

Design of GIS Based Forest Road Layout and Environmental Assessment Tool

by

Xiangfei Meng

Bachelor of Economics, Jilin University, 2001

A report submitted in partial fulfillment of the requirements for the degree of
Master of Forest Engineering (MFE)
in the Faculty of Forestry and Environmental Management

This report is accepted.

.....

Dean of Graduate Studies and Research

THE UNIVERSITY OF NEW BRUNSWICK

October, 2006

© Xiangfei Meng, 2006

ABSTRACT

Poor soil drainage can cause major damage to roads, especially unpaved forest roads. In turn, forest roads can change water flow patterns, which may lead to local soil flooding, ditch erosion, siltation in down-slope water bodies, and the degradation of aquatic habitat in lake and stream bottoms. In this study, I developed an ArcView 3.3 GIS forest road layout tool called FOROAD, and this tool adjusts automatically to specifications about local climate normals, terrain, and depth-to-water below the soil surface. The tool consists of three modules. **ForVisual** is a visualization tool and produces maps of wet areas, culvert distribution and slope. This tool provides users with a general interface of FOROAD for data loading and parameter modification. **ForCulverts** is used to determine the location and the size of cross road drainage culverts. This tool can be used to determine the culvert location of new roads or to assess the suitability of culvert size and location of existing roads. **ForRoutes** assists to identify road layouts across environmentally sensitive areas such as steep slope areas and wet areas. This is done with an automated optimization routine to seek alternative routes according to total road length, construction cost, number of bridges and culverts needed, length of wet and environmentally sensitive areas to be crossed, slopes and slants, and overall earth-moving efforts.

TABLE OF CONTENT

ABSTRACT	II
ACKNOWLEDGEMENT	ERROR! BOOKMARK NOT DEFINED.
TABLE OF CONTENT	III
LIST OF TABLES	V
LIST OF FIGURES	6
CHAPTER 1	
INTRODUCTION	8
REPORT OBJECTIVES.....	10
LITERATURE CITED.....	12
CHAPTER 2	
GETTING STARTED	14
CHAPTER 3	
DETERMINING CULVERT LOCATION AND SIZE.....	21
STUDY AREA	23
Peak Flow	23
Drainage Area	24
Runoff coefficient “C”	24
Extreme Rainfall Events (“I”) Distribution Map.....	25
Sizing Culverts	29
Locating Culverts.....	29
Test Data.....	30
DISCUSSION.....	34
LITERATURE CITED.....	35
CHAPTER 4	
FORROUTES: LAYING OUT FOREST ROAD.....	ERROR! BOOKMARK NOT DEFINED.
STUDY AREA	ERROR! BOOKMARK NOT DEFINED.
METHODS.....	ERROR! BOOKMARK NOT DEFINED.
The Drawing Tool.....	40
The Assessment Tool.....	41
The Optimum Route Tool.....	Error! Bookmark not defined.
Curve Smoothing	Error! Bookmark not defined.
Calculations.....	Error! Bookmark not defined.
SUMMARY	63
LITERATURE CITED.....	ERROR! BOOKMARK NOT DEFINED.
CHAPTER 5	
FOROAD AND OTHER EXISTING ROAD DESIGN TOOLS.....	66
INTRODUCTION.....	66
LITERATURE CITED	82

CHAPTER 6

SUMMARY AND DISCUSSIONS	84
SUMMARY OF THE SOFTWARE	84
CONTINUATION OF THE SOFTWARE	86
LITERATURE CITED	87
APPENDIX A- USER TUTORIAL	88
<i>User Manual</i>	88
APPENDIX B- RAINFALL DATA FOR ATLANTIC CANADA.....	93
APPENDIX C- METHODOLOGY OF MAPPING WET AREAS.....	98
LITERATURE CITED.....	103
APPENDIX D- FOROAD ARCVIEW AVNUE™ CODE	106

LIST OF TABLES

Table 3.1 Determining runoff coefficients based on slope percent	28
Table 3.2 Kirpich adjustment factor (k)	31
Table 4.1 A comparison of model suggested roadway alternatives	60
Table 5.1 Existing Software Packages for Road Design	69
Table 5.2 A summarization of existing transportation design tool	70

LIST OF FIGURES

Figure 1.1 Road damage arising from roadway surface drainage problems	10
Figure 1.2 Roadway and environmental damage arising from subsurface flow drainage problems	11
Figure 1.4 Report out line and relationships between chapters	14
Figure 2.1 The GUI of FOROAD. Users provide data and modify parameters in this window.	20
Figure 2.2 Loading DEM of New Brunswick, Canada	21
Figure 2.3 Loading water features, which include lakes, streams and rives in New Brunswick, Canada.....	21
Figure 2.4 Loading the Depth-to-Surface water map of New Brunswick Figure	22
Figure 2.5 Loading Atlantic Canada weather map	22
Figure 2.6 Loading existing road network for New Brunswick area. Dropdown menus in the top red oval control show three module controls of ForVisual, ForCulverts and ForRoutes	23
Figure 3.1 Processing flowchart for sizing culvert	25
Figure 3.2 Digital map of Rainfall Distribution for Atlantic Canada (30 minutes event with a return period of 25 years)	30
Figure 3.3 Isometric precipitation map, adapted from Rainfall Frequency Atlas for Canada.....	30
Figure 3.4 Topographic map, stream channels and road network map and existing culverts locations on survey roads of CFB Gaagetown in central New Brunswick	34
Figure 3.5 Model predict and existing culverts locations illustrated with ArcView GIS map	36
Figure 3.6 Model predict and actual culverts	37
Figure 3.7 Culvert size and locations are listed in an attribute table	37

Figure 4.1 Forest road network in a portion of northern New Brunswick, Canada	43
Figure 4.2 Principal stages involved in deriving mapped and unmapped flow channels from local DEMs	46
Figure 4.3 A composite illustrating wet-areas mapping applications	47
Figure 4.4 Profile for existing road segment.	49
Figure 4.5 An illustration of resistance map.	50
Figure 4.6 Model structure of the optimum path design	52
Figure 4.7 The relationship between resistance output and depth-to-surface water	53
Figure 4.8 The locations and sizes of potential culverts	56
Figure 4.9 Selected preliminary routes based on resistance values produced by depth-to-Surface water map and culverts map	58
Figure 4.10 A filling up cross section in wet areas	58
Figure 4.11 Calculating the earth moving effort by smoothing road profile	58
Figure 4.12 The cut and fill area for cross sections on the side slope	58
Figure 4.13 Final results: optimum path are selected based on final resistance values produced by depth- to- Surface water, culverts, slope and road length map	61
Figure 4.14 More comparisons of gradients and slants for selected routes	62
Figure 4.15 Comparison between smoothed and original curves	63
Figure 4.16 Road information from the smooth line	64
Figure 4.17 The locations of culverts on selected route	65
Figure 4.18 Culverts and drainage related information for selected route	65

CHAPTER 1

INTRODUCTION

Forest roads are usually designed for low traffic volumes on unpaved surfaces. These roads and their surfaces need to function properly to avoid water logging and ponding, because subsequent water run-off will lead to surface erosion and serious road damage such as wash-outs and gulleys. High moisture content in surface materials and road foundations also reduce bearing capacity, leading wheel ruts, puddling and frost heaving (Figure 1.1).

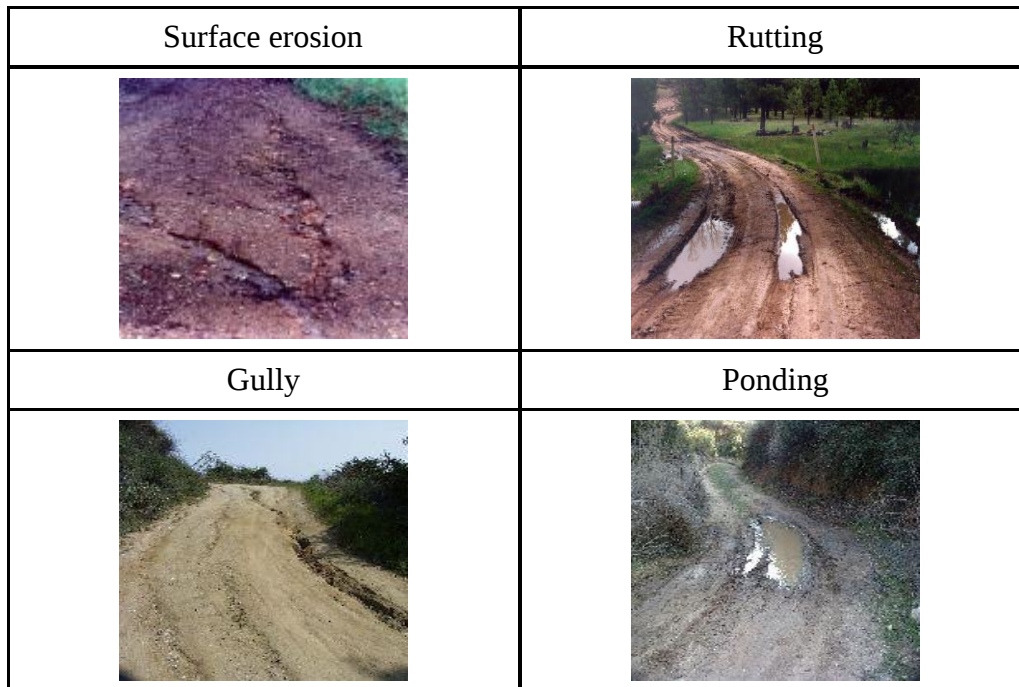


Figure 1.1. Road damage arising from roadway surface drainage problems (adapted from: USDA (2000) Water/road field interactions guide)

Drainage failure of subsurface flows can lead to backslope and fillslope slides, wheel ruts, ditch erosion and culvert failure, and loss of ditch capacity due to sedimentation thereby worsening the situation (Figure 1.2). As a result, improper drainage of forest roads can cause road access and safety problems.

Tension crack	Ditch erosion
	
Culvert washout	Sedimentation
	

Figure 1.2 Roadway and environmental damage arising from subsurface flow drainage problems (adapted from: USDA Water/road field interactions guide)

Improperly designed and maintained forest roads also have critical impacts on streams and lakes: water flow patterns may be changed by forest roads, which could block fish migration, especially by way of poor culvert placing, or mismatching culvert size and strength with actual traffic loads, or underestimating the effects of frost heaving on culvert displacement or upward re-shaping of the culvert ends. Sediments generated via road surface and slope erosion affect water quality and clog otherwise gravelly stream channels.

Proper road planning addresses potential water-road interactions with the aim of minimizing or diverting these as much as possible, and must consider local and regional differences in road hydrological conditions, as modified by topography, climate, summer through winter. Water-road interactions can be minimized by choosing from potential

road locations, and through bridges, culverts, ditches, cross-drain and diversion-channels installations. Ditches are required for proper road surface and roadbed drainages, and for directing run-off water away from roads and direct entry into streams and lakes.

At this stage, existing professional road design software packages mainly target highway and bridge design, and require users to provide detailed and accurate survey data and storm flow information. These software packages are generally expensive and fairly difficult to use in the context of common forest management practices such as designing and implementing often temporary road access to timber resources, cutblock by cutblock, as well as directing in-block harvest and wood-forwarding traffic. As a result, forest managers and engineers still design road layout and large using maps and simple spreadsheet calculations, while detailed decisions about road location and construction are made in the field, based on the experience of field personnel and road-building contractors. Nevertheless, forest road construction continues to deal with many as yet unmapped and non-anticipated situations regarding terrain, substrate and water conditions, thereby requiring last-minute adjustments to road layout and routing. Invariably, responding to these “surprises” is often costly and dealing with them on the spot may not necessarily produce the most desired effective and yet low-cost and low-maintenance road.

REPORT OBJECTIVES

There are three objectives.

1. To optimize the route selection between two end points. This objective deals with formulating a new road design tool that forest managers can use to find an optimum path

between any two given points in forested landscapes. This tool is designed to incorporate the knowledge of the road engineer, has a simple user interface and is easy to use so that it can supplement the existing toolkit of paper-based maps, clinometers, compasses, aerial photography, geographic information systems, and global positioning systems (GPS). The tool is interactive, contains automated optimization logic, and enables the road engineer to change the suggested road-optimization criteria easily. The tool can be used:

- to find the optimum route between two given points;
- to assess the layout of existing roads;
- to provide users who wish to undertake a manual design process with a good visualization and assessment device.

2. To determine number, size and culvert locations along planned forest roads. In

forest road design, the proper design and placement of culverts is key to the maintenance of the roadbed and to the aquatic health and to the attainment of government standards (Sweet, 2004). Some tools have been developed to calculate culvert sizes and determine their most appropriate locations on a given road network, but these tools have some deficiencies. For instance, existing tools do not have a function to reduce redundant culverts through locating alternative routes. The objective is to build a flow-channel crossing tool that can place and size culverts efficiently, by providing the user with analytical capacity:

- to automatically assess the size of the watershed, the dominant vegetation types within the drainage area, as well as the storm flow rate, and
- to determine the location and size of culvert on existing forest roads.

3. To perform the calculations by way of ArcVGiew GIS procedures. The ArcView and DTM-based FOROAD extension tool consists of three modules:

- visualization, or ArcView GUI (ForVisual),
- culvert design (ForCulverts) and
- forest road layout (ForRoutes, see Figure 1.3, top portion of the ArcView menu bar).

ArcView is a popular geographic database and mapping software package in more than 90 countries. It uses the script-programming language “Avenue” for tool developments (compiled and implemented as extension tools). The tools so produced can be used as part of the fully integrated geographic analysis package. This includes using digital terrain models (DTM) to map flow-channels, wet areas, and likely depth to water below the soil surface with operationally sufficient resolution (Appendix).

REPORT ORGANIZATION (Figure 1.3)

The methodology for mapping wet areas is introduced in Chapter 2. The ForCulverts module of FOROAD is described in Chapter 3, followed by the ForRoutes module in Chapter 4. Chapter 5 presents FOROAD in the context of other road design tools. Original contributions of this work and recommendations for future work are summarized in Chapter 6. The appendices include a ForRoad tutorial, the code of this tool, and data for the climate normal in Atlantic Canada.

LITERATURE CITED

Sweet, V., 2004. Culvert Sizing and Placement on Forest Roads with Arcmap. Bachelor thesis, University of New Brunswick, Fredericton, N.B. Canada

Water/Road Interaction Field Guide, 2000. USDA Forest Service, San Dimas Technology and Development Center, San Dimas, California.

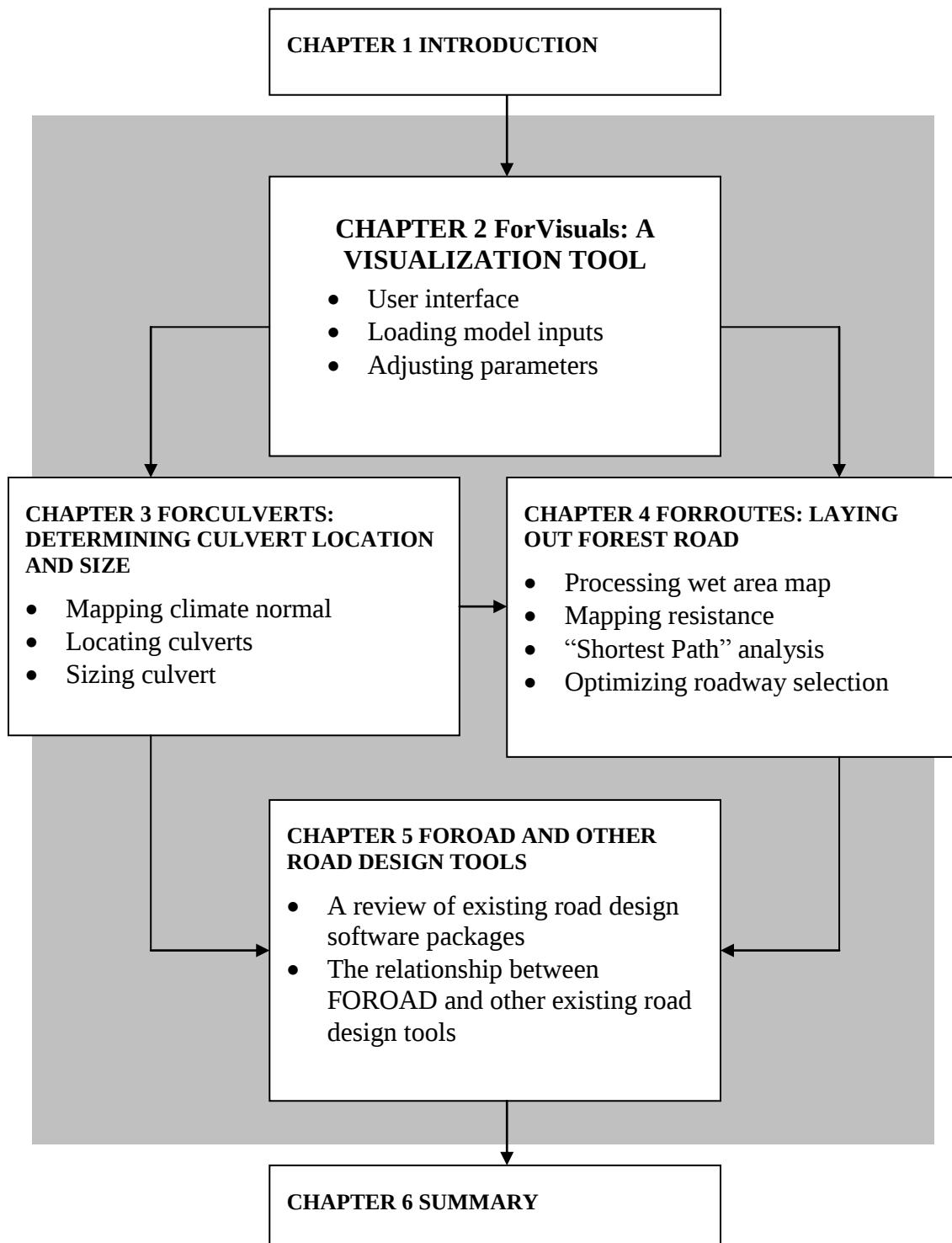


Figure 1.3 Report outline and relationships between chapters

CHAPTER 2

GETTING STARTED

The objective of this chapter is to present the ForVisual component of the ForRoad model, by describing its graphic user interface (GUI) to parametrize the ForCulverts and ForRoutes modules, and to load the maps that are needed to determine road and culvert locations. This includes maps that depict topography by way of a digital elevation model, slope, existing road maps, hydrographic features such as already mapped wetlands, streams, rivers, and shorelines associated with lakes and other surface water features, and likely depth-to-surface water.

This Chapter also describes which particular information and data layers are needed as part of the start-up preparations, and how this information is accessed and displayed by the GUI shown in Figure 2.1. Required inputs refer to ArcView-formatted DEMs (Figure 2.2), hydrographic maps (lakes, streams and rivers, and shorelines, Figure 2.3), depth-to-surface water map (Figure 2.4), weather maps (climate normal, Figure 2.5), and existing or designed road segments (Figure 2.6).

Input dialogue

The DEM and hydrographic and depth-to-water maps are uploaded into ForVisual by way of the functional buttons in the left column of the input dialog (Figure 2.1), i.e., “Load DEM”, “Load Streams”, and “Load Depth-to-water”. The second column of the input dialog (Figure 2.1) includes two check boxes, to choose a 25 or 100 weather operations cycle, and a functional button, to load the “weather map”, which refers to a

geospatially interpolated representation of climate normal (precipitation amounts) for the area of interest. Choosing the 25 year weather period increases the risk of incurring road damage due to improper culvert size. In contrast, choosing the 100 year weather period increases the cost of road construction, by requiring culverts of larger diameter than what would be chosen for the 25 year planning frame. The third column of the GUI shows a list of existing themes. Newly produced road layouts by ForRoutes will be also presented in this list. By selecting a theme from the list, the user can identify the road network for locating and sizing culverts.

Parameter selection

The second section of the GUI deals with selecting parameters (wet areas, topography, and culverts) for road design. The three slide dialogs refer to the change of impact that each factor has on the road construction and maintenance. Sliding the slide dialogs to the right will emphasize the selected parameter, and vice versa. The middle portion of this section allows the user to enter minimum culvert diameters. The three functional buttons on the right refer to:

- “Load Cover Type”, used to specify vegetation data, which can be used to determine the dominant vegetation type which, in turn, is needed to calculate the rainfall runoff coefficient;
- “Load Sensitive Area” , used to identify an area that a designed forest road cannot pass. Such kind of areas could be protected habitats, scenic areas, dangerous district for collapse, etc.

- “Confirm and Close” is used to transfer the selected user input to the computer program, and to close the GUI.

Module specifications

The third section of the GUI is for reading, focussing on input/output specifications and related data requirements. The following is the complete text.

Model dependencies:

3D analysis, Spatial analysis, Av.DLL.

Input:

Visual Module requires DEM, mapped streams datalayers

Culvert Module requires DEM

Route Module requires DEM, mapped stream (or depth- to-water index), existing road network

- All maps must have same projection system.
- Single input is required. Multi-choice selections in an input dialog might result in a wrong calculation.
- Loaded maps and input numbers and change of parameters will be applied directly. Redoing these inputs will replace previous action.
- The tool for cover type is not integrated, because the data is not ready yet.
- The reset button disables the inputs from this panel, and clears all invisible variables.

xVisual:

Flow accumulation and Flow direction must be figured out before mapping culvert.

Road theme can be either identified from list window or loaded from 'Load Existing Road' button.

As required input for Culvert module, weather maps must be loaded, and return the period must be entered.

xRoute:

To smooth a road line, the theme must be editable and the line feature must be selected manually.

Smoothing can be undone, by selecting <undo> in the edit menu.

Make a single selection when comparing two different routes in the same location.

Activate the fixed DEM theme when identifying the starting point of a planned route.

The three modules, ForVisual, ForCulverts and ForRoutes will be ready to run after the required data are loaded and optional parameters are confirmed. The tool menus of these modules are shown in the top of Figure 2.6.

The screenshot displays the FOROAD GUI, which is organized into three main sections:

- Input Dialog:** This section contains three columns of controls. The first column has three buttons: "Load DEM", "Load Streams", and "Load Depth-to-surface water". The second column, titled "Choose flood Period", includes a checked checkbox for "100 year" and an unchecked checkbox for "25 Year", with a "Load Weather Map" button below them. The third column, titled "Choose Road Theme", features a list box with options: "100yr30min", "Bbw_wtbl83", "Flow Accumulation", and "\$Filled", along with a "Load Existing Roads" button.
- Interactive Parameters:** This section includes three horizontal sliders for "Wet area", "Culvert", and "Slope". To the right of these sliders is a "Minimum Culvert (cm):" input field with the value "40". Further right are three buttons: "Load Cover Type", "Load Sensitive Area", and "Confirm and Close".
- Tutorial:** This section contains a text area with the following content:

Model dependencies include: 3D, Spatial anylysis, Av.DLL

INPUT:

 - Visual Module requires DEM, Mapped stream
 - Culvert Module requires DEM
 - Route Module requires DEM, Mapped stream(or Depth - to -surface water)

All maps must have same projection system

Loaded maps and input numbers and change of parameters will be applied to workdirectly. Redoing these input will replace prvious action

Single input is required. Multi-choice in a input dialoq might result in a wrong calculation

Figure 2.1 the GUI of FOROAD. Users provide data and modify parameters in this window. The bottom tutorial window was designed to explain this dialog.

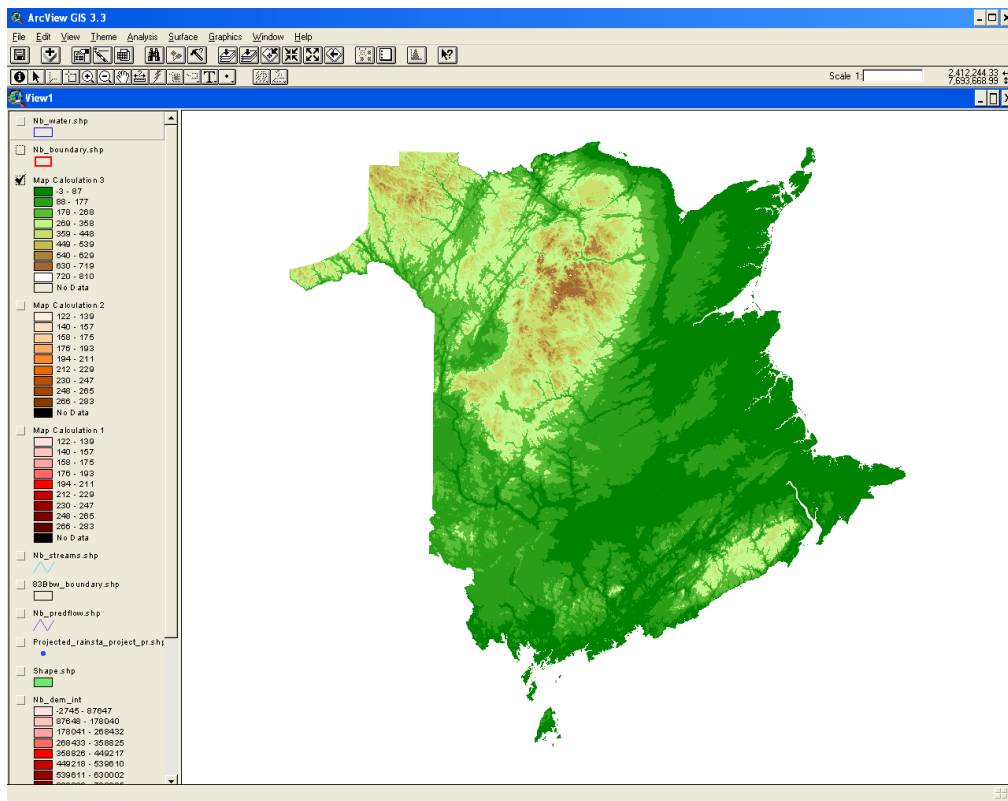


Figure 2.2 Loading DEM of New Brunswick, Canada

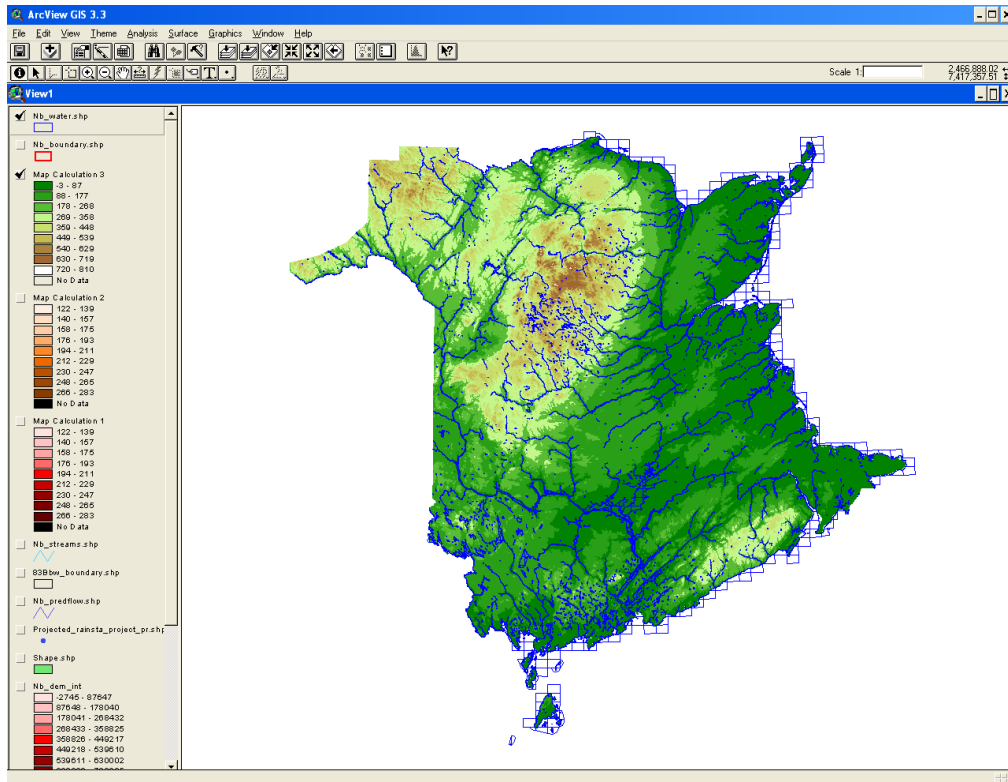


Figure 2.3 Loading water features, which include lakes, streams and rivers in New Brunswick, Canada.

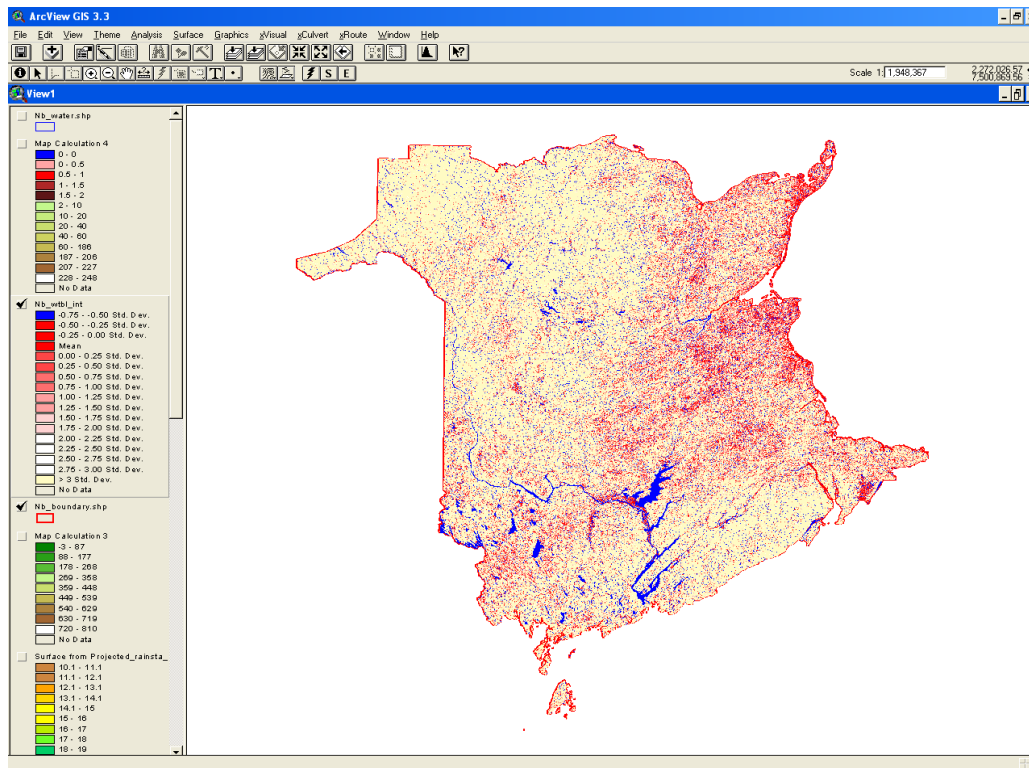


Figure 2.4 Loading the Depth-to-Surface water map of New Brunswick

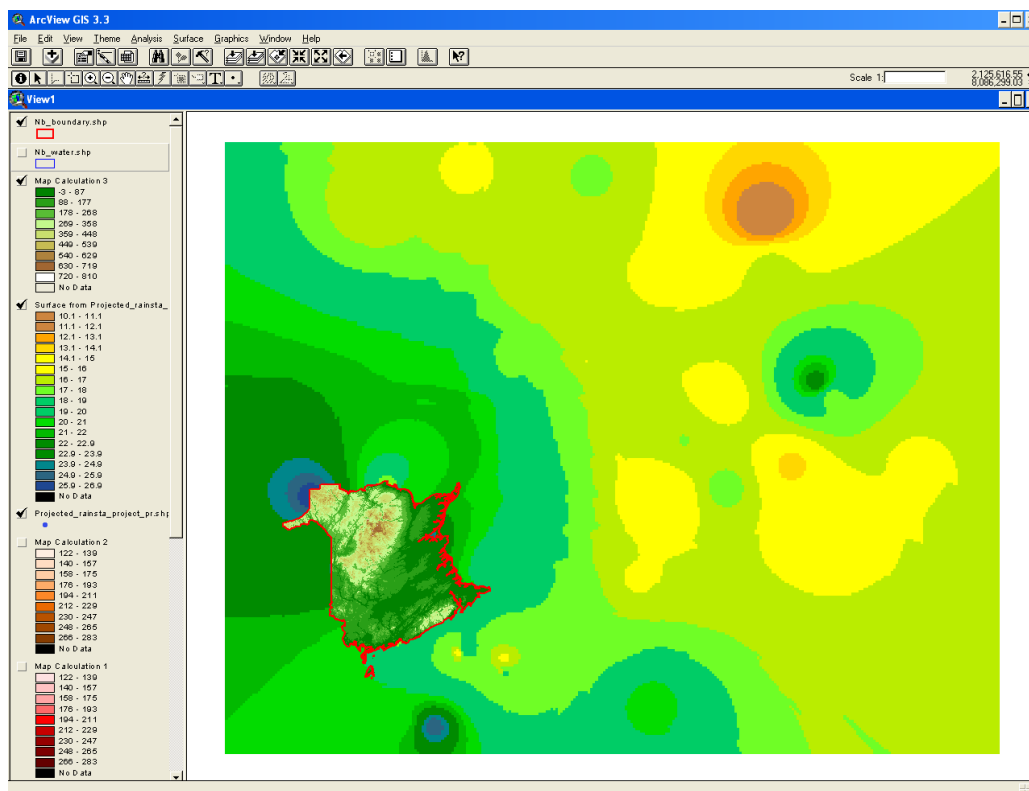


Figure 2.5 Loading Atlantic Canada weather map. The methodology for making this map is shown in Chapter 4.

