**The Influence Of Cut-Block Size And Adjacency
Rules On Harvest Levels And Road Networks**

by

Steven Terence Finn

B.S.F., The University Of British Columbia, 1987

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF FORESTRY**

in

THE FACULTY OF GRADUATE STUDIES

Department of Forestry

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

October 1994

© Steven Terence Finn, 1994

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

(Signature)

Department of FORESTRY

The University of British Columbia
Vancouver, Canada

Date OCT 5 1994

ABSTRACT

Previous harvesting practices in coastal British Columbia generally favoured large opening sizes with relatively short regeneration periods before adjacent blocks were removed. These large openings were both criticized for being aesthetically unpleasing and for not providing a proper age class distribution for wildlife habitat. To combat these problems, smaller opening sizes with longer times between harvesting of adjacent blocks have been proposed.

It is believed that smaller opening sizes and more rigid adjacency constraints will impact cutting levels and harvesting and transportation costs. An area-based scheduling model was developed between 1989 and 1991 which quantifies these impacts and provides a better understanding of the harvest levels and costs associated with different cut block sizes and adjacency constraints.

The effect on the Annual Allowable Cut (AAC) due to block size and exclusion period showed that the delay period had a greater effect on the AAC than did block size. As the block size decreased while keeping the delay period constant, the reduction in AAC ranged from 6.0% to 23.5%.

As the delay or exclusion period increased, adjacency constraints limited the number of blocks available for harvest and therefore reduced the harvest volume. Increasing the delay period from 1 to 2 and from 1 to 3 caused volumes to drop an average of 18.6% and 40% respectively. Increasing the delay period and decreasing the block size resulted in large reductions in harvest volume of up to 50%.

Another effect of increasing the delay period and decreasing the block size is that a large number of blocks may never be harvested. Values ranged from 6.8% to 43.8% of blocks not eligible for harvesting.

A stumpage calculation showed that there is a drastic difference in the amount of stumpage payable when comparing the different sized blocks and adjacency requirements. Values ranged from \$119.7 million to \$52.8 million, a reduction of nearly \$67 million as the block size was decreased and the adjacency requirements increased.

TABLE OF CONTENTS

ABSTRACT	ii
LIST OF TABLES	iv
LIST OF FIGURES	v
ACKNOWLEDGEMENTS	vi
INTRODUCTION	1
PROBLEM BACKGROUND	1
OBJECTIVES	1
LITERATURE REVIEW	2
LITERATURE RELATING TO STRATA BASED PLANS	3
LITERATURE RELATING TO AREA BASED PLANS	5
METHODOLOGY	6
DATA FORMULATION AND COMPILATION	6
MCIP FORMULATION / INFORMATION REQUIRED	8
THE RANDOM SEARCH TECHNIQUE - BASIC STEPS	12
RESULTS AND DISCUSSION	14
PROGRESSIVE CLEAR CUT SOLUTION	14
UNCONSTRAINED ROAD SOLUTIONS	15
CONSTRAINED ROAD SOLUTION	19
CONSTRAINED VERSUS UNCONSTRAINED SOLUTION	23
THE EFFECT ON THE AAC DUE TO BLOCK SIZE AND EXCLUSION PERIOD	25
ECONOMIC ANALYSIS	29
IMPACT ON STUMPAGE	35
CONCLUSION	37
BIBLIOGRAPHY	40
APPENDIX 1. DESCRIPTION OF ANALYSIS AREAS	42
APPENDIX 2. ANALYSIS AREA MAP / ZONE MAP	45
APPENDIX 3. DBASE PROGRAM TO MAKE TABLES OF BLOCK VOLUMES	48
APPENDIX 4. BLOCK LAYOUT AND ROAD NETWORKS BY BLOCK SIZE	52
APPENDIX 5. INPUT FILE EXAMPLES	56
APPENDIX 6. MCIP FLOW CHART	70
APPENDIX 7. MAIN PROGRAM FILE	72
APPENDIX 8. OUTPUT FILE EXAMPLES	94
APPENDIX 9. TYPICAL PROGRAM SOLUTIONS	98

