NCGIA

National Center for Geographic Information and Analysis

VT/GIS The von Thünen GIS Package

by Rustin F. Dodson

National Center for Geographic Information & Analysis
University of California at Santa Barbara
Santa Barbara, CA 93106-4060

Technical Report 91-27

August 1991

Simonett Center for Spatial Analysis
University of California
35 10 Phelps Hall
Santa Barbara, CA 93106-4060
Office (805) 893-8224
Fax (805) 893-8617
ncgia@ncgia.ucsb.edu

State University of New York
301 Wilkeson Quad, Box 610023
Buffalo NY 14261-0001
Office (716) 645-2545
Fax (716) 645-5957
ncgia@ubvms.cc.buffalo.edu

University of Maine
348 Boardman Hall
Orono ME 04469-5711
Office (207) 581-2149
Fax (207) 581-2206
ncgia@spatial.maine.edu

National Center for Geographic Information and Analysis
University of California at Santa Barbara
August, 1991

Published by:

National Center for Geographic Information and Analysis
University of California
Santa Barbara, CA 93106-4060 USA

Telephone 805/893-8224
FAX 805/893-8617

PREFACE

VT/GIS was developed with two main objectives: first, to investigate the possibility of using a commercially-available GIS package to develop an automated tutorial of a classic model in Geography; and second, to provide students with an interactive environment in which to investigate the results of a spatially-relaxed von Thünen model.

One of the constraints on the development of VT/GIS was that it be created from an existing GIS. This situation involves a tradeoff between efficiency and "reinventing the wheel". The use of an existing GIS greatly simplifies the task of creating the tutorial, but at the same time the tutorial becomes subject to the limitations of the GIS engine. Specifically, the tutorial must cope with the algorithms, data handling conventions, and interface of the GIS with which it is coupled. The result of this situation is that VT/GIS has some shortcomings, as the GIS modules it employs are not optimized for the specific task of tutoring students in von Thünen.

These problems could be solved, of course, by programming the entire tutorial from scratch, but that was never the intention. Using an existing GIS can greatly simplify the development of a package such as this one, but at the same time it will always introduce difficulties. Yet in spite of its admitted weaknesses, it is hoped that VT/GIS will make a successful educational tool as well as inspire the development of other such tutorials.

NOTE ABOUT IDRISI

The VT/GIS package is driven by a linked set of modules taken from IDRISI, a grid-based geographic analysis system developed at Clark University. The IDRISI modules are provided courtesy of the IDRISI project, Graduate School of Geography, Clark University, Worcester, Massachusetts, 01610. *The modules are part of a copyrighted set and are only for use with the VT/GIS package as it is distributed.*

ACKNOWLEDGEMENTS

Thanks go to Dr. J. Ronald Eastman for allowing the IDRISI modules to be distributed, and for creating straightforward, functional, and inexpensive GIS software. Thanks also go to Dr. Michael Goodchild and Dr. Waldo Tobler for providing the original impetus for this project, and to Karen Kemp for invaluable editorial help. Support for the NCGIA through the National Science Foundation, grant SES 88-10917, is gratefully acknowledged.

TABLE OF CONTENTS

DESCRIPTION OF COMPONENT DOCUMENTS

This package contains four component documents: an introduction to von Thünen and VT/GIS, a user's manual, and two student exercises. Each document may be reproduced and used separately.

* **Introducing von Thünen's Model and the VT/GIS Package**

  This is a brief introduction for the student regarding concepts relevant to von Thünen's work and to the VT/GIS Package.

* **VT/GIS User's Manual**

  Includes information on installation, hard/software requirements, running the program, and error messages. An index to all menu options is provided at the beginning of this section.

* **Student exercise 1: Introducing VT/GIS**

  This exercise runs the student through the model under various stages of constraint relaxation.

* **Student exercise 2: Advanced activities**

  This section contains several short explorations which go further in the investigation of the model. Some of these activities investigate GIS concepts more than agricultural location theory, and some require access to the entire IDRISI package.

APPENDICES

* **Appendix A: Technical documentation**

  This appendix provides instructions for modifying and enhancing the VT/GIS package, as well as annotated flowcharts which explain exactly how VT/GIS calculates the Thünen model.

* **Appendix B: Index to disk contents**

  This appendix lists and explains the files stored on the included diskette.

INTRODUCING VON THÜNEN'S MODEL

INTRODUCING VON THÜNEN'S MODEL AND THE VT/GIS PACKAGE

INTRODUCTION

Why do things occur where they do? What factors are responsible for the location of features on the physical landscape? In seeking answers to these spatial questions, a number of models of location theory have been developed by geographers and others as early as 150 years ago. These models "are of fundamental importance in understanding the world around us" (Tobler, 1991).

The work of Johann Heinrich von Thünen, published in 1826, makes up one of the classical models of agricultural location theory. In, *The Isolated State*, von Thünen develops a model which determines what types of agricultural land use will occur around a hypothetical town at the center of a vast, fertile plain. Working as he did in an age of low technology, von Thünen's model adopts a very simplified view of reality with numerous limiting assumptions. These assumptions have been the base for many criticisms of von Thünen's work by both economists and geographers (Kellerman, 1989a).

In later sections of his book, von Thünen relaxes some of his assumptions and examines the effects of these relaxations on the outcome of the model. Indeed, much has since been written on the subject of relaxing assumptions of the classical Thünen model; past efforts include computer programs which calculate the model under a relaxed set of spatial assumptions. Marble and Anderson (1972) present a tutorial program which allows for a simple highway network, multiple markets, and a variety of user-defined aspatial parameters. In Tobler (1991) a program written some years ago is described which allows for multiple markets as well as a climatic gradient over the otherwise isotropic plain. Kellerman (1989b) provides an excellent overview of past works which relax one or more of von Thünen's assumptions.

An implementation of von Thünen's theory using Geographic Information System (GIS) technology allows several of the classical model's limiting assumptions to be relaxed simultaneously, and provides an interactive user environment as well as advanced display capabilities. By allowing for such items as a transportation network, multiple towns, and varying soil fertility, a GIS can serve as a powerful tool for visualizing the agricultural landscape of von Thünen's "isolated state" under a variety of initial conditions.

This paper introduces a modular software package, VT/GIS[1], which calculates von Thünen's classical model under a relaxed set of geographic assumptions. VT/GIS allows the user to interactively create a wide variety of initial conditions, and then to calculate the model and display its results graphically. The intention is to provide an instructive tool for visualizing the effects of changing geographical parameters within von Thünen's model. The first section of this paper introduces the basics of von Thünen's classical model, and later sections describe the workings of the VT/GIS package.

---

[1]VT/GIS was created in large part by linking existing modules from IDRISI, a grid-based geographic analysis system developed at Clark University. IDRISI is a registered trademark of Clark University.

VON THÜNEN'S MODEL

[skip to the next section if you are already familiar with the von Thünen model]

Assumptions

The isolated state envisioned by von Thünen consists of a single town at the center of an *isotropic* plain--that is, a plain which is physically the same everywhere. The plain is dotted with small farms which obtain manufactured goods from the town and in turn provide the town with agricultural goods. A large number of landowners exist who rent their farmland to the highest bidder. Soil fertility and climate are unchanging across the plain, and only one mode of transportation[1] exists for moving farm products to the town. Transportation costs are equal in all directions. In addition, farmers and landowners are assumed to be completely rational[2] in their economic behavior.

Under these assumptions, von Thünen attempts to predict the pattern of agricultural land use that will develop around the town. He sees distance from the town (which is proportional to transport cost) as playing a major role in determining the distribution of various land uses. Agricultural products which are very delicate, perishable or bulky and therefore expensive to transport must be grown close to the town in order to be profitable. If not, the cost of transporting these goods will outweigh the revenue fetched by selling them at the town. By this reasoning, von Thünen concludes that the resulting pattern of land use will resemble a series of concentric rings around the town, with each ring containing a different class of crops and/or a different degree of farming intensity.

Land rent

Land rent is the factor which determines which crops will be grown at a given distance from the town. Also called economic rent, bid rent, and location rent (Kellerman, 1989a), land rent refers to "that portion of the farm revenue that is left after deduction of the interest on the value of the buildings, timber, fences, and all other valuable objects separable from the land, that portion which pertains to the land itself..." (von Thünen, 1966 English translation, pg. 18). Land rent represents the maximum amount of money per unit land area a landlord can charge to a farmer for the use of the land. A simplified equation which reflects this definition of land rent is:

$$R = Y(P - C_p) - Y(C_t d) \qquad \text{(equation 1)}$$ [3]

Where:

- R = land rent in dollars per unit land area
- Y = yield in units of commodity per unit land area
- P = market price in dollars per unit commodity
- C_p = production cost in dollars per unit commodity
- C_t = transport cost in dollars per unit commodity per mile
- d = distance from the town in miles

---

[1] von Thünen assumes a horse-drawn cart as the sole means of transportation throughout most of his work.

[2] A rational person seeks to maximize profits, and has all of the information necessary to do so.

[3] Based on a modification of von Thünen's rent equation by Losch (1954, pg. 38-44) and Dunn (1954, pg. 7-8), cited in Kellerman, 1989a.

Thus we see that land rent is analogous to profit. It depends on the revenue gained by selling a crop minus the costs of producing the crop and transporting it to the town. At any location on the plain, any of a variety of crops can be grown. The crop that will be grown at a given location is the one that is most profitable--the crop that produces the highest land rent. Since farmers and landlords are economically rational, they both know what crops are most profitable at a given distance from the town. Therefore the landlord will charge as much rent as the market will bear, and the farmer, in order to afford this rent, will cultivate the most profitable crop possible.

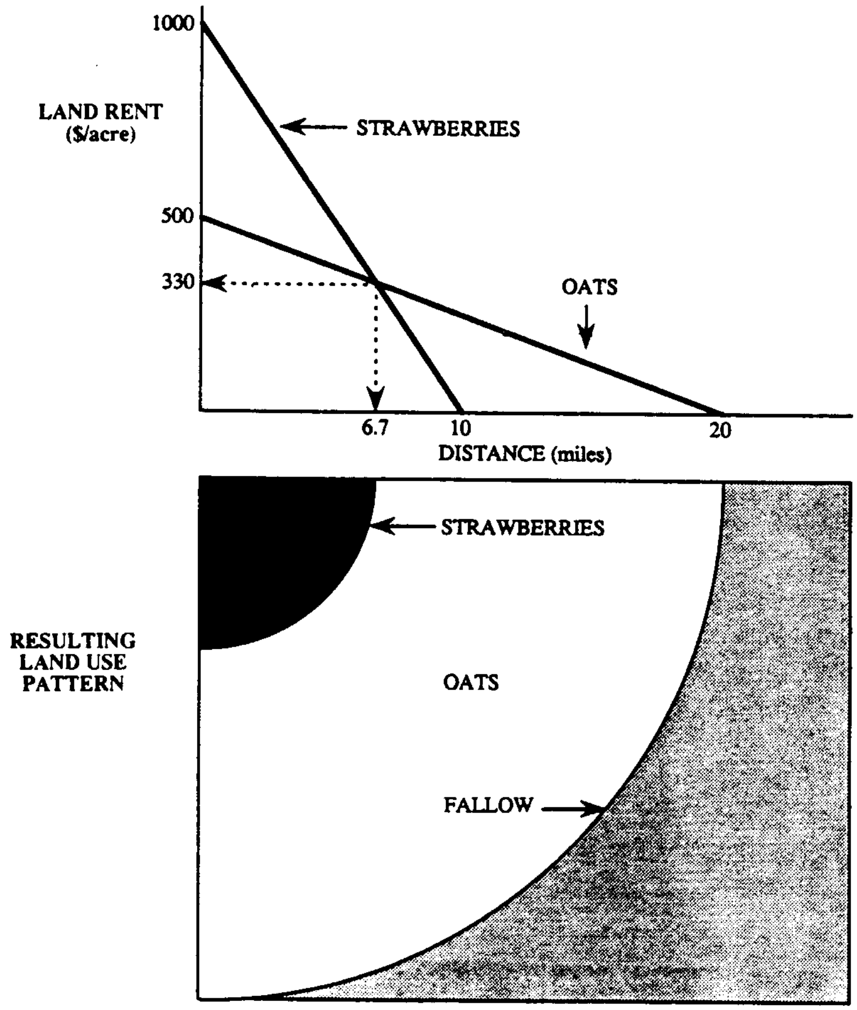
The relationship between the land rents produced by various crops and the resulting rings of land use is shown graphically by means of a *bid-rent curve*. The land rent equation above is essentially the equation for a line of the form (Y = a - bX) where *a* is the Y-axis intercept and *b* is the slope of the line. In equation 1 above, *R* represents the Y variable; the term $Y(P - C_p)$ represents the Y-intercept; the term $Y(C_t)$ represents the slope (which is negative since land rent decreases with increasing distance); and *d* represents the X variable.