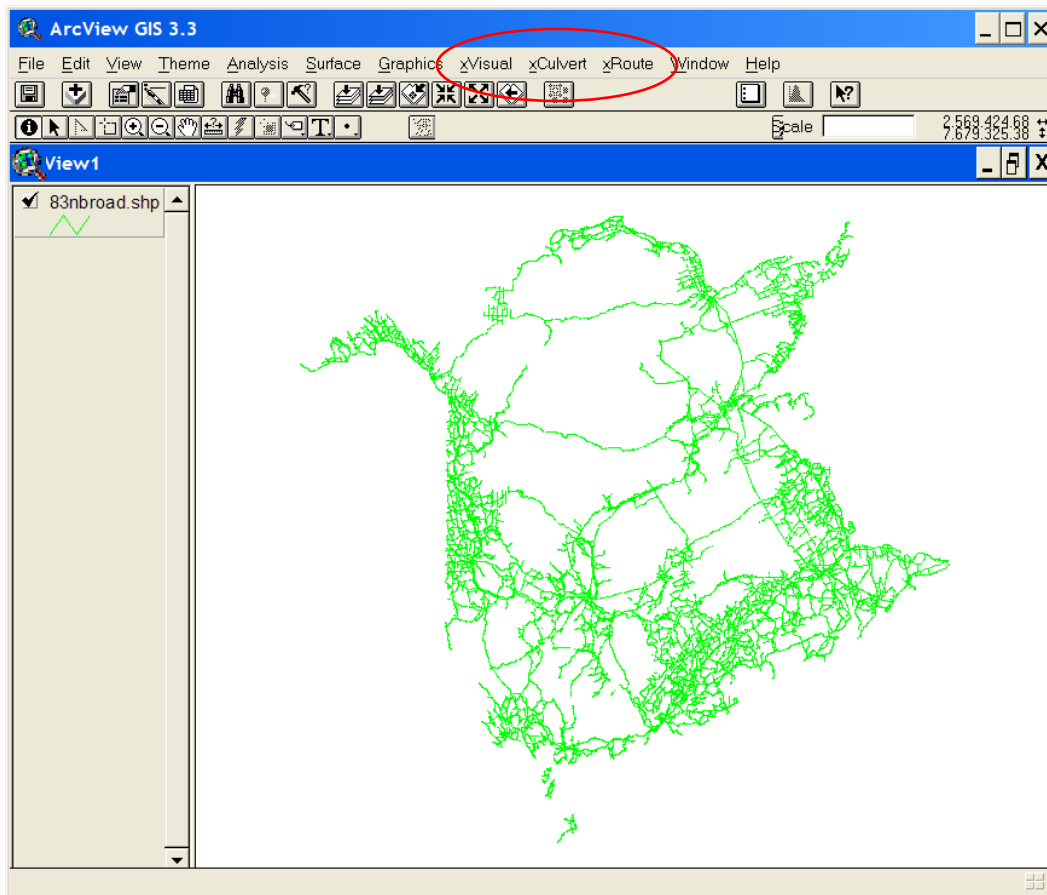


Figure 2.6 Loading existing road network for New Brunswick area. Dropdown menus in the top red oval control three modules of ForVisual, ForCulverts and ForRoutes.

CHAPTER 3

DETERMINING CULVERT LOCATION AND SIZE

INTRODUCTION

Sweet (2004) developed a GIS tool for estimating the location and size of culverts for forest roads. This tool uses a digital elevation model (DEM) as input, and determines the location of each mapped and unmapped flow channel. A digital elevation model (DEM) is a digital representation of ground surface topography or terrain. A DEM can be represented as a raster (a grid of squares) or as a triangular network. DEMs are used as a common basis for digitally-produced relief and flow channel modelling (Wikipedia, 2007). For example, DEM-derived flow channels via the flow accumulation algorithm can be forced to comply with the locations of already mapped flow channels. In turn, the DEM can then be used to locate other flow channels that feed the already mapped channels (Meng et al., 2006).

Automated culvert selection tools assume that culverts should be installed at the intersection points between proposed roads and mapped and unmapped channels. The size of each culvert is then determined by total drainage area above the channel-road intersection point. In addition, culvert size must also be related to anticipated peak flows and peak durations as they would occur over stated time intervals, e.g., 25 or 100 years. This Chapter describes the new ForCulverts module, designed to assist forest managers by determining culvert locations and sizes along unpaved forest roads, based on flow-channel, cover type maps, and regional climate normals, i.e., arithmetic means of a specific climatological elements such as daily precipitation computed over three consecutive decades (Environment Canada, 2007). This tool can be used:

- to automatically determine watershed area above any road-stream intersection point, the dominant vegetation types within the drainage area, as well as the storm flow rate.
- to automatically determine culvert size for each intersection point in a given area, and find out their locations and sizes.
- to evaluate culvert requirements among alternative routes between two endpoints, including existing roads that already connect these points.

The process of determining culvert size is summarized by the following input specifications:

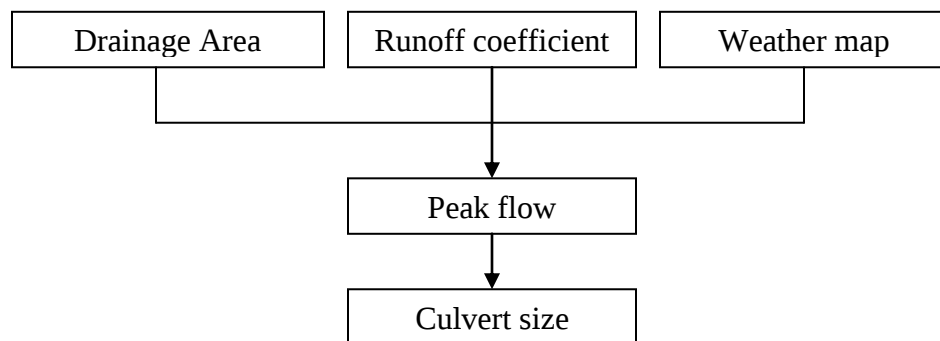


Figure 3.1 Processing flowchart for sizing culvert

STUDY AREA

CFB Gagetown is one of the largest military bases in Canada, with a land base of about 1100 square kilometers. CFB Gagetown is located 20 kilometers east-southeast from the city of Fredericton, New Brunswick. Data layers obtained for this area include DEM, and stream and road map. The data also involve GPS locations for culverts and a long select route of the existing road network.

METHODS

The Rational Method [Eq. 1] and Manning Equation [Eq. 5] are used in this study to calculate peak flows and culvert size. Prospective culvert locations are derived from the DEM for the area using Spatial Analysis tool for flow direction and flow accumulation in ArcView GIS.

Peak Flow

Culvert diameters can be derived from Peak Flow values directly. The algorithm for calculating peak flow by way of the widely used “rational formula” (ConDOT, 2000) is given by

$$[1] \quad Q = C \cdot I \cdot A^{0.76}$$

where:

Q = peak flow (m^3/s)

A = Drainage Area (km^2)

I = Extreme Weather Events (mm/h)

C = Runoff coefficient

The rational formula therefore allows for a direct mathematical evaluation of the peak flow, the drainage area, the extreme weather precipitation rate, and the influence of slope and length of the main channel and roughness of stream channels (by way of the Runoff coefficient). For specifications are required for: the time for water to travel through the watershed, and the effect of different rainfall intensities and duration upon discharge (Rothwell 1978).

Drainage Area

A drainage area (watershed) is an area that drains water and other substances to a common outlet as concentrated drainage (Greenfacts, 2007). ArcView GIS can determine the drainage area above a set of cells in DEM (ESRI, 2001). In this tool, the original DEM is modified with the existing stream and road network data in ArcView GIS, to ensure that the flow direction and flow accumulation algorithms comply with already mapped flow channels and ditches. This is necessary because the topography around a road site is changed after new roads are constructed, and the flow patterns adjacent to the road (built or to be built) is modified accordingly.

Runoff coefficient “C”

The runoff coefficient C is the variable depending on the soil type, watershed slope and land use. Thus far, **ForCulverts** calculates the runoff coefficient based on the soil type and watershed slope alone. Since watersheds often contain more than one soil types, it is necessary to determine the runoff coefficient on an area weighted basis based on method developed by UW-M (2007), i.e.;

$$[2] \quad C = \frac{\sum C_i A_i}{\sum A_i}$$

where C_i (Table 3.1) is the coefficient applicable to the area A_i . In case information data for “C” is not available, the model default value range for “C” is 0.12 (Milam, 2007), which is a commonly used value for forested areas.

Table 3.1 Determining runoff coefficients based on slope percent
(Frevert et al. 1955)

Slope (%)	Open Sandy Loam (C_i)	Clay And Silt Load (C_i)	Tight Clay (C_i)
0-5	0.10	0.30	0.40
5-10	0.25	0.35	0.50
10-30	0.30	0.50	0.60

Extreme Rainfall Events (“I”) Distribution Map

To determine the extreme weather condition, a series of rainfall events maps are interpolated from historic precipitation data using the surface tool in ArcView. Raw data (Appendix B) were collected from Environment Canada including the locations of rainfall monitoring stations (in terms of longitude and latitude), rainfall accumulation, concentration time and return period.

In ArcView GIS, the maps are created with the Inverse Distance Weighted (IDW), which assumes that each rainfall datum has a local influence that diminishes with distance; therefore, a raster-based surface is generated by interpolating the dispersed rainfall data, with every cell receiving an estimated rainfall value (Figure 3.2). The conventional rainfall map from the Rainfall Frequency Atlas for Canada (Figure 3.3) can also be used to estimate the largest amount of precipitation likely to fall at a point. However, estimating probable maximum precipitation (PMP) from this map is not

automatic, i.e., the specific location of interest needs to be identified on the map manually.

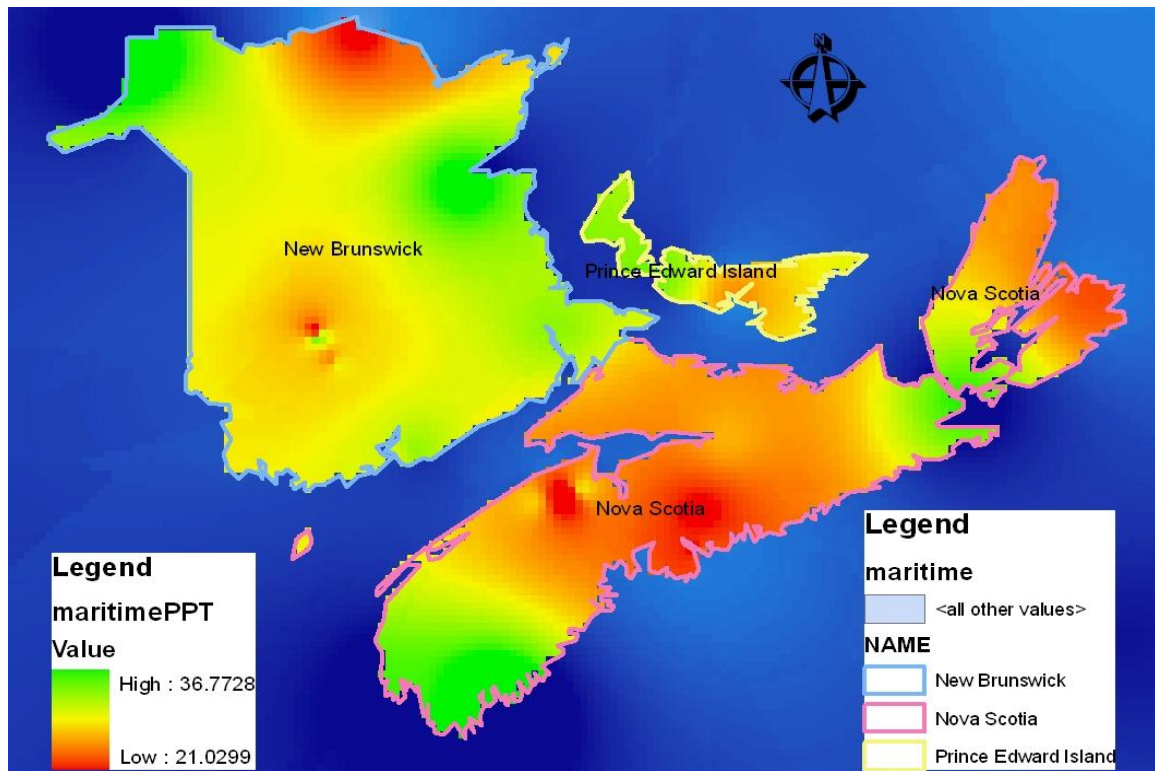


Figure 3.2 Digital map of Rainfall Distribution for Atlantic Canada (30 minutes event with a return period of 25 years). This map was interpolated with the data of Atlantic Canada climate normal collected from Environment Canada

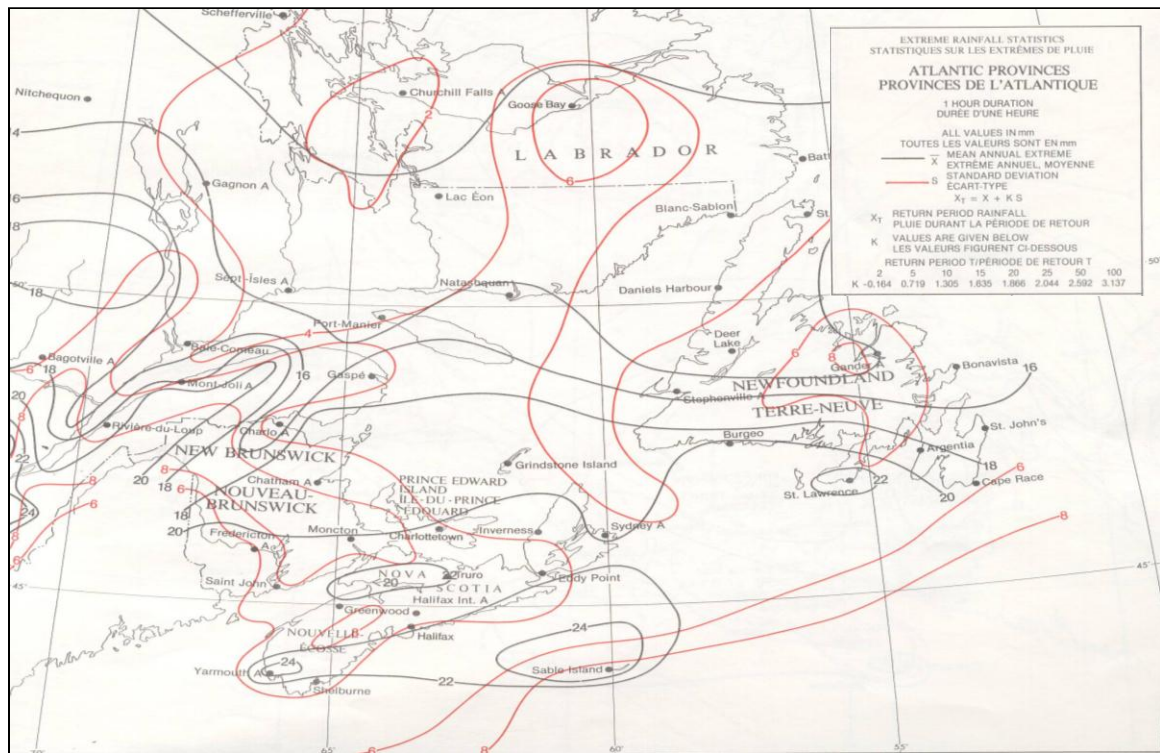


Figure 3.3 Isometric precipitation map, adapted from Rainfall Frequency Atlas for Canada (Hogger, 1985)

Time of Concentration

Concentration time is the time required for a particle to travel from the watershed divide to the watershed outlet. The Revised Kirpich equation is used for the calculation of this concentration time, i.e.:

$$[3] \quad t = 55.704 k (A / S)^{0.385}$$

where:

k = Kirpich adjustment factor (Table 3.2)

A = Area of watershed, Ha

S = Average slope of the watercourse, m/m

t = Time of concentration, minutes

This equation was developed from data obtained in seven rural watersheds in Tennessee, USA (Table 3.2).

Table 3.2 Kirpich adjustment factor (k)

Ground Cover	Kirpich Adjustment Factor, k (Chow et al., 1988)
General overland flow and natural grass channels	2.0
Overland flow on bare soil or roadside ditches	1.0
Overland flow on concrete or asphalt surfaces	0.4
Flow in concrete channels	0.2

Return period

The return period describes the frequency of extreme weather events, which allows us to quantify acceptable risk. The return period of rainfall is a common criterion of culvert design. The formula used to estimate return period is given by:

$$[4] \quad m = (N+1)/ T$$

where:

T = return period, recurrence interval in years of a given flow

N = number of years of record (determined by existing database)

m = the rank of the rainfall density value from the biggest to the smallest

The rainfall intensity is the average rainfall rate (in mm/h) for a duration that is equal to the time of concentration for a selected return period (RTI, 2007). Once a

particular return period has been identified for design, the rainfall intensity data for each watershed can be chosen from a built-in rainfall-intensity map according to the concentration time. The default value of return period in this model is 100 years.

Sizing Culverts

The Manning equation is used to calculate the diameter of the culverts (Rothwell, 1978), as follows:

$$[5] \quad D = \left(\frac{Q^{3/4} * n^{3/4}}{S^{3/8} * \pi^{3/4}} \right)$$

where:

D = diameter of culvert (m)

S = slope of culvert, generally ranges from 1 - 2%, depending on user choice;

1.5% was selected by default

n = roughness coefficient for culvert, set at 0.021 (Chow, 1959)

Q = discharge rate (m³/s), obtained from Equation [1].

The Manning Equation was developed for uniform steady state and open channel flows. This semi-empirical equation is a commonly used method for simulating water flows in channels and culverts where the water is open to the atmosphere. This equation was first presented by Robert Manning in 1889.

Locating Culverts

As stated in “Drainage Area”, both DEM and flow patterns are affected by roads once constructed. In Arcview, the flow direction and flow accumulation calculations are modified accordingly, and the positions for potential culverts are obtained by

automatically locating all subsequent channel-road intersections. Once located, the calculations then proceed, again automatically, by determining watershed borders and size above the intersection points. A minimum distance of 20 meters was used as default value for distance between any two adjacent culverts along the same road. The redundant culvert locations within this distance were merged onto merged into one location, at center. This minimum length is a user-defined parameter, usually set at 100 m.

Test Data

The ForCulverts calculations were tested using GPS-tracked locations for existing culverts at select routes in CFB Gagetown, New Brunswick. The calculations were based on the overlay of the existing DEM, road network and stream data layers for the area (Figure 3.4).

RESULTS

As displayed on Figure 3.5, the model suggested that there should be 99 culverts along the selected route, but 14 of these were removed due to the redundancy criterion of not having more than 1 culvert within 100 meters from another culvert along the same road. Hence, the model favors fewer culverts but with larger diameter. In addition, the model calculations indicate that 70% of the modelled culvert locations lie within 50 meters of the actual culvert locations. Suggested culvert sizes are listed within the FOROAD-produced output table in Figure 3.7).

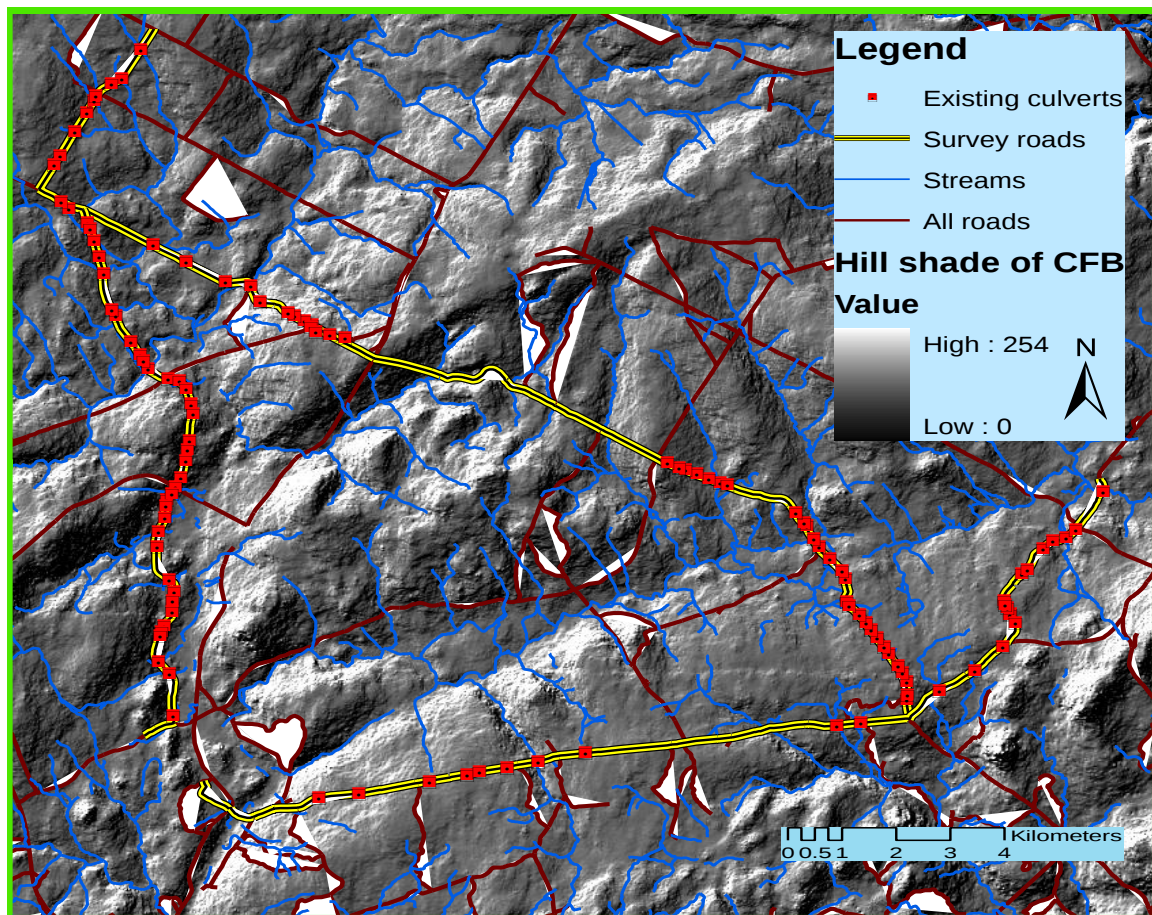


Figure 3.4 Topographic map (background shape file), stream channels (blue line features) and road network map (brown line features), and existing culverts locations (red square features) on survey roads (yellow line features) of CFB Gagetown in central New Brunswick.

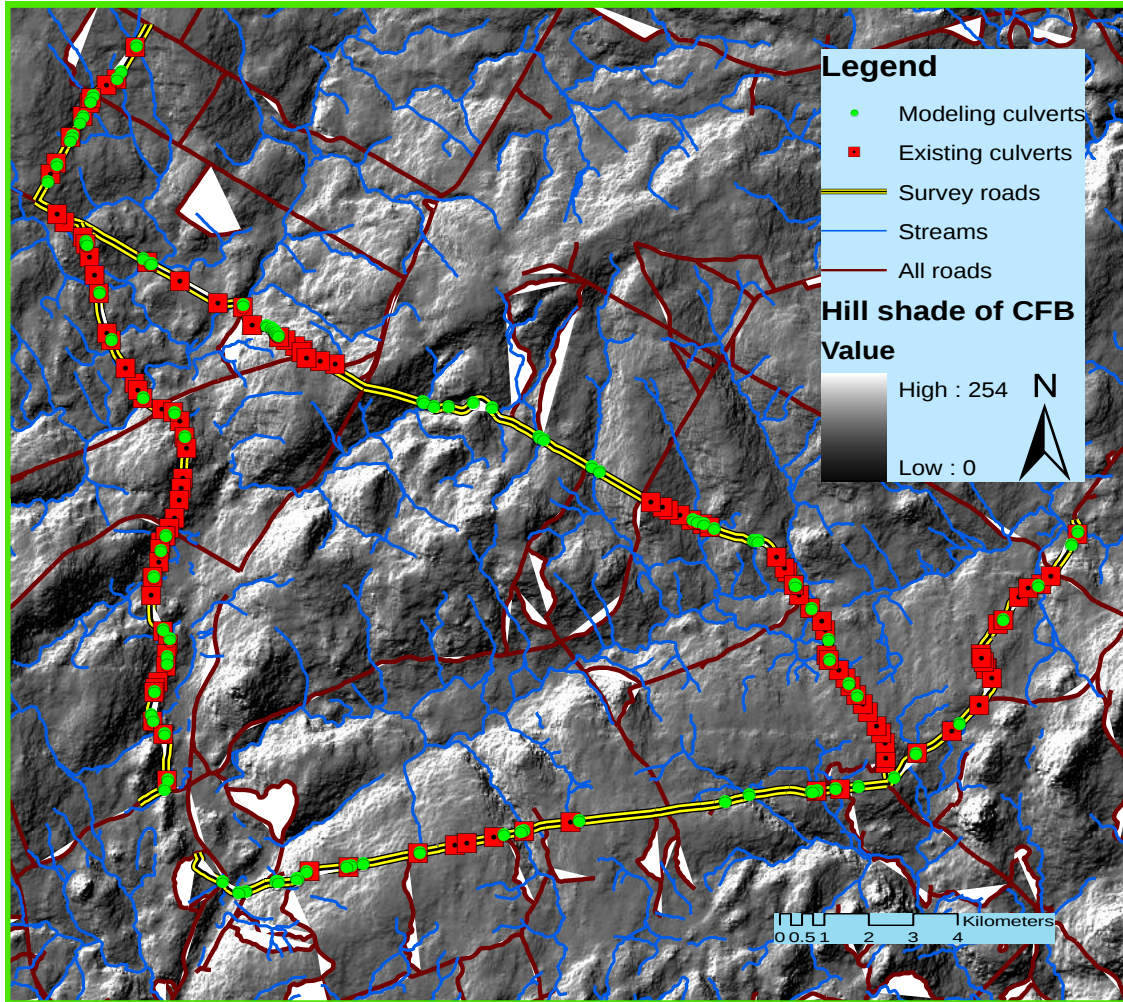


Figure 3.5 Model predict and existing culverts locations illustrated with ArcView GIS map

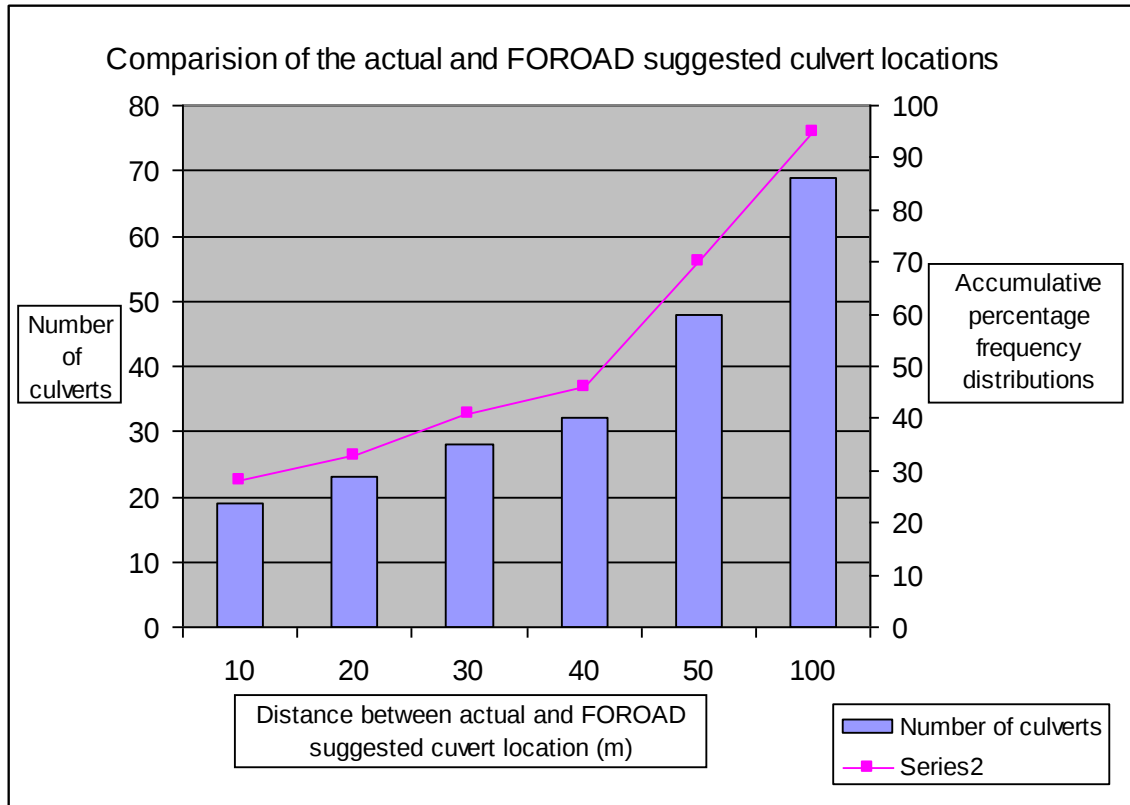


Figure 3.6 Model predicted and actual culverts

Shape	PointID	X'	Y'	Diameter
Point	39	707761.8	5052781.1	0.64
Point	71	716401.8	5050401.1	0.39
Point	147	709601.8	5045161.1	0.35
Point	43	708231.8	5052001.1	2.41
Point	28	707551.8	5053541.1	0.96
Point	1	707971.8	5056961.1	0.35
Point	72	708331.8	5050321.1	0.15
Point	119	717051.8	5046631.1	0.88
Point	105	719391.8	5047281.1	0.53
Point	116	717961.8	5046701.1	0.39
Point	64	721611.8	5050511.1	3.15
Point	27	708291.8	5053891.1	2.07
Point	80	721091.8	5049741.1	1.25
Point	125	713831.8	5046011.1	0.82
Point	110	708691.8	5047231.1	0.53
Point	112	718201.8	5046741.1	0.47
Point	5	707751.8	5056541.1	0.43
Point	94	718091.8	5048631.1	0.36
Point	8	707381.8	5056221.1	0.34
Point	132	713681.8	5045971.1	0.34
Point	24	707401.8	5054051.1	0.33
Point	142	710641.8	5045231.1	0.32
Point	31	708031.8	5053351.1	0.32

Figure 3.7 Culvert size and locations are listed in an attribute table

DISCUSSION

Forest roads will inevitably intersect streams and natural drainage areas at some points. As stated, the flow and presence of water along roads have major impacts on forest roads and vice versa. Correctly locating culverts and minimizing the number of these through proper road routing is therefore important avoid road-water contacts, which – in turn – should lead to significant reductions in negative road and water impacts, while – at the same time – reduce overall road construction and maintenance costs. In this context, forest roads are already of low grade, and often designed and built with considerable time and budget constraints. However, extensive user training requirements, acquisition of accurate field surveys, and working with high-level road-design software packages tends to limit the use of such packages to the design of major arterial highways. In forestry, sizes of the larger culverts are often determined by simple spreadsheet calculators. and the locations and sizes of small culverts are generally determined in the field, based on and limited by the experiences of the field personnel and the road-building contractors.

In this paper, a simple tool (**ForCulverts**) has been described that can be used to assist forest managers and planners to determine culvert locations and sizes for forest roads automatically from digital elevation data, and from already mapped information about stream and actual and potential road locations. The program should be tested and verified further, but even in its default formulation can provide forest engineers with a planning layer to locate and size culverts across the terrain along already existing or contemplated routes. In detail, the functions of **ForCulverts** are listed as follows:

- The tool can be used by forest managers for road design. **ForCulverts** can be used to determine locations and sizes of culverts required by a particular road layout.
- The tool could also be used to assess the locations and sizes of existing culverts on forest roads. With DEM and actual or desired road locations, **ForCulverts** can be used to generate reasonable road alternatives.
- The tool determines the size of the watershed, the dominant vegetation types with the drainage area, and anticipated storm flow rate during extreme events. Drainage areas and peak flows are also determined. **ForCulverts** provides output for any of these parameters.
- The tool provides an automated procedure to calculate all stream crossings for a given area; doing this manually is also an option.
- The tool provides an assessment of side slope and soil erosion potential. Soil loss concentrated at road-stream crossings are estimated based on watershed area and slopes.