LIST OF TABLES

Table 1: Summary Of Files Used In The MCIP Model	9
Table 2: Zone Number And Earliest Possible Entry	13
Table 3: Average Harvest Volumes And Percent Reductions For The Unconstrained Road Construction Solutions	16
Table 4: Average Harvest Volumes And Percent Reductions For The Constrained Road Construction Solutions	19
Table 5: Difference In Volumes Harvested Between The Constrained And Unconstrained Solutions	23
Table 6: Total Amount Of Road Constructed (In Kilometers) For Block Size And Delay Period Combinations	24
Table 7: Percentage Of Blocks Not Harvested During The Planning Period	26
Table 8: Range Of Road Construction, Hauling And Maintenance Costs Per Cubic Meter Harvested By Block Size And Delay Period	29
Table 9: Range Of Tree To Truck Costs Per Cubic Meter By Block Size And Delay Period	30
Table 10: Range Of Total Delivered Cost Of Wood (\$/M3) By Block Size And Delay Period	30
Table 11: Range Of Operating Costs (\$/M3) By Block Size And Delay Period	35
Table 12: Range Of Indicated Stumpage (\$/M3) By Block Size And Delay Period	36
Table 13: Range Of Stumpage Revenues (in millions of \$) By Block Size And Delay Period	37

LIST OF FIGURES

Figure 1: Road Construction Schedule For The Progressive Clear-Cut Solution	15
Figure 2: Road Maintenance Schedule For The Progressive Clear-Cut Solution	15
Figure 3: Road Construction Schedule For The Unconstrained Road	
Solution - 1 Decade Exclusion Period	16
Figure 4: Road Construction Schedule For The Unconstrained Road	
Solution - 2 Decade Exclusion Period	17
Figure 5: Road Construction Schedule For The Unconstrained Road	
Solution - 3 Decade Exclusion Period	17
Figure 6: Road Maintenance Schedule For The Unconstrained Road	
Solution - 1 Decade Exclusion Period	18
Figure 7: Road Maintenance Schedule For The Unconstrained Road	
Solution - 2 Decade Exclusion Period	18
Figure 8: Road Maintenance Schedule For The Unconstrained Road	
Solution - 3 Decade Exclusion Period	19
Figure 9: Road Construction Schedule For The Constrained Road	
Solution - 1 Decade Exclusion Period	20
Figure 10: Road Construction Schedule For The Constrained Road	
Solution - 2 Decade Exclusion Period	20
Figure 11: Road Construction Schedule For The Constrained Road	
Solution - 3 Decade Exclusion Period	21
Figure 12: Road Maintenance Schedule For The Constrained Road	
Solution - 1 Decade Exclusion Period	21
Figure 13: Road Maintenance Schedule For The Constrained Road	
Solution - 2 Decade Exclusion Period	22
Figure 14: Road Maintenance Schedule For The Constrained Road	
Solution - 3 Decade Exclusion Period	22
Figure 15: Total Delivered Cost By Decade - 80 ha Blocks, 1	
Decade Exclusion Period	31
Figure 16: Total Delivered Cost By Decade - 80 ha Blocks, 2	
Decade Exclusion Period	31
Figure 17: Total Delivered Cost By Decade - 80 ha Blocks, 3	
Decade Exclusion Period	32
Figure 18: Total Delivered Cost By Decade - 40 ha Blocks, 1	
Decade Exclusion Period	32
Figure 19: Total Delivered Cost By Decade - 40 ha Blocks, 2	
Decade Exclusion Period	33
Figure 20: Total Delivered Cost By Decade - 40 ha Blocks, 3	
Decade Exclusion Period	33
Figure 21: Total Delivered Cost By Decade - 20 ha Blocks, 1	
Decade Exclusion Period	34
Figure 22: Total Delivered Cost By Decade - 20 ha Blocks, 2	
Decade Exclusion Period	34
Figure 23: Total Delivered Cost By Decade - 20 ha Blocks, 3	
Decade Exclusion Period	35

ACKNOWLEDGEMENTS

I would like to thank the following people for their help with this thesis. First, I would like to thank Dr. John Nelson of the Faculty of Forestry at UBC for his initial topic suggestion, computer programming skills, proof reading of drafts and finally his guidance and support throughout my graduate work at UBC. Second, Ian Thomas and Dave Daust of the UBC Harvest Research Group were of great assistance with their skills in computers and Geographical Information Systems.

