined before deciding upon the method outlined in this paper. For example, despite the shortcomings of the graphics subroutine library available transportation corridors can be displayed in that fashion. Similarly, the calculation of the distance between a node and the transportation link being analyzed can be performed by transforming the locational data back into spherical coordinates and using the equation of great circle distance.

Defining a problem and finding a solution which works provides the opportunity of looking back, examining the methodology, and deciding what could be done to make the problem easier to solve. In this case, knowledge of spherical geometry and the use of spherical coordinates throughout would shorten the solution to the problem by removing the need to transform the locational data into a rectangular system. Another advantage of spherical geometry is that the need for adjustment factors, due to map projection distortion, would be eliminated. A methodology using spherical geometry would require an algorithm to find the shortest great circle path between a node and the link, and another algorithm to calculate the distance along that great circle.

The examples used in this research were from the African transportation database and throughout the text possible uses of a transportation corridor in Africa, particularly with respect to famine relief, have been outlined. However, great pains were taken throughout this research to ensure that the method of extracting transportation corridors would be effective across the globe, and not just in the test cases of Central and West Africa. In these regions one degree of latitude and one degree of longitude are approximately equal and the adjustments to compensate for the distortion of the globe are virtually redundant. The question now remains, how

can transportation corridors be used in other parts of the world?

One of the most obvious uses of transportation corridors is in conjunction with techniques which define routes through a network on the basis of constraints. Two examples of these types of procedures are shortest path algorithms, and techniques which minimize or maximize a given value along a route.

Shortest path algorithms find the shortest route between two points in a network. Once this route has been found, a transportation corridor can be extracted to provide the network user with information on the particular modes of transport which link the two points and other points of interest within a user specified distance of the route.

Other procedures define routes by maximizing or minimizing an attribute value along a route. For example, during the transportation of radioactive waste materials a route may be chosen which minimizes the number of people who live on or near suitable transportation links. Transportation corridors can be used in this case to determine which population centers are within a given distance, or increasing distances (100, 200, or 300km) of the route chosen. In this way, towns and cities at greatest risk, in the event of an accident, can be determined.

Transportation databases are a part of the realm of geographic databases, and, similar to the application of the techniques developed in this thesis with transportation modelling and transportation databases, the extraction of corridors can be applied to any locational database. For example, corridors containing polygons of land within a user-specified distance of a feature of interest could be extracted from a land-use database. This application may be useful to land-use planners and, given the location of the center of each polygon, would only require minor adjustments to the methodology outlined in this paper.

Global applications of extracting corridors from geographic databases do exist, and, having developed a technique of extracting and displaying such corridors, a possible extension of this research would be to couple the techniques documented with the other transportation or geographic applications outlined above.

BIBLIOGRAPHY

BIBLIOGRAPHY

Aronson, P. "Attribute Handling For Geographic Information Systems", Proceedings: Eighth International Symposium Computer-Assisted Cartography Baltimore, Maryland, March 29-April 3, 1987.

Carter, J. R. Computer Mapping: Progress in the 80's. Resource Publications in Geography, Association of American Geographers, Washington, D.C., 1984.

Dangermond, J. "G.I.S. Trends and Experiences" Proceedings: Second International Symposium on Spatial Data Handling, Seattle, Washington July 10-15, 1986.

Dyck, V. A., J. D. Lawson, and J. A. Smith. FORTRAN\77 An Introduction to Structured Problem Solving. Reston Publishing Company, Inc., Reston, Virginia, 1984.

Electric Power Research Institute, Transportation Network Models For Energy Supply Analysis, Vol. 3, Palo Alto, EPRI, 1982.

Horowitz, E. and S. Sahni. Fundamentals of Data Structures, Computer Science Press, Rockville, Maryland, 1983.

Marble, D. F. and L. Sen. "The Development of Standardized Benchmarks for Spatial Database Systems", Proceedings: Seventh International Symposium on Computer-Assisted Cartography, Washington, D.C., March 11-14, 1985.

Snyder J. P. Map Projections Used by the U.S. Geological Survey, 2nd Edition. U.S. Government Printing Office Washington, D.C., 1982.

Waugh, T. C. and R. G. Healey. " The GEOVIEW Design: a relational database approach to geographic database handling", Proceedings: Second International Symposium On Spatial Data Handling, Seattle, Washington, July 15-10, 1986.

Webster, R. Prodesign II: The Easy To Use CAD System American Small Business Computers, Inc., Pryor, Oklahoma, 1986.

White, M. S. "Technical Requirements and Standards for a Multipurpose Geographic Data System", The American Cartographer, Vol 11, No. 1, 1984.

Young, T. L. and M. L. Van Woert. PLOT88 Software Reference Manual, 2nd ed., PLOTWORKS, Inc., La Jolla, California, 1987.

APPENDICES

APPENDIX ONE

African Transportation Network Database:

Database Structure

The following are the countries in the African transporation network database, and the country code assigned to each one.