LITERATURE CITED

- Brinker, R. and Tufts, R. 1995. Forest Roads and Construction of Associated Water Diversion Devices. Retrieved May 29, 2007 from <http://www.aces.edu/pubs/docs/A/ANR-0916/ANR-0916.pdf?PHPSESSID=4dd00077338d5bc5706800cbb1ae4835>)
- Chow, V.T. 1959. Open-channel hydraulics. McGraw-Hill, New York, NY, 680 p.
- Chow, V. T. Maidment, D. R. and Mays, L. W. 1988. Applied Hydrology. McGraw-Hill.
- ConDOT. 2000. Chapter 11.5-1 Storm Drainage System: Hydrology, Department Of Transportation, Connecticut, United States of American
- Environment Canada. 2007. National Climate Archive: Climate Data Online. http://www.climate.weatheroffice.ec.gc.ca/climate_normals/climate_info_e.html

- ESRI, 2001. Arcview 3.3 software help files. Copyright 1999-2002: Environmental System Research Institute Inc.
- GreenFacts. 2007. GreenFacts Glossary. <http://www.greenfacts.org/glossary/def/drainage-area.htm>
- Meng, F., Castonguay, M., Ogilvie, J., Murphy, and P. A. Arp. 2006. Developing a GIS-based flow channel and wet-areas mapping framework for precision forestry planning. IUFRO Precision Forestry Conference 2006
- Milam, 2007. Runoff Coefficient. Retrieved May 29, 2007 from <http://www.geocities.com/Eureka/Concourse/3075/coef.html> (Copyright © 1998 Donald R.Milam, PE)
- Rothwell, R.L., Schlabach, G.O. Deminster, T.W. and Barnes, K.K.1955. Soil and Water Conservation Engineering. John Wiley and Sons, Inc. New York
- RTI. 2007. RTI: An overview of CULSED. Retrieved May 30, 2007 from <http://www.ruraltech.org/tools/culsed/index.asp>
- Sweet, V. 2004. Culvert Sizing and Placement on Forest Roads with Arcmap. Bachelor thesis, University of New Brunswick, Fredericton, N.B. Canada
- Hogg, W.D. 1985. Rainfall Frequency Atlas for Canada. Minister of Supply and Services Canada
- USDA.. 2000. Water/Road Interaction Field Guide. United States Department of Agriculture Forest Services Technology &Development Program. Retrieved May 30, 2007 from <http://www.fs.fed.us/eng/pubs/html/00771803/00771803.html>
- UW-M. 2007. Hydraulic principles: Chapter 3 Runoff coefficient. Biological Systems Engineering, University of Wisconsin-Madison
http://209.85.165.104/search?q=cache:dMxUztufvP8J:bse.wisc.edu/courses/472/Lecture_Notes_03_Ch3.doc+runoff+coefficient&hl=en&ct=clnk&cd=7
- Wikipedia. 2007. Digital elevation model. Wikipedia: the free encyclopedia. Adapted on June11, 2007 from http://en.wikipedia.org/wiki/Digital_elevation_model

CHAPTER 4

ForRoutes: LAYING OUT FOREST ROAD

INTRODUCTION

In forested areas, designing a roadway to connect two points sounds like a simple task. Certainly, a straight line between two points will result in the shortest road to build. Some forest road networks have been planned in this way, as shown in Figure 4.1 (left). Such roads do have some obvious benefits:

- Minimizing the length of a forest road would perhaps result in the lowest construction costs.
- The transportation cost should be lower because the traveling distance between two points could be shortest.
- The road network is geometric, and therefore easily registered and managed.
- The wood-forwarding distances could be even with even road spacings.

However, straight line road networks (Figure 1, bottom left) are quite rare in forestry, with a network similar to what is shown in Figure 4.1 (top right) being more common. One of the main reasons as to why some road networks do not follow straight lines refers to topography, and related flow channel and soil drainage regimes, on account of avoiding areas that are either too wet or too steep. Other reasons may refer to land ownership, avoiding areas of unique values in their pristine states, or areas set aside for wilderness or conservation.



Figure 4.1 Forest road network in a portion of northern New Brunswick, Canada (left: straight forest roads on J D Irving Ltd; Right: Forest road on crown land next to private land. (Source from: <http://maps.google.com/>)

In detail, topographic conditions, as these may vary along existing or planned routes, affect road design and layout the most. Among existing design tools for low volume roads (LVRs), ROUTES (Sessions, 2006.) and PEGGER (Rogers, 2005) solely focus on topographic road designing, especially the designing of road slopes and related limitations.

The pattern of local soil moisture and water drainage regimes also has major impacts on forest road planning and design: time periods with high moisture contents in the road surface materials or foundations will reduce the overall load-bearing capacity of the roots, which then will induce road rutting when wet, and frost heaving during winter through spring. The drainage failure of subsurface flow could also lead to backslope and fillslope slides, ditch erosion and culvert failures.

With a recently developed depth-to-surface water map, the distribution of wet soil conditions across entire forest regions can now be displayed with reasonable resolution (10 m) and reliability. This Chapter reports on the **ForRoutes** tool which enables the user to apply the depth-to-surface water map to the forest road design and layout task. In general, **ForRoutes** can be used to find the optimum route between two given points within any forested area of interest. The purpose of this tool is to locate the route with the “least cost resistance”, through automated or manual means. “Least cost resistance” refers to optimizing route location(s) while restricting:

- the overall road length through wet soil areas,
- the number of streams to be crossed,
- the placement of road segments along slopes that are too steep.

The manual mode allows the road planner / designer to plot likely route locations by hand, based on professional experiences and judgment. **ForRoutes** then ignores the automated road-optimization routine, and restricts itself to the visualization of the route within its topographic, flow-channel and wet-areas setting, and to summarizing the details of this setting, such as the combined length of all wet-area crossings, the number of all channel crossings, and the elevational profile along the selected route(s).

STUDY AREA

The Black Brook Basin (BBB) is approximately 14.5 km² and drains into the Saint John River at Grand Falls, New Brunswick. It is located between 47°05' and 47°09'N and between 67°43' and 67°48'W. Elevation ranges from 180 to 260 m above mean sea level in Black Brook Basin (Fowler, 2003). Most of the Black Brook Basin topography is undulating to gently rolling with slopes of 1-6% in the upper portions and slopes of 4-9% in the central parts. In the lower portions, slopes are more strongly rolling at 5-16% (Mellerowicz et al., 1993).

METHODS

The ForRoutes Drawing Tool

This module provides the user with a display of the detailed depth-to-surface water map as a base layer, on which manual drawings of contemplated road segments can be made. This process is illustrated in Figure 4.2.

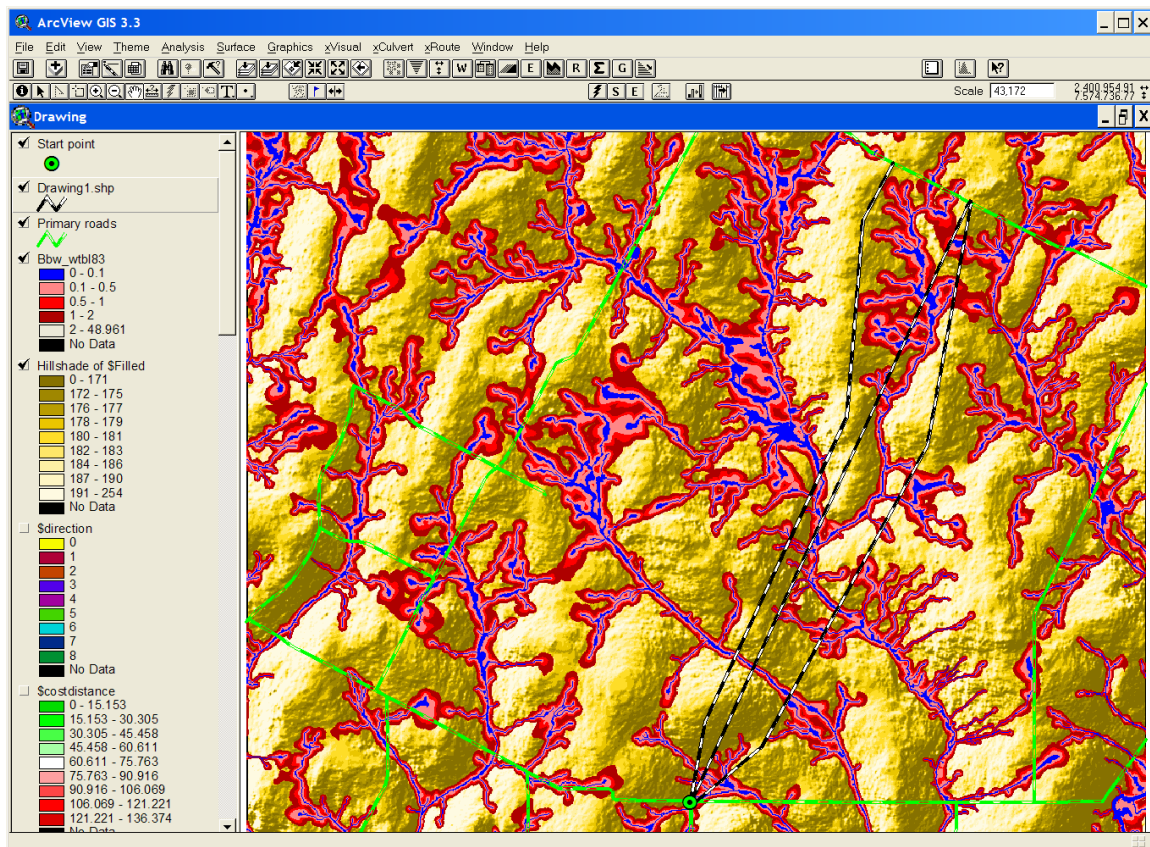


Figure 4.2 Drawing connections between two points on the wet area map. The proposed road lines (black and white dashed lines) can be drafted on the background maps, showing hill-shaded topography (yellow), water (blue) and wet soil features (pink to red, from shallow to deeper, respectively).

The Assessment Tool

This module deals with determining the elevational road profile, and summarizes combined road lengths across wet areas and culvert placements for existing or planned routes.

Road Profile. In ArcView GIS, imported road line features via a shapefile are converted into raster data and then disintegrated into a number of point features. The distance from every point on the road segment to the start point is calculated and the elevations of every point are automatically determined from the DEM. Then, a road profile is plotted on a

graph showing the natural elevation profile along the existing or contemplated route, versus distance from the route starting point (Figure 4.3).

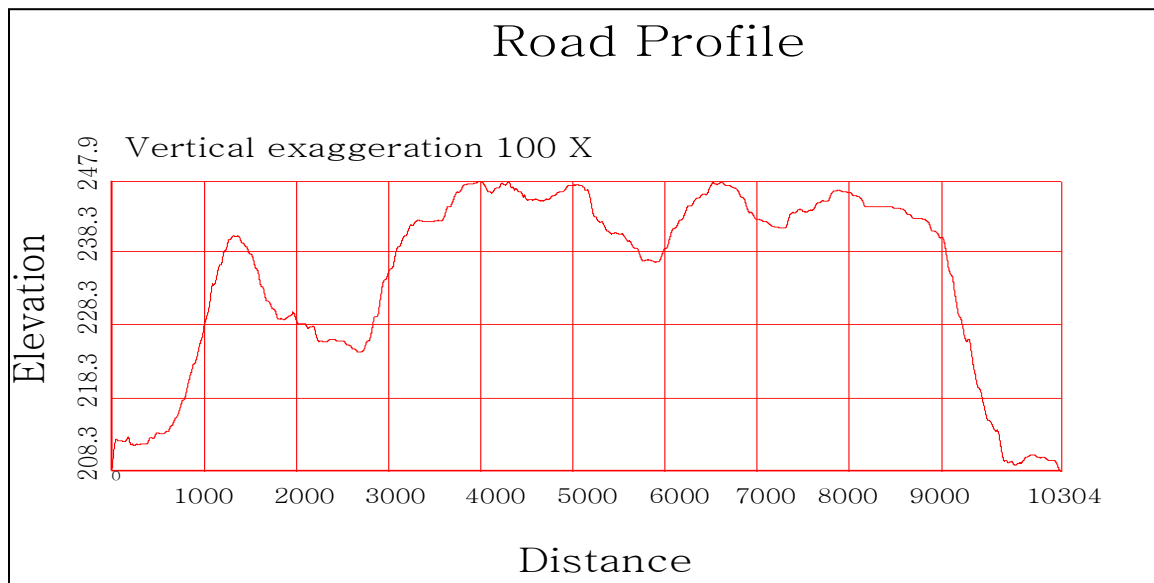


Figure 4.3 Elevation profile for an existing road segment.

Road length across wet areas. Using the depth-to-surface water map, ForRoutes automatically determines the length of road segments across wet soil areas for existing or a planned routes. In this, wet areas are defined by their likely depth-to-water when all the surface-water features nearest to the road are filled to their shorelines, as mapped. The depth-to-water threshold that defines a wet area is user controlled. It is generally $< 1\text{m}$.

Culvert evaluation. This component is used to determine culvert locations along existing or contemplated routes, using the ForCulverts module (preceding Chapter), and to profile the number and sized of culverts needed for particular road segments.

A visual application of depth-to-surface water map on forest road layout

Laying a road in wet areas will result in high construction and maintenance costs. With hand-held devices like maps, compass and rulers, detailed field reconnaissance to locate wet areas is often time-consuming. However, due to usually limited design and construction time, the road layout is often decided in the forest field, based on the experience of the field personnel.

Thus, lacking full field survey could lead roads to be positioned in wet area (Figure 4.3 red line in left picture). With a depth-to-surface water map, foresters could easily find saddle points between wet areas (Figure 3. middle and right pictures)

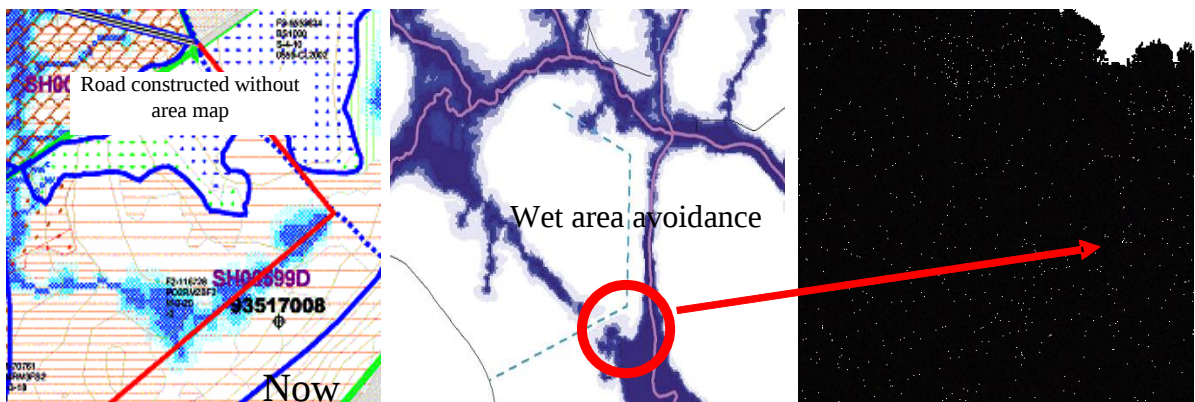


Figure 4.3 A composite illustrating wet-areas mapping applications: left picture shows an existing road that lines up with an unmapped but field-confirmed wet area; middle and right picture show how a road was constructed to pass through an un-mapped saddle point while staying clear of a mapped stream (line feature within in dark blue shading are mapped streams) (Meng etc. 2006).

The Assessment Tool

This module consists of the calculations of road profile, summarization of road length in different wet areas, and culvert information for an existing or planned road. The module results provide users with critical evaluations of existing and planned road segments.

1. Road Profile

Depending on DEM resolutions, in ArcView GIS, road line features are converted into raster data and then disintegrated into a number of point features. The distance from every point on the road segment to the start point is calculated and the elevations of every point are determined from the DEM. Then a road profile is plotted on a graph with coordinates of Elevation vs. Road distance (Figure 4.4).

2. Road length in wet area

Using the depth-to-surface water map, this module can calculate the length of road segments through wet soil areas for an existing or a planned road. By determining the ratio of road length in wet area over total length, foresters are able to determine how well the road is designed.

3. Culvert evaluation

This module can suggest alternatives for existing culverts on a particular road segment. Specific method for this function has been introduced by Sweet (2005), while a new tool (ForCulverts) was developed to locate and evaluate culverts on forest roads.

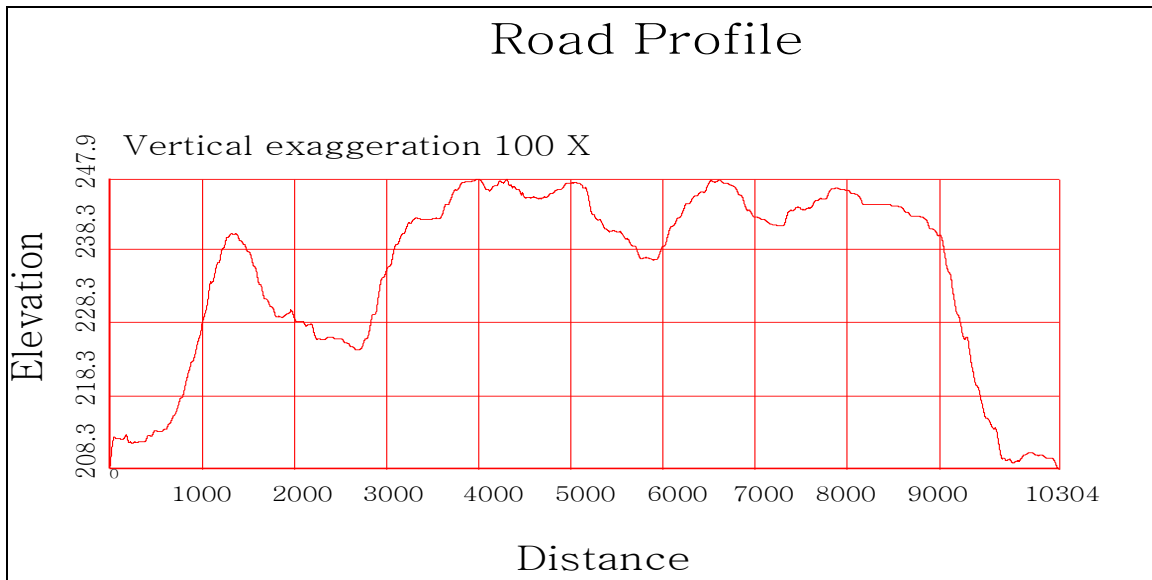


Figure 4.4 Profile for existing road segment.

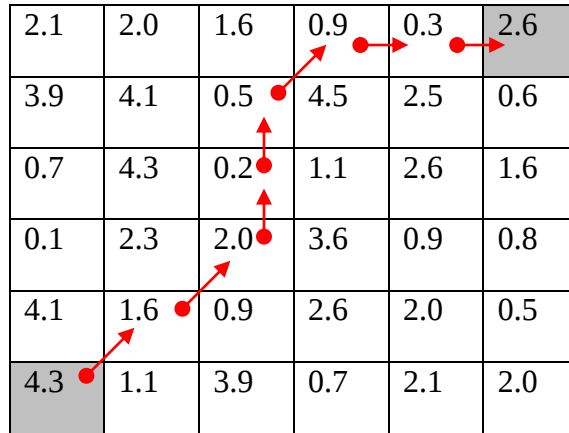
The Optimum Route Tool

1. Algorithm

This module focuses on defining the optimum route between two given points. Essentially, the method to find the optimum route is the “Shortest Route” problem in Management Science (Anderson, 1997), although, in this case, the “shortest” route may not be the one of the shortest length. For raster data applied in ArcView GIS, the path lines and nodes on “Shortest Route” graphs are replaced by grids. As stated in the tutorial of ArcView GIS, creating an accumulative minimum-distance grid using graph theory can be viewed as an attempt to identify the lowest value cell and add it to an output list (ESRI, 2001).

In this module, a raster-based “resistance map” is developed to judge the negative impacts on environment and economics for every point through which the planned road will pass. The number assigned to each cell in the raster-based resistance map shows the extent of negative impacts on road construction. According to the number value in the

resistance map, a series of “easiest” cells are selected and connected, and an optimum route is created (Figure 4.5).



Source cell: the user defined points which need to be connected

Figure 4.5 an illustration of resistance map. The assumed numbers, referring negative impacts on road construction, are accumulated along with any possible connection routes between given two points, and finally the optimum routes is determined.

2. Model structure

The processing procedures are shown in Figure 4.6.

The inputs of **ForRoutes** include mapped water flows, DEM, culvert distribution map and user-defined points that need to be connected. Then, in ArcView GIS, four data layers are created, which are raster-based maps of depth-to-surface water, slope, road length and culverts. For each of these data layers, a punishment function is designed to simulate the extent of resistance. The result produced by each punishment function is made into a raster-based sub-resistance map.

The last step of data processing is to overlay all sub-resistance maps to obtain the final resistance map and apply the “Shortest Route” algorithm to connect given points. The optimization is theoretically accomplished by conquering difficulties from designed

punishment functions. However, all the designed punishment functions in this paper are as yet unverified by field data. Output from **ForRoutes** is a 3D-line feature representing the optimum route connection.

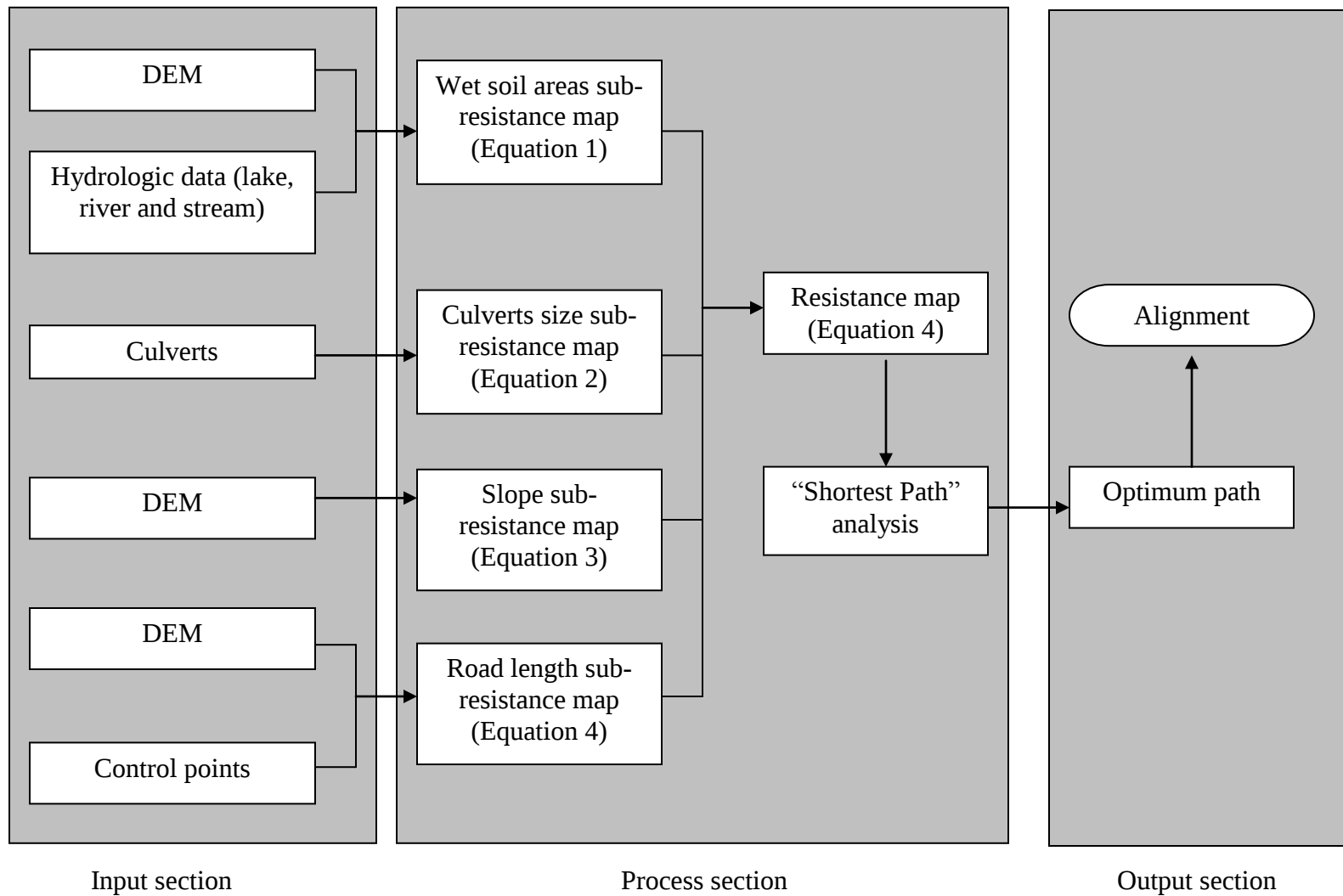


Figure 4.6 Model structure of the optimum path design.

3. Impact evaluations

Wet-area and depth-to-water index. Equation [1] is designed to index the negative impact of wet areas on forest road construction (wet-areas penalty).

$$[1] \quad F_{(w)} = \frac{k_{wet} k_{d2w}}{w}$$

where:

$F_{(w)}$ = potential difficulty of constructing a road segment on a pixel water at or close to the surface, in relation to the depth-to-water index w
 w = depth-to- water index (in m)
 k_{wet} = 0 if $w > 1$, otherwise 0.25; k_{d2w} = 0 if $w = 0$, otherwise 1

The output from this equation is presented in Figure 4.7. The entire raster-based wet-area map is evaluated cell by cell by way of Equation [1].

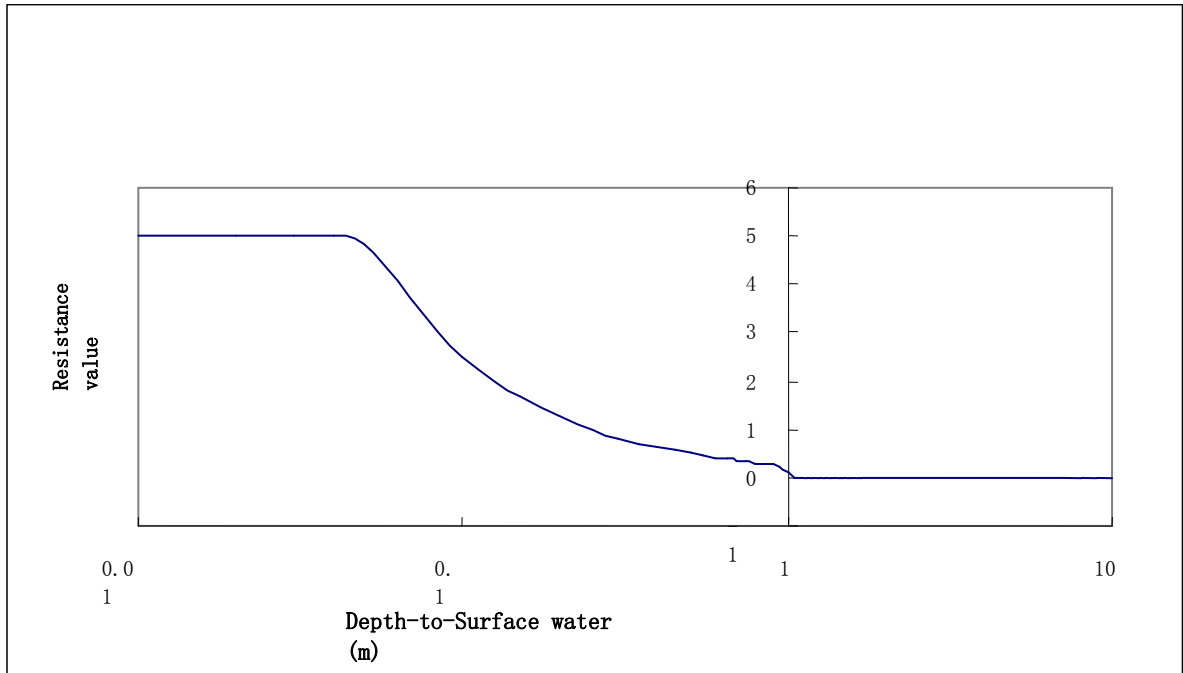


Figure 4.7. The trend line from equation [1] shows the relationship between resistance output and depth-to-surface water. By default in ForRoutes, the wet area is defined as 1-meter depth- to- surface water area (a parameter that can be changed). As illustrated in above diagram, non-wet area (right side of Y-axis) does not have any impacts on forest road construction; the impacts (left side of Y-axis) increase with decrement of depth-to-surface water in wet area and then remain constant.

Culvert requirements and size. In Chapter 2, a map of potential culvert locations and culvert size has been developed based on DEM and local climate normals. Equation [2] determines the relative weight (penalty) of negative culvert impacts on forest road construction for each pixel:

$$[2] \quad F_{(c)} = 2 + (K_{culvert} + K_{maxc})C$$

where:

$F_{(c)}$ = potential culvert penalty

C = diameter of culvert (in m)

$K_{culvert}$ = culvert penalty coefficient (0.1 by default or user defined)

K_{maxc} = 0.7 when $C \leq 2.8$ m, else 1.2. This value also sets up that the program default maximum culvert diameter.

For example, culvert diameters need to be as large as 2.8 m when the drainage area above the culvert is 81 ha with a maximum rainfall intensity of 60 mm/hour, which has been the peak value in Atlantic Canada for at least 50 years. For watersheds with drainage area greater than 81 ha, circular culverts are not sufficient to accommodate the peak flow that would be generated from that rainfall intensity. Instead, bridges or other open-channel structures need to be installed.

Slopes. The raster- based slope is also derived from the DEM by way of ArcView GIS map operations. Equation [3] is used to evaluate and track negative slope impacts on forest roads construction, i.e.,

$$[3] \quad F_{(S)} = K_{slope} \arctan(S)$$

where:

$F_{(S)}$ = the cell value on the weighting map of slope

S = percentage of slope

K_{slope} is an indicator of the maximum acceptable gradient of the road, = 0 when $S \leq 12\%$, otherwise 1; if the cell has a slope value less than 12%, it is considered

to make no contribution to the resistance map; otherwise the resistance value is an inverse function of the tangent of the slope.

Road Length. In DEM, the total length of the road was calculated by counting the number of number of cells that the route passes through, thus the effect of road length on road construction costs is simply given by

$$[4] \quad F_{(L)} = P_c$$

where P_c is used as an index to track the penalty of road construction cost with increasing road length. The default value of this index is set at 1.

Final resistance map. The final road resistance map is produced by adding the sub-resistances (indices) as follows:

$$[5] \quad road_resistance_index = F_{(w)} + F_{(c)} + F_{(s)} + F_{(L)}$$

Results

Applying equations [1] and [2] to every flow-channel pixel generated a map for potential culvert size as illustrated in Figure 4.8 for a section within the Black Brook Watershed (BBW) area.