Figure 1 shows the relationship between two agricultural products, strawberries and oats. Let's assume that an acre of land can produce a ton of strawberries or a ton of oats; the price paid in town is $1000 per ton of strawberries and $500 per ton of oats; and finally that transport costs for the two crops are $100 per ton-mile[1] for strawberries and $25 per ton-mile for oats. For this example we will ignore costs of production (by assuming they are equal for both crops) and only look at transport costs. Which areas will grow strawberries and which will grow oats?

---

[1] A ton-mile represents the movement of one ton of commodity for one mile.

Figure 1
Bid-rent curve for strawberries and oats



From figure 1 we see that at the town (where distance is equal to zero), growing strawberries is far more profitable than growing oats. Given their higher market price, strawberries will be grown in the areas closest to the town. However, strawberries, being both perishable and delicate, are more expensive to transport than oats. For each mile that strawberries must be transported, profit (and land rent) decreases by $100. So, at a distance of ten miles from the town, the cost of transporting strawberries will equal the revenue obtained at the market, and profit/land rent will equal zero.

Oats will bring about less revenue when grown near the town, however they are cheaper to transport, so their slope on the bid-rent curve is less steep. For this reason, oats are profitable to farm out to a distance of 20 miles as compared to the 10 mile limit for strawberries. In addition, while strawberries will produce a profit at distances from 6.7 to 10 miles from the town, oats will actually be cultivated at these distances because they are more profitable (6.7 miles represents the point on the graph where land rent is equal for both crops). Thus, by simply examining the bid-rent curve, we can see the extent of the resulting land use rings by looking at the lines on the graph associated with the highest land rents for all distances from the town.

USING GIS WITH VON THÜNEN'S MODEL

With the classical model, a single calculation is all that is needed to define the land use pattern for a given commodity. In the above example, by calculating the intersection of the graphs for oats and strawberries on the bid rent curve, the resulting distance value, 6.7 miles (with an associated land rent of $330 per acre), defines the radius of a simple land use ring. However, when working with the relaxed geographic assumptions of the VT/GIS package, a single number (a ring radius) is no longer sufficient for defining the spatial distribution of land use for a commodity. The introduction of variable costs and yields over space can cause the classical ring pattern to warp, bend, and in some cases break down completely. Thus land rents for each commodity must be calculated *at each point in space* before the commodity which produces the highest rent at that point can be identified. This is a task well-suited for a computer. However, as a computer can not calculate values over an infinite set of points, space is discretized into finite units (called *cells* or *pixels*) which are used to represent continuous space.

The purpose of this package is not to provide a modern version of von Thünen's model which can be applied to real-world situations. Rather, the intention is to relax some of the model's assumptions about space and calculate the model in a more complex and in a slightly more realistic geographic environment. GIS technology allows us to explore the nineteenth century world of von Thünen using a modern set of analytical tools. These tools enable us to visualize the landscape and to investigate more complex spatial relationships that were not easily examined in von Thünen's day.

Relaxed assumptions: convoluting the isotropic plain

Von Thünen was forced to assume a homogeneous land surface. Transport costs and soil fertility/productivity are assumed constant everywhere, and only one town exists for the sale of crops. The VT/GIS package, on the other hand, relaxes these assumptions by introducing several features which convolute the original, unvarying plain.

* *Railroads*: By introducing railroads to the plain, transportation costs are no longer constant. The VT/GIS program assumes that rail transport rates are ten times cheaper than overland (horse wagon) transport. Therefore farms near the railroad network will be able to produce expensive-to-transport goods at greater distances from the town than farms which are not near a railroad line.

* *Lakes*: Lakes function as barriers to transportation. In situations where a lake is between a farm and a town, goods must be transported around the lake. Exception: A railroad can be made to cross over a lake. In these cases it is assumed that the portion of the railroad which crosses a lake is a bridge.

* *Multiple towns*: Up to ten towns can be used with the VT/GIS demo. These can be put anywhere on the plain.

* *Yield weights*: Variations in crop yield over the plain can be accounted for by creating yield weight surfaces. These surfaces assign to each location on the plain a weight value ranging from zero to one. These numbers represent the proportion of the optimum crop yield a given area is able to produce. The yield weights can be used to account for several types of variation within the plain which would affect crop yields--soil fertility, temperature, rainfall, slope, cost of irrigation water, etc.

Assumptions of the VT/GIS program

While VT/GIS does away with some of von Thünen's original assumptions, the package retains the Thünen model's economic assumptions and introduces a few of its own.

* *Discrete space*: The relaxed geographic assumptions of VT/GIS require that calculations be made at each point in space. This is clearly impossible, thus VT/GIS uses a 100 x 100 grid of *cells*, with each cell representing nine square kilometers on the ground. The 100 x 100 grid appears coarse and blocky on the screen, but is necessary in keeping computation times reasonably short. In transforming the plain into a regular grid of cells, movement along the plain is restricted to eight directions--from any cell it is only possible to move to one of the cell's eight neighbors. For this reason, the discrete space assumption introduces some error into the calculations of the shortest-path distance to market and therefore transport cost.

* *Three bands (milk, grain, livestock)*: In his book, von Thünen describes six rings which are associated with various sets of commodities and different farming practices (intensities of land input). For reasons of simplicity, VT/GIS collapses the original six rings into three *bands* which are given short names representing a key commodity occurring in that band. Thus when the documentation refers to "milk" it actually is referring to the commodities and farming practices associated with von Thünen's "free cash cropping" ring.

TABLE 1
The three bands of VT/GIS

VT/GIS band	Thünen ring	Example Commodities
MILK	Free cash cropping	milk, cauliflower, lettuce, strawberries
GRAIN	Forestry	fuel, timber, charcoal
	Crop alteration system	potatoes, rye, oats, clover, wheat
	Improved system	rye, barley, oats, ley grass
	Three field system	rye, oats
LIVESTOCK	Stock farming	cattle, butter, wool, fat meat

* *Transportation costs* (C_t): Railroad travel is ten times cheaper than overland (horse wagon) travel. To determine the transportation cost for a given crop at a particular point, VT/GIS first finds the least-cost path to the nearest town and tallies the cost of traversing the cells in that path. This results in a 100 x 100 cell cost surface in which each cell's value represents the cost of transporting to the nearest town. Each cell in the cost surface is then multiplied by a transportation cost factor for each commodity in order to create three transportation cost surfaces (one per commodity).

* *Constant production costs* (C_p): Production costs are assumed to be the same for all commodities at all locations on the plain. This was done to limit the number of user-defined parameters and hence the complexity involved in operating VT/GIS. Also, ignoring production costs allows the analysis to concentrate on the spatial parameters, as these are considered more interesting from a geographic standpoint.

* *Constant optimal yield value* (Y): The optimal yield value, Y, represents the amount of commodity which can be produced under perfect climatic, soil, and topographic conditions. This value is fixed at one ton of commodity per cell for all commodities. Y can be made to

vary over space by assigning different yield weights to locations on the plain. Y can be made to vary for each crop by using a different maximum weight for each crop. For example, if a given area can produce a ton of oats but only half a ton of strawberries, oats can be given a weight of 1 and strawberries a weight of 0.5.

The assumptions adopted by von Thünen were designed so that his model could concentrate on the effect of distance as the sole determinant of land rent and therefore land use. VT/GIS maintains von Thünen's economic assumptions, but relaxes many of those which pertain to space. Thus VT/GIS creates a modelling environment which allows the analysis to focus on spatial parameters--not simply distance, but position with respect to the transportation network, barriers to transportation, and a plain of variable fertility. In doing this, VT/GIS attempts to present the classical model of von Thünen in a manner which is of particular interest to geographers.

REFERENCES

Kellerman, A., 1989a, "Agricultural Location Theory 1: Basic Models", *Environment and Planning A*, 21 1381-1396.

Kellerman, A., 1989b, "Agricultural Location Theory 2: Relaxation of Assumptions and Applications", *Environment and Planning A*, 21 1427-1446.

Marble, D. P., and Anderson, B. M., 1972, "LANDUSE, A Computer Program for Laboratory Use in Economic Geography Courses", *Commission on College Geography Technical Paper No. 8*, Association of American Geographers, Washington, D.C.

Tobler, W., 1991, Paper presented at the NSF-sponsored GIS and Social Science Conference, Santa Barbara, CA, March, 1991.

von Thünen, J. R., *Der Isolierte Staat*, English translation, 1966, *von Thünen's Isolated State*, Ed. P. Hall, translated by C. M. Wartenberg, Pergamon Press, Oxford.

USER'S MANUAL

VT/GIS USER'S MANUAL

INTRODUCTION

This manual first provides information on hardware requirements and installation of VT/GIS. The next section is organized by screen menu (MAIN, YIELD, and RUN) and covers every menu option found in the package. A final section provides error message and troubleshooting information.

HARDWARE AND SOFTWARE REQUIREMENTS

The VT/GIS package requires a PC running DOS version 3.3 or higher with at least 512k of RAM, a hard disk drive, a color monitor, and a Microsoft-compatible mouse. When uncompressed and installed from the enclosed diskette, the package takes up about **1.5 megabytes** of disk space. This space requirement can increase if you save images with the archive utilities.

INSTALLATION

VT/GIS directory structure

When installed, VT/GIS is organized into three directories. The *VTGIS main directory*, assumed to be called "**VTGIS**" (see below), is the directory under which the entire package is stored. This directory contains several files and two subdirectories: "**VTGIS/BIN**" contains batch files, programs, and image files which make up most of VT/GIS. The subdirectory "**VTGIS/ARCHIVE**" contains any displays which were saved using the archive utilities. When using VT/GIS, you should be in the VTGIS main directory, not in the BIN or ARCHIVE subdirectories.

Normal installation: on D:\VTGIS

Although VT/GIS can be installed on any disk drive in any subdirectory, the version included here is configured to go on the D: drive in a directory called VTGIS. If you want it to reside elsewhere, see the next section.

The VT/GIS package is stored on the enclosed diskette as a compressed archive file, VTGIS.EXE. To install the package, create a directory on your D: drive called VTGIS (type "MD VTGIS"). Then, after inserting the diskette, type "INSTALL NORMAL" at the A:\> prompt. The compressed files and subdirectories stored in VTGIS.EXE will be uncompressed and stored in the VTGIS directory on the D: drive (the compressed file VTGIS.EXE will not be altered). Then change to the VTGIS directory and type "START" to update your path and bring up the main menu.

D:\> **MD VTGIS**
A:\> **INSTALL NORMAL**
D:\VTGIS\> **START**

NOTE: When you type "START", an additional item, "**.\BIN**", is added to your system's search path so that VT/GIS commands can be found by the system. The START command need not be used again unless your system has been rebooted or turned off. Use the MENU command at the

DOS prompt to display a menu. To remove the change made by VT/GIS to your system environment, reboot your system.

Installing on other drives/subdirectories

To install on a directory other than D:\VTGIS, first follow the above procedure, substituting the desired pathname for "NORMAL" (e.g. type "INSTALL C:\PROGRAMS\THUNEN" to install the package on that drive and path. Remember that you first need to create the desired directory or subdirectory.). Then you must use a text editor to modify a file called IDRISI.ENV which is located in your VTGIS main directory.

NOTE: If you own a copy of IDRISI, you can accomplish the following task by running **ENVIRON** from the VTGIS\BIN directory and setting up the appropriate data path.

The file to be modified, "IDRISI.ENV", is located in the target directory you specified with the INSTALL command--this is your VTGIS main directory. You need to modify the following two lines:

Default data drive: *d:*
Default data path: *\vtgis\bin*

so that they include the desired data drive and path. For example, if VT/GIS was installed in the C:\PROGRAMS\THUNEN subdirectory, change the above lines of the file C:\PROGRAMS\THUNEN\IDRISI.ENV so that they read:

Default data drive: *c:*
Default data path: *\programs\thunen\bin*

(Remember to enclose the pathname above with backslashes, "\").

Now change to your VTGIS main directory and type "START" to update your path and bring up the main menu. See the note above regarding the START command.

C:\PROGRAMS\THUNEN\> **START**

DELETING THE ENTIRE VT/GIS PACKAGE

The archive file (VTGIS.EXE) on the enclosed diskette is not altered during the installation process. The VT/GIS package can be easily re-installed again and again. Therefore, if you need to free up some hard-disk space, or if you accidentally delete files so that the program no longer runs properly, you can quickly and easily delete the entire VT/GIS package and then re-install it from the diskette when desired.