Country	Country Code
Angola	ANG
Benin	BEN
Botswana	BOT
Burkina Faso	BKF
Burundi	BUR
Cameroon	CAM
Central African Rep.	CAR
Chad	CHD
Congo	CON
Djibouti	DJI
Equitorial Guinea	EQG
Ethiopia	ETH
Gabon	GAB
Gambia	GAM
Ghana	GHA
Guinea	GUI
Guinea Bissau	GUB
Ivory Coast	IVC
Kenya	KEN
Lesotho	LES
Liberia	LIB
Malawi	MAL
Mali	MAI
Mauritania	MAU
Mozambique	MOZ
Namibia	NAM
Niger	NIG
Nigeria	NIA
Rwanda	RWA
Senegal	SEN
Sierra Leon	SLN
Somalia	SOM
South Africa	RSA
Sudan	SUD
Swaziland	SWZ
Tanzania	TAN
Togo	TOG
Uganda	UGA
Zaire	ZAI
Zambia	ZAM
Zimbabwe	ZIM

The following is an outline of the structure and content of the Node, Link, and Transshipment files of the African transportation database.

NODE FILE.

Column	Data Entry
1	N (Node)
3-12	Latitude
12-22	Longitude
26	Mode
28-30	Node Class Name
32-37	Centroid Flag
44-53	Nodename
55	Flag
57-59	Country Code
61-70	Place Name

Latitude and Longitude:

These values are listed in degrees and their decimal parts. A negative latitude indicates a point north of the equator. A negative longitude indicates a point west of prime meridian.

Mode:

D = Deep Draft Ports
B = Barge
H = Highway
R = Railway - 1.067m gauge
N = Railway - 1.000m gauge
S = Railway - 0.600m gauge
T = Railway - 0.610m gauge
U = Railway - 0.615m gauge
V = Railway - 0.765m gauge
W = Railway - 1.055m gauge
X = Railway - 1.432m gauge
Y = Railway - 1.435m gauge
P = Pipeline

Node Class Name:

NR0 or NR1 - Rail Nodes
NH0 or NH1 - Highway Nodes
NB - Barge Nodes
ND - Deep Draft Nodes
NP - Pipeline Nodes

The Node Class Name is a three character set. The N represents a node element, the second character the mode, and the third character in the rail and highway modes a particular node attribute. If the third character in the rail node class is equal to 1 then the node is in South Africa. If the third character is equal to 0 then the node is not in South Africa. If the third character in the highway node is equal to one then the node is a border crossing. If the third character is equal to 0 then the node is not a border crossing.

Flag:

A value of 3 indicates a border crossing.

LINK FILE.

Column	Data Entry
1	L (Link)
3-12	A Nodename (Start Point)
13-22	B Nodename (End Point)
24-28	Length of Link (Kilometers)
30-35	Link Class Function Name
41-45	Link Capacity Class Name
54-61	Digitizing Flags
65	Mode
67-69	Country Code

Link Class Function Name:

LR - Rail Links
LH - Highway Links
LB - Barge Links
LP - Pipeline Links

These class functions indicate whether the railway track is a single or double track, the highway surface material, and the direction of barge movements upstream, downstream, or lake barge.

Capacity Class Names:

This class function only exists for the rail links, and includes the railway gauge (the same descriptors are used to describe the railroad gauge as are used to describe the mode in a node file), the train length, the opertaing days of the railroad and the hours of operation, and whether or not the railroad is electrified.

Mode:

The same mode descriptors are used in the link files as are used in the node files.

TRANSSHIPMENT FILE.

Column	Data Entry
1	T (Transshipment)
3-12	A Nodename
13-22	B Nodename
48	A Mode
53	B Mode
55-57	Country Code

All Transshipments are treated as two-way links.

e.g., if a transshipment point has a rail and a highway mode, it is possible to transfer from rail to highway, and from highway to rail.

APPENDIX TWO

Prodesign II Drawing File Structure

The following file is an example of a Prodesign II drawing file. Each line is annotated to describe its function in the generation of a drawing. In an actual .PD1 file spaces do not exist between each line.

```
0.00 0.00 1000.0 1000.0 0  - Header Line

21 1 0 0 0                 - Command Line. Specifies the
                             layer (#1).

1 2 1 0 8                  - Command Line. Describes a
                             vector drawn by the
                             connection of two points
                             in color #8 (brown).

100.0 200.0
200.0 100.0                - Point Lines, X and Y values.

21 2 0 0 0                 - Command Line. Specifies new
                             layer (#2).

16 2 1 0 8                 - Command Line. Describes a
                             circle defined by its
                             center and a point on the
                             circumference. Color #8.

100.0 300.0
100.0 320.0                - Point Lines, X and Y values.
```

VITA

Graham M. Barrowman was born in Dumbarton, Scotland on September 8, 1964. After attending primary and secondary schools near Glasgow, Scotland, he attended Glasgow University where he captained the varsity golf team in 1986, and, in June of the same year received a Bachelor of Science (Hons.) degree in Geography. In September of 1986, he accepted a teaching assistantship in the Department of Geography at The University of Tennessee, Knoxville, and crossed the Atlantic to pursue an interest in computer cartography. He received a Master of Science degree in Geography in August 1988.

The author is currently employed as a Research Analyst with PlanGraphics, Inc. in Silver Spring, Maryland. PlanGraphics, Inc. is a Geographic Information System (G.I.S.) consulting firm.