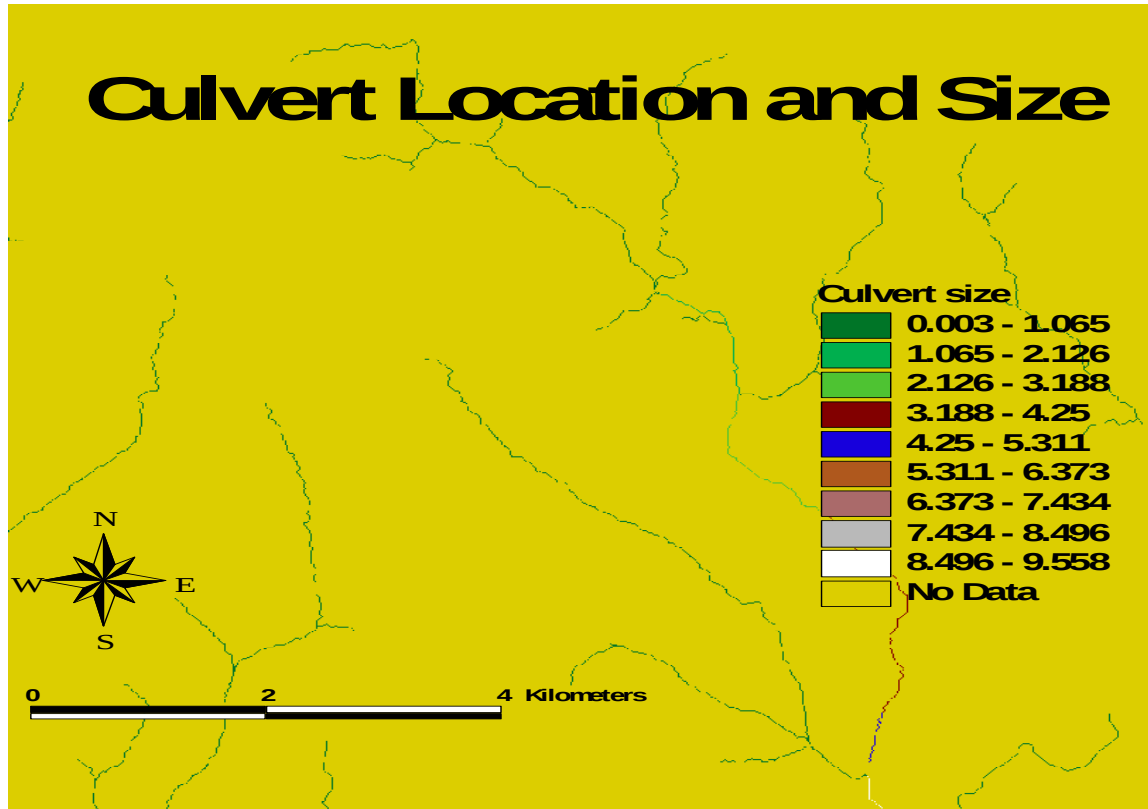


Figure 4.8 This map shows the locations and sizes of potential culverts. This map consists of a collection of colored points and when a planned road intersects one of these points, a culvert needs to be installed, with the size shown in the map legend. Blank area refers where culverts are not needed.

For the stream network map in Figure 4.8, each flow-channel pixel has a value the suggested culvert size at the location of the pixel, according to Equations [1] and [2]. Then, the wet-areas map with its depth-to-water index from 0 to 2 m (Figure 4.9) and the corresponding $F_{(w)}$ map are overlaid. The program then searches for and displays routes with the minimum cumulative pixel values for $F_{(w)} + F_{(c)}$ by, e.g.:

1. using the ForRoutes default parameters (green line in Figure 4.9, with a low penalty factor for culvert placement) and
2. increasing the culvert penalty coefficient $K_{culvert}$ in Equation [2] from 0.1 to 0.15 by 50% (blue line in Figure 4.9 – 4.12).

As shown, the green line intersects two flow channels and the associated wet areas, whereas the blue route by-passes these channels and areas altogether, but at the cost of a longer road length.

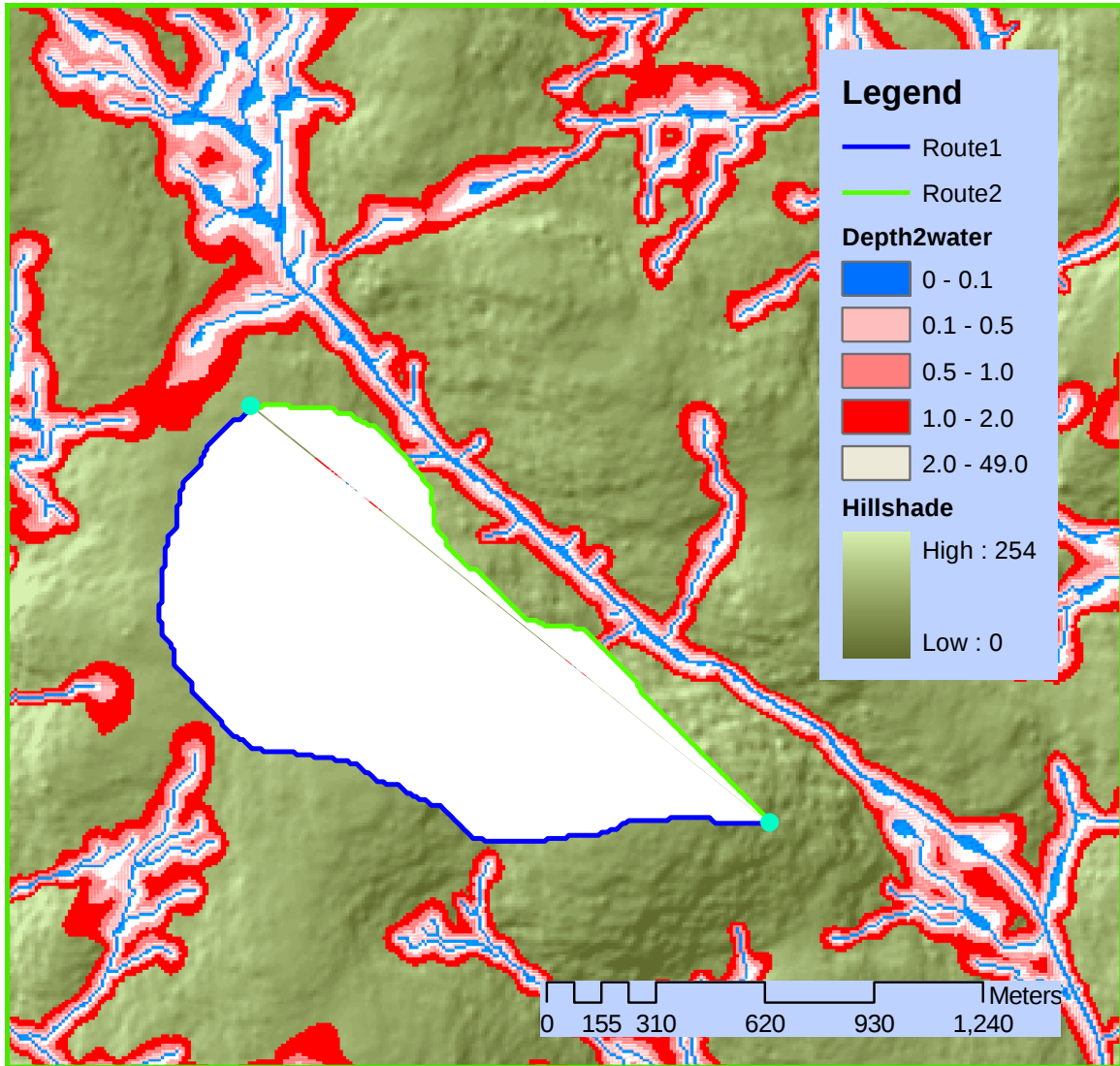


Figure 4.9 Preliminary routes are selected based on resistance values produced by depth-to- Surface water map and culverts map. The background is depth-to- water map and legend shows red and pink area on the map is under 1 m depth-to-surface water area. First roadway with a steam crossing (green line) is selected based on model default setup and the second bypassing wet area (blue line) is produced by increasing k value in Equation [2].

Adding the additional penalty for road slope and length generates two additional routes, namely

3. the green line in Figure 4.13 , with a high slope penalty
4. the yellow line in Figure 4.13, 4.11, 4.12, with a low slope penalty

The entire penalty setting for the 4 routes in Figure 4.13 are summarized in Table 4.1.

The proposed width of these roads is 4 meters.

Table 4.1 A comparison of model suggested roadway alternatives

Route color/ number	Parameter change from default	Road length in wet areas (m)	Number of culverts	The steepest slope	Total earthmoving efforts (m3)	Total Length (m)
Blue/1	$K_{culvert} + 50\%$	0	0	0.104	2306	3334
Green/2	$K_{slope} + 50\%$	115.3	2	0.046	2134	2429.8
Yellow/3	$K_{culvert} + 50\%$	0	0	0.076	3456	3565
Red/4	$K_{slope} + 50\%$	109.7	2	0.084	3863	2649.6

The other parameters used for this calculation are model default values, namely

$$\begin{aligned}
 k_{wet} &= 0.25 && \text{Equation [1]} \\
 k_{d2w} &= 1 && \text{Equation [1]} \\
 K_{slope} &= 1 && \text{Equation [3]} \\
 K_{culvert} &= 0.1 && \text{Equation [2]} \\
 K_{maxc} &= 0.7 && \text{Equation [2]}
 \end{aligned}$$

The components of total earthmoving efforts (the sixth column in Table 4.1), are earthworks for filling up wet areas, cut and fill earthworks for profile and cross sections.

This tool defines that areas with a value of depth to surface water less than 1 meter (could be changed, Figure 4.10) are wet areas, and such areas will contribute difficulties to road infrastructures. A filling-up earth moving effort thus will be accumulated and recorded by the program when a road is planned to pass wet areas, namely

$$\text{Earthwork Volume} = \text{Road Width} \times \text{Road Length} \times \text{Fill-up Depth}$$

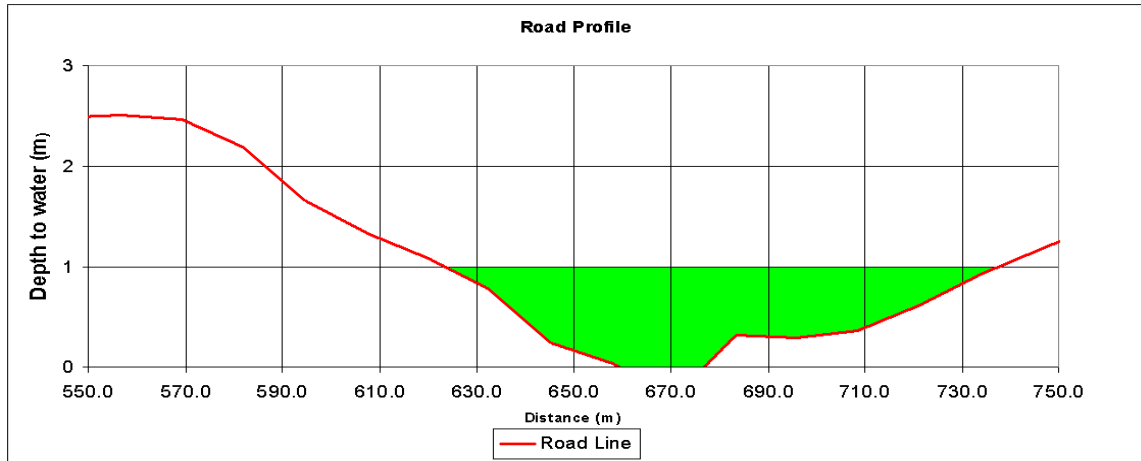


Figure 4.10 A filling up cross section in wet areas.

To calculate cut and fill earthworks for road profile, the road line will be firstly smoothed by averaging adjacent five elevation values along the roadway on DEM. Then, the earth moving efforts can be calculated by integrating the green shadow areas (Figure 4.11), namely

$$\text{Earthwork Volume} = \text{Road Width} \times \int_0^{\text{length}} (\text{curve1} - \text{curve2})$$

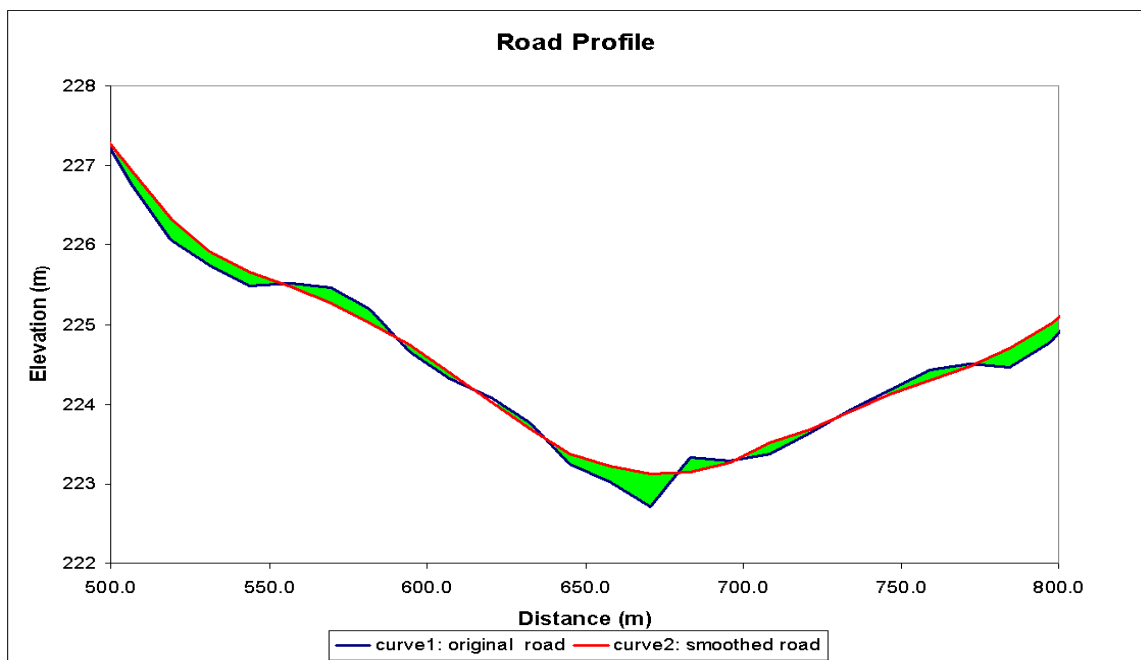


Figure 4.11 Calculating the earth moving effort by smoothing road profile.

The earth moving efforts for cross sections focus on the excavating actions on the side slopes (Figure 4.12). The side slopes are calculated based on topographic characters in DEM. By default, 4 meters is set as the road width and 120° is set as back slope.

According to trigonometric function,

$$\text{the cross area} = \text{Road Width}^2 \times (0.375\text{Cos}(\text{side slope}) - 0.215\text{Sin}(\text{side slope}))$$

and

$$\text{Earthwork Volume} = \text{Road Width}^2 \times (0.375\text{Cos}(\text{side slope}) - 0.215\text{Sin}(\text{side slope})) \times \text{Road length}$$

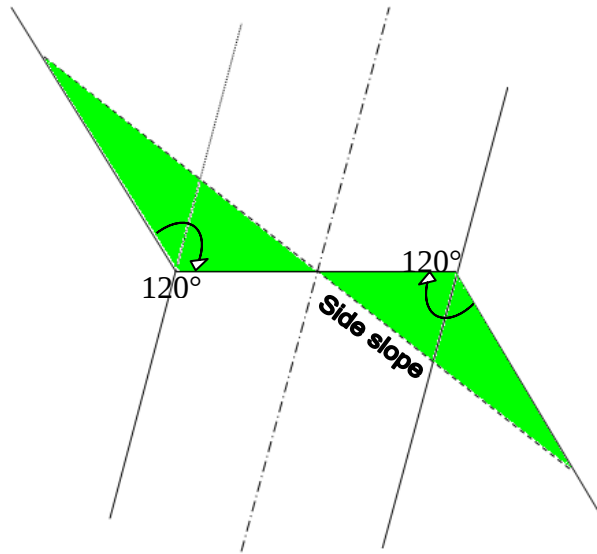


Figure 4.12 The cut and fill area for cross sections on the side slope.

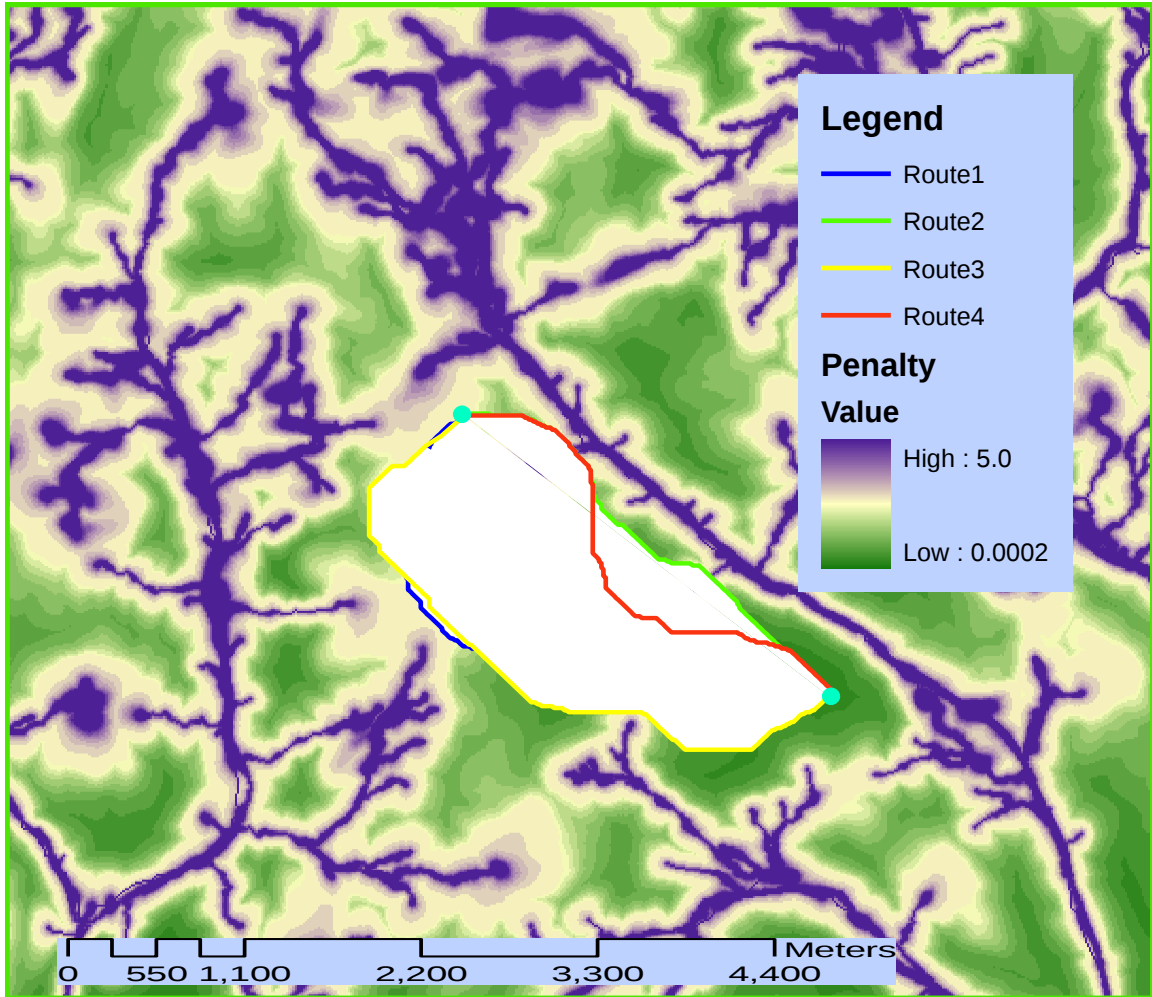


Figure 4.13 The background is the final weighting map made from 4 overlaid factors, which are slope, culvert, wet areas and length, with deeper greenness referring bigger penalty value. The optimum paths are selected based on final resistance values denoted on the final weighting map. First roadway with a stream crossing (green line) is selected based on model default setup and the second bypassing wet area (blue line) is produced by increasing $K_{culvert}$ value in Equation [2]. The result presented in Figure 9 is overlaid here for comparison. The difference on locations between stream-crossing lines (red and green lines) and between wet areas avoiding lines (yellow and blue lines in Figure 4.14) is because the slope resistance has limited and affected optimum choice of route selection.

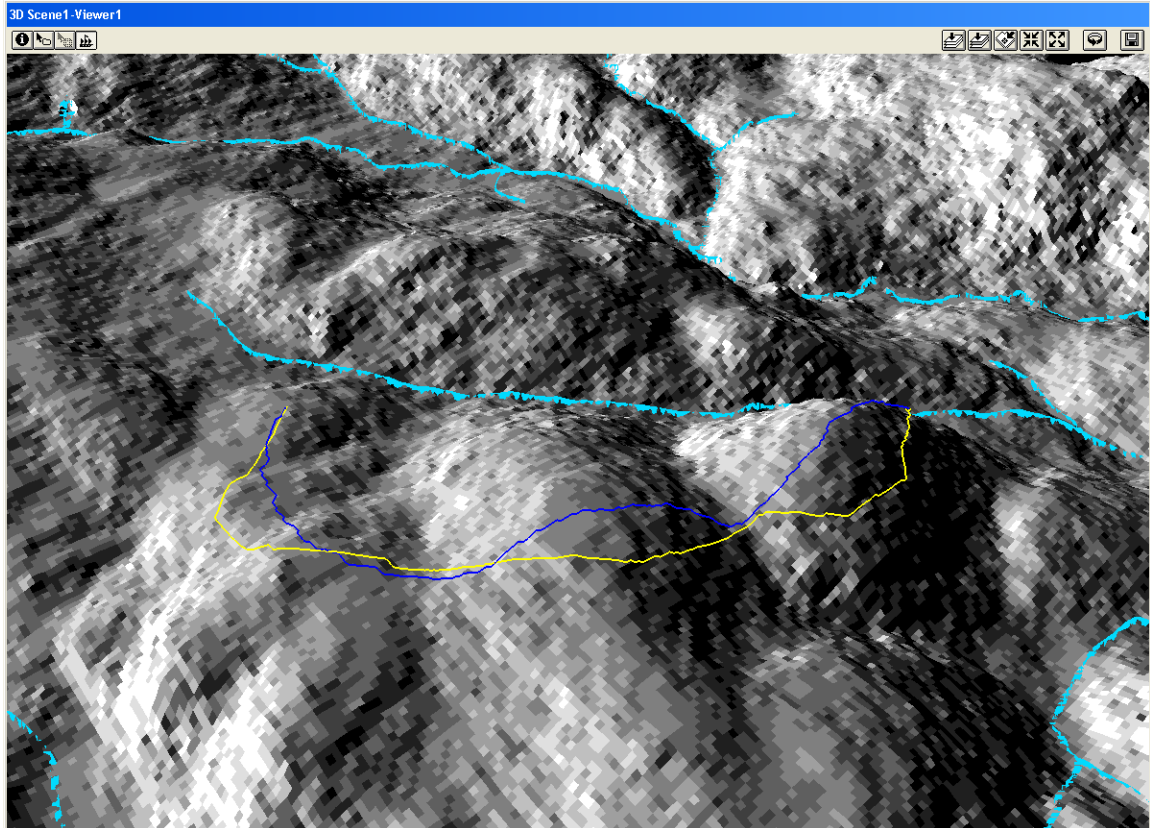


Figure 4.14 A 3-D show of two wet-area avoiding lines (route 1 and 3 in Figure 4.13). Base on the current parameter setup, model produced route1 (blue) was projected along the hill ridge. Both lines do not have any river crossing points. Due to different slope coefficients, the blue line was projected on the hill ridge and the route2 (yellow) was mostly projected along the same normal contour. The yellow one with gentler slope is longer than blue one.

The accurate, detailed slope and slant information (Figure 4.15) was derived from selected routes and DEM in ArcView. This avails further comparison of these alternatives (all line in Figure 4.13) and decision making. The top graph in Figure 4.15 is plotted with x-axis of Distance and y-axis of Elevation. As a result, the profiles of given routes were presented on this Figure. From this graph, we can see the shortest green line has a fluctuated slope change. The yellow line has an optimum slope fluctuation but is the longest one. The bottom graph is plotted with x-axis of Distance and y-axis of Slant. Besides being used to calculating earth movement for road construction, this graph is

useful to predict potential soil erosion locations. Comparing Figure 4.14 with Figure 4.15, the blue line (along with the ridge) has much gentler slants than the yellow line (in the half way of down slope of the hill). The earth movement for blue line is also much less than yellow line (2306:3456 m³, Table 4.1). All these information are valuable for forest road identification and the functions designed in this section can assist forest engineers in planning roads.

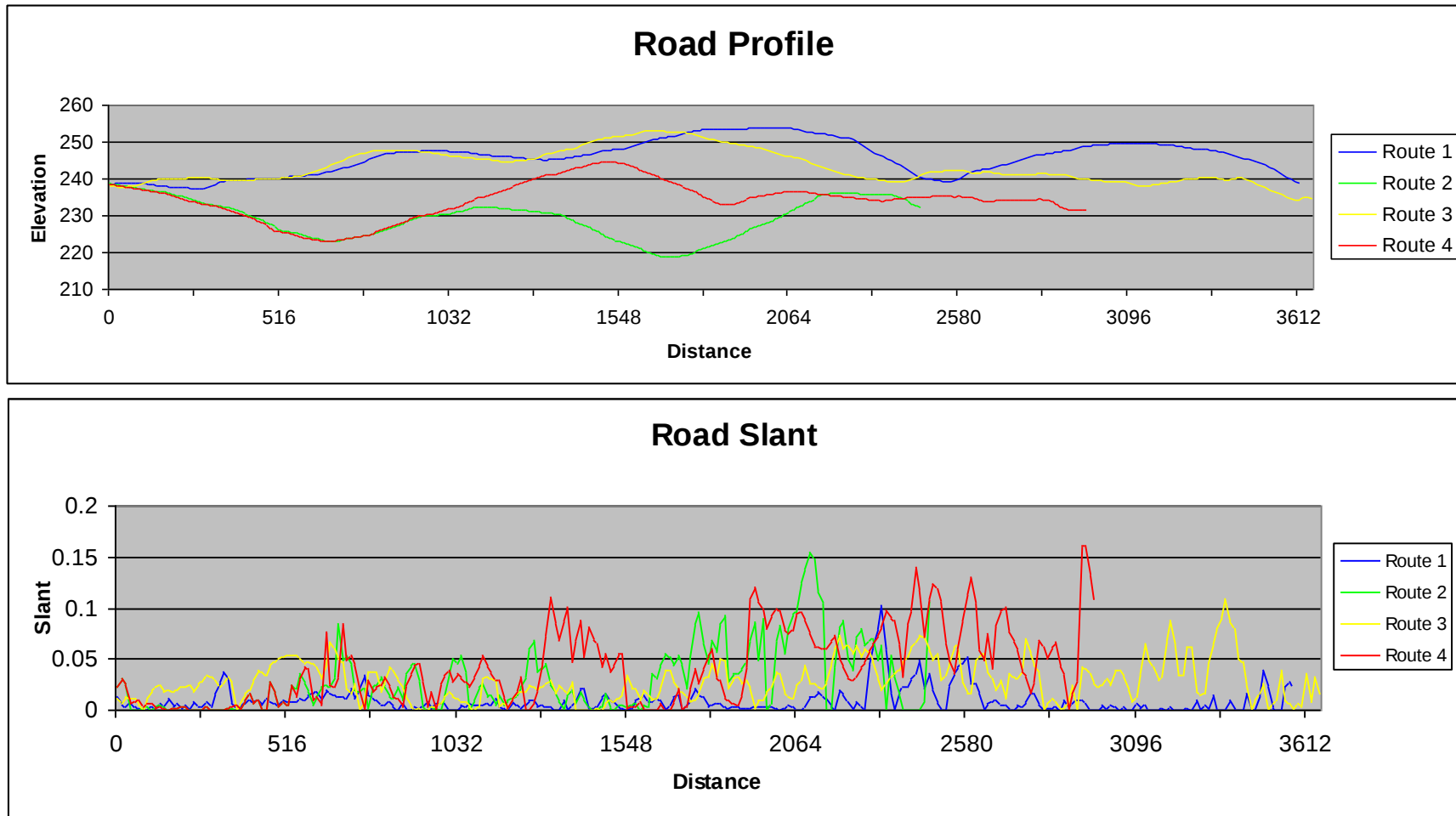


Figure 4.15 The top graph takes into account elevations and gradients between adjacent grid points along the selected route. The bottom graph shows the gradients between relevant grid points across the route line in every about 10 meters, which could be used to estimate the slant situations.

Curve Smoothing

Test results indicated the route selected by this model generally had many sharp angles (Figure 4.16), which result in an unnecessary increase of the road length. Thus the curve smoothing function is designed to smooth the selected route so as to calculate a more reasonable length and practical shape.

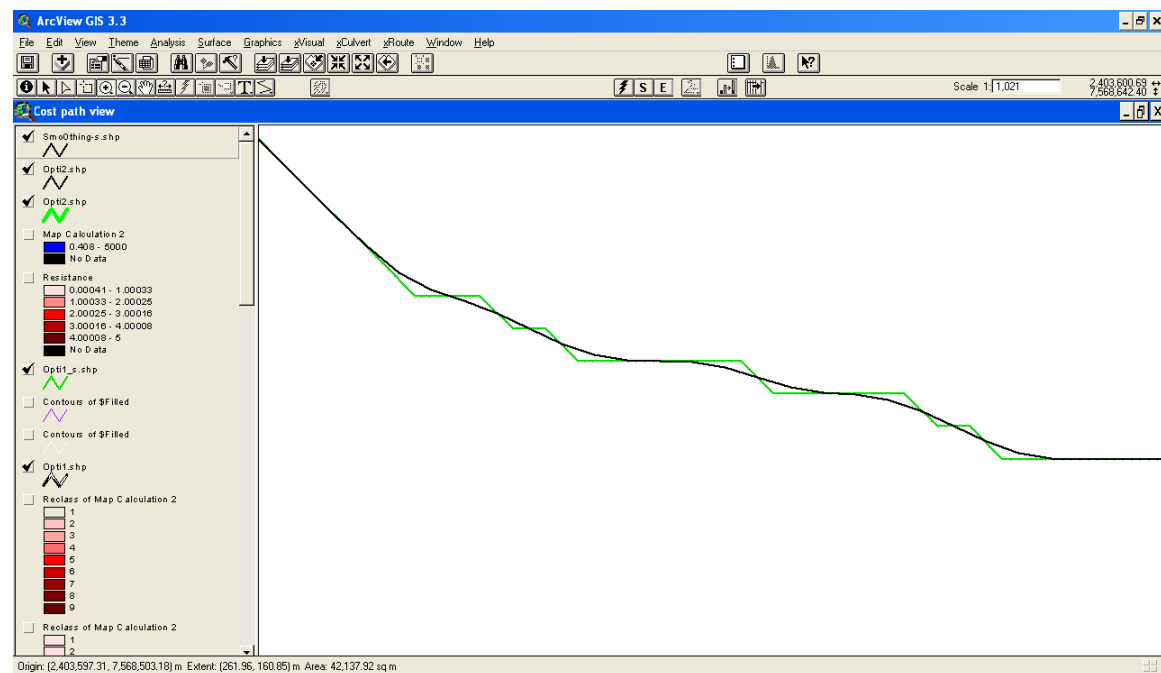


Figure 4.16 Comparing two routes, we can see that the curves have replaced the sharp angles and the route has been smoothed.

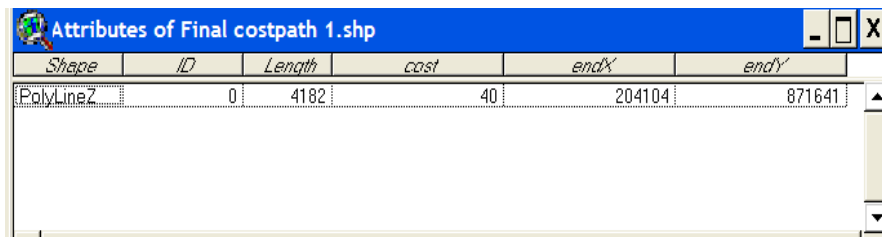
Calculations

Base on the result above, some useful calculations that are related to road construction and assessment can be done by this model.

Road Length

After the sharp angles are removed, the newly produced route is good for length calculating, and the result is shown in an attached table of selected route

(Figure 4.17). The information in this table also includes the ending location of selected route.



Shape	ID	Length	cost	endX	endY
PolyLineZ	0	4182	40	204104	871641

Figure 4.17 Road information from the smooth line

Culverts

The following calculations will give culvert information for a newly selected route. As discussed in Chapter 3, ForCulverts will evaluate the drainage areas around the selected route and suggest the appropriate locations and sizes for culverts.

Figure 4.18 shows the culverts suggested by FOROAD for a newly designed road layout. A table including the exact location (x-y value) and size (feet) of culvert shows on Figure 4.19. Additional information in this table includes the drainage area (acres) and rainfall concentrating time (hours) for every culvert.