From the VTGIS main directory, type "DESTROY". The system will then ask if you really want to delete the package. If you want to abort the destruction of VT/GIS, press **Ctrl+c** and enter **y** to terminate the batch job. If you want to continue, press any key and then enter **y** in response to the three "*Are you sure? (Y/N)*" prompts

RUNNING THE VT/GIS PACKAGE--HELPFUL TIPS

* RUNNING THE PROGRAM: Type **start** from your VTGIS main directory (assumed to be D:\VTGIS).

* DEFAULT MODEL: When first installed, VT/GIS contains the classical Thünen model as a default. Press **8** at the main menu to view it. From here, try modifying a few parameters at a time and then re-computing and displaying the model.

* DISPLAYING THE MENU: Type **menu** at the DOS prompt to see a list of menu options. The **start** command need only be used after installation of the VT/GIS program or after the computer has been turned off or re-booted.

* RE-COMPUTE TO CHANGE THE SCREEN DISPLAY: When you modify any of the model parameters, the screen display of the model (main menu option **8**) will not change until you re-compute the model (main menu option **7**) using the newly-created parameters.

* CLEARING THE SCREEN DISPLAY: If you are looking at a color display and nothing appears to be happening, press **Enter**. This will clear the display from the screen and allow the program to continue.

* ADDING NEW FEATURES: If you want to add to an existing set of features, (e.g., adding a fourth town to a current set of three), you have to re-enter the existing ones. The best way to do this is to display the existing features and then trace directly over them.

* FILENAMES TYPED IN BY THE USER: Some menu options require you to type in the names of certain files. These must be typed in correctly in order for the program to work. When possible, VT/GIS displays a verification image of what it thinks you entered. It this image doesn't look right, run the menu option again.

* ESCAPE TO THE DEFAULT MODEL: If your parameter settings get "messy" (e.g., there are too many railroads or all three commodities are not visible), you can always return to the default, classical model with option **10** from the main menu.

* IMAGE TITLES: Many of the image titles for the screen displays are generated automatically and are not meaningful. If a title doesn't make sense, ignore it.

MAIN MENU

MENU OPTIONS

NOTE: For reasons of simplicity, menu options are explained in the context of von Thünen's classical circular rings--i.e. by changing the parameters associated with a menu option, how will the resulting ring pattern be affected? However, when adding railroads, lakes, and yield weights to the plain, the model will no longer produce regular, concentric rings. It remains as a learning exercise to infer the results of changing various parameters in more complex spatial situations.

--- ASPATIAL PARAMETERS ---

1. Enter new prices

This option allows you to change the market price for one or more goods. An increase in the market price of a good would make that good more profitable to produce and thus would enlarge the ring associated with that good. In terms of the bid-rent curve, the price increase raises the intercept of a good's graph while leaving the slope unchanged.

2. Enter new transportation cost factors

The VT/GIS program calculates a transportation cost surface based on distance and the assumption that overland transport is ten times more expensive than railroad transport. A cost surface for each good is then produced by multiplying the basic cost surface by the transportation cost factor for each good. For example, by assigning the weights 4, 2, and 1 to milk, grain, and livestock respectively, a situation is created in which milk is twice as expensive to transport as grain and four times as expensive to transport as livestock. Goods with higher transportation cost factors are less profitable to produce with increasing distance from the town. Thus, land rents become lower and the rings become smaller (as the slope of the bid-rent curve becomes steeper).

--- SPATIAL PARAMETERS ---

3. Enter town locations

This option allows you to place points on the screen which represent towns. This is done by clicking on the point locations using the mouse and the screen cursor. The procedure requires several steps. (After a few repetitions, this process will become intuitive.)

When you see an instruction screen entitled "VON THÜNEN DEMO", press **enter** to display the graphics screen on which you will enter features.

Viewing the positions of existing railroads and lakes

If you wish to see where the current railroads and lakes are (this is helpful in avoiding such things as putting a town inside of a lake), simply press the **v** key (v stands for "vector file"). In the lower-left corner of the screen, the program will then ask for a filename. Type **rails** and then press **enter** in response to the remaining prompts. You should now see the current railroad network on the screen. To view the lakes follow the same procedure (press **v** and enter **lakes**). **NOTE:** If you wish to view existing features, it must be done **before** entering any towns.

Entering towns with the mouse

You should now be looking at a blue screen entitled "PRESS "d" TO BEGIN DIGITIZING[1]..." (If you want to view the railroads and lakes follow the above procedure first.) In the following section, **bold** items are typed in by you and *italic* items are prompts from the program. "**<cr>**" refers to a carriage return (labeled "Enter" on some keyboards).

TO START: press **d**	(for "digitize")
Enter feature type: (P,L,A) **p**	(for "point")
Enter a start ID: **1**[2] **<cr>**	(a screen cursor appears)

Now move the cursor to the desired location and click the **left** mouse button to enter towns (up to ten). Click the **right** mouse button to quit entering towns. The computer responds with:

Keep these points? (Y/N) **y**	(or **n** to start over)
<cr>	(the blue screen disappears)
Enter a name for the vector file just created: **towns**	
Enter a title for the image just created: **<cr>**	(no title is needed)

Ignore the warning message about mismatching file types--it is intentional.

Now you are prompted to verify these features. They will appear coarse or blocky because they were transformed to a 100x100 grid. If they don't look right (aside from being coarse), run this menu option again.

---

[1] *Digitizing* refers to the process of converting non-numeric information into numeric information which is readable by a computer. In this case, you are converting screen locations (pointed to with the mouse) into numeric (X,Y) coordinates.

[2] This number is simply the ID number for the first town. Any additional towns will be stored internally as 2, 3, 4, etc..

--- SPATIAL PARAMETERS (CONT'D) ---

4. Create a railroad network

This option allows you to draw lines on the screen which represent railroad lines. This is done by clicking the locations of line segments using the mouse and the screen cursor. The procedure requires several steps:

When you see an instruction screen entitled "VON THÜNEN DEMO", press **enter** to display the graphics screen on which you will enter features.

Viewing the positions of existing towns and lakes

If you wish to see where the current towns and lakes are, press the **v** key (v stands for "vector file"). In the lower-left corner of the screen, the program will then ask for a filename. Type **towns** and then press **enter** in response to the remaining prompts. You should now see the current town locations on the screen. To view the lakes follow the same procedure (press **v** and enter **lakes**). **NOTE:** If you wish to view existing features, it must be done **before** entering any railroads.

Entering railroad lines with the mouse

You should now be looking at a blue screen entitled "PRESS "d" TO BEGIN DIGITIZING..." (If you want to view the towns and lakes follow the above procedure first.) In the following section, **bold** items are typed in by you and *italic* items are prompts from the program. "**<cr>**" refers to a carriage return (labeled "Enter" on some keyboards).

TO START: press **d**	(for "digitize")
Enter feature type: (P,L,A) **l**	(for "line")
Enter a feature ID: **1**[1] **<cr>**	(a screen cursor appears)

Now click the **left** mouse button to begin entering a railroad line. Move the mouse a short distance and click **left** each time you want to enter a line segment. After creating the desired rail line as a series of straight line segments, click the **right** mouse button. The computer responds with:

Keep this feature? (Y/N) **y**	(or **n** to ignore this feature)

Repeat the above procedure for each railroad line you wish to create, always using a feature ID of **1**. When finished, type:

<cr>	(the blue screen disappears)
Enter a name for the vector file just created: **rails**	
Enter a title for the image just created: **<cr>**	(no title is needed)

[1]The feature-ID of 1 causes railroad cells to be coded with a value of 1. Since VT/GIS codes non-R/R cells with a value of 10, the result is that railroad travel is ten times cheaper than overland travel.

Now you are prompted to verify these features. They will appear coarse or blocky because they were transformed to a 100x100 grid. If they don't look right (aside from being coarse), run this menu option again.

5. Create lakes

This option allows you to draw areas on the screen which represent lakes. This is done by clicking the locations of line segments along the perimeter of the lake using the mouse and the screen cursor. The procedure requires several steps.

When you see an instruction screen entitled "VON THÜNEN DEMO", press **enter** to display the graphics screen on which you will enter features.

Viewing the positions of existing towns and railroads

If you wish to see where the current towns and railroads are, press the **v** key (v stands for "vector file"). In the lower-left corner of the screen, the program will then ask for a filename. Type **towns** and then press **enter** in response to the remaining prompts. You should now see the current town locations on the screen. To view the railroads follow the same procedure (press **v** and enter **rails**). **NOTE:** If you wish to view existing features, it must be done **before** entering any lakes.

Entering lakes with the mouse

You should now be looking at a blue screen entitled "PRESS "d" TO BEGIN DIGITIZING..." (If you want to view the towns and railroads follow the above procedure first.) In the following section, **bold** items are typed in by you and *italic* items are prompts from the program. "**<cr>**" refers to a carriage return (labeled "Enter" on some keyboards).

TO START: press **d**	(for "digitize")
Enter feature type: (P,L,A) **a**	(for "area")
Enter a feature ID: **1000 <cr>**	(a screen cursor appears)

(NOTE: It is important that you enter **1000** here and not **1** as you did for towns and railroads.)[1]

Now click the **left** mouse button to begin entering a lake boundary. Move the mouse a short distance and click **left** each time you want to enter a line segment. After creating the desired lake as a series of straight line segments, click the **right** mouse button to "close up" the boundary. The computer responds with:

Keep this feature? (Y/N) **y**	(or **n** to ignore this feature)

Repeat the above procedure for each lake you wish to create, always using a feature-ID of **1000**. When finished, type:

<cr>	(the blue screen disappears)
Enter a name for the vector file just created: **lakes**	
Enter a title for the image just created: **<cr>**	(no title is needed)

[1]The feature-ID of 1000 tells the program to make lakes prohibitively expensive for the transport of goods. The effect of this is that lakes are barriers to travel.

Now you are prompted to verify these features. They will appear coarse or blocky because they were transformed to a 100x100 grid. If they don't look right (aside from being coarse), run this menu option again.

--- SPATIAL PARAMETERS (CONT'D) ---

6. Create yield weight surfaces

This option takes you to the YIELD menu, which is described in a later section. Options in the YIELD menu allow for the creation and display of yield weight surfaces for each of the three commodities.

--- DISPLAY / RUN ---

7. Compute a new model

This option takes you to the RUN menu, which is described in a later section. The RUN menu allows for a new model to be computed under a variety of options.

8. Display the results of the last computed model

This option displays two versions of a land use map as well as area statistics corresponding to the last model which was computed. The first map displayed is a conventional flat view, and the second is a three-dimensional perspective view of the first map draped over a land rent surface. On the second map, the elevation of each cell is proportional to the cell's land rent. Peaks and ridges represent areas of higher land rent (usually around towns and railroads), and low areas represent areas of lower land rents. Lakes and fallow regions appear flat since land rent is zero for those areas. (The 3-D display isn't perfect: sometimes lakes appear to "creep" up on bounding cliffs.)

Before displaying the second (3-D) map, the program displays the total areas for the various land use types and then allows you to abort the display by pressing **[Ctrl+c]** and then **y** to terminate the batch program. Remember to type **menu** to display the menu.

9. Review the current parameter settings

This option displays, one screen at a time, all of the current model parameter settings. The first screen shows the numerical parameters, and the next screens show the locations of the spatial parameters. If you don't want to view every screen, this option can be terminated by pressing **[Ctrl c]** and then **y** to terminate the batch program. Then type **menu** to display the menu options.

NOTE: This option shows the most recently entered parameters. These are not necessarily the parameters that were used in the last model computation.

10. Reset all parameters to the classical (basic) model

This option clears any parameter changes you may have made and returns to the basic Thünen model that existed when VT/GIS was installed. Specifically, the basic model includes no railroads, no lakes, no yield weights (yield weights are optimal [10] everywhere), and one town at the center of the plain. The numerical parameters for this model are as follows:

Parameter	Milk	Grain	Livestock
Price	600	525	200
Trans. Cost Factor	5.25	3.75	0.75

After using this option, option **8** can be used to display the classical model. There is no need to re-compute first.

--- ARCHIVE ---

11. Describe the ARCHIVE UTILITIES (to save, delete, or display images)

This option displays a screen which explains how to use the archive utilities. The archive utilites allow you to save the current model in the archive directory, and to display or delete images in the archive directory.

YIELD MENU

This menu covers the creation and display of yield weight surfaces, and is reached from main menu option 6. The addition of yield weights to the von Thünen model allows for land productivity to vary across the plain. This variation can be in response to one or a combination of factors including soil fertility, rainfall, temperature, etc.