Appreciation is also extended to the Staff and Management of the Renewable Resources Technology at the British Columbia Institute of Technology for their computer and educational leave support.

Finally, to my wife and children, I thank you for your never ending patience and support.

INTRODUCTION

PROBLEM BACKGROUND

Previous harvesting practices in coastal British Columbia generally favoured large opening sizes with relatively short regeneration ("free to grow") periods before adjacent blocks were removed. Sometimes viewed as a progressive clear-cut, these large openings were criticized for being aesthetically unpleasing and for not providing a proper age class distribution for wildlife habitat. With the growing importance of environmental and other non-timber issues, it is likely that today's harvesting practices will have to be changed or modified to reflect these concerns. In some areas this may mean a switch to an alternative to clear-cutting, such as selection cutting. However, in many areas of the coast selection cutting is not viable from a silvicultural prospective. Therefore, clear-cutting will likely continue but with smaller opening sizes and with longer times between harvesting of adjacent blocks.

OBJECTIVES

It is believed that smaller opening sizes and more rigid adjacency constraints will impact cutting levels and harvesting and transportation costs. An area-based scheduling model was developed which quantifies these impacts and provides a better understanding of the harvest levels and costs associated with different cut block sizes and adjacency constraints.

Using a case study approach, this paper will quantify the effect on the Annual Allowable Cut (AAC) due to the reduction of block size in conjunction with more rigid adjacency constraints when:

- 1) there is no limit on road construction and;
- 2) there is a road construction limit.

Specifically, this paper will:

- 1) compare the constrained and unconstrained road budget model solutions with regard to volume harvested and road construction and maintenance schedules.
- 2) determine the effect on the AAC due to block size and exclusion period through a comparison of volume harvested;
- 3) quantify the total delivered wood cost for combinations of block and exclusion periods and ;
- 4) determine the impact of smaller cut-blocks on stumpage revenues.

LITERATURE REVIEW

Forest planning models can be segregated into two main categories according to the length of the planning horizon and the level of detail required to address resource allocation issues (Nelson and Brodie, 1990). First, there are the long-range strategic or strata based plans that look ahead over one or more rotations. Strata based plans give minimal area resolution with no operational detail. Second, there are the tactical or area based plans that cover a shorter planning horizon than those of the strata based plans. Area based plans are area specific with operational detail.

LITERATURE RELATING TO STRATA BASED PLANS

The most widely used technique for timber harvest scheduling in the United States is linear programming (LP) (O'Hara et al., 1989). One of the earliest models developed was Timber RAM (Resource Allocation Model) which did not include any spatial considerations. This was followed by other models such as MUSYC (Multiple Use Sustained Yield Calculator) and FORPLAN (Forest Planning Model) which attempted to deal more effectively with site-specific environmental questions.

Armel (1986) noted that one of the most frequently asked questions by forest managers concerns how the allocations represented by the standard, strata based FORPLAN solution, in which homogeneous forest units are aggregated, can be implemented within a heterogeneous area represented by a given parcel of national forest land. Complications result from the specific placement and management of habitats for wildlife such as spotted owls or pileated woodpeckers, and from the need to consider harvest adjacency constraints. The standard FORPLAN solution does not consider these factors. Approaching the problem by the means of the "coordinated allocation choices" option in FORPLAN version II does not lead to a satisfactory solution to this problem because representation of the problem at the harvest-unit level produces a problem of unmanageable size.