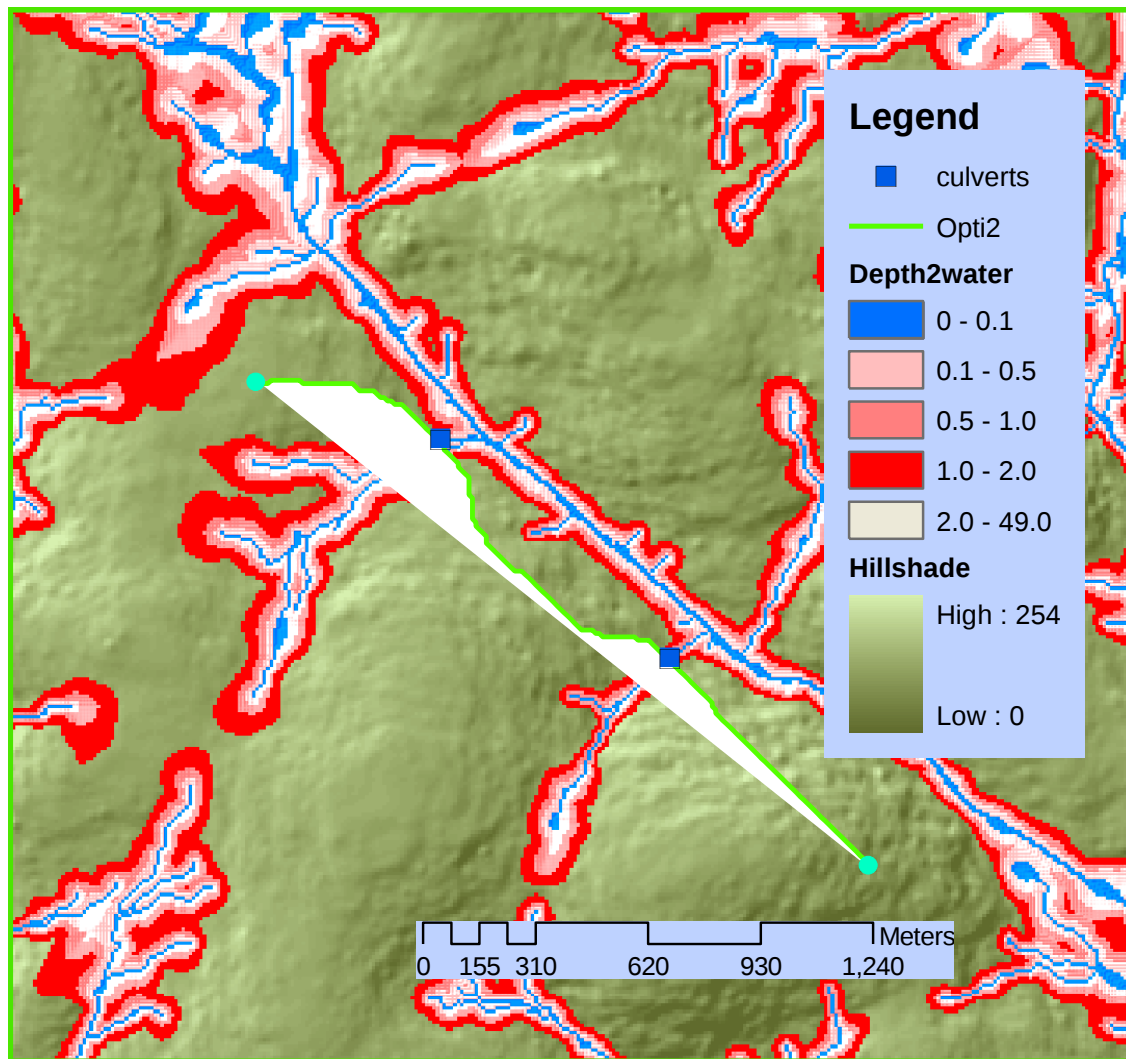


Figure 4.18 the locations of culverts on selected route

Attributes of Culverts8905.shp							
Shape	Pointid	Grid_code	X	Y	DrainageArea	Diameter	ConcentrationHour
Point	1	1.0000	2403839.3	7569867.8	75.510	0.78	0.57
Point	2	1.0000	2404489.3	7569237.8	35.460	0.45	0.40

Figure 4.19 Culverts and drainage related information for selected route

SUMMARY

Forest roads will inevitably intersect streams and natural drainage areas at some points. As stated, soil moisture and water drainage has major impacts on forest roads and vice versa. In Atlantic Canada, where annual precipitation is substantially higher than potential evapotranspiration, the landscape is characterized by gentle

slopes and vast wet soil areas. Interactions between water and roads must be concerned through all stages of forest roads planning, construction and maintenance. Properly locating forest roads will minimize the damage to both forest area and forest roads during water/road interactions.

Topographically-based wet-areas mapping is beginning to prove useful to identify areas of risk with regard to the construction and maintenance of forest roads (Meng etc. 2006). The availability of this map makes **ForRoutes** possible, which is a new forest road design tool. Integrated with slope and drainage calculation, **ForRoutes** could be used to:

- find the optimum route between two given points, and
- assess existing forest roads, and
- visualize the distribution of wet soil area.

LITERATURE CITED

- Anderson, D.R., Sweeney, D.J. and Williams, A.S. 1997. An Introduction to Management Science: quantitative approaches to decision making. Chapter 9: Network Models. NY: Minneapolis/ST. Paul
- Douglas, R.A. 1999. Forest Roads/ resource access roads Delivery. University of New Brunswick, Pp. 49
- ESRI. 2001. Arcview 3.3 software help files. Copyright 1999-2002: Environmental System Research Institute Inc.
- Fowler, C. 2003. Modeling Watershed Responses to Agriculture and Forestry in the potato belt of northwestern New Brunswick, Master thesis, University of New Brunswick
- Mellerowicz, K.T., Rees, H.W., Chow, T.L. and Ghanem, I. 1993. Soils of the Black Brook Watershed, St. Andre Parish, Madawaska County, New Brunswick. New

Brunswick Dept of Agriculture and Agriculture Canada, Fredericton, N.B., Canada.

Meng, F., Castonguay, M., Ogilvie, J., Murphy, P. and Arp, P. 2006. Developing a GIS-based flow channel and wet-areas mapping framework for precision forestry planning. IUFRO Precision Forestry Conference 2006

Stone, R. P. and Hilborn, D. 2000. Universal Soil Loss Equation (USLE), Factsheet, Agriculture and Rural, Ontario

Elliot, W. J., Ballerini, M. and Hall, D.E. 2003. Simplified methods for evaluating road prism stability. Transportation Research Record, Transportation Research Board, National Academies of Science, Washington, DC. 1819(2):95-100.

CHAPTER 5

FORROAD AND OTHER EXISTING ROAD DESIGN TOOLS

INTRODUCTION

Forest management involves strategic, tactical, and operational planning levels (Boyland, 2003). At the strategic level, planners deal with long-term harvest-block selection and scheduling, and assessing harvest-block access based on existing and potentially needed forest road network extensions and alterations. At the tactical level, planners detail and cost specific harvest modes and access requirements, e.g., which harvest methods to use and which specific road segments to upgrade, maintain and/or construct. At the operational level, planners require spatially explicit information as to what, where, when, how and with whom to conduct field operations, and to build, maintain or decommission specified road segments. This Chapter focuses on strategic, tactical, and operational forest road design considerations, and reviews a number of models used for forest management planning in general, and for road layout specifically (Table 5.1 and 5.2). The entries in these tables either deal with road network design (FORPLAN, PLANEX, SNAP, UWTHPS, NETWORK 2000&NETWORK 2001), or single road design tools (RoadEng, Roadpac, pegger, Novapoint, Routes). Among single road design tools, PEGGER and ROUTES are useful for preliminary route selection among alternative routes and road profiles. RoadEng, Roadpac and Novapoint deal with specific road design and related road construction specifications.

Table 5.1 Existing Software Packages for Road Design

Application	Models	Function
Strategic level	FORPLAN; PLANEX	Road network optimization design
Tactical level	SNAP; UWTHPS; NETWORK 2000, 2001	
Operational level	RoadEng; ROADPAC; TRACER; ROUTES; PEGGER; Novapoint	Road segment optimal design

Table 5.2 A summarization of existing transportation design tool

Models	Intended use and objectives	Algorithm	Programming Code/ Framework	Inputs	Outputs	Road segment selection platform
FORPLAN (Johnson, 1986; Weintraub etc., 1994)	Forest management planning: harvest-block scheduling and road access	Linear programming, coupled with mixed-integer and goal programming techniques	FORTRAN 77	Data layers: Forest inventory (stand type, habitat type, and location); road network. Objective functions: optimizing goals concerning timber, wildlife and other resources under sustainability and socio-economic constraints	Cut-block allocations with corresponding road network spanning, say, 30 to 100 years; minimizing overall forest management costs	Polygon network based on cut-block allocations (Figure 5.2a)
PLANEX (Epstein, 2001; Sessions, 2006)	Same as above	Linear programming mixed with heuristic decision making	FORTRAN 77	Same as above, with coarse-gridded topography data used for harvest method and road selection	Road network delineation; minimizing overall forest management costs	DTM square grid (Figure 5.2b)
SNAP (Sessions, 1988)	Forest management planning: balancing environmental impacts with operational costs	Same as above	C, C ⁺⁺	Same as above	Same as above	DTM square grid 20 m (Figure 5.2c)

Models	Intended use and objectives	Algorithm	Programming Code/ Framework	Inputs	Outputs	Road segment selection platform
UWTHPS (Schiess, 1995)	Same as above	Similar to above	C, C ⁺⁺ and ARC INFO	Same as above	Same as above	DTM Square grid
NETWORK 2000 (Chung, 2000, 2001; Sessions, 2006)	Same as above.	Similar to above	VC ⁺⁺ ; AutoCAD	Data layer: Harvest plan map; round trip haul cost, road cost, road length Objective function: minimize road system length or other road segment attributes	Road access network with construction attributes specified, including costs	DTM Square grid (Figure 5.2d)
ROUTES (Reutebuch, 1988; Tucek and Pacola, 1999)	Develop and evaluate alternative routes for logging; minimize transport costs, including road construction costs	Contour-based road-slope design	BASIC 2.0	Digitalized air photos and DTM	Road segments with attributes such as side slope, profile and skidding distances specified and evaluated	DTM Square grid 40 feet (Figure 5.2e)
Pegger (Rogers, 2005)	Automated road selection process based on design specification	Contour-based road-slope design	Avenue; ArcView	DTM	Road segment and profile selection	DTM grid 1m, 10m (Figure 5.2f)

Models	Intended use and objectives	Algorithm	Programming Code/ Framework	Inputs	Outputs	Road segment selection platform
ROADPAC (Keays, 2007)	Road selection, including design of cross-section and profile	Cut and fill Algorithm	AutoCAD	DTM	Road segment and profile selection; with cut and fill specifications	DTM Square grid
RoadEng (2005)	Same as above	Cut and fill Algorithm	VC ++	DTM and GPS data	Same as above	Traverse information; Profile; Plan; Multi-Plot (Figure 5.2g)
Novapoint (Vianova, 2006)	Same as above	Cut and fill Algorithm	AutoCAD	Topographic data, and AutoCAD drawings	Same as above, with 3D visualizations, and spreadsheet report	DTM grid (Figure 5.2h)

The FOREst PLANning (FORPLAN) software is a strategic forest-level planning and optimization tool developed by the US Forest Service. It uses linear, integer programming, mixed-integer and goal programming techniques to select road segments from a region-wide road network, actual or contemplated. This tool was first developed to provide an analytical structure to capture and portray multiple road-use interactions in determining a range of optimal and potentially feasible choices for road locations (Johnson, 1986). As such, FORPLAN is aimed at making strategic road-network choices rather than designing specific road segments. In practice, FORPLAN output is often counter-intuitive due to many interacting constraints and complexities (Hoekstra, 1986).

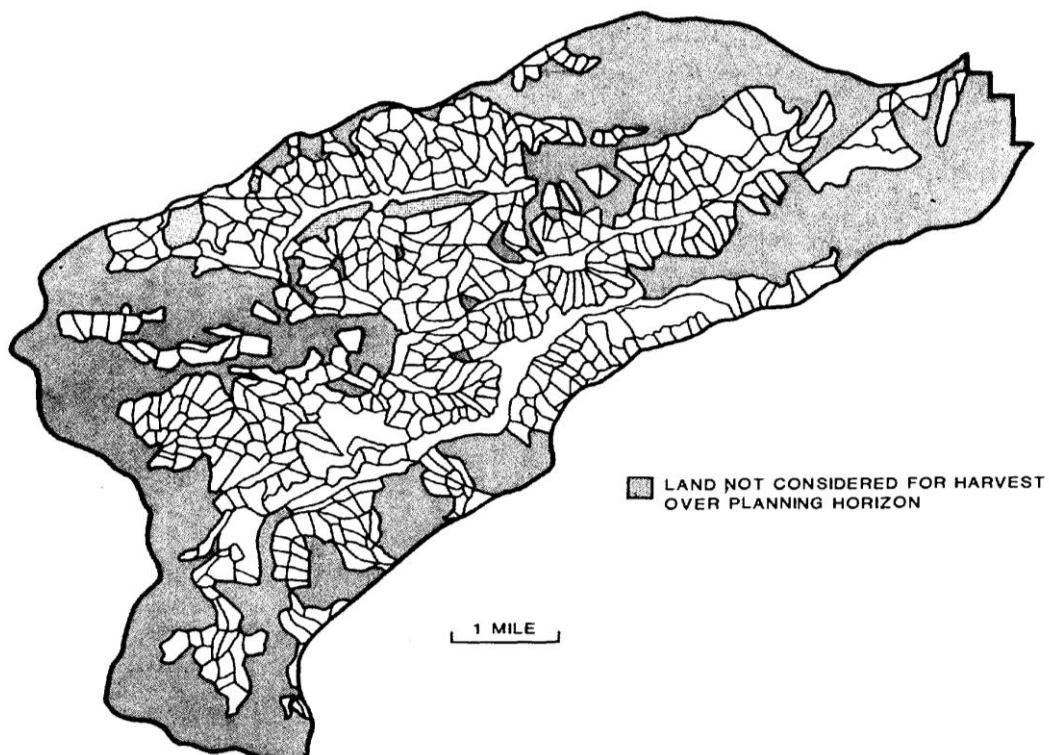


Figure 5.2a FORPLAN: Forest harvest area with polygons representing scheduled harvest blocks, each requiring access roads or trails. FORPLAN automatically devises harvest-block and road segment selection (Sessions, 2006.)

The PLANEX model by Epstein (2001) can be used to solve strategic road networking problems in terms of assessing specific road construction costs in reference to harvest and timber yard locations, all within the context of strategic forest management decision making. This model uses coarse-gridded topographic data to identify and locate individual road segments as needed for the overall transportation network. Anderson and Nelson (2004) used vector data instead of raster data to create “road networks that are suitable for strategic planning” prior to operational planning.

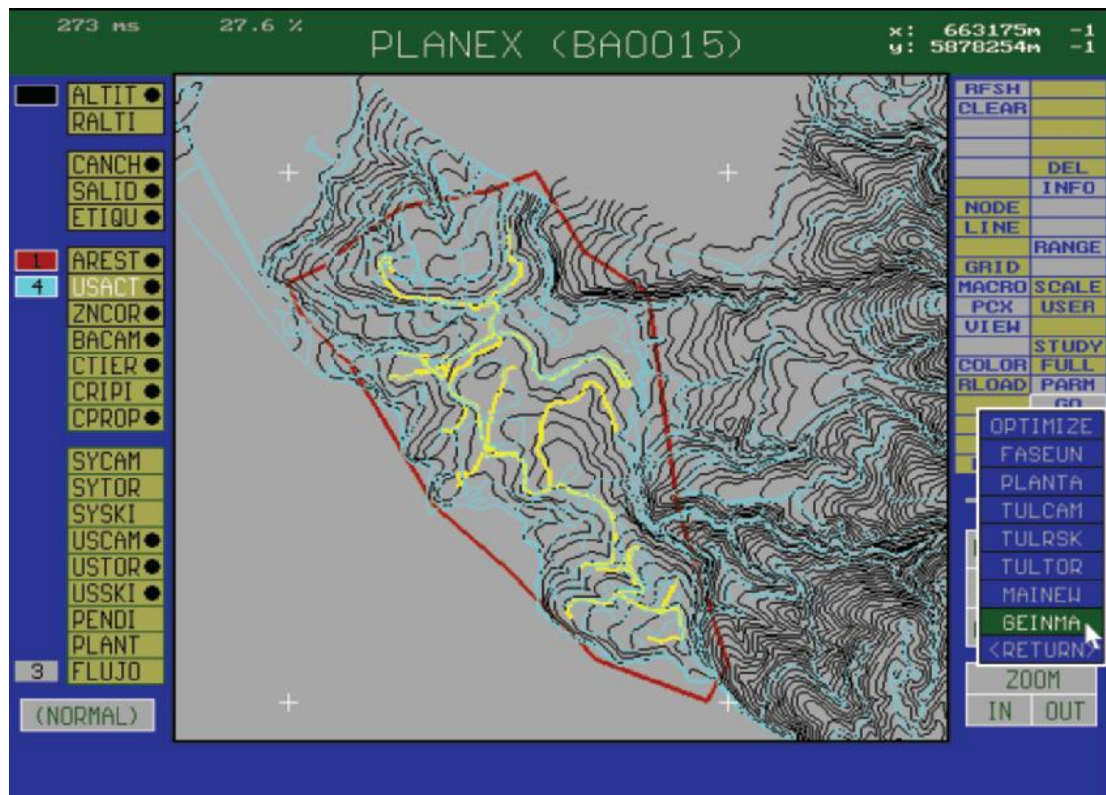


Figure 5.2b PLANEX selects harvest blocks and appropriate harvesting methods according to topography (ground slopes), and sequentially determines landing and road locations while minimizing overall logging costs including road-building and transportation (Sessions, 2006).

The other strategic planning tools that are relevant are:

1. the SNAP and UWTHPS (University of Washington Timber Harvest Planning System) models, which are geospatially explicit tools for operations planning,

specifically designed to assist in forest-harvest transportation planning (Sessions, 1988; Sessions, 1992; Andersson, 2005; Schiess and Lawrence, 1995)

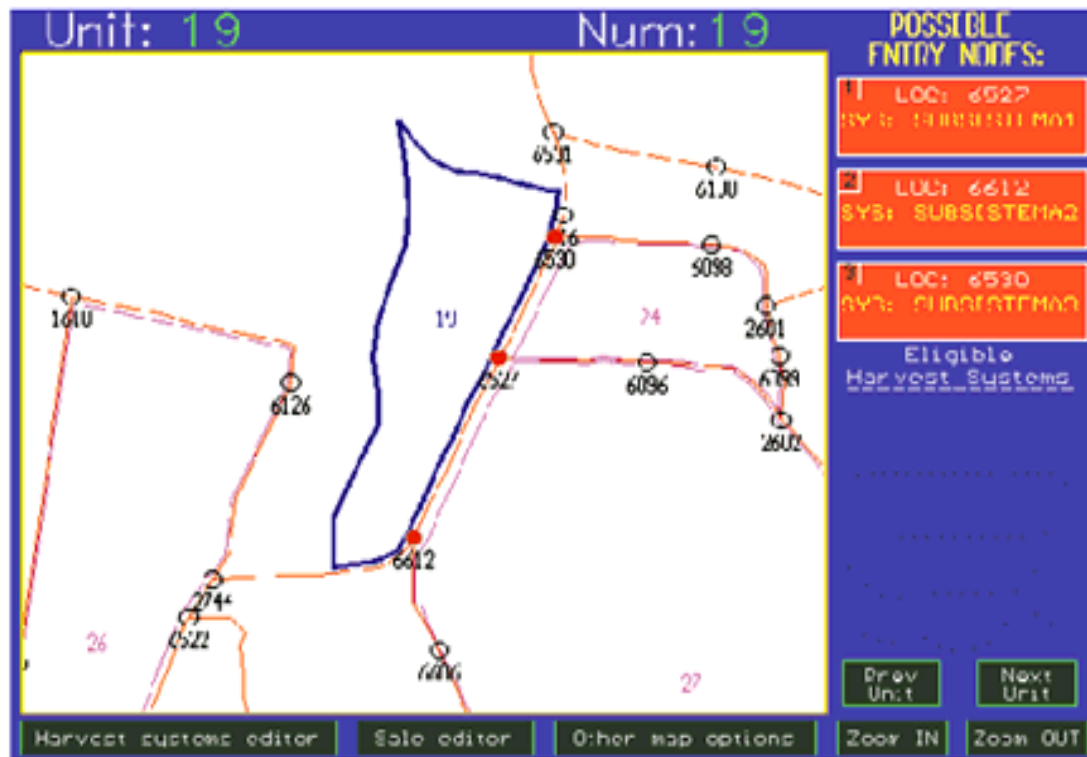


Figure 5.2c SNAP: timber deposit localization, harvest costs, maximum economic extraction distances and road quality and cost parameters were considered to define the technically and economically viable harvest subsystems in each period. The polygons to be harvested were defined along with their respective points of timber deposition (red points), as illustrated in this figure. Observing this example, knots 6512, 6527 and 6530 were selected as possible timber deposit points in polygon 19. (http://www.scielo.br/scielo.php?pid=S0100-67622003000600009&script=sci_arttext)

2. The NETWORK 2000 (Chung and Sessions, 2000) program, which deals with variable and fixed cost optimization within the multi-period transportation context. As such, NETWORK 2000 contains designed algorithms to determine shortest paths, and probabilistic algorithms for determining minimum road construction cost (Woodam and Sessions, 2003). However, rather complex data requirements, lack of site-specific controls, and

time required to work with these software tools have severely limited practical use and further developments (Rogers, 2005).

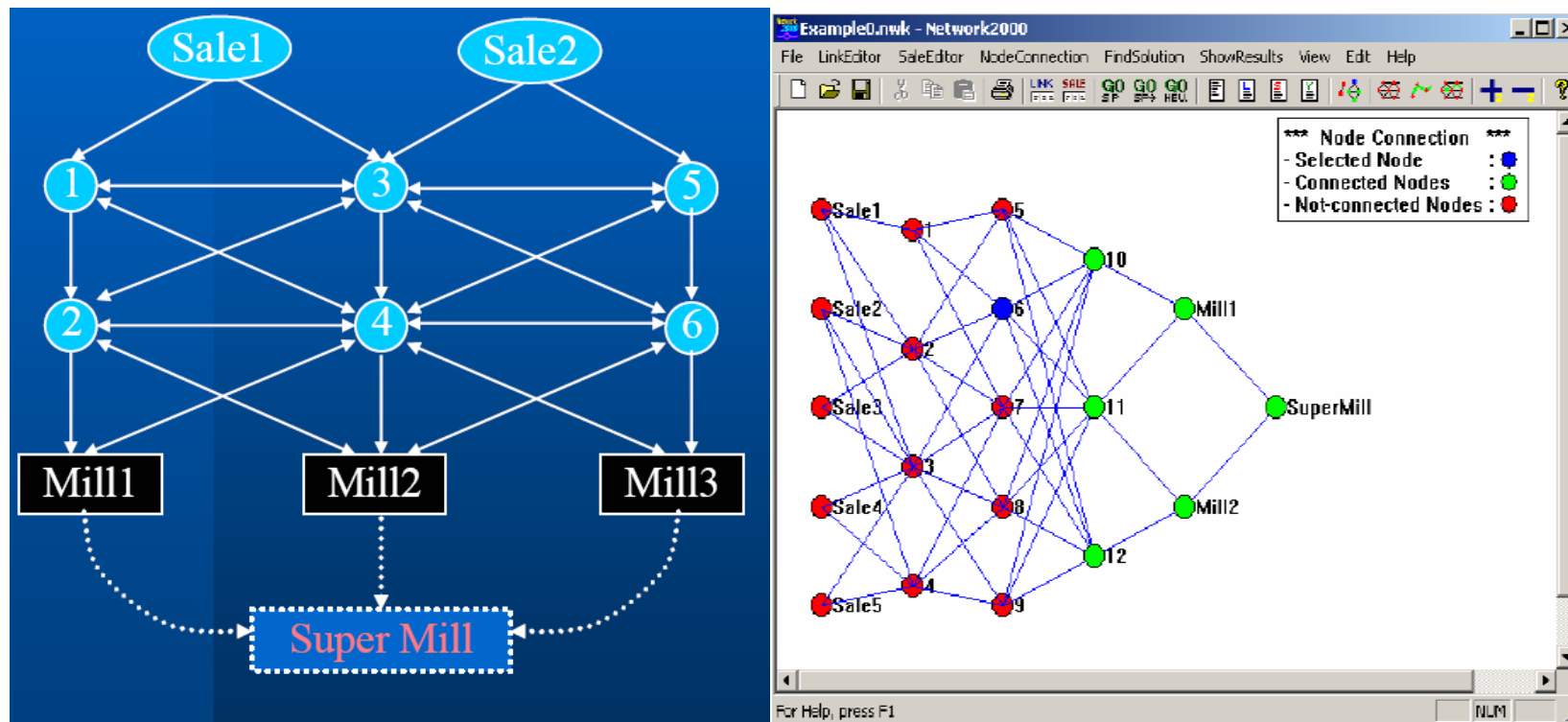


Figure 5.2d NETWORK2000 has been used to determine resource-transportation routes, as well as road locations using the vector representation of alternative road (Sessions, 2006). The model theory is shown in the graph (left): multiple objectives of Network2000 are minimizing costs and maximizing profits. The result shown in the window (right) is a case study based on the constraints: transportation and road costs, open road mileage, sediment yields and environmental hazard probability.

Kobayashi (1973) developed the earliest operationally useful road design tool, which did not have an intuitive or “user-friendly” user interface. The more advanced harvest- and transportation- planning software package called ROUTES was published 15 years later by Reutebuch (1988), with focus on selecting particular road segments according to required grade and side slopes specifications. However, heavy reliance on air-photo digitization, an unusual programming code (HP 9000) and general lack of a graphical user interface (GUI) discouraged general adoption and use (Rogers, 2005).

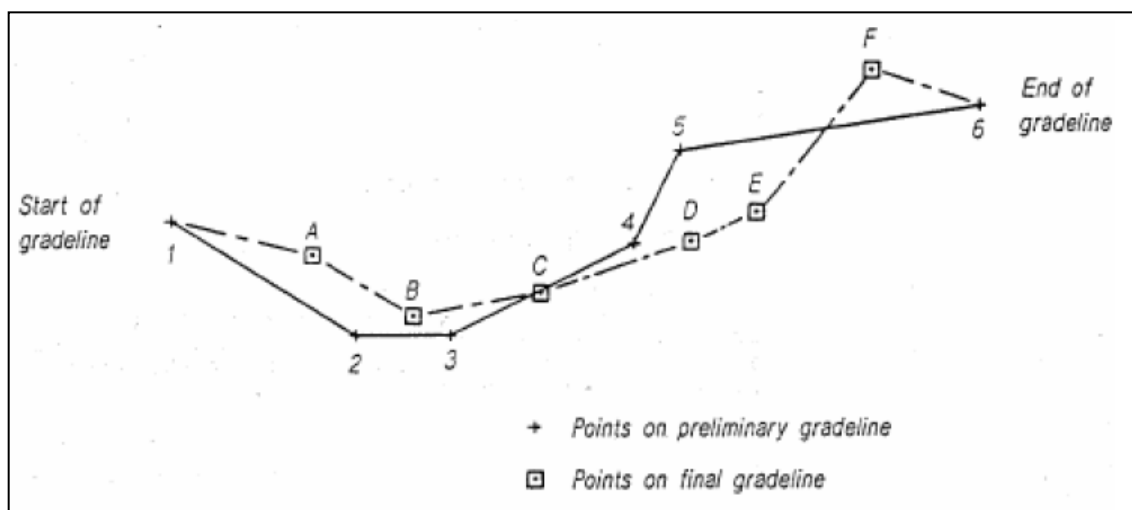


Figure 5.2e ROUTES assists planners in finding and evaluating route locations based on digital terrain data. It takes into account gradients between relevant grid points and other topographical features which makes them applicable in mountainous environments. As illustrated in figure above, the final gradeline is superimposed on the preliminary gradeline. The routine always forces the final gradeline to terminate at the last point on the preliminary gradeline, so a cut or fill is often needed to get on-grade at the end of the route (Tucek, 1999; Reutebuch, S. 1988.).

In contrast, the more recently developed “Novapoint Road Professional” program has become a popular tool for road and highway design. This program uses a CAD engine and a well designed graphic interface, including a “design wizard” for

automatically determining potential routes based on specific design standards, and for quick 3D visualizations (Vianova, 2006). Its “super-elevation” features automatically tilt the roadway to help offset vehicular curve-induced centripetal forces.

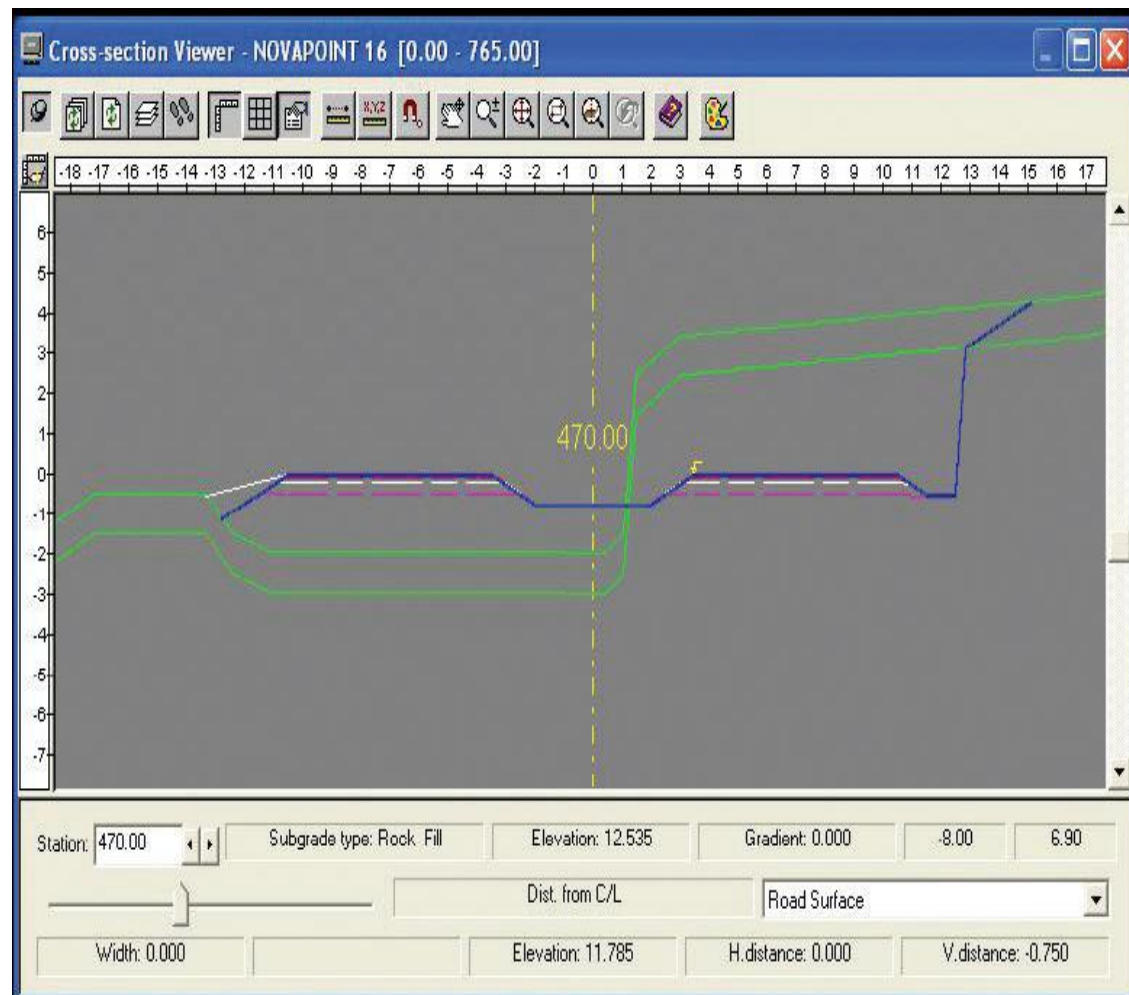


Figure 5.2h. Novapoint: the cross-section viewer gives designed information on the measurement of width, distances, slope, elevation and coordinates.
(<http://www.novapoint.com/novus/upload/file/products/Road%20professional.pdf>)

RoadEng is a versatile software package that performs tasks, from road design to mapping (2005). It provides an editor for detailed culvert design, and reports on suggested culvert size, length, skew, and gradient. Both Novapoint and RoadEng are powerful road design tools but require much training to address all design components accurately and effectively.



Figure 5.2g The output of RoadEng includes four kinds of road design information: plan, Profile, Cross Section, Data and Multi-Plot. This figure shows profile information produced by RoadEng program. (http://www.softree-espanol.com/products/location_forest_pro.html).

At the operational level of forest management, Pegger focuses on the automation of projection process of road design and can give different route alternatives on the basis of variable slope tolerances. Pegger is practical and can find alternatives quickly on contour maps; however, its inability to handle other environmental and economic considerations limits its utility.