When creating a yield weight surface, the user divides the plain into regions or polygons which are assigned values ranging from 0 to 10. VT/GIS then transforms these values so that they range from 0 to 1 (This operation is necessary as the IDRISI software only allows integer input for *FEATURE ID's*). The resulting values reflect *the proportion of the optimal yield value* (Y) a position on the plain is able to produce for a given commodity. Areas which are not specifically assigned a yield weight are assumed to produce the optimal yield value.

--- CREATE YIELD WEIGHTS ---

a. Create yield weights for MILK

This option allows you to draw areas on the screen and assign values to them representing the yield weights. This is done by clicking the locations of line segments along the perimeter of the area using the mouse and the screen cursor. The procedure requires several steps.

When you see an instruction screen entitled "VON THÜNEN DEMO", press **enter** to display the graphics screen on which you will enter features.

Viewing the positions of existing features

If you wish to see where the current features (towns, railroads, lakes, or yield weights) are, press the **v** key (v stands for "vector file"). In the lower-left corner of the screen, the program will then ask for a filename. Type the desired filename (**towns**, **rails**, **lakes**, **milk**, etc.) and then press **enter** in response to the remaining prompts (when viewing yield weights, you might want to enter **y** at the "*plot outlines only?*" prompt). You should now see the current feature locations on the screen. To view another feature follow the same procedure (press **v** and enter the filename).

Entering yield weights with the mouse

You should now be looking at a blue screen entitled "PRESS "d" TO BEGIN DIGITIZING..." (If you want to view other existing features, follow the above procedure first.) In the following section, **bold** items are typed in by you and *italic* items are prompts from the program. "**<cr>**" refers to a carriage return (labeled "Enter" on some keyboards).

TO START: press **d**	(for "digitize")
Enter feature type: (P,L,A) **a**	(for "area")
Enter a feature ID: **[desired number from 0-10] <cr>**	(a screen cursor appears)

Now click the **left** mouse button to begin entering an area boundary. Move the mouse a short distance and click **left** each time you want to enter a line segment. NOTE: you must create an area (or polygon), rather than simply "cutting off" a corner of the plain. You can, however, allow your area boundary to fall outside of the plain. You could, for example, enclose the entire plain in one large box and assign a single weight to the entire plain. After creating the desired area as a series of straight line segments, click the **right** mouse button to "close up" the boundary. The computer responds with:

Keep this feature? (Y/N) **y**	(or **n** to ignore this feature)

Repeat the above procedure for each area you wish to create. When finished, type:

<cr>	(the blue screen disappears)
Enter a name for the vector file just created: **milk**	
Enter a title for the image just created: **<cr>**	(no title is needed)

Now you are prompted to verify these features. They will appear coarse or blocky because they were transformed to a 100x100 grid. If they don't look right (aside from being coarse), run this menu option again.

b. Create yield weights for GRAIN

This option uses the same procedure as option **a.** above. You need only make the following change:

> *Enter a name for the vector file just created:* **grain**

c. Create yield weights for LIVESTOCK

This option uses the same procedure as option **a.** above. You need only make the following change:

> *Enter a name for the vector file just created:* **stock**

d. Apply the MILK yield weights to all commodities

This option causes the current yield weight surface for MILK to be applied to GRAIN and LIVESTOCK as well. This is useful if you are assuming that all commodities respond in the same way to a varying yield surface. NOTE: When this option is used, any existing yield weights for GRAIN and LIVESTOCK will be deleted.

--- DISPLAY YIELD WEIGHTS ---

e. View the yield weights image for MILK

f. View the yield weights image for GRAIN

g. View the yield weights image for LIVESTOCK

These options display a yield weights image on the screen. This can be helpful in assessing the current state of the model, as it can be difficult to keep track of the yield surfaces when they are different for each commodity.

The yield weight surfaces will be displayed without a legend. However, you can verify weight values by using the mouse and screen cursor. When you see the graphic display of the weight surface, type **c** to get a screen cursor. Using the mouse, move to the desired location and click **LEFT** to lock the cursor and display the weight value in the lower-left screen corner. Click **LEFT** again to unlock the cursor and query other locations. Click **RIGHT** to make the cursor go away.

NOTE: Yield weights will be presented as real numbers ranging from 0.0 (worst) to 1.0 (best).

RUN MENU

This menu provides four different ways to run the model. The first two options ignore any yield weights which may have been created, and the second two use the yield weights images created within the YIELD menu. It is often instructive to run the model first ignoring yield weights, and then to run it a second time using yield weights. The first run can be saved for future reference using the ARCHIVE UTILITIES.

The run options **o2** and **v2** are useful when you have established a "base map" of towns, railroads and lakes. You can then vary the prices, transportation costs, and yield weights and use **o2** and **v2** to re-compute the model much faster.

--- USING OPTIMAL YIELD WEIGHTS ---

o1. Compute a new model with the current settings

This option calculates the model using the current configuration of towns, railroads, and lakes and the current values for price and transportation cost factor. Yield weights are assumed optimal everywhere. This option takes several minutes of run-time.

o2. Compute a new model using only changes in price and/or cost factor.

This option allows for a faster calculation[1] (about half the time as **o1**) of the model assuming you have only changed the numerical parameters (price and/or transportation cost factor--main menu options 1 and 2). If you made any changes in the positions of towns, railroads, or lakes, this option will ignore them. Yield weights are assumed optimal everywhere.

--- USING VARIABLE YIELD WEIGHTS ---

v1. Compute a new model with the current settings

This option is identical to **o1** above, but it uses variable yield weights created by the user. If no variable yield weights were created, this option will produce the same results as **o1** above.

v2. Compute a new model using only changes in price, cost factor, or yield weights.

This option is identical to **o2** above, but it uses variable yield weights created by the user. If no variable yield weights were created, this option will produce the same results as **o2** above. The **v2** option may be used when yield weight surfaces have changed.

---

[1]This option is faster because, since only numerical parameters were changed, VT/GIS does not have to calculate a new cost surface.

ARCHIVE UTILITIES

These utilities allow you to save the current model display and to display, delete, and remove images corresponding to past runs of the model. The images are stored in the ARCHIVE directory of your VTGIS main directory.

SAVE: Save the current model. (SAVE "filename")

SAVE makes a copy of the last computed model and stores it in the ARCHIVE directory. To save a model, type **SAVE "filename"** where "filename" is a name of your choosing (8 characters maximum; do not use a filename extension[1]). NOTE: This utility saves the "flat" view only, not the 3-dimensional view.

SHOW: Display an archived model. (SHOW "filename")

This utility allows you to view any of your archived images. If you know the name of the image you want to view, type **SHOW "filename"** where "filename" is the name of the archived image (don't use a filename extension). If you don't know the filename, type **LISTALL** to list the archived images.

REMOVE: Delete an archived model. (REMOVE "filename")

Use this utility to free up disk space by deleting unwanted archived images. Type **REMOVE "filename"** where "filename" is the name of the archived image you want to delete (don't use a filename extension).

LISTALL: List the available archived images. (LISTALL)

This utility displays a list of images in the archive. The filenames are displayed with their ".IMG" filename extensions, although you never need to type these extensions. This utility is equivalent to typing "DIR ARCHIVE*.IMG" from the VTGIS main directory.

---

[1] A *filename extension* consists of one to three characters following the main body of a filename. The filename and its extension are separated by a period.

TROUBLESHOOTING INFORMATION

* THE COMPUTER LOCKS UP DURING INSTALLATION: Most likely, the target directory was not created before the installation attempt. Create the directory first using **MKDIR** and then install the program.

* *"OUT OF ENVIRONMENT SPACE :"* Your DOS system environment buffer is full. The START command tries to add ".\BIN" to your system search path. This will not happen with a full environment buffer. You need to free up some space by either dropping an item or two from your search path or removing any unnecessary system variables which are set in your AUTOEXEC.BAT file. After you free up some environment space, re-boot your machine and try the **START** command from your VTGIS directory.

* *"DATA PATH NOT FOUND :"* Most likely, the file <dir>\IDRISI.ENV does not contain the correct pathname ("<dir>" refers to the directory on which you installed VT/GIS). See the section on installation at the beginning of the manual to see how to modify this file.

* *"IMAGE FILE <filename> DOES NOT EXIST ON <path> "*: If <path> does not match the BIN subdirectory of your VTGIS main directory, modify your IDRISI.ENV file as described in the installation section of the manual. If <path> is correct, then <filename> may have been inadvertently deleted. Try running **DESTROY** and then re-installing VT/GIS from the diskette.

* *"IDRISI ENVIRONMENT FILE MISSING :"* Make sure you are in the correct directory when you type menu options. You should be in your VTGIS main directory, NOT in the BIN directory.

STUDENT EXERCISE 1

STUDENT EXERCISE 1

INTRODUCTION

This exercise introduces the operation of the VT/GIS package: setting numerical and spatial parameters, and running and displaying the model. The exercise assumes a knowledge of the fundamentals of von Thünen's theory of agricultural location. A brief introduction to this material can be found in the document entitled *Introducing von Thünen's Model and the VT/GIS Package.*

The model presented by VT/GIS is not meant to be a realistic predictor of agricultural land use. While it allows for a slightly more realistic spatial environment than von Thünen was able to provide in his work, VT/GIS is chock full of simplifications, generalizations, assumptions, and aggregations. The intention is not to perfectly model reality; rather it is to provide a conceptual framework within which students can investigate the von Thünen model and an associated set of spatial relationships.

Requirements

This exercise requires the *VT/GIS User's Manual* and a PC running the VT/GIS package.

Notational conventions

Menu options are presented in the form "menu name(option)". For example, **MAIN MENU(8)** refers to option 8 on the MAIN menu. To run this option, type **8** at the DOS prompt from the MAIN menu and then press **return** (or **Enter**).

Throughout the exercise text, items in **bold** are entered by the user. Responses and prompts from the computer are in *italics*.

PART ONE: NUMERICAL PARAMETERS

If the VT/GIS package has just been installed, go to the next paragraph. Otherwise, you need to reset the current model parameters to those of the classical Thünen model. Use **MAIN MENU(10)** (main menu option 10) to do this.

Viewing current numerical parameters

First use **MAIN MENU(9)** to review the existing parameter settings. The first screen shows the numerical parameters: price and transportation cost factor. Price refers to the price paid per unit commodity at the town, and transportation cost factor is a relative indication of the transportation cost for each commodity.

The next screen shows the locations of the current towns, railroads, and lakes. Since the classical model doesn't include railroads or lakes, you will just see a dot in the center of the screen which represents the town.

After viewing this screen, you might want to press **Ctrl+c** and then **y** to terminate the parameter review option (then type **menu** to see the main menu). Otherwise you will be shown several empty screens which pertain to parameters which you haven't yet entered.

Viewing the classical model

Use **MAIN MENU(8)** to display the classical von Thünen model. The first screen display shows the location of the single town, where commodities are sold, and the various agricultural land uses which occur on the plain surrounding the town. Notice how the areas of different land use take the form of rings (often called *Thünen rings*) or bands around the town.

Press **Enter** and you will see areal statistics for the various land use types occurring on the plain. Write these down for future comparison:

Land use area statistics

Milk:
Grain:
Livestock:

Press **Enter** again to see the second display which gives a 3-dimensional view of the land use "draped" over a land rent surface. The land rent surface has nothing to do with the topography of the region--it represents the land rent which is charged by a landlord at a given location on the plain[1]. The "elevation" anywhere on the plain is proportional to the land rent at that location.

1. What is located at the highest point on the land rent surface? Why?

Changing the price for livestock

Now we will change some numerical parameters and see how they affect the distribution of land use across the plain. If you want to

2. If the market price of livestock were to double, making it twice as profitable to raise livestock, how would you expect the resulting Thünen rings to change?

Now use **MAIN MENU(1)** to make this change. Set the price of livestock to 400, and leave the prices for milk and grain unchanged (at the prompts, type **600**, **525**, and **400**, respectively). To see the effect of the change in livestock price, we must now recompute the model. Use **MAIN MENU(7)** to go to the **RUN MENU** and view the available options. Since we have only changed price, a numerical parameter, we can get a faster computation of the model by using **o2** instead of **o1** (although either will work). (NOTE: The last two run options, **v1** and **v2**, will be used later when yield weights enter the picture).

---

[1] In reality, an infinite number of points exist in any given area. The VT/GIS package divides the Thünen plain into a 100x100 grid of *cells* or *pixels* which are used to represent continuous space.

Compute the new model with **RUN MENU(o2)**. It will take a couple of minutes to compute the model. When finished, the computer will display the final land use map. Press **Enter** when you want to see the next screen.