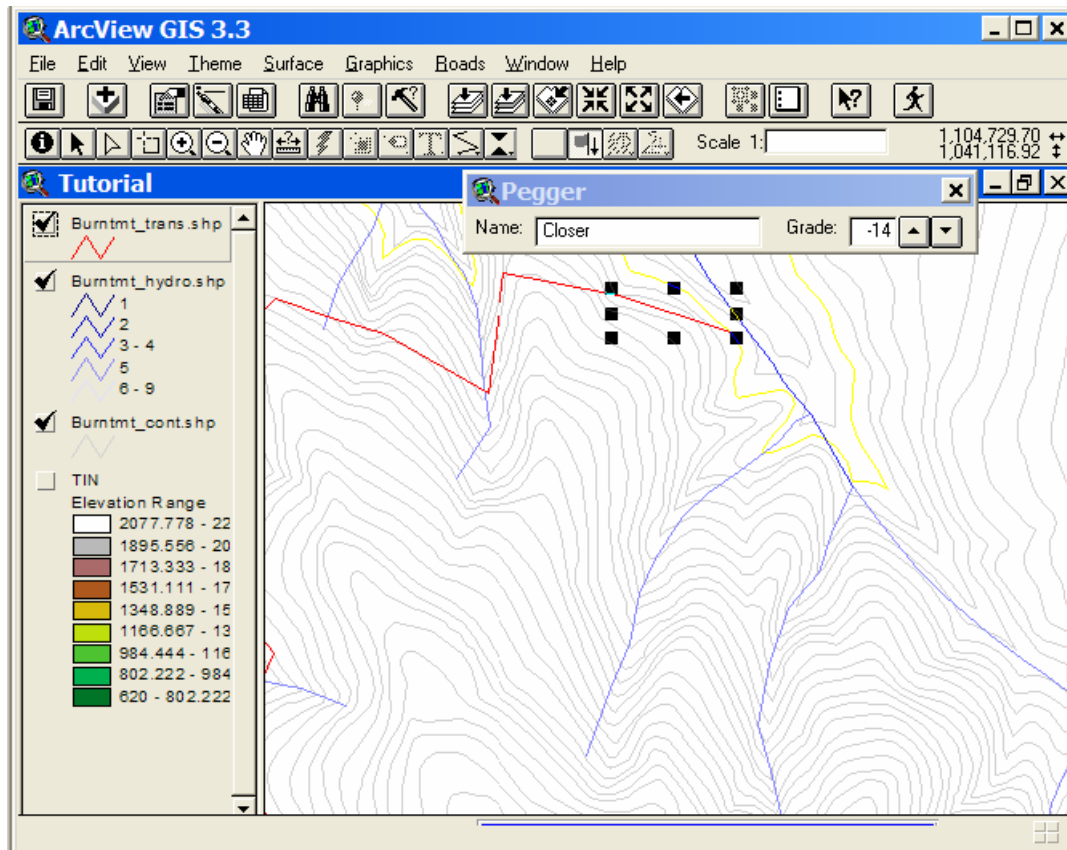


Figure 5.2f PEGGER was developed to automate initial forest road design and creates and connects road segments on the contour map (Rogers, 2005). While the maximum allowed slope is set up by user, selected contour lines will be connected by the shortest road segment (red line) with the slope no more than preset value.

ROUTES is another DTM-based route selection tool, but one that takes in to account harvesting skidding distance in its analysis of alternative routes. (Tucek, J. and Pacola, E. 1999).

In mountainous areas (e.g. western North America), topography is key factor in road design, and profile and cross sections must be well designed. On the other hand, eastern Canada consists of many flat or gently rolling areas, requiring that road designs focus on water conditions rather than topography. However, the previously described operational-level road planning tools do not address concerns related to wet area and water conditions.

One of the primary issue in road design is the selection of road segments, in relation to terrain conditions as these vary with topography and substrate type, from rocky to bogs, from flat to mountainous, from wet to dry, from frictional (coarse soil texture) to cohesive (fine soil texture), and from frequent potential road-stream crossing to few or none. Each of the above-mentioned road-design tools deals with these variations differently. For example, FORPAN ignores these variations altogether, assuming access to be equally costly everywhere. Topographic variations and related road profile variations including cut and fill requirements are considered with slope index. Road-stream crossings and related requirements for culverts and bridges are part of culvert index.

The innovations of mapping topographically-derived soil wetness indices (Murphy, 2007), and the mapping of hydrological risks in general, as defined by DEM-derived flow-channels, wet-areas and likely depth-to-water variations across the landscape at high 5 to 10 m resolution, have opened up further avenues of further improving the road design process. It is suggested that these tools could become components of advanced road designed tools such as RoadEng and Novapoint. In the mean time, there is a need to explore ways and means by which soil wetness and

hydrological risk maps can further assist in automating and optimizing forest operations including road segment and trail selection as part of timber resource access planning, through digital means. The tool (FOROAD) proposed in this report was developed to address this need, by delineating the hydrological risks (potential and actual flow channels, wet areas, and likely depth to water) across the forest terrain based on DEM modeling. The FOROAD software is developed within the ArcView GIS frame, and has several modules, such as those related to sizing and locating culverts, and road layout. Model inputs such as DEM weather data and road selection parameters such as maximum gradient can be input or modified by the user.

LITERATURE CITED

- (2005). Documentation for Forestry Supported Software Helpdesk> Documentation> Forestry applications >RoadEng.
<http://wwwdata.forestry.oregonstate.edu/helpdesk/docs/>
- Akay, A.E., Karas, I.R., Sessions, J. Yukesel, A., Bozali, N., and Gundogan, R., 2004. Using High-Resolution Digital Elevation Model for Computer-Aided Forest Road Design. Geo-Imagery Bridging Continents, Istanbul, Turkey. The International Society for Photogrammetry and Remote Sensing.
- Anderson, A.E. and Nelson, J.D. 2004. Projecting vector-based road networks with a shortest path algorithm. Can J For. Res. 34: 1444-1457.
- Anderson, D. 2005. Approaches to Integrated Strategic/Tactical Forest Planning Faculty of forest science. Licentiate thesis. Department of Forest Resource Management and Geomatics, Swedish University of Agricultural Sciences.
- Boyland, M. 2003. Hierarchical Planning in Forestry. ATLAS/SIMFOR Project Technical Report. Vancouver, B.C., University of British Columbia, Canada
- Chung, W. and Sessions, J. 2001. Designing a forest road network using heuristic optimization techniques. Proceedings of the 24th Meeting of the Council on Forest Engineering, July 15-19, Snowshoe, West Virginia.
- Epstein, R., Sessions, J. et al. 2001. PLANEX: A system to identify landing locations and access. 11th International Mountain Logging and Pacific Northwest Skyline Symposium, Seattle, Washington, USA.
- Hoekstra, T.W., Dyer, A.A. and Le Master, D.C. 1986. FORPLAN: an evaluation of a forest planning tool: proceedings of a symposium. Denver, Colorado, US. Pp. 49.
- Johnson, K.N. 1986. FORPLAN version 1: an overview. Washington, D. C.: USDA Forest Service, Land Management Planning Systems Section, Washington, USA
- Keays software. 2007. The Integrated Solution for Civil Engineering and Surveying Software
<http://www.keays.com.au/root.php?path=field&m0=1&m1=2>
- Kobayashi, H. 1973. A Study on Automatic Processing in Forest Road Design Mainly Concerning the Earthwork, Government Forest Experimental Station. Tokyo, Japan.
- Murphy P.N.C., Ogilvie J, Connor K & Arp P.A. 2007. Mapping wetlands: a comparison of two different approaches for New Brunswick, Canada. Wetlands, **in print**.

- Reutebuch, S. 1988. ROUTES: A Computer Program for Preliminary Route Location. Pacific Northwest Research Station, Forest Service, Department of Agriculture, U.S.
- Rogers, L. 2005. Automating Contour-Based Route Projection for Preliminary Forest Road Designs using GIS. Master thesis, University of Washington.
- Schiess, P. and O'Brien, L.M. 1995. The application of Geographic Information Systems to Forest Operations: The Integration of Cable Setting Design into GIS. Second Brazilian Symposium on Timber Harvesting and Forest Transportation, Salvador, Bahia, Brazil.
- Sessions, J., Akay, A. and Murphy, G. 2006. Chapter 5: Road and Harvesting Planning and Operations. Computer Applications in Sustainable Forest Management. Springer. Netherlands, Pp. 83-99
- Sessions, J. and Sessions, J.B. 1988. SNAP - A Scheduling and Network Analysis Program for Tactical Harvest Planning. International Mountain Logging and Pacific Northwest Skyline Symposium. Oregon, USA.
- Sessions, J. and Sessions J.B., 1992. Post new forestry tools for tactical forest planning. Western Wildlands Winter: 39-44.
- Tucek, J. and Pacola, E. 1999. Algorithms for Skidding Distance Modeling on a Raster Digital Terrain Model. International Journal of Forest Engineering, January, 1999, vol.10 no.1
- Vianova Systems AS Company. 2006. Introduction of products: The most complete infrastructure design system.
[Http://www.novapoint.com/index.asp?id=23421&it=13](http://www.novapoint.com/index.asp?id=23421&it=13)
- Weintraub, A., Jones, G., Magendzo, A., Meacham, M. and Kirby, M. 1994. A Heuristic System to Solve Mixed Integer Forest Planning Models. Operations Research, Nov. - Dec., 1994, Vol. 42, No. 6. Pp. 1010-1024.
- Woodam, C. and Sessions, J. 2003. Documentation for Forestry Supported Software: What is Network 2000? Oregon State University, United States of American
<http://wwwdata.forestry.oregonstate.edu/helpdesk/docs/application.php?id=54>

CHAPTER 6

SUMMARY AND DISCUSSIONS

SUMMARY OF THE SOFTWARE

While computerized route location has been applied to forest area for many years, it has never become a popular tool for forest professionals. With FOROAD, the road designer can quickly evaluate route locations within a GIS framework. Specific functions and capabilities incorporated in the two modules of FOROAD include those related to:

- determine the optimum route between two given points;
- find roadway alternatives;
- visualization of the distribution of wet soil areas;
- determining the locations and sizes of culverts required by a particular road layout;
- assessing the locations and sizes of existing culverts on forest roads;
- determining the size of the watershed, dominant vegetation types with the drainage area, and storm flow rate;
- providing an automatic procedure to calculate all stream crossings for a given area, or optionally figure out an outlet for a drainage area manually;
- providing an assessment of sideslope soil erosion potential.

Other capabilities of FOROAD include the ability to create flow accumulation maps and depth-to-surface water maps for land base, and establish road profile graphs.

The tool was designed to work in situations where a minimal investment in design time and cost is warranted. The forest planner can locate a road network in a few hours on a PC applying their own knowledge with FOROAD. The free options of key

parameters give users more confidence in the locating process because the alternatives can be found out easily.

Ease of installation and a simple user interface make the program very user-friendly. Using FOROAD to select feasible preliminary routes can save planners considerable time and reduce the risk of road failures and environmental damage. For an area of 100 km², alternative routes can be identified in a few hours in the office.

Contributions of this Report

- a depth-to-surface water map was applied as an index on the forest road layout
- an extreme rainfall events distribution map was created and integrated with FOROAD and , hence, culvert sizes were calculated accurately
- a practical culvert and watershed tool (ForCulverts) was designed
- water/road interaction was firstly applied as a primary factor to a forest road layout tool
- a computer framework for forest road layout was created. Further study will make this

Limitations of FOROAD

- FOROAD is aimed at supporting the forest road design problem in the context of non-mountainous conditions
- The module for culvert design should not be used to size bridges because the Rational Method was designed for the drainage of small watersheds

CONTINUATION OF THE SOFTWARE

Culverts, depth-to-water and slope information can be used to calculate rough cost estimates related to the construction of each alternative route. Several models have been developed to calculate partial cost of the road construction such as earthwork volumes calculation (Aruga et al., 2003) using the Pappus-based method (Easa, 2003).

Aruga's program (2003) works well when calculating earthwork volumes for curved roadways in mountainous areas. Then it estimates construction and maintenance costs using the USDA Forest Service Region 6 Cost Estimating Guide (1999). However, many uncertainties are inherent in the cost calculation process. Even though data quality can be guaranteed, the diversity of the design standard in different areas makes the road construction costs in wet areas difficult to be figured out.

In this work, a few penalty functions were designed as slope and wet area constraints to find out the optimum route. The parameters related to those equations are key points to alternate users' choice. An attractive objective of the future work is to endow these parameters with practically economic values, such as unit cost and unit price of construction materials and labors. The further work of FOROAD will include a calculation platform based on which a rough estimate of road construction cost can be done. The platform will provide algorithms for earthwork volumes and wet areas conditions, and user-defined cost parameters.

LITERATURE CITED

- Aruga, K., Sessions, J. and Miyata, E.S., 2003 Tabu search optimization of forest road alignments combined with shortest path and cubic Spline
- Easa, S.M., 2003 Estimating Earthwork Volumes of Curved Roadways: Simulation Model. *J. Surveying Engineering* 129(1), Pp. 19-27
- Douglas, R. A. 1999. Forest Roads/ resource access roads Delivery. University of New Brunswick, Pp. 49
- Fowler, C., 2003. Modeling Watershed Responses to Agriculture and Forestry in the potato belt of northwestern New Brunswick, Master thesis, University of New Brunswick

Appendix A- User Tutorial

User Manual

This tutorial is written to help new users install and run the FOROAD model in ArcView. First, load the extension file into extension folder and enable the Spatial Analyst and 3D Analyst Extension together with FOROAD Extension by selecting Extensions under the File menu and selecting 3D, Spatial Analyst and xFOROAD (Fig. 1).

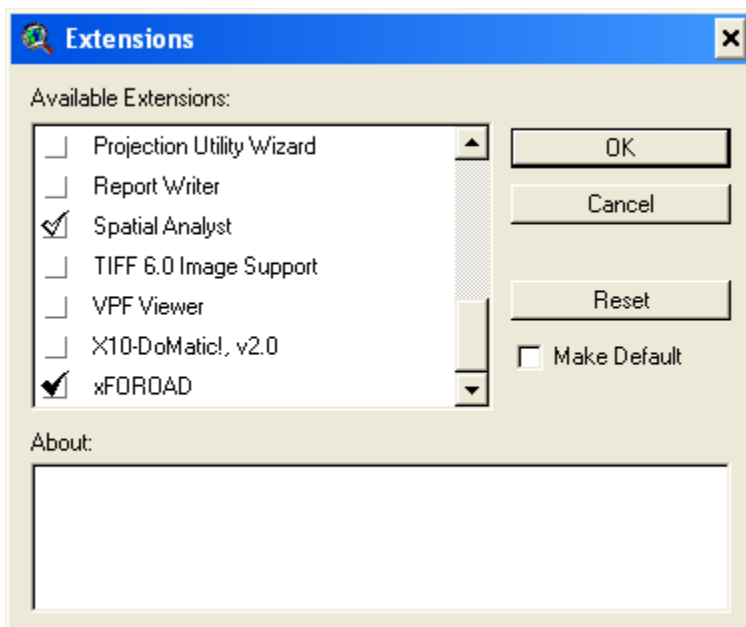


Figure 1

After being activated, the FOROAD model is ready to run. Three dropdown menus (Fig. 2) and 3 buttons will be available on user interface

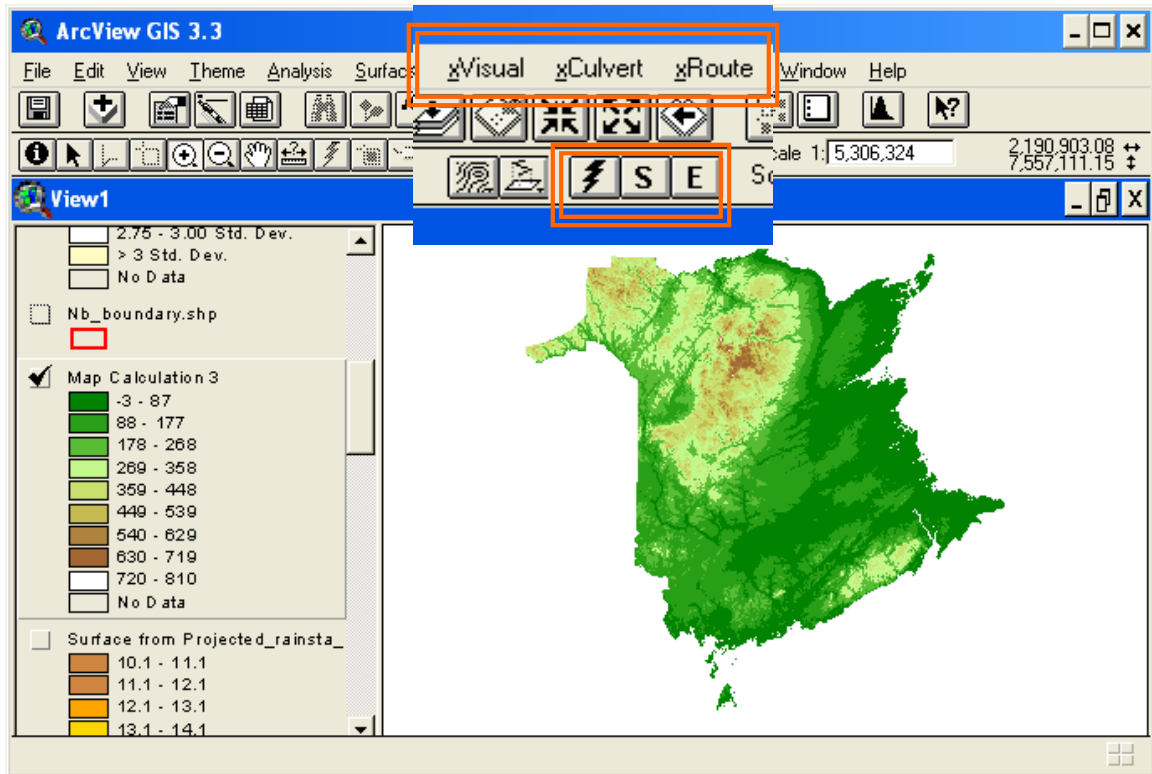


Figure 2

The running procedure of the model is straightforward. Three modules of FOROAD, ForVisuals, ForCulverts and ForRoutes are corresponding to three dropdown menus, xVisual, Xculvert and xRoute. The xVisual menu also includes a graphic user interface of FOROAD. For starting the program, users select “Setup” from dropdown list first (Fig. 3) and a control panel (Fig. 4) will show up. The required input can be loaded and optional parameters can be changed from this panel. Detailed introduction of data input for this GUI can be found in Chapter 2.

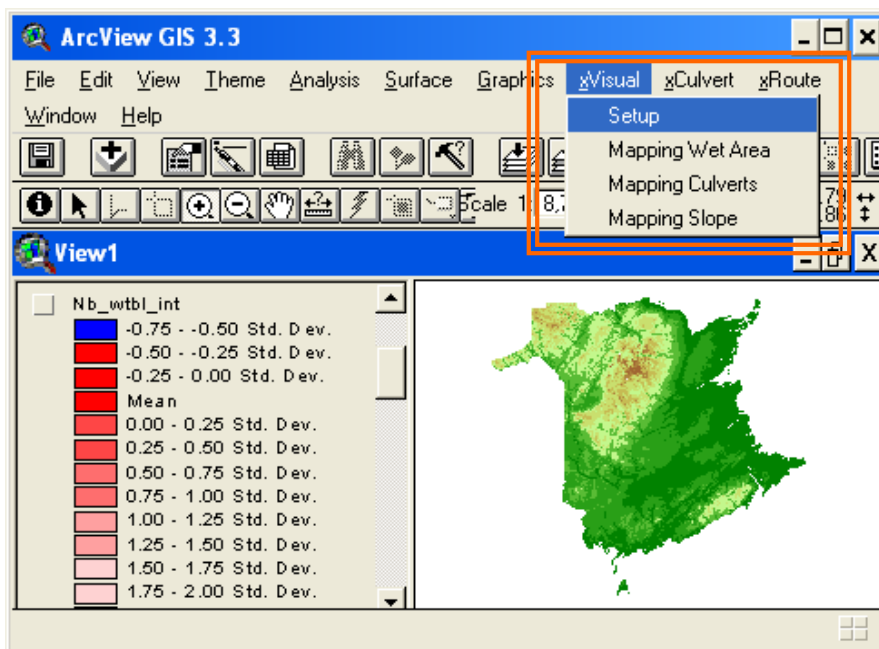


Figure 3

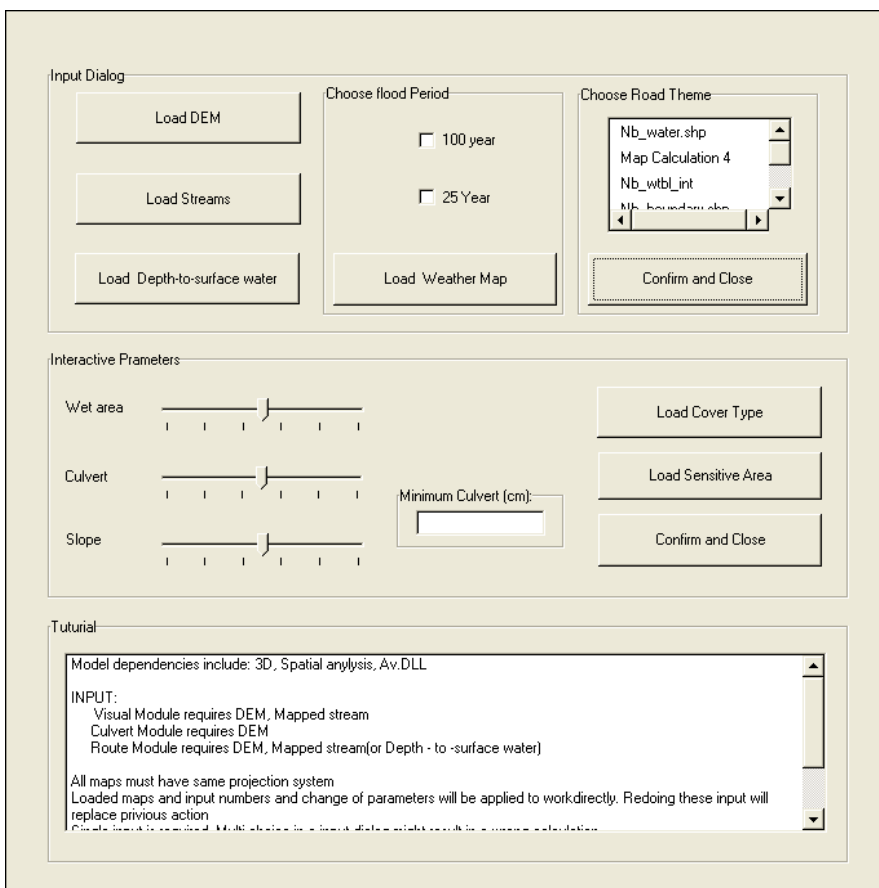


Figure 4

After required data are inputted from the GUI, the menus and buttons are activated. The ForVisuals module provides three visualization maps, which can be made by selecting the function buttons from the dropdown menu (Figure 3). The functions for sizing and locating culverts designed in the ForCulverts module (Chapter 3) can be triggered in the xCulvert menu (Fig. 4) and a button  (Fig. 2). This button is designed for watershed analysis.

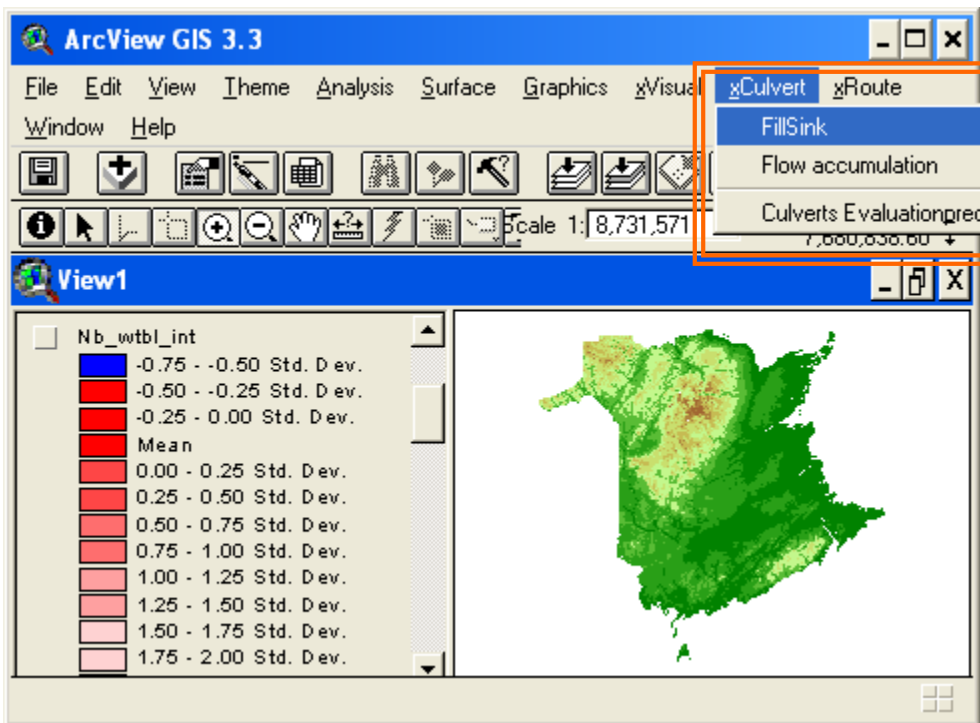


Figure 4

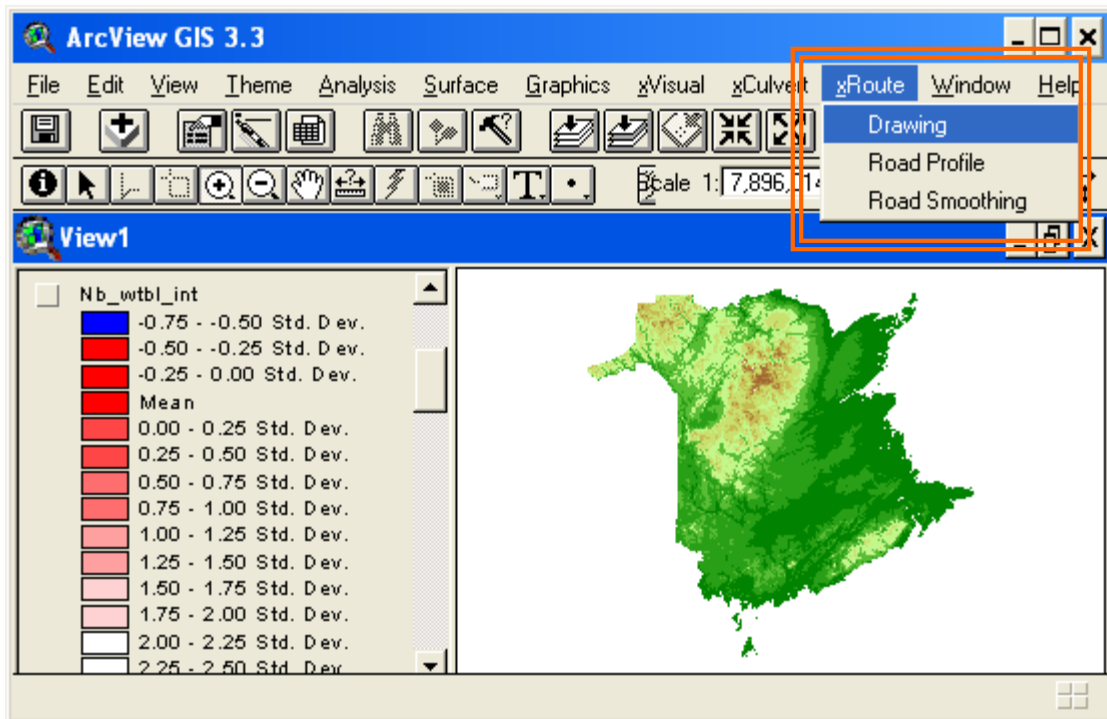


Figure 5

To plot the preliminary designed route described in Charter 4, the **S** **A** (Fig. 2) buttons are used to identify the start point and end point for the road on the map. The xRoute menu includes several applications of the ForRoutes module, such as manual drawing of forest road, road profile and curve smoothing (Fig. 5).

Appendix B- Rainfall Data for Atlantic Canada

Station Id	Station Name	Prov	Longitude	Latitude	Start Date	End Date	25/5	25/10	25/15	25/30	25/1h	25/2h
8104800	SAINT JOHN	NB	-66.08	45.28	1924	1950	8.9	14.2	18.1	26.9	37.3	48.6
8104900	SAINT JOHN A	NB	-65.88	45.33	1958	1994	11.5	16.8	19.2	28.4	40.5	59.4
8101600	FREDERICTON CDA	NB	-66.61	45.91	1959	1998	12.1	17.8	20	24	29.7	40.4
8101500	FREDERICTON A	NB	-66.53	45.86	1984	994	9.5	15.6	19.2	26.4	26.9	35.3
8104480	ROYALI ROAD	NB	-66.71	46.05	1966	1992	13.6	19.8	24.1	31.4	36.4	42.2
8104482	ROYAL ROAD WEST	NB	-66.73	46.08	1966	1978	13.6	16.7	18	19.1	27.4	36.5
8103200	MONCTON A	NB	-64.68	46.10	1946	1998	12.3	17.7	21.6	28.6	38.4	48.2
8100512	BEECHWOOD	NB	-67.66	46.53	1959	1969	12.9	19.2	22.4	26.5	28.2	31.9
8101000	CHATHAM AIRPORT	NB	-65.45	47.01	1964	1990	12.4	20	24.7	31.7	34.9	41
8105100	SUMMIT DEPOT	NB	-68.33	47.78	1955	1973	14.8	22.1	26.7	33	44.1	56.2
8100514	BELLEDUNE	NB	-65.83	47.90	1971	1989	10	16.4	19.6	23.5	26.3	31.3
8100880	CHARLO A	NB	-66.33	47.98	1959	1998	10.1	13.7	16.1	21.5	25.8	30.6
8104928	ST LEONARD A	NB	-67.83	47.15	1985	1998	12.9	18.3	23.3	27	27.9	33.3
8205126	SHELBURNE	NS	-65.25	43.71	1973	1988	14.2	20.5	25.3	35.8	46.2	58.5
8206500	YARMOUTH A	NS	-66.08	43.83	1971	1996	10.2	16.9	23	31.1	41.8	51.9
8204700	SABLE ISLAND	NS	-60.01	43.93	1962	1995	12.4	17.1	20.7	29.6	40.3	51.3
8205090	SHEARWATER A	NS	-63.50	44.63	1955	1997	10.9	15.3	18	24.5	32.2	51.3
8202200	HALIFAX	NS	-63.56	44.65	1941	1973	11.1	13.9	15.8	22.6	32.9	50.1
8202250	HALIFAX INT'L AIRPORT	NS	-63.51	44.88	1977	1988	9.5	13	16.1	22.4	37.2	49.9
8202000	GREENWOOD A	NS	-64.91	44.98	1964	1997	10.2	14.5	18.6	26.5	37.5	46.7
8205085	SHARPE BROOK	NS	-64.63	45.01	1968	1977	8.1	12.8	14.9	20	31.8	54.8

Station Id	Station Name	Prov	Longitude	Latitude	Start Date	End Date	25/5	25/10	25/15	25/30	25/1h	25/2h
8104800	SAINT JOHN	NB	-66.08	45.28	1924	1950	8.9	14.2	18.1	26.9	37.3	48.6
8104900	SAINT JOHN A	NB	-65.88	45.33	1958	1994	11.5	16.8	19.2	28.4	40.5	59.4
8101600	FREDERICTON CDA	NB	-66.61	45.91	1959	1998	12.1	17.8	20	24	29.7	40.4
8101500	FREDERICTON A	NB	-66.53	45.86	1984	994	9.5	15.6	19.2	26.4	26.9	35.3
8104480	ROYALI ROAD	NB	-66.71	46.05	1966	1992	13.6	19.8	24.1	31.4	36.4	42.2
8104482	ROYAL ROAD WEST	NB	-66.73	46.08	1966	1978	13.6	16.7	18	19.1	27.4	36.5
8103200	MONCTON A	NB	-64.68	46.10	1946	1998	12.3	17.7	21.6	28.6	38.4	48.2
8100512	BEECHWOOD	NB	-67.66	46.53	1959	1969	12.9	19.2	22.4	26.5	28.2	31.9
8101000	CHATHAM AIRPORT	NB	-65.45	47.01	1964	1990	12.4	20	24.7	31.7	34.9	41
8105100	SUMMIT DEPOT	NB	-68.33	47.78	1955	1973	14.8	22.1	26.7	33	44.1	56.2
8100514	BELLEDUNE	NB	-65.83	47.90	1971	1989	10	16.4	19.6	23.5	26.3	31.3
8100880	CHARLO A	NB	-66.33	47.98	1959	1998	10.1	13.7	16.1	21.5	25.8	30.6
8104928	ST LEONARD A	NB	-67.83	47.15	1985	1998	12.9	18.3	23.3	27	27.9	33.3
8205126	SHELBURNE	NS	-65.25	43.71	1973	1988	14.2	20.5	25.3	35.8	46.2	58.5
8206500	YARMOUTH A	NS	-66.08	43.83	1971	1996	10.2	16.9	23	31.1	41.8	51.9
8204700	SABLE ISLAND	NS	-60.01	43.93	1962	1995	12.4	17.1	20.7	29.6	40.3	51.3
8205090	SHEARWATER A	NS	-63.50	44.63	1955	1997	10.9	15.3	18	24.5	32.2	51.3
8202200	HALIFAX	NS	-63.56	44.65	1941	1973	11.1	13.9	15.8	22.6	32.9	50.1
8202250	HALIFAX INT'L AIRPORT	NS	-63.51	44.88	1977	1988	9.5	13	16.1	22.4	37.2	49.9
8202000	GREENWOOD A	NS	-64.91	44.98	1964	1997	10.2	14.5	18.6	26.5	37.5	46.7
8205085	SHARPE BROOK	NS	-64.63	45.01	1968	1977	8.1	12.8	14.9	20	31.8	54.8

Station Id	Station Name	Prov	Longitude	Latitude	Start Date	End Date	25/5	25/10	25/15	25/30	25/1h	25/2h
8202800	KENTVILLE CDA	NS	-64.48	45.06	1960	1995	10.8	15.9	18.4	27.1	33.5	45.7
8205990	TRURO	NS	-63.26	45.36	1958	1985	9.2	14.6	17.5	25.9	33	45.7
8201716	EDDY POINT	NS	-61.25	45.51	1972	1985	10.3	14.1	19.4	31.4	36.5	46.6
8205700	SYDNEY A	NS	-60.05	46.16	1961	1997	9	13.1	16.6	23.8	33.6	45.7
8300400	CHARLOTTETOWN CDA	PEI	-63.13	46.25	1967	1990	11.2	15.2	17.8	23.2	32.6	49.5
8300700	SUMMERSIDE AIRPORT	PEI	-63.83	46.43	1964	1990	10.6	15.3	19	28.3	37	43.3
8403615	ST LAWRENCE	NFLD	-55.38	46.91	1969	1990	7.8	12.7	16	24.2	36.9	52.7
8402975	PORT AUX BASQUES	NFLD	-59.16	47.56	1975	1990	8.6	11.8	15.9	23.8	31.7	50.3
8400798	BURGEO	NFLD	-57.61	47.61	1967	1990	10.9	14.6	18.2	24.2	31.8	45.8
8403506	ST JOHN'S AIRPORT	NFLD	-52.73	47.61	1949	1990	9.2	12.8	15.8	22	28.6	41.7
8403290	ST ALBANS	NFLD	-55.85	47.86	1969	1983	7.8	11.3	14.3	18.6	25.8	35.4
8403800	STEPHENVILLE AIRPORT	NFLD	-58.55	48.53	1967	1990	7.5	13.6	16.8	23.7	30.3	39.1
8401700	GANDER INT'L AIRPORT	NFLD	-54.56	48.95	1939	1990	9.4	13.7	16.6	23.1	26.8	31.7
8401501	DEER LAKE AIRPORT	NFLD	-57.40	49.21	1966	1990	9.1	12.1	13.9	16.9	21.2	33.4
8401259	COMFORT COVE	NFLD	-54.88	49.26	1967	1990	11.5	17.6	21.6	26.1	29.7	35.6
8401400	DANIELS HARBOUR	NFLD	-57.58	50.23	1969	1990	9.9	12.7	14.5	18.8	26.9	33.5
8403401	ST ANTHONY	NFLD	-55.63	51.36	1971	1990	8.5	13.1	15.4	17.5	20.4	28
8500398	BATTLE HARBOUR LOR	NFLD	-55.60	52.25	1972	1983	7.3	8.1	9.6	11.5	15.5	21.9
8504175	WABUSH LAKE AIRPORT	NFLD	-66.86	52.93	1974	1990	8.1	13	16	21.4	26.4	29.6
8501900	GOOSE AIRPORT	NFLD	-60.41	53.31	1961	1990	8.2	13.4	15.7	20.1	22.9	28.8
8501132	CHURCHILL FALLS AIRPORT	NFLD	-64.10	53.55	1969	1990	9	12.7	15.2	18.2	18.9	21.1

Station Id	Station Name	25/6h	25/12h	25/24h	100/5	100/10	100/15	100/30	100/1h	100/2h	100/6h	100/12h	100/24h
8202800	KENTVILLE CDA	72.9	86.1	102.4	13.5	19.8	22.6	33.9	41.3	56.3	88.9	103.6	125
8205990	TRURO	67.1	77.8	97.4	11.2	17.7	21.2	31.9	40.1	55.8	79.9	92.1	116.6
8201716	EDDY POINT	68.7	84.1	99.3	12.7	17.1	23.6	39.1	44.2	54.9	80.7	98	114.3
8205700	SYDNEY A	76.2	88.9	103.9	10.9	15.8	20.2	29.1	41.3	55.5	91.9	105.1	122.3
8300400	CHARLOTTETOWN CDA	72.2	88.6	93	14.2	18.9	22	28.3	39.9	61.2	88.2	108.2	111.9
8300700	SUMMERSIDE AIRPORT	52.4	60.4	70.1	13.3	19	23.6	35.5	46.4	53	62	71.2	83
8403615	ST LAWRENCE	81.4	92.3	107.1	9	14.8	19	29	44.6	63.6	96.2	107.5	125.6
8402975	PORT AUX BASQUES	80.8	98	121	10.7	14.2	19.3	29.1	38.6	61.8	97.6	117.1	145.2
8400798	BURGIO	72.7	90	109.6	13.3	17.7	22	28.9	37.4	53.7	84.6	103.9	127.3
8403506	ST JOHN'S AIRPORT	66.1	81.7	96.8	11.4	15.8	19.4	26.8	34.4	51	79	96.4	113.6
8403290	ST ALBANS	62.1	98.9	149.8	9.5	13.5	17.2	21.6	30.1	39.8	69.3	115.7	182.6
8403800	STEPHENVILLE AIRPORT	63.3	72.1	99.2	9.2	17.1	21.1	30.1	37.6	48.1	77	85.9	120.1
8401700	GANDER INT'L AIRPORT	47.8	59.1	78.5	12	17.4	21.1	29.7	33.8	38.8	57.5	70.5	95.7
8401501	DEER LAKE AIRPORT	43.1	53.9	63.2	11.5	15.2	17.2	20.4	25.3	41.1	50.4	63	74.1
8401259	COMFORT COVE	53.3	65.4	73.8	15.1	23.2	28.3	33.5	37.6	44.4	65.7	79.8	89
8401400	DANIELS HARBOUR	54.6	76.8	109.2	12.8	16.1	18.1	23.7	33.7	40.7	65.4	93.9	136.5
8403401	ST ANTHONY	46.9	61.8	80.1	10.9	16.9	19.6	21.3	24	32.7	54.5	72.4	95.2
8500398	BATTLE HARBOUR LOR	38.4	53	72.5	9.3	10.1	11.9	13.8	18.2	25.8	45.1	61.7	86.2
8504175	WABUSH LAKE AIRPORT	33.6	43.8	59.5	10.1	16.4	20.3	26.9	33.2	36.1	39.2	50.9	70.5
8501900	GOOSE AIRPORT	44.9	55.9	68.2	10.3	17.2	20.2	26	29.2	36.2	55	67.5	82.3
8501132	CHURCHILL FALLS AIRPORT	34.1	45.9	55.1	11.1	15.9	19.1	22.4	22.8	24.8	40.9	54.9	64.8

Appendix C- METHODOLOGY OF MAPPING WET AREAS

DEMs can be used to derive information on soil moisture or drainage conditions (Moore et al. 1991; Gessler et al. 2000; Ryan et al. 2000; Meng et al. 200?). These can help in the delineation of riparian zones and wetlands in the landscape (Murphy et al. 200?). In this chapter, wet soil areas were expressed as depth-to-surface water map layers. The following describes the DEM process to generate the depth-to-surface water index for use by the tools described in this report.

Two components are needed for this type of mapping: a local digital elevation model (DEM), and the local hydrographic features map (depicting all lakes, streams, shorelines, etc.).

The artificial pits were removed from the raw DEM using ArcView FILL tool and a depressionless DEM was created. A depression grid, identifying areas of the DEM which slope inward to a point from all directions, which were not artificial pits, was derived from the DEM. Then the flow direction algorithm D8 (deterministic-8 node algorithm) (Hornberger and Boyer 1995;) was used to yield the topographically derived flow accumulation pattern. A slope grid, derived from the DEM, and the hydrographic source grid were used as input to derive depth-to-surface water (m).

For each cell in the landscape, an iterative function in ArcView GIS, COSTDISTANCE, finds the source cell with the lowest cumulative slope value between it and the cell in the landscape. The cumulative slope value associated with the least slope path is multiplied by side length of the grid cells (m) and assigned to the cell as a value of the depth-to-surface water (m).

The resultant value approximates the elevation difference between the point in the landscape and the assigned water source cell (Fig. 2.2). Lower values indicate a shallower depth-to-water, and thus, wetter soils.

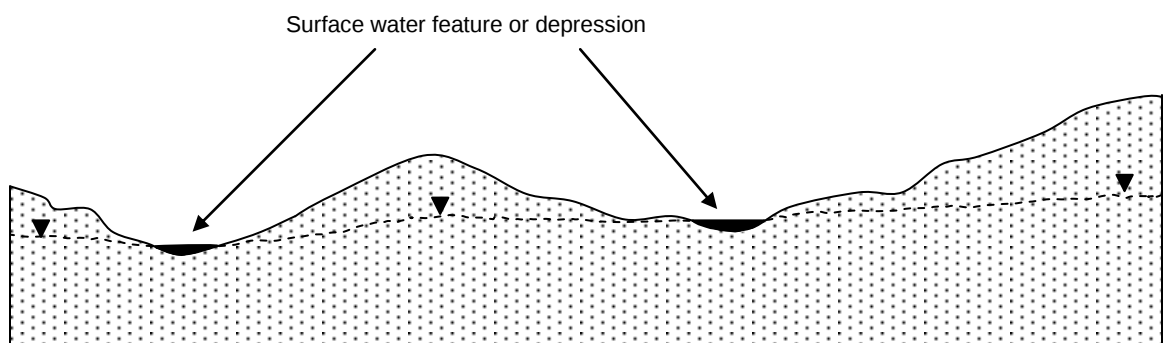
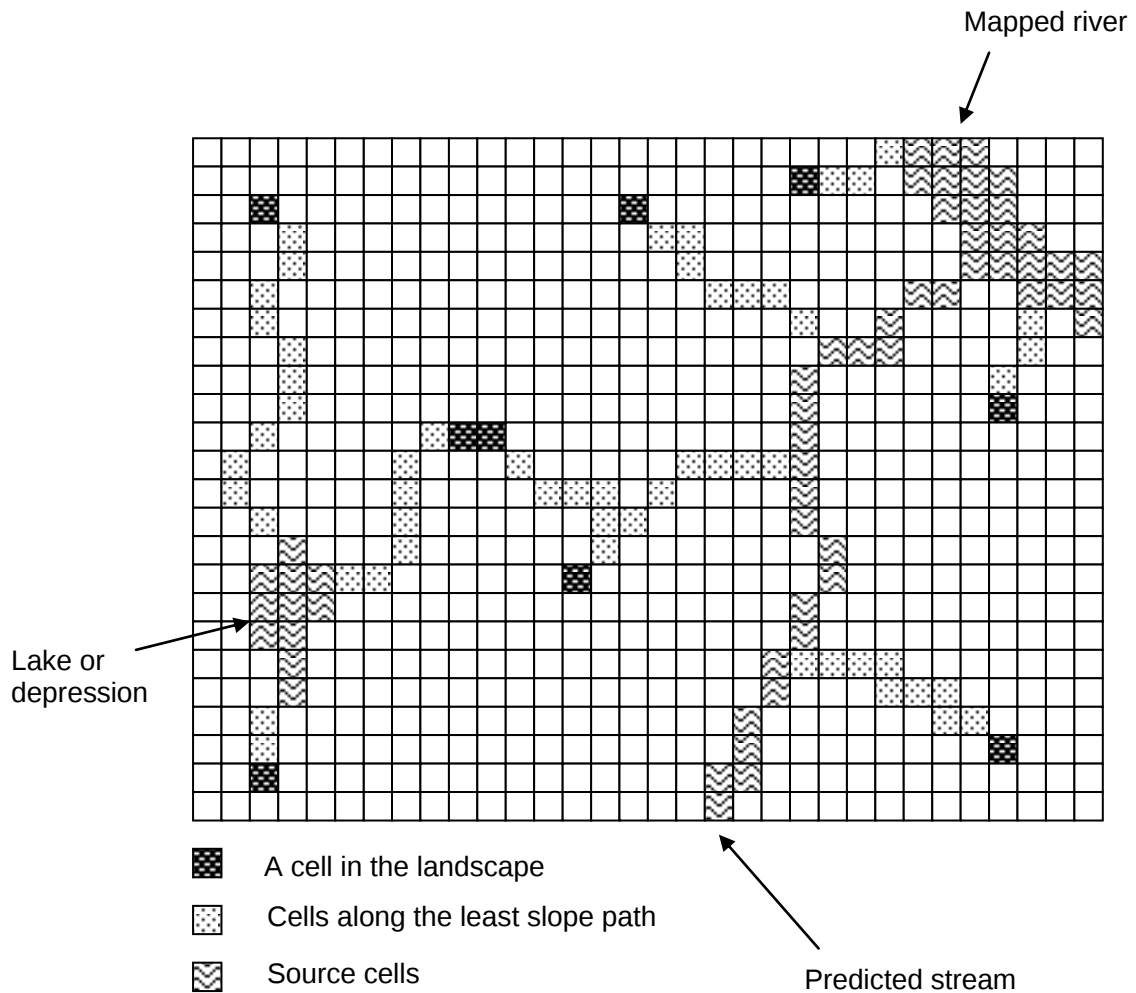


Figure 2.2 The assignment of cells in the landscape to source cells and least slope paths illustrated for a selection of cells. All cells are assigned a depth to water table based on this, defining an approximate water table surface as shown in cross-section. (This figure was adapted from *FORWARD-UNB Wet-area Mapping Project* report, compiled by Dr. Arp et al.)

The depth-to-water value reflects both the vertical distance from a source (surface-water feature or depression) and the slope of the land surface between the landscape cell and the source.

Therefore, the index reflects topographic and hydrologic aspects down slope from a point in the landscape. Depth-to-water will increase, indicating drier soils, with increasing distance from a source and with increasing slope values between the landscape cell and the source (Fig. 2.2). All source cells have a value of 0 because there is 0 distance and 0 slope to a source cell (water table is at the surface). Values increase more rapidly in steeper terrain (higher slope values) and more slowly in flatter terrain (lower slope values).

Approximate depth-to-water is broken up into classes of values for display in the map. It is the areas of shallow water table which are of most concern operationally. Field experience has shown this classification to give a good indication of the distribution of wet areas. The classes can be interpreted on the map as indicating the tendency of soils to be saturated. Areas mapped in the two shallowest classes can be considered highly likely to be wet or saturated for large parts of the year. Forest operations may be planned accordingly.

The principal stages, and detail from the final map of wet areas in this process are illustrated in Figure 2.3.

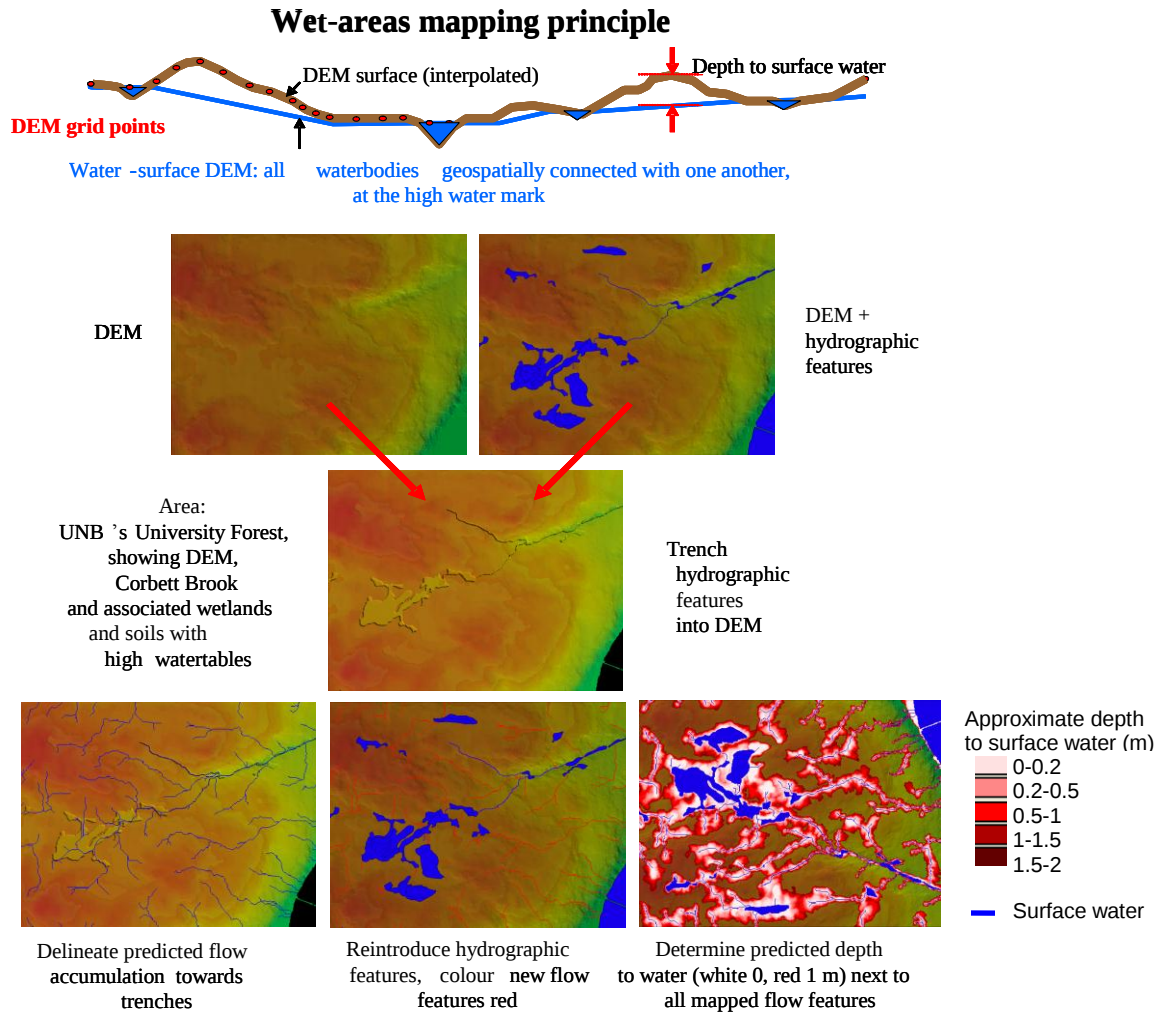


Figure 4.2 Principal stages involved in deriving mapped and unmapped flow channels from local DEMs, using already existing hydrographic information to force the digitally derived flow channels and surface water bodies to comply with this information (resource: “*Developing a GIS-based flow channel and wet-areas mapping framework for precision forestry planning*” IUFRO, 2006).

LITERATURE CITED

- Boerner, R.E.J., S.J. Morris, E.K. Sutherland and T.F. Hutchinson. 2000. Spatial variability in soil nitrogen dynamics after prescribed burning in Ohio mixed-oak forests. *Landscape Ecology* 15: 425-439.
- Case, B.S., F.-R. Meng and P.A. Arp. 2005. Digital elevation modelling of soil type and drainage within small forested catchments. *Can. J. Soil Sci.* 85: 127-137.
- Clarke, S. and K. Burnett. 2003. Comparison of digital elevation models for aquatic data development. *Photo. Eng. Rem. Sens.* 69: 1367-1375.
- Fall, A. and D. Morgan. 2000. Ecological soil moisture prediction models. B.C. Forest Service, Research Branch. Downloaded from <http://www.for.gov.bc.ca/hre/dulp/char/dulpc04.htm>.
- Garbrecht, J., F.L. Ogden, P.A. DeBarry and D.R. Maidment. 2001. GIS and distributed watershed models. I: data coverages and sources. *J. Hydr. Eng.* 6: 506-514.
- Gessler, P.E., O.A. Chadwick, F. Chamran, L. Althouse and K. Holmes. 2000. Modeling soil-landscape and ecosystem properties using terrain attributes. *Soil Sci. Soc. Am. J.* 64: 2046-2056.
- Grunwald, S. 2006. What do we really know about the space-time continuum of soil-landscapes? In S. Grunwald (ed.) *Environmental soil-landscape modeling: geographic information technologies and pedometrics*. pp. 3-36. CRC Press, New York, NY, USA.
- Güntner, A., J. Seibert and S. Uhlenbrook. 2004. Modeling spatial patterns of saturated areas: an evaluation of different terrain indices. *Water Resources Research* 40: 1-19.
- Hornberger, G.M. and E.W. Boyer. 1995. Recent advances in watershed modelling. *Rev. Geophys.* 33: 949-957.
- Iverson, L.R., M.E. Dale, C.T. Scott and A. Prasad. 1997. A GIS-derived integrated moisture index to predict forest composition and productivity of Ohio forests (U.S.A.). *Landscape Hydrology* 12: 331-348.
- Johnston, C.A. and P. Meysembourg. 2002. Comparison of the Wisconsin and National Wetlands Inventories. *Wetlands* 22: 386-405.
- Kent, B.J. and J.N. Mast. 2005. Wetland change analysis of San Dieguito Lagoon, California, USA: 1928-1994. *Wetlands* 25: 780-787.

- Meng, F., Castonguay, M., Ogilvie, J., Murphy, P. and Arp, P. 2006 Developing a GIS-based flow channel and wet-areas mapping framework for precision forestry planning. IUFRO Precision Forestry Conference 2006
- Mizgalewicz, P.J. and D.R. Maidment. 1996. Modeling agrichemical transport in midwest rivers using geographic information systems. CRWR Online Report 96-6. Center for Research in Water Resources, The University of Texas at Austin, Austin.
- Moore, I.D., R.B. Grayson and A.R. Ladson. 1991. Digital terrain modelling: a review of hydrological, geomorphological, and biological applications. *Hydrol. Process.* 5: 3-30.
- Murphy, P.N.C., M. Castonguay, J. Ogilvie, T. Connors and P.A. Arp. 2006a. Forest operations planning based on high resolution wet areas mapping: verifications. International Precision Forestry Symposium, Stellenbosch, South Africa. Stellenbosch University.
- Murphy, P.N.C., J. Ogilvie, M. Castonguay, F.-R. Meng and P.A. Arp. 2007. Verifying calculated flow accumulation patterns of mapped and unmapped forest streams by culvert location. *The Forestry Chronicle* 83: 198-206.
- Murphy, P.N.C., J. Ogilvie, M. Castonguay, T. Connors, F.-R. Meng and P.A. Arp. 2006b. DEM-derived flow channel and wet area mapping: a new tool for forest operations planning. Sustainable Forest Management Network, Fourth International Conference, Edmonton, AB, Canada. Sustainable Forest Management Network.
- Olivera, F. 1996. Spatial hydrology of the Urubamba river system in Peru using geographic information systems (GIS). Center for Research in Water Resources, The University of Texas at Austin, TX, USA. Downloaded from www.ce.utexas.edu/prof/maidment/CE397/urubamba/peru.htm.
- Saunders, W.K. 1999. Preparation of DEMs for use in environmental modeling analysis. 1999 ESRI User Conference, San Diego, California. July 24-30.
- Saunders, W.K. and D.R. Maidment. 1996. A GIS assessment of nonpoint source pollution in the San Antonio-Nueces coastal basin. Center for Research in Water Resources, The University of Texas at Austin, TX, USA. Downloaded from <http://www.ce.utexas.edu/centers/crwr/reports/online.html>.
- Simley, J. 2004. The geodatabase conversion. USGS National Hydrography Newsletter 3 (4). USGS.
- Strager, M.P., J.J. Fletcher and C.B. Yuill. 1998. Hydrologic modeling for acid mine drainage in West Virginia. 1998 ESRI International Users Conference, San Diego, California. ESRI.

Tomer, M.D., D.E. James and T.M. Isenhardt. 2003. Optimizing the placement of riparian practices in a watershed using terrain analysis. *Journal of Soil and Water Conservation* 58: 198-206.

U.S.F.W.S. 2007. National Wetlands Inventory Metadata. United States Fish and Wildlife Service. Downloaded from http://www.fws.gov/nwi/downloads/metadata/nwi_meta.txt.

Appendix D- FOROAD ArcView Avnue™ Code

The following ArcView Avnue code comprises the main parts of FOROAD that creates functions described in this report. Much of the code is not directly related to the main task and is not included here.

```
' FOROAD.RoadMenu.Culverts Locating and Sizing
'
' Created By:  Xiangfei Meng
'              Foresty and Environmental Management
'              University of New Brunswick
'              Fredericton, New Brunswick, Canada
'
' Description: The script fixes raw DEM
' Calls:
' Returns:

'*****
theView = av.GetActiveDoc

' fill input GTheme

theThemeList = av.GetActiveDoc.GetThemes.Clone
' Use these lines to prompt the user for input parameters...
theTheme = (MsgBox.List(theThemeList,"Raw DEM","Please select a grid
theme (Original map)..."))
if (theTheme = nil) then exit end

' fill sinks in Grid until they are gone
elevGrid = theTheme.GetGrid
tempGrid = elevGrid

sinkCount = 0
numSinks = 0
while (TRUE)
    flowDirGrid = elevGrid.FlowDirection(FALSE)
    sinkGrid = flowDirGrid.Sink
    if (sinkGrid.GetVTab = NIL) then
        ' check for errors
        if (sinkGrid.HasError) then return NIL end
        sinkGrid.BuildVAT
    end
    ' check for errors
    if (sinkGrid.HasError) then return NIL end

    if (sinkGrid.GetVTab <> NIL) then
        theVTab = sinkGrid.GetVTab
        numClass = theVTab.GetNumRecords
```

```

        newSinkCount = theVTab.ReturnValue(theVTab.FindField("Count"),0)
    else
        numClass = 0
        newSinkCount = 0
    end
    if (numClass < 1) then
        break
    elseif ((numSinks = numClass) and (sinkCount = newSinkCount)) then
        break
    end
    waterGrid = flowDirGrid.Watershed(sinkGrid)
    zonalFillGrid = waterGrid.ZonalFill(elevGrid)
    fillGrid = (elevGrid
<(zonalFillGrid.IsNull.Con(0.AsGrid,zonalFillGrid))).Con(zonalFillGrid,
elevGrid)
    elevGrid = fillGrid
    numSinks = numClass
    sinkCount = newSinkCount
end

' rename data set
aFN = av.GetProject.GetWorkDir.MakeTmp("fill", "")
elevGrid.Rename(aFN)
DepGrid = elevGrid - tempGrid

' create a theme
theGTheme = GTheme.Make(elevGrid)

' set name of theme
theGTheme.SetName("$Filled")

' add theme to the view
theView.AddTheme(theGTheme)

sinkGrid = elevGrid - tempGrid
' create a theme
theGTheme = GTheme.Make(DepGrid)

' set name of theme
'theGTheme.SetName("Depression")

' add theme to the view
'theView.AddTheme(theGTheme)

```

```

' FOROAD.RoadMenu.Culverts Locating and Sizing
'
' Created By: Xiangfei Meng
'             Foresty and Environmental Management
'             University of New Brunswick
'             Fredericton, New Brunswick, Canada
'
' Description: The script find the locations along road layer where
'drainage area is greater than 10 ha (can be changed)
'             Check the active list, if there is a road layer
'             2 adjacent culverts will be merged; new culvert is
located at the place
'             The bigger one was laid
' Calls:
' Returns:
'*****

theView = av.GetActiveDoc
theThemeList = av.GetActiveDoc.GetThemes.Clone
theRoadTheme = (MsgBox.List(theThemeList,"The road layer","Please
select a line theme (Road)..."))
if (theRoadTheme = nil) then exit end

'theRoadFtab = theRoadTheme.GetFTab
'Find flow accumulation and flow direction grid
theFlowDir = theView.FindTheme("Flow Direction").GetGrid
theAccum = theView.FindTheme("Flow Accumulation").GetGrid

'Obtain the cell size and extend of the grid
TheCellsize = theAccum.GetCellSize
TheExtent = theAccum.GetExtent
'Create a grid with the road layer
aPrj = theView.GetProjection
'Grid.MakeFromFTab (anFTab, aPrj, aField, gridSize)
RoadGrid =
Grid.MakeFromFTab(theRoadTheme.GetFTab,aPrj,NIL,{theCellSize,theExtent}
)
'
'Calculate the point on the road need coulvet
theAcummHa= theAccum*TheCellsize*theCellsize/10000
'Require a watershed size requires coulvet
watershed =5
aAGrid = theAcummHa>=(watershed.asGrid)

RoadlGrid = RoadGrid*0.asgrid+1.asgrid*aAGrid
'theCrossing = GTheme.Make(aAGrid)
'theView.AddTheme(theCrossing)
'theCrossing = GTheme.Make(RoadlGrid)
'theView.AddTheme(theCrossing)
RoadlGrid = RoadlGrid.log10+1
'theCrossing = GTheme.Make(RoadlGrid)
'theView.AddTheme(theCrossing)

'*****calculate the DrainageArea and Culvert diameter

DrainageArea = theAcummHa/100
'A = DrainageArea; I = rainfall intensity (mm/hour 100yr)

```

```

'I = (MsgBox.List(theThemeList,"The rainfall layer","Please select a
grid theme (rainfall)..."))
'if (I = nil) then exit end
I = 45.7.asgrid
'C = MsgBox.Input("Enter The Value of Runoff Coefficient", "the
proportion of the rainfall", "0.75").asnumber
C = 0.3

myList={"100", "50", "25"}
aSlection = MsgBox.ChoiceAsString( myList, "Choose flood period:",
"Years" )
if (YourChoice = "100") then
Wmap = _BBW50
elseif (aSlection = "50") then
Wmap = _BBW100
else Wmap = _BBW25
end
Q100 = C.asgrid*theAcummHa*I/360
'culvert Diameter
Diameter = Q100^2*Wmap
'*****End to calculate the DrainageArea Culvert diameter
'aFileName = "c:\temp\point_C.shp".AsFileName
'Create a Point file contains coulvet location and watershed size
CrossingFtab = Road1Grid.AsPointFTab
("Culvert_location.shp".AsFileName, aPrj)

'Make a theme of road crossing points in the view
theCrossing = FTheme.Make(CrossingFtab)
theView.AddTheme(theCrossing)
'$theView.AddTheme(theCrossing).setVisible(true)

'Create a point layer with road crossing, calculate the X, Y Corrdinate
'Calculate the drainage area
'Calculate the coulvet size "not done yet"
'Label the points of road crossing with the coulvet size "current
drainage area"
'
' Make the FTAB editable and add appropriate fields into the results
table
'
CrossingFtab.SetEditable(TRUE)
if (CrossingFtab.FindField("X") = nil) then
theXField = Field.Make("X",#FIELD_DOUBLE,16,1)
CrossingFtab.AddFields({theXField})
else
theXField = CrossingFtab.FindField("X")
end

if (CrossingFtab.FindField("Y") = nil) then
theYField = Field.Make("Y",#FIELD_DOUBLE,16,1)
CrossingFtab.AddFields({theYField})
else
theYField = CrossingFtab.FindField("Y")
end
'*****DrainArea start
if (CrossingFtab.FindField("DrainArea") = nil) then

```

```

        theDrainageAreaField = CrossingFtab.FindField("DrainageArea")
Field.Make("DrainageArea", #FIELD_DOUBLE, 16, 3)
        CrossingFtab.AddFields({theDrainageAreaField})
    else
        theDrainageAreaField = CrossingFtab.FindField("DrainageArea")
    end
    '*****DrainArea end
    '*****Diameter start

    if (CrossingFtab.FindField("Diameter") = nil) then
        theDiameterField = Field.Make("Diameter", #FIELD_DOUBLE, 16, 2)
        CrossingFtab.AddFields({theDiameterField})
    else
        theDiameterField = CrossingFtab.FindField("Diameter")
    end
    '*****Diameter end

    '*****Calculate the concentration time
    'Formula: Vmax = (2*g*Distance*Sin(Angle))^(1/2)/Roughness
        'Vaverage = 1/2*Vmax
        'ConcentrationTime = Distance/Vaverage

    'where, Distance is between the furthest point to concentration
point and concentration center
    'Angle is the inclination angle of drainage region
    Roughness = 0.2
    Distance = theAcummHa*10000/3.14159.asgrid.pow(0.5)
    Distance=Distance
    Vmax = Distance*0.2
    Vmax = Vmax*19.6*(0.1.Sin)
    Vmax = Vmax.pow(0.5)

    Vaverage = Vmax*0.5
    ConcentrationTime = Distance/Vaverage/3600

    if (CrossingFtab.FindField("ConcentrationHour") = nil) then
        theConcentrationHourField = CrossingFtab.FindField("ConcentrationHour")
Field.Make("ConcentrationHour", #FIELD_DOUBLE, 16, 2)
        CrossingFtab.AddFields({theConcentrationHourField})
    else
        theConcentrationHourField = CrossingFtab.FindField("ConcentrationHour")
    end

    '*****the concentration time end

    '
    ' Loop through the FTAB and find the each shape and set the field
values appropriately.
    '

    theShape = CrossingFtab.ReturnValue(CrossingFtab.FindField("shape"), 0)

    For Each rec in CrossingFtab
        CrossingFtab.QueryShape(rec, aPrj, theShape)
        theX = theShape.GetX