NOTE: If you want to save any of the models you compute in this exercise, use the ARCHIVE UTILITIES, which are described by pressing **MAIN MENU(11)**.

3a. How did the pattern of land use change? Did it match your expectations?

3b. **Land use area statistics**
Milk:
Grain:
Livestock:

4. Make a rough sketch of the bid-rent curve for the three commodities. With a dashed line, show how the position of the graph for livestock changed with the change in price.

Changing the transportation cost factor for grain

In addition to our assumption that the price of livestock has doubled, let's assume that the transport cost for grain has been cut by 50%. Use **MAIN MENU(2)** to change the transportation cost factor of grain to **1.88** (at the prompts, enter **5.25**, **1.88**, and **0.75** respectively). Now recompute the model using **MAIN MENU(7) & RUN MENU(o2)**.

5a. Describe any changes you see in the land use pattern. Was the total fallow area affected? What about the area devoted to milk production?

5b. **Land use area statistics**
Milk:
Grain:
Livestock:

Experiment on your own

If you wish, use **MAIN MENU(1)** and **MAIN MENU(2)** to experiment with different numerical parameters. Recompute the model with **MAIN MENU(7) & RUN MENU(o2)** (main menu option 7 and then run menu option o2). What would you have to do to reverse the order of the Thünen rings? (i.e., to make livestock occur closest to the town and milk occur farthest from town)

PART TWO: SPATIAL PARAMETERS

In the last section, we looked at the effects of changing prices and transportation cost factors on the agricultural landscape. This was done under the assumption of an *isotropic* plain--one in which transportation cost rates, soil fertility, climate, etc. are everywhere the same. Now we will relax some of von Thünen's original assumptions about space and look at how this affects the pattern of land use.

Adding railroads

In his original model, von Thünen considered only one mode of transportation: the horse cart. In this environment, transportation costs depend only on distance from the town. This causes the classic Thünen rings to be circular--at all locations which are a given distance from the town, the price of transporting a good to the town is the same.

6. What are some problems with this assumption?

The VT/GIS package allows for the addition of a railroad network to the plain and assumes that rail transport is ten times cheaper than overland (horse cart) transport. By introducing a railroad network, the situation is created where transportation costs depend not only on distance from the town, but on distance from the nearest railroad and the position of the railroad with respect to the town. Thus the Thünen rings will not necessarily be circular.

7. Sketch the resulting land use pattern assuming a railroad line were built which starts at the town and extends northward to the edge of the plain.

Now we will add the railroad. First, use **MAIN MENU(10)** to reset the numerical parameters to their original values. View the original model with **MAIN MENU(8)** to refresh your memory of what the base model looks like. Press **MAIN MENU(4)** to begin the process of entering railroads. The first screen provides some instructions. Press **Enter** to see the next screen.

Viewing the town's location

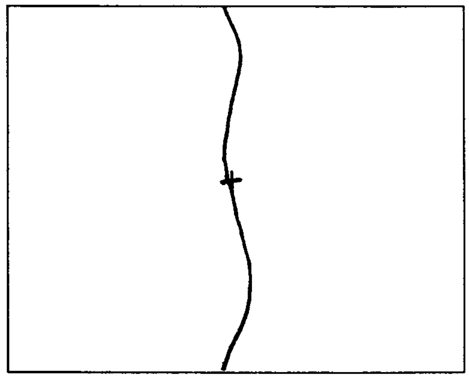
You should now see a blue screen entitled "PRESS "d" TO BEGIN DIGITIZING...". First, since we are going to add a railroad which goes northward from the town, we need to view the town's position. Press **v** and enter **towns** when you see the "*Enter file name:*" prompt at the lower-left screen corner. Now press **Enter** in response to the "*Enter color code:*" prompt[1]. A small green cross should appear at the center of the screen, marking the town's position.

---

[1]An integer color code (0-15) is not needed here since IDRISI displays vector point data as green crosses.

Using the mouse to enter line features

Now that the town is visible, we are going to digitize[3] a line feature on the screen which will represent the railroad. Refer to the MAIN MENU section of the VTIGIS *User's Manual* entitled **"4. Create a railroad network".** Following the procedure described in the manual, create a railroad which meanders from the town northward and southward to the edge of the plain (add more railroad lines if you wish). Your railroad should look something like this:



Now use **MAIN MENU(7) & RUN MENU(o1)** to recompute the model. The **o1** run option takes roughly twice as long as the **o2** option. **o2** can't be used in this instance because a spatial parameter was changed. When the new model has been computed, answer the following questions:

8. First write down the areal statistics:

   <u>Land use area statistics</u>
   Milk:
   Grain:
   Livestock:

   On the three-dimensional display, explain the presence of the land rent "ridge".

9. Notice the increase in cultivated area as compared to the original model. Imagine what would happen to the cultivated area if a series of ten railroad lines were built which each extended in a different direction from the town to the plain's edge. Assuming that the population of the town remains the same, would the resulting land use pattern be economically realistic in terms of total cultivated area? Why or why not?

[3] *Digitizing* refers to the process of converting non-numeric information into digital information which is readable by a computer. In this case, you are converting screen locations (pointed to with the mouse) into numeric (X,Y) coordinates.

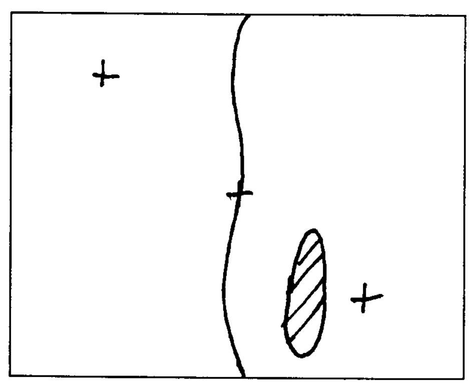
Adding multiple towns

Now we will add a few more towns to the landscape. Use **MAIN MENU(3)** to do this. Following the procedure outlined in the manual under **"3. Enter new town locations"**, first display the railroad[s] and then add three towns to the plain. Put one at the center of the plain (where the original town was); put another in the southeast corner of the plain, and put a third in the northwest comer about an inch or so from the railroad line (see the diagram below).

Adding lakes

Before recomputing the model, we'll add one more feature to the plain. Refer to the manual section **"5. Create new lakes"** to do the following: Using **MAIN MENU(5)**, first display the existing towns and railroads, and then add an elongated lake near the town in the southeast comer (see the diagram below). The lake will function as a barrier to transportation.

Your setup should now look something like this: (You can verify the locations of these spatial features with **MAIN MENU(9))**



Now recompute the model with **MAIN MENU(7) & RUN MENU(o1)**. This will take several minutes.

10. What effect does the lake have on the surrounding pattern of land use?

<u>Land use area statistics</u>
Milk:
Grain:
Livestock:

11. What would happen to the land use pattern if a bridge were built which connected the eastern and western shores of the lake? (OPTIONAL: Add the bridge—as a short railroad segment--and recompute the model.)

Adding variable yields to the model

Now that we have convoluted the original isotropic plain with additional spatial features, we will relax the classic Thünen assumption of constant yield across the landscape. This will be done by creating *yield weight surfaces*: each point on the plain will be assigned a weight value representing the productivity of the land at that point. The weight values will range from 0 (no yield) to 10 (optimal yield--ideal conditions). For this first use of yield surfaces, we will assume a very simple case in which every commodity responds to the same pattern of varying yield. Further, the yield surface will consist of only two regions.

Let's assume that the western half of the plain is indeed a flat, fertile plain, but the eastern half is rugged and mountainous. Not only is it difficult to grow crops on rugged terrain, steep slopes typically have thin, poor-quality soils as well. On the basis of these conditions, we will assume that yields are optimal in the West and 60% of optimum in the East.

Use **MAIN MENU(6)** & **YIELD MENU(a)** to create a yield weight surface for MILK. Following the procedure in the YIELD MENU section of the manual entitled "**a. Create yield weights for milk**", draw a box around the eastern half of the plain and assign it a feature ID of **6** (yield weights are entered in the range: 0=worst to 10=optimal). Remember that you can draw area boundaries "off the edge" of the plain--there is no need to carefully trace the plain boundaries. There is no need to do the western half of the plain, as it will automatically be assigned a value of 10.

When you are finished, the program will display the yield surface for verification. Using the legend on the screen, make sure that the weights are correct (west=100%, east=60%). If everything looks good, use **YIELD MENU(d)** to apply the MILK yield surface to GRAIN and LIVESTOCK. (If the yield surface doesn't look right, run option **a** again.)

Now press **r** to get back to the MAIN menu and compute a new model with **MAIN MENU(7)** & **RUN MENU(V2)**. (Remember that when using yield weights you must use **v1** or **v2**--whichever is appropriate. Here we can use **v2** because the town, railroad, and lake features have not been changed since the last model computation.)

12. Describe how the agricultural landscape has changed.

<u>Land use area statistics</u>
Milk:
Grain:
Livestock:

13. On the three dimensional display, are all of the land rent "peaks" at the same height? Why or why not?

14. Do you see any problems in the way that varying yields can be represented with VT/GIS? For example, how would you create a yield weight surface which, due to variations in temperature, changed gradually from 50% in the North to 100% in the South?

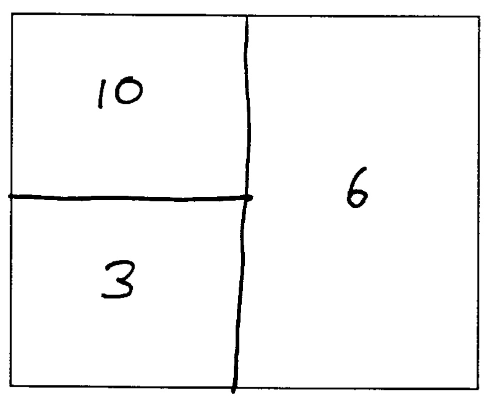
Using different yield surfaces for each commodity

We now come to the most convoluted form of the model allowed by the VT/GIS package: the case where each commodity responds differently to the landscape and thus requires a separate yield surface.

MILK

For MILK, we'll assume the same east/west variation in yield due to topography and soils, and in addition we will assume that the southwest quadrant of the plain is not near any sources of irrigation water. Since the MILK ring is characterized by intensive land use (e.g., irrigation, heavy fertilizer and labor inputs)[4], the southwest region will be assigned a non-optimal yield weight.

Use **MAIN MENU(6) & YIELD MENU(a)** to make the yield surface. Refer to the manual section under YIELD WEIGHTS entitled "a. **Create yield weights for milk".** Now use the mouse and screen cursor to draw (digitize) a box around the eastern half of the plain and give it a weight (feature ID) of **6**. Add another box around the southwest quadrant and assign a weight of **3** to it. The northwest quadrant will default to **10**. Your weight surface for MILK should now look like this:

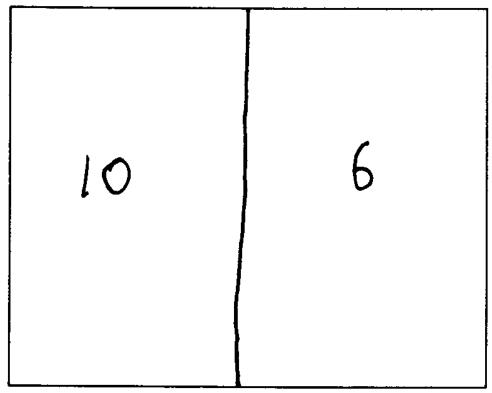


NOTE: If yield weight region boundaries overlap (for example the 5/6 boundary above), the last region entered will "cover" any regions beneath it.

[4] See the document *Introducing von Thünen's Model and the VT/GIS Package* for a description of how von Thünen's original rings were aggregated to the symbolic MILK, GRAIN, and LIVESTOCK categories used by VT/GIS.

GRAIN

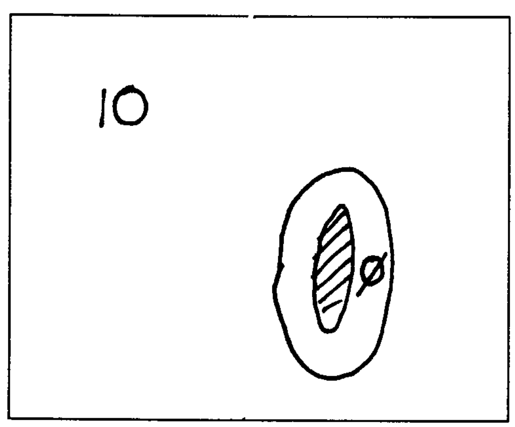
For GRAIN, let's assume that no irrigation is necessary. We will use the current yield surface which divides the plain into eastern and western regions. You shouldn't have to change this surface from the last setting. Use the **YIELD MENU(b)** option to verify that the GRAIN yield surface looks like this:



LIVESTOCK

For LIVESTOCK, let's assume that yield is optimal in the rugged eastern area (instead of the previous weight of 6), as grasses and grazing animals can typically thrive in hilly terrain. However, let's assume that a zoning law prohibits grazing activity near the lakeshore.

Using the manual and the **YIELD MENU(c)** option, first display the lake by pressing **v** and entering **lakes** as the filename. Then draw a yield area around the lake, approximately 1/2 inch from the lakeshore. Give this area a weight of **0** since grazing is not permitted here at all. Your YIELD weight surface should look like this:



If all of your yield weight surfaces look correct, press **r** to return to the MAIN menu. Run the model with the **MAIN MENU(7) & RUN MENU(v2)** options.

15. Interpret the results of this model. What effect does each yield surface have on the land use pattern?

PART THREE: FURTHER EXPLORATION

This exercise led you through some relatively simple spatial environments. Feel free to create new environments by changing the parameters any way you wish and recomputing the model. Here are some possible scenarios you might want to try:

* A regular grid of railroad lines.

* A town at the end of a long, thin peninsula on a lake.

* Approximate a continuous yield surface by making a series of long, thin yield weight areas ranging from 0 to 10.

* Create different classes of railroads or roads by assigning different impedance values (*feature ID* numbers) to different linear features (This is done during the creation of railroads when the computer prompts you to enter a feature ID). For example, you could assume that highways have an impedance of **1** and that minor roads have an impedance of **5**. VT/GIS allows linear feature ID numbers to range between 1 and 999. Remember that VT/GIS assigns an impedance of **10** to cells which are neither railroad nor lake.

STUDENT EXERCISE 2

STUDENT EXERCISE 2: Advanced activities

NOTE TO INSTRUCTORS

The following activities are further investigations into the Thünen model and GIS modeling in general. These advanced activities, except where noted, require access to the complete IDRISI package, an IDRISI manual, and a basic knowledge of the IDRISI system. The activities may be given to students as written here, or you may wish to employ them as exercises in GIS by assigning students the following tasks without giving them the detailed procedures below.

Software configuration

The following activities require access to IDRISI modules which are not included with VT/GIS. In order to run these modules from the VTGIS main directory, you need to modify your system's search path to include the directory in which your IDRISI modules are stored. This can be done by adding the following line to the end of your AUTOEXEC.BAT file:

SET PATH=%PATH%;C:\IDRISI

where "c:\idrisi" represents the drive and directory in which your IDRISI modules are stored.

In case of disk space problems or a corrupted VT/GIS package

When the following activities are completed, additional IDRISI image files are created in the VTGIS/BIN directory. If these files are not cleaned up, they can overrun your hard drive's storage space. In addition, if an existing filename is used, VT/GIS files can be overwritten, causing the VT/GIS package to no longer operate correctly.

If hard disk space is running out, or if the package doesn't seem to be working properly (or at all), it is recommended that a fresh copy of VT/GIS be installed. First run the **DESTROY** utility (described in the installation section of the user's manual) and then reinstall the package from the original compressed archive disk.

ACTIVITY 1: Approximating a continuous yield surface

The VT/GIS package allows the creation of areas of varying crop yield as homogeneous groups of cells. This situation often does not correspond well with reality, as many factors relating to crop yield, for example, slope and rainfall, tend to vary gradually across the landscape rather than occur in discrete areas with sharp breaks. (Of course, the division of the plain into a 100 by 100 cell grid is itself a discretization into homogeneous areas with sharp breaks, but at a finer resolution.)

To better approximate a gradually-varying, or *continuous* yield surface, we will create one in IDRISI based on slope and then insert it into the VT/GIS package for incorporation into the model. Two procedures for doing this are provided below, at two levels of focus. As an enhancement to the Thünen model, do the *detailed procedure* below. As an advanced GIS exercise, do the *GIS procedure* below.

NOTE: This activity gives the best output when there are several towns dispersed throughout the plain. Add towns with **MAIN MENU(3)** (and any other desired features) and then compute the model with **RUN MENU(o1)**.

Detailed procedure (does not require IDRISI)

The following should be executed from your VTGIS main directory, assumed here to be D:\VTGIS.

1. Two images are provided for this exercise: one is a digital elevation model, called DEM, and the second is a slope image of the DEM in degrees. The second image is called SLOPE.

   To get an idea of what these images contain, view them as both flat and 3-dimensional maps by using the commands **FLAT** and **3D**, followed by the appropriate filename (e.g., type "**FLAT DEM**" to see a flat representation of the DEM).

2. Now you will reclassify the slope image, which contains values from 0 to 90 degrees, into yield weights which vary from 1=best to 0=worst. Flat areas (where slope = 0) will be given an optimal weight value of 1.

   You must decide on the threshold slope value above which all weights will be zero. For example, if you think that slopes 30 degrees and above are completely unusable, then your threshold value would be 30 and slopes between 0 and 30 degrees would be linearly assigned a weight from 1 to 0 (e.g., a 15 degree slope would get a weight of 0.5).

   When you have decided on your threshold value, use the command **SLP2WT** to convert the degree slopes into yield weight values. This command requires two parameters: first, the name of the slope image, and second, the threshold slope value. For example, typing:

   slp2wt slope 45

   would reclassify the SLOPE image into a yield weight surface for MILK, using a threshold value of 45 degrees.

3. Now type "MENU" to see the VT/GIS menu. Use **YIELD MENU(d)** to apply this weight surface to all three commodities.

(OPTIONAL: Use a different weight surface for each commodity by specifying different slope threshold values. First, run **SLP2WT** with the desired threshold for LIVESTOCK. Then, to apply this surface to LIVESTOCK, type **COPY BIN\YLDWT1.* BIN\YLDWT3.***. Run **SLP2WT** again using a different threshold value for GRAIN; then apply the newly-created surface to GRAIN by typing **COPY BIN\YLDWT1.* BIN\YLDWT2.***. Now run **SLP2WT** a third time for MILK. This will overwrite the previously-created YLDWT1. Now proceed to the next step--DO NOT run option **d** from the YIELD MENU.)

4. The yield weights are now in place. Re-compute the model with **RUN MENU(v2)** (or use v1 if you modified the towns, railroads, or lakes).

5. After the model has been calculated, you will see the usual flat view of land use followed by the land use draped over the land rent surface. To see yet another view of the model, use the command **LNDSCAPE** to see the land use draped over the DEM (Type "LNDSCAPE DEM").

GIS procedure (for students knowledgeable in IDRISI)

The following should be executed from your VTGIS main directory, assumed here to be D:\VTGIS.

1. Obtain or create a 100 by 100 cell digital elevation model (DEM). You can create one in IDRISI by digitizing points on-screen from the **COLOR** module and then using **INTERPOL** to generate the DEM surface.

2. Use **SURFACE** to create a percent slope image.

3. Reclassify the slope image so that slopes have a value from 0 to 1, based on some criteria such as "flat areas have a weight of 1; slopes of 30 degrees and above have a weight of 0; slopes in between are scaled accordingly". (NOTE: this procedure will take several steps.)

4. Copy the resulting image of weights (real numbers from 0 to 1) into the VTGIS/BIN directory with the name "YLDWT1". This is the name for the yield weight surface for MILK. (An IDRISI image is made up of two files. You need to copy to the files YLDWT1.IMG and YLDWT1.DOC)

5. Use **YIELD MENU(d)** to apply this weight surface to all commodities (or create a separate weight surface for each commodity, if you wish, by overwriting, as in step 4, the files YLDWT2 [grain] and YLDWT3 [livestock], both with .IMG and .DOC extensions).

6. Re-compute the model using **RUN MENU(v1 or v2)**, whichever is appropriate (v2 is faster).

7. View the resulting land use maps, and then use **ORTHO** to drape the land use image DISPLAY2 over your DEM image, using the user-defined color palette called STUFF (The supplied command **LNDSCAPE** will do the ortho/drape for you).

Questions

* Describe the resulting land use pattern. Do the land uses appear as contiguous belts? Why?

* Describe how you could approximate a continuous yield surface based on precipitation. How could this surface be combined with the surface you created above in order to obtain a composite yield surface which accounts for both slope and precipitation. (OPTIONAL: Do it.)

* How sensitive is your model of predicted land use to small changes in threshold slope value? How did you assess this sensitivity?

* Earlier we assumed that a continuous approximation of a surface was a better alternative to a surface with abrupt boundaries. This is not always the case. Can you think of some instances where distinct breaks occur in natural phenomena?

ACTIVITY 2: Delineating trade area boundaries for each town (requires IDRISI)

When computing the Thünen model using multiple towns, the resulting pattern of land use often makes it difficult to discern *trade areas*. These are areas which trade with a given town, based on distance and transportation cost. A trade area for a given town consists of all areas from which transport of goods is cheaper than for any other town.

In this section you will use the IDRISI **ALLOCATE** module to determine which areas on the plain are serving which towns. Then you will extract the boundary information and add it to the land use map produced by VT/GIS.

Procedure

The following should be executed from your VTGIS main directory, assumed here to be D:\VTGIS.

1. In order to have trade areas, there needs to be more than one town on the VT/GIS plain. If your current land use model only contains one town, use **MAIN MENU(3)** to add multiple towns to the plain (add at least five in order to get an interesting result). Now re-compute the model with either **o1** or **v1** from the **RUN MENU**.

2. Before running IDRISI's **ALLOCATE** module, you need a target image which contains a unique integer for each town location. The image MARKETB was created within VT/GIS (by the **MAIN MENU(3)** option), and contains a one at each town location and zeros elsewhere.

   To assign each town a unique value, run **GROUP** on MARKETB to create a new image called TOWNS. Display TOWNS with **COLOR** and verify that each town has a unique value and that the background is composed of zeros.

   You need one more image to use with **ALLOCATE**. This image is a cost surface relative to the towns, and was also created internally in VT/GIS. Display the image COST with **COLOR A** and again with **ORTHO**, so that you have a feel for what it represents (or use the supplied macros **FLAT** and **3D**).

3. Now run **ALLOCATE**, using COST as the *distance image* and TOWNS as the *target image*. Name the output allocation image AREAS. Display AREAS with **COLOR**.

4. You now have an image (AREAS) which contains one field of identical cell values (one "polygon") for each town. These polygons represent the trade areas for each associated town. The next step is to extract only the boundary information from this image so that it can be overlayed with the land use map.

   Display AREAS with **ORTHO** (or **3D**). Note the slopes which occur within the polygons and also at their boundaries. We will make use of this situation to extract only the boundary locations from the AREAS image. Run **SURFACE** on AREAS to get a *slope* image in *degrees* called SLPBOUND. Display SLPBOUND with **COLOR A**.

5. Now run **RECLASS** on SLPBOUND to make a new image called BOUNDS. You need to reclassify all values greater than zero to a new value of **7** (use a small number such as 0.00001 for the lower range).

6. Run **OVERLAY** and *cover* BOUNDS over DISPLAY2 (the land use map from VT/GIS). Call the new image FINALMAP. Now insert FINALMAP back into VT/GIS for display by typing:

COPY BIN\FINALMAP.IMG BIN\DISPLAY2.IMG

NOTE: Don't copy over the DISPLAY2.DOC file or you will lose the legend and title of the land use map.

Questions

* Explain what happened when you used **SURFACE** in step 4 above. How did this procedure extract the boundary information? What would have happened if you used an aspect image instead of a slope image?

* How accurate do you think the trade area boundaries are? Can you think of any factors that might cause the actual trade regions to differ from the trade areas you found above?

Further activities (OPTIONAL)

* The trade area boundaries you created above are rather coarse--several cells wide in places. How might you thin the boundaries so that they are only one or two cells wide? (You might experiment with the **FILTER** command.)

* Several methods can be used to extract boundary information from the AREAS image (trade area "polygons"). Instead of using **SURFACE** to make a slope image, use **POLYVEC** and **LINERAS** to create an image of trade area boundaries. (You should run **INITIAL** first to create a 100 by 100 image of zeros to be updated by **LINERAS**.) Then proceed from step 5 above.

Files to be cleaned up (all are in the BIN directory)

The following files were created during this exercise. You may want to delete them.