```

```

theY = theShape.GetY
thePoint =Point.Make(theX, theY)
DrainageArea = theAcummHa.CellValue (theShape, aPrj)
Diameter1 = Diameter.CellValue(theShape, aPrj)
Concen_Time = ConcentrationTime.CellValue (theShape, aPrj)

CrossingFtab.SetValue(theXField,rec,theX)
CrossingFtab.SetValue(theYField,rec,theY)
CrossingFtab.SetValue(theDrainageAreaField,rec,DrainageArea)
CrossingFtab.SetValue(theDiameterField,rec,Diameter1)
CrossingFtab.SetValue(theConcentrationHourField,rec,Concen_Time)

'
'Label the crossing points
'
'txtSet = {}
'txtSymbol = TextSymbol.Make
'Diameter1.SetFormat( "dd.dd" )
'gl
GraphicLabel.MakeWithSym(Diameter1.AsString,thePoint,txtSymbol,txtSet) =
' The MakeWithSym request automatically adds gl to txtSet
'theView.GetGraphics.Add(gl)
'gl.Invalidate

end

'*****Calculate the distance between two close culverts
'Remove smaller one

thePrj = av.GetActiveDoc.GetProjection
for each t in CrossingFtab
'#for each i in theView.GetVisibleThemes '###

' Get the point location you are measuring FROM.
' If the view is projected, get the the point location in projected
units.
theFromShape =
CrossingFtab.ReturnValue(CrossingFtab.FindField("shape"),t)
'record the diameter of t culvert
FromD = Diameter.CellValue (theShape, aPrj)

for each j in CrossingFtab
if (j >= (t+1))then
'Get it in projected units, if the view is projected.
theToShape =
CrossingFtab.ReturnValue(CrossingFtab.FindField("Shape"),j)

'record the diameter of t culvert
ToD = Diameter.CellValue (theShape, aPrj)

' Calculate the distance between the two points.
' Add the value to the output (TO) branch table.

'***End of Romoving label
end

```

```

end
end

'*****End calculation of the distance between two close culverts

Filled = theView.Findtheme("$Filled").GetGrid
'***Road increased by 10
NewDEM = RoadGrid*Filled*0 + Filled
NewDEM = NewDEM.merge({Filled})
CulvertSink = NewDEM - (Road1Grid + 9)
NewDEM = CulvertSink.merge({NewDEM})
'NewDEM = Gtheme.make(NewDEM)
'theview.addtheme(NewDEM)
ReFlowDirec = NewDEM.FlowDirection(false)
ReFlowAccum = ReFlowDirec.FlowAccumulation(Nil)

'aReflowAccum= Gtheme.make(ReFlowAccum)

'theview.addtheme(aReFlowAccum)

'*****End of recal flowaccumulation

'*****Update drainage area (explanation: see above)
theAccum = ReFlowAccum
theAcummHa= theAccum*TheCellsize*theCellsize/10000
'Require a watershed size requires coulvet
aAGrid = theAcummHa>=(watershed.asGrid)

DrainageArea = theAcummHa/10
Q100 = C.asgrid*DrainageArea*I/360

Diameter = Q100^2*Wmap

For each rec in CrossingFTab
    CrossingFtab.QueryShape(rec,aPrj,theShape)
    DrainageArea1 = DrainageArea.CellValue (theShape, aPrj)
    Diameter1 = Diameter.CellValue(theShape, aPrj)
    'Concen_Time = ConcentrationTime.CellValue (theShape, aPrj)

    CrossingFtab.SetValue(theDrainageAreaField,rec,DrainageArea1)
    CrossingFtab.SetValue(theDiameterField,rec,Diameter1)
    'CrossingFtab.SetValue(theConcentrationHourField,rec,Concen_Time)

    ,
    'Label the crossing points
    ,

    theX = theShape.GetX
    theY = theShape.GetY
    thePoint =Point.Make(theX, theY)

    txtSet = {}
    txtSymbol = TextSymbol.Make
    Diameter1.SetFormat( "dd.dd" )
    gl
    GraphicLabel.MakeWithSym(Diameter1.AsString,thePoint,txtSymbol,txtSet)
    'The MakeWithSym request automatically adds gl to txtSet

```

```
        theView.GetGraphics.Add(gl)
        gl.Invalidate

end

'*****End of updating drainage area

CrossingFtab.SetEditable(false)
```

```

' FOROAD.RoadMenu.Profile
'
' Revised By:  Xiangfei Meng
'              Foresty and Environmental Management
'              University of New Brunswick
'              Fredericton, New Brunswick, Canada
'
' Description: The script create profile for new designed route.
'Original script is from 3D.profile compile by ArcView company
' Calls:
' Returns:

*****

theView = av.GetActiveDoc

' t = TemplateMgr.Show

    l = Layout.Make

    l.SetName("<Road Profile>")

    l.GetWin.Open

' 3D.ProfileGraphTool

theLayout = l

'*****      profileDialog  =  av.GetProject.FindDialog("ESRI
Profile Graph")

' return list from dialog will be:
'   0 : viewname (String)
'   1 : title status (Boolean)
'   2 : title (String or NIL)
'   3 : x label status (Boolean)
'   4 : x label (String or NIL)
'   5 : y label status (Boolean)
'   6 : y label (String or NIL)
'   7 : x grid status (Boolean)
'   8 : x grid (String or NIL)
'   9 : y grid status (Boolean)
'  10 : y grid (String or NIL)
'  11 : vert status (String)
'  12 : vert status (String or NIL)

'*****      profileOptions = profileDialog.Open
profileOptions              = {theview,true,"Road
Profile",true,"Distance",true,"Elevation",true,"1000",true,"10","", "100
"}

```

```

if (profileOptions = NIL) then
    return NIL
end

' unselected any selected graphics in layout
theLayout.GetGraphics.UnselectAll

' find view and get profile line
if (profileOptions.Get(0) = "<Empty View>") then
    theProfileList = NIL
else

' return is list of shape and symbol or NIL
'*****

    theProfileList
av.Run("RoadTools.$ProfileGraph.GetProfile",{profileOptions.Get(0)}) =
end

' set scale based on vertical exaggeration and profile line
xShift = 747
yShift = 747
theWidth = 7.5
theHeight = 5.5

'*****
'MsgBox.Info( theOrigin.GetX.AsString, "xShift")
'MsgBox.Info( yShift.AsString, "yShift")
'MsgBox.Info( theWidth.AsString, "theWidth")
'MsgBox.Info( theHeight.AsString, "theHeight")

theExtremeList = av.Run("ProfileGraph.GetExtremes",{theProfileList})
totalDistance = theExtremeList.Get(0)
zShift = theExtremeList.Get(1)
theZRange = theExtremeList.Get(3)
if (profileOptions.Get(11) = "Auto") then
    xScale = theWidth / totalDistance
    yScale = theHeight / theZRange
    verEx = yScale / xScale
else
    verEx = profileOptions.Get(12).AsNumber
    xScale = theWidth / totalDistance
    yScale = xScale * verEx
    if ((yScale * theZRange) > theHeight) then
        yScale = theHeight / theZRange
        xScale = yScale / verEx
    end
end

' set up symbols
titleSymbol = Symbol.Make(#SYMBOL_TEXT)
titleSymbol.SetFont(Font.Make("#FONT_COURBI","Normal"))
titleSymbol.SetSize(127)

```

```

titleSymbol.SetColor(Color.Getgreen)
labelSymbol = Symbol.Make(#SYMBOL_TEXT)
labelSymbol.SetFont(Font.Make("Helvetica","Normal"))
labelSymbol.SetColor(Color.Getgreen)
axisSymbol = Symbol.Make(#SYMBOL_PEN)
axisSymbol.SetColor(Color.getblack)
axisSymbol.SetSize(2)
subTitleSymbol = Symbol.Make(#SYMBOL_TEXT)
subTitleSymbol.SetFont(Font.Make("Helvetica","Normal"))
subTitleSymbol.SetColor(Color.Getred)
graphSymbol = Symbol.Make(#SYMBOL_PEN)
graphSymbol.SetColor(Color.GetGray)
numberSymbol = Symbol.Make(#SYMBOL_TEXT)
numberSymbol.SetFont(Font.Make("Helvetica","Normal"))
numberSymbol.SetColor(Color.Getred)
lastSize = 127

' create x and y axis for profile
x      = GraphicShape.Make(Line.Make((xShift)@(yShift),(xShift      +
(totalDistance * xScale))@(yShift)))
x.SetSymbol(axisSymbol)
x.SetSelected(TRUE)
y      = GraphicShape.Make(Line.Make((xShift)@(yShift),(xShift)@(yShift  +
(theZRange * yScale))))
y.SetSymbol(axisSymbol)
y.SetSelected(TRUE)
theLayout.GetGraphics.Add(x)
theLayout.GetGraphics.Add(y)

' place title
if (profileOptions.Get(1) and (profileOptions.Get(2) <> "")) then
    theTitle = GraphicText.Make(profileOptions.Get(2),0@0)
    theTitle.SetSymbol(titleSymbol)
    theTitle.SetSelected(TRUE)
    theLayout.GetGraphics.Add(theTitle)

' set title symbol font so that the title is 2/3rds the width of
graph
theSize = 127
maxSize = 127
minSize = 4
titleWidth = theTitle.GetExtent.GetX - theTitle.GetOrigin.GetX
minWidth = (totalDistance * xScale) * (2 / 3)
theCount = 0
while ((titleWidth - minWidth).Abs > (minWidth * 0.1))
    theDiff = (maxSize - minSize) / 2
    if (titleWidth > minWidth) then
        theSize = theSize - theDiff
        maxSize = theSize
    else
        theSize = theSize + theDiff
        minSize = theSize
    end
    theTitle.GetSymbol.SetSize(theSize)
    theTitle.Invalidate
    titleWidth = theTitle.GetExtent.GetX - theTitle.GetOrigin.GetX
    theCount = theCount + 1

```

```

        if (theCount > 10) then
            break
        end
    end
    lastSize = theTitle.GetSymbol.GetSize
end

' place y label
labelSymbol.SetSize((lastSize - 2) Max 4)
if (profileOptions.Get(5) and (profileOptions.Get(6) <> "")) then
    yAxisLabel = GraphicText.Make(profileOptions.Get(6), 0@0)
    yAxisLabel.SetAngle(90)
    yAxisLabel.SetSymbol(labelSymbol)
    yAxisLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(yAxisLabel)

    ' set axis symbol font so that the label is not longer than width
    of graph
    if (labelSymbol.GetSize > 4) then
        labelWidth = yAxisLabel.GetExtent.GetY -
yAxisLabel.GetOrigin.GetY
        yAxisWidth = (theZRange * yScale)
        if (labelWidth > yAxisWidth) then
            theSize = labelSymbol.GetSize
            maxSize = theSize
            minSize = 4
            theCount = 0
            while ((labelWidth - yAxisWidth).Abs > (yAxisWidth * 0.1))
                theDiff = (maxSize - minSize) / 2
                if (labelWidth > yAxisWidth) then
                    theSize = theSize - theDiff
                    maxSize = theSize
                else
                    theSize = theSize + theDiff
                    minSize = theSize
                end
                yAxisLabel.GetSymbol.SetSize(theSize)
                yAxisLabel.Invalidate
                labelWidth = yAxisLabel.GetExtent.GetY -
yAxisLabel.GetOrigin.GetY
                theCount = theCount + 1
                if (theCount > 10) then
                    break
                end
            end
        end
        lastSize = yAxisLabel.GetSymbol.GetSize
    end

    ' place x label
    if (profileOptions.Get(3) and (profileOptions.Get(4) <> "")) then
        xAxisLabel = GraphicText.Make(profileOptions.Get(4), 0@0)
        xAxisLabel.SetSymbol(labelSymbol)
        xAxisLabel.SetSelected(TRUE)
        theLayout.GetGraphics.Add(xAxisLabel)
    end
end

```

```

' set axis symbol font so that the label is not longer than width
of graph
  if (labelSymbol.GetSize > 4) then
    labelWidth = xAxisLabel.GetExtent.GetX -
xAxisLabel.GetOrigin.GetX
    xAxisWidth = (totalDistance * xScale)
    if (labelWidth > xAxisWidth) then
      theSize = yAxisLabel.GetSymbol.GetSize
      maxSize = theSize
      minSize = 4
      theCount = 0
      while ((labelWidth - xAxisWidth).Abs > (xAxisWidth * 0.1))
        theDiff = (maxSize - minSize) / 2
        if (labelWidth > xAxisWidth) then
          theSize = theSize - theDiff
          maxSize = theSize
        else
          theSize = theSize + theDiff
          minSize = theSize
        end
        xAxisLabel.GetSymbol.SetSize(theSize)
        xAxisLabel.Invalidate
        labelWidth = xAxisLabel.GetExtent.GetX -
xAxisLabel.GetOrigin.GetX
        theCount = theCount + 1
        if (theCount > 10) then
          break
        end
      end
      yAxisLabel.GetSymbol.SetSize(xAxisLabel.GetSymbol.GetSize)
      yAxisLabel.Invalidate
    end
  end
  lastSize = xAxisLabel.GetSymbol.GetSize
end

' place vertical exaggeration text

subTitleSymbol.SetSize((lastSize - 2) Max 4)
if (verEx < 0.000001) then
  format = "d.ddddddd"
elseif ((verEx > 0.000001) and (verEx < 0.00001)) then
  format = "d.ddddddd"
elseif ((verEx > 0.00001) and (verEx < 0.0001)) then
  format = "d.ddddd"
elseif ((verEx > 0.0001) and (verEx < 0.001)) then
  format = "d.dddd"
elseif ((verEx > 0.001) and (verEx < 0.01)) then
  format = "d.ddd"
elseif ((verEx > 0.01) and (verEx < 0.1)) then
  format = "d.dd"
elseif ((verEx > 0.1) and (verEx < 10)) then
  format = "d.d"
else
  format = "d"
end
verEx.SetFormat(format)

```

```

    verExString = verEx.AsString
    subtitleLabel = GraphicText.Make("Vertical exaggeration" ++
verExString ++ "X",0@0)
    subtitleLabel.SetSymbol(subTitleSymbol)
    subtitleLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(subtitleLabel)

    ' place x axis numbers
    numberSymbol.SetSize((subtitleLabel.GetSymbol.GetSize - 2) Max 4)
    if (totalDistance < 10) then
        xFormat = "d.ddd"
    elseif ((totalDistance > 10) and (totalDistance < 100)) then
        xFormat = "d.dd"
    elseif ((totalDistance > 100) and (totalDistance < 1000)) then
        xFormat = "d.d"
    else
        xFormat = "d"
    end
    xEndLabel =
GraphicText.Make(totalDistance.SetFormat(xFormat).AsString,0@0)
    xEndLabel.SetSymbol(numberSymbol)
    xEndLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(xEndLabel)
    xStartLabel = GraphicText.Make("0",0@0)
    xStartLabel.SetSymbol(numberSymbol)
    xStartLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(xStartLabel)

    ' place y axis numbers
    if ((zShift + theZRange) < 10) then
        yFormat = "d.ddd"
    elseif (((zShift + theZRange) > 10) and ((zShift + theZRange) < 100))
then
        yFormat = "d.dd"
    elseif (((zShift + theZRange) > 100) and ((zShift + theZRange) <
1000)) then
        yFormat = "d.d"
    else
        yFormat = "d"
    end
    yEndLabel =
theZRange).SetFormat(yFormat).AsString,0@0)
    yEndLabel.SetSymbol(numberSymbol)
    yEndLabel.SetAngle(90)
    yEndLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(yEndLabel)
    yStartLabel =
GraphicText.Make(zShift.SetFormat(yFormat).AsString,0@0)
    yStartLabel.SetSymbol(numberSymbol)
    yStartLabel.SetAngle(90)
    yStartLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(yStartLabel)

    ' find size of font for numbers of each grid line

    if (numberSymbol.GetSize > 4) then

```

```

' set number symbol font so that the number is not longer than
2/3rds
' of x or y grid distance

if (profileOptions.Get(7) and (profileOptions.Get(8) <> "")) then
  xGap = profileOptions.Get(8).AsNumber
  numberWidth = xEndLabel.GetExtent.GetX - xEndLabel.GetOrigin.GetX
  xGridWidth = (xGap * xScale) * (2 / 3)
  theXSize = numberSymbol.GetSize
  if (numberWidth > xGridWidth) then
    maxSize = theXSize
    minSize = 4
    theCount = 0
    while ((numberWidth - xGridWidth).Abs > (xGridWidth * 0.1))
      theDiff = (maxSize - minSize) / 2
      if (numberWidth > xGridWidth) then
        theXSize = theXSize - theDiff
        maxSize = theXSize
      else
        theXSize = theXSize + theDiff
        minSize = theXSize
      end
      xEndLabel.GetSymbol.SetSize(theXSize)
      xEndLabel.Invalidate
      numberWidth = xEndLabel.GetExtent.GetX -
xEndLabel.GetOrigin.GetX
      theCount = theCount + 1
      if (theCount > 10) then
        break
      end
    end
  end
end

if (profileOptions.Get(9) and (profileOptions.Get(10) <> "")) then
  yGap = profileOptions.Get(10).AsNumber
  numberWidth = yEndLabel.GetExtent.GetY - yEndLabel.GetOrigin.GetY
  yGridWidth = (yGap * yScale) * (2 / 3)
  theYSize = numberSymbol.GetSize
  if (numberWidth > yGridWidth) then
    maxSize = theYSize
    minSize = 4
    theCount = 0
    while ((numberWidth - yGridWidth).Abs > (yGridWidth * 0.1))
      theDiff = (maxSize - minSize) / 2
      if (numberWidth > yGridWidth) then
        theYSize = theYSize - theDiff
        maxSize = theYSize
      else
        theYSize = theYSize + theDiff
        minSize = theYSize
      end
      yEndLabel.GetSymbol.SetSize(theYSize)
      yEndLabel.Invalidate
      numberWidth = yEndLabel.GetExtent.GetY -
yEndLabel.GetOrigin.GetY
      theCount = theCount + 1
      if (theCount > 10) then

```

```

        break
    end
end
end
end
if ((profileOptions.Get(7) and (profileOptions.Get(8) <> "")) and
    (profileOptions.Get(9) and (profileOptions.Get(10) <> "")).Not)
then
    numberSymbol.SetSize(theXSize)
    elseif ((profileOptions.Get(7) and (profileOptions.Get(8) <>
"")).Not and
            (profileOptions.Get(9) and (profileOptions.Get(10) <> "")))
then
    numberSymbol.SetSize(theYSize)
    elseif ((profileOptions.Get(7) and (profileOptions.Get(8) <> ""))
and
            (profileOptions.Get(9) and (profileOptions.Get(10) <> "")))
then
    numberSymbol.SetSize(theXSize Min theYSize)
end
end

' adjust axis number fonts and positions
xStartLabel.GetSymbol.SetSize(numberSymbol.GetSize)
xStartLabel.Invalidate
yStartLabel.GetSymbol.SetSize(numberSymbol.GetSize)
yStartLabel.Invalidate
xEndLabel.GetSymbol.SetSize(numberSymbol.GetSize)
xEndLabel.Invalidate
yEndLabel.GetSymbol.SetSize(numberSymbol.GetSize)
yEndLabel.Invalidate
xOrigin = ((totalDistance * xScale) + xShift) -
((xEndLabel.GetExtent.GetX - xEndLabel.GetOrigin.GetX) / 2)
yOrigin = yShift - ((xEndLabel.GetExtent.GetY -
xEndLabel.GetOrigin.GetY) * 0.75)
numberPoint = Point.Make(xOrigin,yOrigin)
xEndLabel.SetOrigin(numberPoint)
xOrigin = xShift
yOrigin = yShift - ((xStartLabel.GetExtent.GetY -
xStartLabel.GetOrigin.GetY) * 0.75)
numberPoint = Point.Make(xOrigin,yOrigin)
xStartLabel.SetOrigin(numberPoint)
xOrigin = xShift - ((yEndLabel.GetExtent.GetX -
yEndLabel.GetOrigin.GetX) * 1.1)
yOrigin = (yShift + (theZRange * yScale)) -
((yEndLabel.GetExtent.GetY - yEndLabel.GetOrigin.GetY) / 2)
numberPoint = Point.Make(xOrigin,yOrigin)
yEndLabel.SetOrigin(numberPoint)
xOrigin = xShift - ((yStartLabel.GetExtent.GetX -
yStartLabel.GetOrigin.GetX) * 1.1)
yOrigin = yShift - ((yStartLabel.GetExtent.GetY -
yStartLabel.GetOrigin.GetY) / 2)
numberPoint = Point.Make(xOrigin,yOrigin)
yStartLabel.SetOrigin(numberPoint)

' place x grid
if (profileOptions.Get(7) and (profileOptions.Get(8) <> "")) then

```

```

xGrid = profileOptions.Get(8).AsNumber
xDistance = xGrid
while (xDistance < totalDistance)
    pointA = Point.Make(((xDistance * xScale) + xShift),yShift)
    pointB = Point.Make(((xDistance * xScale) + xShift),yShift +
(yScale * theZRange))
    aLine = GraphicShape.Make(Line.Make(pointA,pointB))
    aLine.SetSymbol(graphSymbol)
    aLine.SetSelected(TRUE)
    theLayout.GetGraphics.Add(aLine)
    xLabel
GraphicText.Make(xDistance.SetFormat(xFormat).AsString,0@0)
    xLabel.SetSymbol(numberSymbol)
    xLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(xLabel)
    xOrigin = ((xDistance * xScale) + xShift) -
((xLabel.GetExtent.GetX - xLabel.GetOrigin.GetX) / 2)
    yOrigin = yShift - ((xLabel.GetExtent.GetY -
xLabel.GetOrigin.GetY) * 0.75)
    numberPoint = Point.Make(xOrigin,yOrigin)
    xLabel.SetOrigin(numberPoint)
    xDistance = xDistance + xGrid
    if ((xDistance - totalDistance).Abs < (xGrid / 2)) then
        break
    end
end
pointA = Point.Make(((totalDistance * xScale) + xShift),yShift)
pointB = Point.Make(((totalDistance * xScale) + xShift),yShift +
(yScale * theZRange))
aLine = GraphicShape.Make(Line.Make(pointA,pointB))
aLine.SetSymbol(graphSymbol)
aLine.SetSelected(TRUE)
theLayout.GetGraphics.Add(aLine)
end

' place y grid
if (profileOptions.Get(9) and (profileOptions.Get(10) <> "")) then
yGrid = profileOptions.Get(10).AsNumber
yDistance = yGrid
while (yDistance < theZRange)
    pointA = Point.Make(xShift,yShift + (yDistance * yScale))
    pointB = Point.Make(xShift + (xScale * totalDistance),yShift +
(yDistance * yScale))
    aLine = GraphicShape.Make(Line.Make(pointA,pointB))
    aLine.SetSymbol(graphSymbol)
    aLine.SetSelected(TRUE)
    theLayout.GetGraphics.Add(aLine)
    yLabel = GraphicText.Make((zShift
yDistance).SetFormat(yFormat).AsString,0@0)
    yLabel.SetSymbol(numberSymbol)
    yLabel.SetAngle(90)
    yLabel.SetSelected(TRUE)
    theLayout.GetGraphics.Add(yLabel)
    xOrigin = xShift - ((yLabel.GetExtent.GetX -
yLabel.GetOrigin.GetX) * 1.1)
    yOrigin = (yShift + (yDistance * yScale)) -
((yLabel.GetExtent.GetY - yLabel.GetOrigin.GetY) / 2)

```

```

    numberPoint = Point.Make(xOrigin,yOrigin)
    yLabel.SetOrigin(numberPoint)
    yDistance = yDistance + yGrid
    if ((yDistance - theZRange).Abs < (yGrid / 2)) then
        break
    end
end
end
pointA = Point.Make(xShift,yShift + (theZRange * yScale))
pointB = Point.Make(xShift + (xScale * totalDistance),yShift +
(theZRange * yScale))
aLine = GraphicShape.Make(Line.Make(pointA,pointB))
aLine.SetSymbol(graphSymbol)
aLine.SetSelected(TRUE)
theLayout.GetGraphics.Add(aLine)
end

' place profiles
if (theProfileList <> NIL) then
    for each p in theProfileList
        if (p.Get(0).GetName <> "Line of Sight Graphic") then
            theOffset = 0
            profileLinePoints = {}
            theLines = p.Get(0).AsList
            lastPoint = theLines.Get(0).Get(0)
            for each l in theLines
                tempLine = PolyLine.Make({l})
                theDistance = tempLine.ReturnLength
                theOffset = theOffset + lastPoint.Distance(l.Get(0))
                linePoints = {}
                for each pt in l
                    theX = (((tempLine.PointPosition(pt) / 100) * theDistance)
+ theOffset) * xScale) + xShift
                    theY = ((pt.GetZ - zShift) * yScale) + yShift
                    linePoints.Add(Point.Make(theX,theY))
                end
                profileLinePoints.Add(linePoints)
                lastPoint = l.Get(l.Count - 1)
                theOffset = theOffset + theDistance
            end
            profileGraphic
GraphicShape.Make(PolyLine.Make(profileLinePoints))
            profileGraphic.SetSymbol(p.Get(1))
            profileGraphic.SetSelected(TRUE)
            theLayout.GetGraphics.Add(profileGraphic)
        else
            visLines = p.Get(0).GetGraphics.Get(0)
            nonVisLines = p.Get(0).GetGraphics.Get(1)
            theLine = p.Get(0).GetObjectTag.Get(0)
            observer = p.Get(0).GetObjectTag.Get(1)
            target = p.Get(0).GetObjectTag.Get(2)
            if (visLines.GetShape.IsNull.Not and
                (visLines.GetShape.AsList.Get(0).Get(0).GetX
theLine.ReturnStart.GetX) and
                (visLines.GetShape.AsList.Get(0).Get(0).GetY
theLine.ReturnStart.GetY)) then
                firstLines = visLines.GetShape.AsList
                firstSymbol = visLines.GetSymbol
            end
        end
    end
end

```

```

        secondLines = nonVisLines.GetShape.AsList
        secondSymbol = nonVisLines.GetSymbol
    else
        firstLines = nonVisLines.GetShape.AsList
        firstSymbol = nonVisLines.GetSymbol
        secondLines = visLines.GetShape.AsList
        secondSymbol = visLines.GetSymbol
    end
    theFirstCount = 0
    theSecondCount = 0
    theOffset = 0
    drawFirstLines = TRUE
    moreToDraw = TRUE
    while (moreToDraw)
        if (drawFirstLines) then
            drawSymbol = firstSymbol
            l = firstLines.Get(theFirstCount)
            theFirstCount = theFirstCount + 1
        else
            drawSymbol = secondSymbol
            l = secondLines.Get(theSecondCount)
            theSecondCount = theSecondCount + 1
        end
        theSegment = PolyLine.Make({l})
        theDistance = theSegment.ReturnLength
        profileLinePoints = {}
        for each pt in l
            theX = (((theSegment.PointPosition(pt) / 100) *
theDistance) + theOffset) * xScale) + xShift
            theY = ((pt.GetZ - zShift) * yScale) + yShift
            profileLinePoints.Add(Point.Make(theX,theY))
        end
        profileGraphic =
GraphicShape.Make(PolyLine.Make({profileLinePoints}))
        profileGraphic.SetSymbol(drawSymbol)
        profileGraphic.SetSelected(TRUE)
        theLayout.GetGraphics.Add(profileGraphic)
        theOffset = theOffset + theDistance
        firstGap = -1
        secondGap = -1
        if (theFirstCount < firstLines.Count) then
            firstGap =
firstLines.Get(theFirstCount).Get(0).Distance(l.Get(l.Count - 1))
        end
        if (theSecondCount < secondLines.Count) then
            secondGap =
secondLines.Get(theSecondCount).Get(0).Distance(l.Get(l.Count - 1))
        end
        if (firstGap = -1) then
            if (secondGap = -1) then
                moreToDraw = FALSE
            else
                drawFirstLines = FALSE
                theOffset = theOffset + secondGap
            end
        elseif (secondGap = -1) then
            drawFirstLines = TRUE
        end
    end

```

```

        theOffset = theOffset + firstGap
    else
        if (firstGap < secondGap) then
            drawFirstLines = TRUE
            theOffset = theOffset + firstGap
        else
            drawFirstLines = FALSE
            theOffset = theOffset + secondGap
        end
    end
end
thePointSymbol = Symbol.Make(#SYMBOL_MARKER)
theFont = Font.Make("ESRI Geometric Symbols","Normal")
thePointSymbol.SetFont(theFont)
thePointSymbol.SetCharacter(35)
thePointSymbol.SetColor(Color.GetBlack)
thePointSymbol.SetSize(8)
theLOSSymbol = Symbol.Make(#SYMBOL_PEN)
theLOSSymbol.SetPattern({3,3})
theLOSSymbol.SetColor(Color.GetGray)
theLOSSymbol.SetSize(2)
profileGraphic
GraphicShape.Make(Line.Make(Point.Make(xShift,((observer.GetZ - zShift)
* yScale) + yShift),Point.Make(xShift + (totalDistance *
xScale),((target.GetZ - zShift) * yScale) + yShift)))
profileGraphic.SetSymbol(theLOSSymbol)
profileGraphic.SetSelected(TRUE)
theLayout.GetGraphics.Add(profileGraphic)
profileGraphic
GraphicShape.Make(Point.Make(xShift,((observer.GetZ - zShift) * yScale)
+ yShift))
profileGraphic.SetSymbol(thePointSymbol)
profileGraphic.SetSelected(TRUE)
theLayout.GetGraphics.Add(profileGraphic)
profileGraphic = GraphicShape.Make(Point.Make(xShift +
(totalDistance * xScale),((target.GetZ - zShift) * yScale) + yShift))
profileGraphic.SetSymbol(thePointSymbol)
profileGraphic.SetSelected(TRUE)
theLayout.GetGraphics.Add(profileGraphic)
end
end
else
    MsgBox.Error("Profiles not found
in"++profileOptions.Get(0),"Profile Graph")
end

' move text to proper places
if (profileOptions.Get(1) and (profileOptions.Get(2) <> "")) then
    xOrigin = (xShift + ((totalDistance * xScale) / 2)) -
((theTitle.GetExtent.GetX - theTitle.GetOrigin.GetX) / 2)
    yOrigin = (yShift + (theZRange * yScale)) +
((theTitle.GetExtent.GetY - theTitle.GetOrigin.GetY) / 2) +
(subtitleLabel.GetExtent.GetY - subtitleLabel.GetOrigin.GetY)
    numberPoint = Point.Make(xOrigin,yOrigin)
    theTitle.SetOrigin(numberPoint)
end
if (profileOptions.Get(5) and (profileOptions.Get(6) <> "")) then

```

```

        xOrigin      =      xShift      -      ((yAxisLabel.GetExtent.GetX      -
yAxisLabel.GetOrigin.GetX) * 2)
        yOrigin      =      (yShift      +      ((theZRange      *      yScale)      /      2))      -
((yAxisLabel.GetExtent.GetY - yAxisLabel.GetOrigin.GetY) / 2)
        numberPoint = Point.Make(xOrigin,yOrigin)
        yAxisLabel.SetOrigin(numberPoint)
    end
    if (profileOptions.Get(3) and (profileOptions.Get(4) <> "")) then
        xOrigin      =      (xShift      +      ((totalDistance      *      xScale)      /      2))      -
((xAxisLabel.GetExtent.GetX - xAxisLabel.GetOrigin.GetX) / 2)
        yOrigin      =      yShift      -      ((xAxisLabel.GetExtent.GetY      -
xAxisLabel.GetOrigin.GetY) * 1.5)
        numberPoint = Point.Make(xOrigin,yOrigin)
        xAxisLabel.SetOrigin(numberPoint)
    end
    xOrigin      =      (xShift      +      ((totalDistance      *      xScale)      /      2))      -
((subTitleLabel.GetExtent.GetX - subTitleLabel.GetOrigin.GetX) / 2)
    yOrigin      =      (yShift      +      (theZRange      *      yScale))      +
((subTitleLabel.GetExtent.GetY - subTitleLabel.GetOrigin.GetY) / 2)
    numberPoint = Point.Make(xOrigin,yOrigin)
    subTitleLabel.SetOrigin(numberPoint)

' group graphics
theLayout.GetGraphics.GroupSelected

```