TOWNS.* AREAS.* SLPBOUND.* BOUNDS.* FINALMAP.*

ACTIVITY 3: Non-linear transportation costs

VT/GIS assumes that all transportation costs are linear with respect to distance. That is, the cost per unit distance for shipping a commodity will be the same regardless of the length of the trip. This is a fair assumption for some modes of transportation, however several examples exist where transportation costs are distinctly non-linear. (Based on Figure A)

Take airline travel as an example: a round-trip ticket from Los Angeles to San Diego might cost about $200 for the 200 mile trip, or $1.00 per mile. A ticket from

LA to Denver would cost around $300 for a 2000 mile flight, or about $0.15 per mile. Finally, a 6000 mile round-trip flight from LA to New York would cost about $500, or approximately $0.08 per mile.

So we see that the transportation cost rate ($ per mile) is not constant with distance. Rather, per-mile costs tend to decrease with increasing distance (although total cost--the price of the ticket--increases with increasing distance). If you were to graph the per-mile airfares against distance, the resulting plot would take the shape of a curve which slopes gently downward to the left:

In this exercise we will introduce the type of non-linearity discussed above to the transportation cost surface of the VT/GIS model. Note that the objective of this activity is to introduce the modeling of non-linearity and not necessarily to create a more realistic environment for the Thünen model.

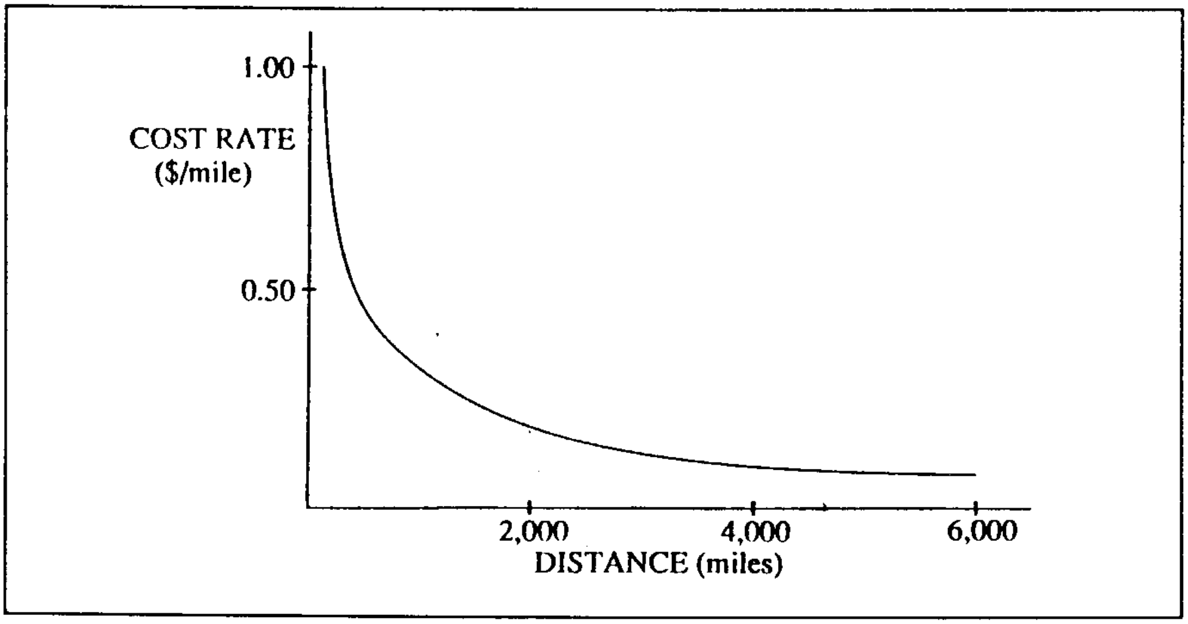
Procedure

The following should be executed from your VTGIS main directory, assumed here to be D:\VTGIS.

1. In VT/GIS, calculate a model to be used with this exercise. Include any desired spatial features. When this has been done, write down the land use area totals below for later comparison (these statistics are given by the **MAIN MENU(8)** option).

Figure A

Air travel cost rate versus distance



Land use area statistics
Milk:
Grain:
Livestock:
Fallow:

2. To introduce the non linearity, we will be altering the cost surface of the model you just computed. Before doing this, make a backup copy of this cost surface. You will need it later. To make the backup, type:

   COPY BIN\COST.IMG BIN\ORIGCOST.IMG
   COPY BIN\COST.DOC BIN\ORIGCOST.DOC

   NOTE: For the above operation, don't use "COST.*" as this will rename an important file called "COST.EXE".

3. Now use **SCALAR** to *exponentiate* the image called COST. The value of the exponent chosen will determine the shape of the resulting non-linear cost surface. For the first run, use an exponent of **0.75**. Call the new image NONLIN.

4. The exponentiation to a power less than one caused all values in the original cost surface to decrease. The next step is to calibrate the two surfaces to a common value. The non-linear surface will now be rescaled so that its maximum value is equal to the maximum value of the original cost surface.

   To do this, you need to know the maximum value occurring in each image. Use **DESCRIBE** on both COST and NONLIN to get these values (remember that they will be in scientific notation). You don't need a lot of precision--up to the first two decimal places is fine.

   Old maximum value (from COST):
   New maximum value (from NONLIN):

   Note that the new maximum should be equal to the old maximum taken to the power of the exponent you used in step 3.

5. **RESCALE** transforms the new surface by multiplying it by the ratio of the old to the new maxima. The **RESCALE** command requires the two maxima as arguments:

   RESCALE <oldmax> <newmax>

6. The **RESCALE** operation has now altered your nonlinear cost surface to have the appropriate maximum cell value, and copied the resulting image to the file COST.IMG (this is the file that VT/GIS uses for the transportation cost surface). Now re-compute the model using **RUN MENU(o2)** (or v2 if appropriate).

   NOTE: DO NOT use option **v1** or **o1** here, as these options will create a new cost surface and overwrite the existing one.

7. Note the change in the land use pattern. Write down the areal statistics:

Land use area statistics
Milk:
Grain:
Livestock:
Fallow:

Questions

* How did the non-linearity affect the final pattern of land use?

* Is this effect apparent on the three-dimensional land use/land rent map? What does it look like?

* Perform this operation (steps 3-7) again, using different exponent values (use values less than one) in step 3. (Remember to exponentiate the ORIGCOST image, as the current COST image has been altered.) How do different exponent values affect the land use pattern?

* Conceptually, what would happen to transportation costs if the cost surface were exponentiated to a power *greater* than one?

Files to be cleaned up (all are in the BIN directory)

The following files were created during this exercise. You may want to delete them.

ORIGCOST.* NONLIN.*

APPENDIX A: TECHNICAL DOCUMENTATION

This section contains two parts. The first is a set of instructions on how to modify images within VT/GIS in order to expand on the complexity of the model. The second part consists of annotated flowcharts which explain exactly how VT/GIS calculates the Thünen model.

General instructions for enhancing the use of the VT/GIS package (for IDRISI users)

As in the previous advanced activities, the VT/GIS package can be enhanced and refined by processing VT/GIS images using any desired IDRISI functions. These modified images can then be copied back to their appropriate VT/GIS names, and the model can be re-calculated. The following is a description of some key VT/GIS images that you may wish to modify.

All IDRISI image files are stored in the **BIN** directory.

* The friction image

VT/GIS makes an image called FRICTION which represents the cost impedances associated with travel through each cell. These impedances reflect the feature ID values of the railroad and lake features (cells which are neither railroad nor lake are assigned an impedance of 10). You can create a friction surface based on other factors and then calculate a new model by executing the batch file **DOCOST** (just type "docost") and then running **RUN MENU(o2 or v2)**.

NOTE: In making the final land use map, the VT/GIS run options require feature ID's for railroads to range from 1 to 999, and for lakes to range from 1000 to 30,000. If these constraints are not met, then the color assignments of the final land use map will be wrong. The 1 to 999 range for railroad ID's means that you can create different classes of linear features. For example, you might give railroads and main highways an impedance of 1 and minor roads an impedance of 5. You might also assign a linear feature an impedance of 200, and assume it is a tariff associated with a political boundary.

* The cost surface

VT/GIS runs the IDRISI **COST** module with MARKETB as the *source image* and FRICTION as the *friction image*. The output image is called COST. If you modify COST you need to calculate your new model with **RUN MENU(o2 or v2)**, as o1 and v1 will overwrite your cost image.

* Yield surfaces

To alter the yield surfaces, change the images YLDWT1, YLDWT2, and YLDWT3, which represent MILK, GRAIN, and LIVESTOCK respectively. These images should contain values ranging from 0.0 (worst) to 1.0 (best). Use **RUN MENU(v1 or v2)** to calculate the model.

* The final land use map

The final land use map generated by VT/GIS is called DISPLAY2. You may want to run additional analytical procedures on it (e.g., overlay it with a previous run of the model in order to discern areas of change).

* **The final land rent surface**

  This image is called MAXPROF2. You may want to run additional analytical procedures on it (e.g., calculate cumulative land rent totals by commodity type).

FLOWCHARTS OF VT/GIS FUNCTIONS

This section documents how VT/GIS works. VT/GIS calculates the Thünen model by executing a series of functions, each of which is a DOS batch file. The batch files execute DOS commands, IDRISI modules, and C-programs written specifically for VT/GIS. The flowcharts in this section outline the procedures performed by each batch file. These batch files are ASCII text and may be viewed directly in the DOS environment.

Flowchart notation

* An *ellipse* indicates the batch file which created the image pointed to with a *dashed line*.

* A *box* indicates an IDRISI image.

* *Arrows* connecting boxes indicate actions performed on one or more images.

* *Text above* these arrows shows the IDRISI or DOS command which is executed, and *text below* the arrows shows the IDRISI or DOS command parameters.

* For the RECLASS module parameters, values to the *left* of the equal sign represent new cell values. Values to the *right* of the equal sign (in parentheses) represent the range of old values which are reclassified. For example, the expression "5 = (1 to 1,000)" assigns a new value of five to all old values ranging from one to just less than one thousand.

* A *dotted line* indicates a continuation of the flowchart to another location on the page.

FRICT.BAT (Figure 1)

This function produces a cost impedance surface, which is used in computing the transportation cost surface. It combines raster images of the railroads and lakes and then reclassifies all zero cells (cells which are neither lake nor railroad) to a value of ten. Thus, on the final impedance surface, overland cells have a cost of ten, railroad cells have a cost of one, and lake cells (barriers) have a cost of one thousand (or you may have used other impedance values for lakes and railroads).

DOCOST.BAT (Figure 2)

This function produces the Thünen model's transportation cost surface. The image MARKETB contains the town locations and the image FRICTION contains the cost impedances associated with travel through each cell. Each cell of the final cost image (COST) contains the total cost associated with travel to the nearest town (nearest in this sense refers to the least cost distance, not the least euclidean distance).

MAK_PROF.BAT (Figure 3)

This function inputs a single cost surface and outputs an image containing the maximum possible land rent at each cell. The transportation cost factors (noted as <*commodity* ***costfac***>) are stored in the file VTGIS\COSTFILE. The values indicated by <*commodity* ***price***> represent revenues, since yield is assumed to be one unit of commodity per unit land area, and revenue equals price times yield. These values are stored in the file VTGIS\REVFILE. The files REVFILE and COSTFILE are maintained by programs called from the functions 1.BAT and 2.BAT, respectively.

NOTE: The images PROF1, PROF2, and PROF3 contain *negative* land rent values (cost minus revenue). This is done to achieve faster processing by reducing the number of steps taken by the function. The final step multiplies by negative one so that the output image contains positive land rent values.

YMAK_PRO.BAT (Figure 4)

This function is identical to MAK_PROF.BAT above, except it uses yield weight images to modify the commodity revenues (prices). The resulting images, YIELD1, YIELD2, and YIELD3 contain the modified revenue values.

ID_PROF.BAT (Figure 5)

This function takes the maximum land rent surface (MAXPROF) and identifies which commodity belongs to each of its grid cells (i.e., it identifies the most profitable commodity at each cell). This is done by subtracting an individual land rent image and reclassifying the resulting zero cells to an arbitrarily high value (10,000; 20,000; or 30,000). This procedure is done three times and the resulting images are overlayed and reclassified to produce a land use image, called FINAL, where the values 0, 1, 2, and 3 represent the land uses fallow, milk, grain, and livestock, respectively.

NOTE: The images PROF1, PROF2, and PROF3 contain negative land rent values (see the note for MAK_PROF.BAT above).

MDRAPE.BAT (Figure 6)

This function creates the images used in the final screen display of the model. First, the image MAXPROF2 is created to serve as the three-dimensional display surface. The rest of the function combines the towns, lakes, and railroads with the classified land use image (FINAL). The output of this procedure is the final "flat" model display, with a legend and title. The last operation of this function is a DOS copy command which associates the .IMG file of the final display image with the name of a standard .DOC file containing a legend and title.

DISP.BAT

This function is not documented by a flowchart. It first uses **COLOR** to show the "flat" land use image DISPLAY2. Then it uses **ORTHO** to show a perspective view of the land rent surface MAXPROF2 with the DISPLAY2 image draped over it.

HOW VT/GIS USES THESE BATCH FILES

Within VT/GIS, the Thünen model is calculated from the RUN MENU options **o1**, **o2**, **v1**, **or v2**. This section shows how the batch file functions described above are called by the VT/GIS menu options:

Using ***optimal*** *yield weights*

o1.bat	**o2.bat**
call frict.bat	call mak_prof.bat
call docost.bat	call id_prof.bat
call mak_prof.bat	call mdrape.bat
call id_prof.bat	call disp.bat
call mdrape.bat	
call disp.bat	

Using ***variable*** *yield weights*

v1.bat	**v2.bat**
call frict.bat	call ymak_pro.bat
call docost.bat	call id_prof.bat
call ymak_pro.bat	call mdrape.bat
call id_prof.bat	call disp.bat
call mdrape.bat	
call disp.bat	

FLOWCHARTS

Figure 1

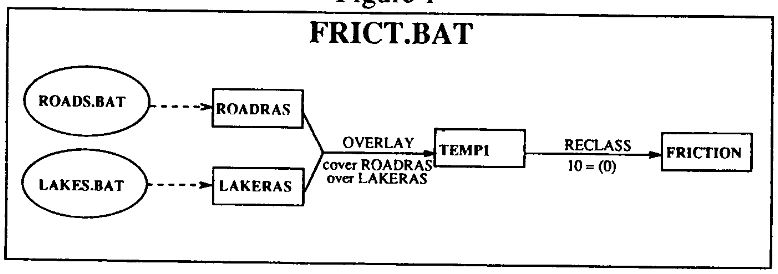


Figure 2

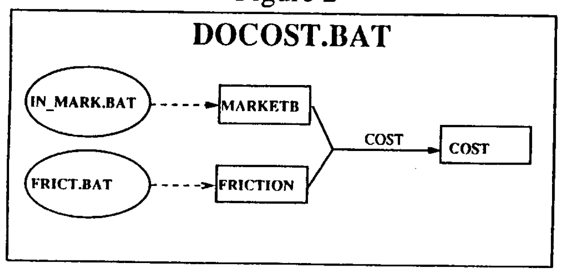


Figure 3

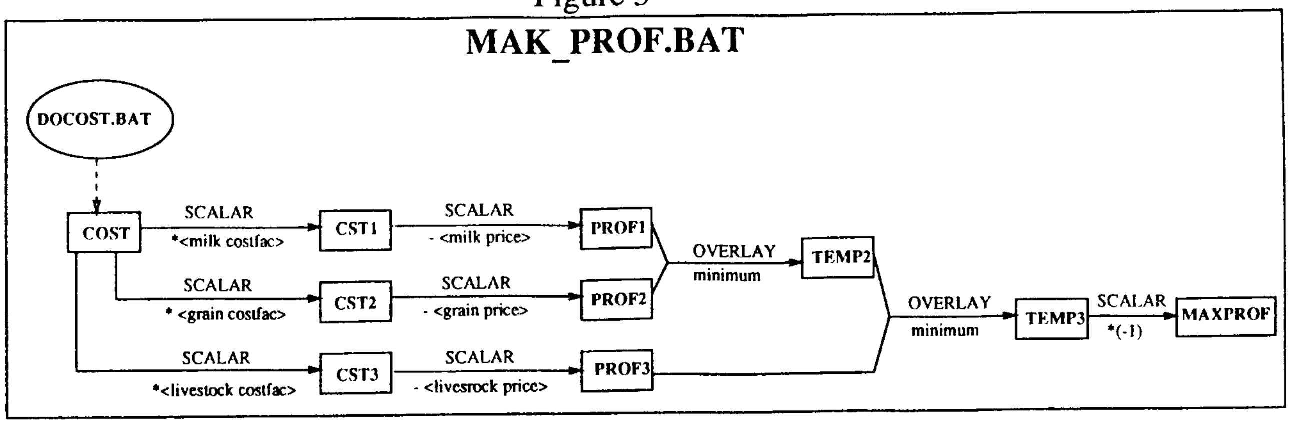


Figure 4

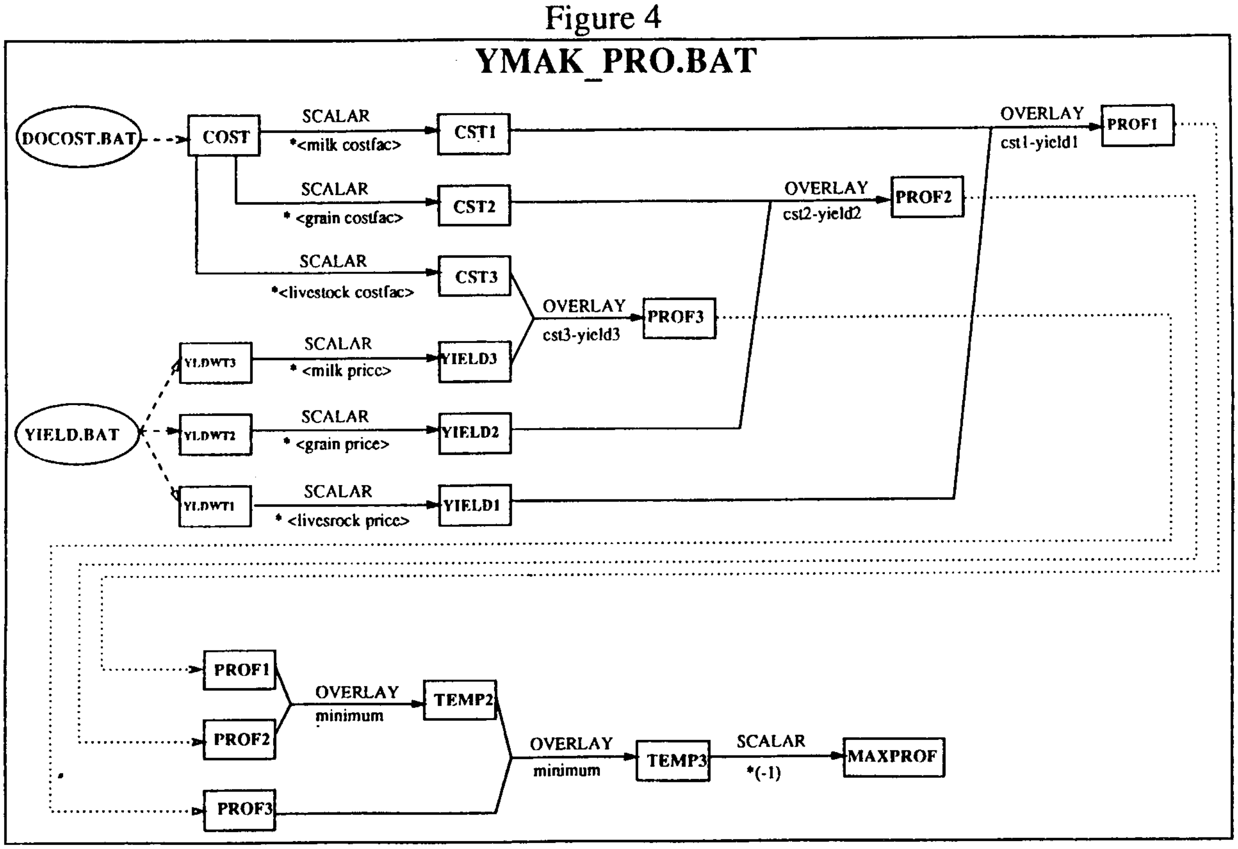


Figure 5

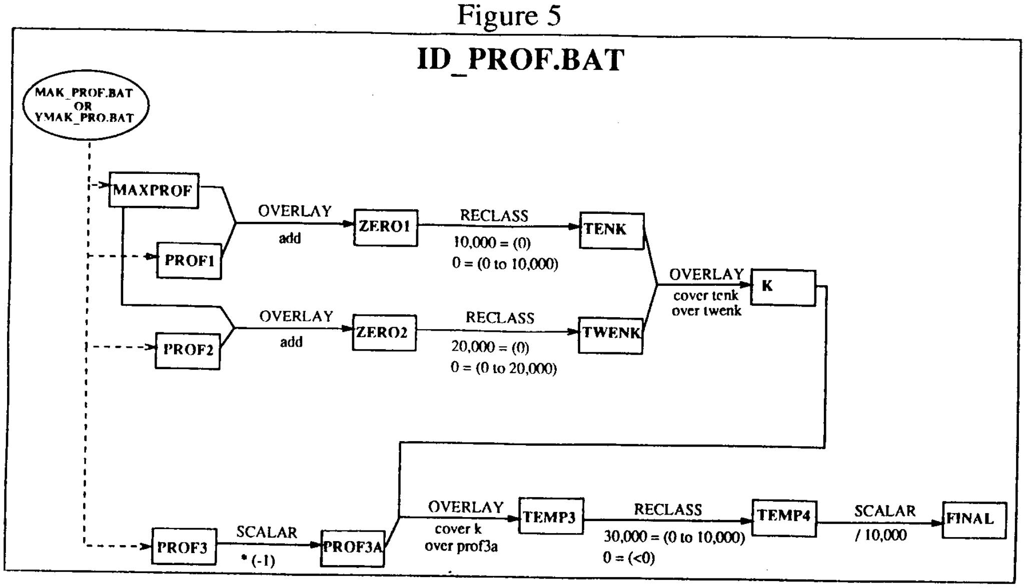
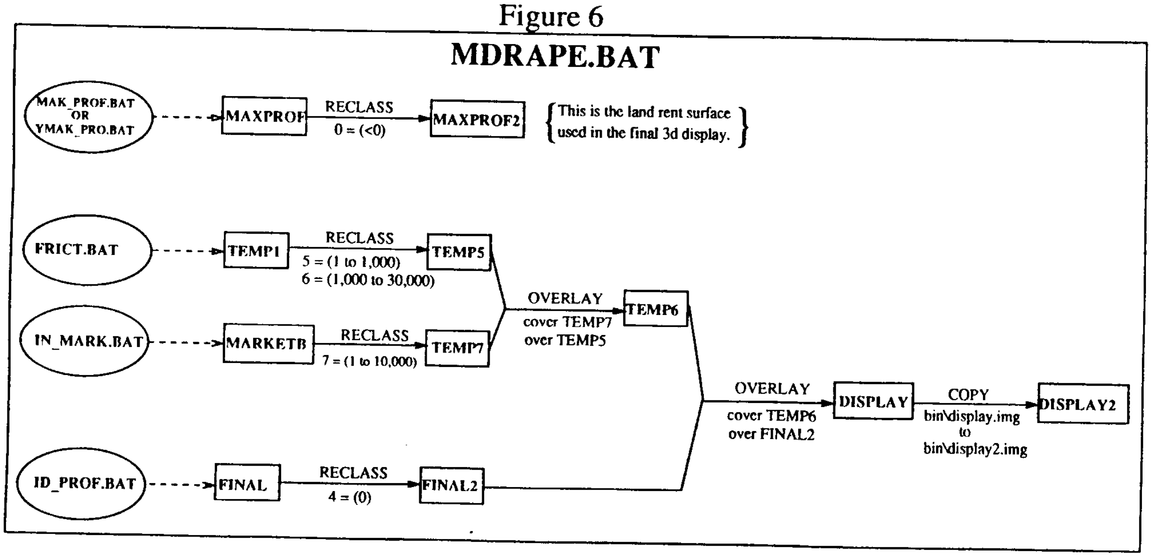


Figure 6



APPENDIX B: INDEX TO DISK CONTENTS

The diskette included with VT/GIS contains several files:

* **VTGIS.EXE** is a self-extracting compressed archive file which was created with PKZIP [1] file compression software. This archive file contains the files and programs that make up the VT/GIS package.

* **INSTALL.BAT** is a batch file which handles the uncompressing of the VTGIS.EXE file.

The next files are in a subdirectory called **TEXT**:

* **README** contains general information on installing the VT/GIS package. This information is taken directly from the VT/GIS manual--it is provided here as a quick reference. You can view it by typing "MORE < README".

* **VTGIS.DOC** is a file in Microsoft Word 5.0 format which contains all the VT/GIS documentation (not including the graphics).

* **VTGIS.ASC** is a file in generic ASCII format which contains all the VT/GIS documentation (not including the graphics).

---

[1] PKZIP is a registered trademark of PKWARE, Inc., 7545 N. Port Washington Rd., Suite 205, Glendale, WI 53217.