O'Hara et al. (1989) state that a problem shared by all LP approaches to solving spatially constrained timber harvest scheduling problem is that the solutions are not integral. Commonly units are split to meet the spatial constraints. In a mathematical sense, the constraints are met, but in practical terms, field implementation of the solution is not possible unless the solution is integral.

Bare et al. (1984) note that explicit recognition of spatial relationships and the consequent constraints on timber harvest scheduling are important for a number of reasons. First, management for multiple use requires that managers know the geographic location of specific outputs and how much of the output to produce in a given time period. Second, failure to recognize spatial relationships can also result in environmental problems. Where a buffer strip is to be retained, such as around streams, lakes, recreation areas, roads, or wilderness areas, or where the buffer strip is necessary for regeneration purposes, the spatial relationship between the timber scheduled for harvest and the adjacent stand or feature needs to be recognized. Third, the spatial relationship between harvest units must be recognized so that specific wildlife habitat objectives, such as maintenance of adequate degrees of habitat diversity can be met.

The lack of spatial resolution and site specific data is the major disadvantage of strata based models (Nelson and Brodie, 1990). They further state that the principal advantage of strata based models is that individual timber stands with similar physical and economic characteristics can be aggregated, making forest wide planning a task that is computationally feasible. Therefore these models are only adequate for setting strategic harvest goals within a temporal context.

LITERATURE RELATING TO AREA BASED PLANS

In an area based plan individual harvest units (blocks) and road systems are specified creating the ability to define a spatially feasible solution.

Since area based plans require integer solutions, these problems are generally solved using mixed integer programming (MIP) techniques (Nelson, 1988). Integer variables are needed to specify if a road is or is not built, and to specify if a harvest unit is or is not cut. This alleviates the problem of splitting harvest units between time periods. However, due to these integer restrictions only very small problems can be solved using MIP.

As an alternative to MIP and simulation, a random search algorithm called Monte-Carlo Integer Programming (MCIP) has been used to generate feasible solutions to area based problems (Nelson and Brodie 1990). According to Clements et al. (1990), "a typical MCIP algorithm begins by generating random solutions to a mixed-integer programming (MIP) problem. These solutions are tested against a set of spatial and temporal constraints, and solutions meeting all of the constraints are designated feasible. Each feasible solution is evaluated relative to an objective function. After a large number feasible solutions have been identified, the solutions best satisfying the objective function are selected for further analysis." Unlike MIP, MCIP does not guarantee optimality, however, it is capable of quickly generating feasible solutions to the complex integer problems associated with planning. Nelson and Brodie (1990) were

able to find a MCIP solution with a objective function solution that was within 3% of the true optimum. They concluded that by using MCIP, it is relatively easy to find several solutions with values that lie within 10% of the optimum.

METHODOLOGY

A 4500-hectare subunit of MacMillan Bloedel's Stillwater Logging Division was used as a case study area. This study is broken into two major components, the first being the data formulation and compilation and the second being the MCIP analysis.

DATA FORMULATION AND COMPILATION

Nelson (1988) utilized the same study area and therefore most of the inventory data used were taken from his paper. There are 109 individual stands grouped into 62 analysis areas (AA's). An analysis area is a timber stratum based on zone, logging system, site, species and age. See Appendix 1 and 2, pages 42 and 45 for a complete description of the analysis areas and an analysis area map.

Using a 1:20,000 scale map of the study area in conjunction with the analysis areas, cut blocks of 80-hectares were formulated using the following guidelines:

- average yarding distance of 200-m.,
- maximum yarding distance of 250-m.,
- blocks to be made up of similar age classes.

The 80 ha. blocks were split in half to make 40 ha. blocks. The 40 ha. blocks were subsequently split in half to make 20 ha. blocks. The blocks were split so that the same logging system, and the same main roads could be used in all cases. The results were the formulation of forty-eight 80 ha. blocks, ninety-six 40 ha. blocks and one hundred ninety-two 20 ha. blocks. (Block layout and road network maps are included in Appendix 4, page 52). The blocks were then input into a Geographic Information

System (GIS). GIS processing was carried out with the result being themes that give the total area of each block and the area of each analysis area within the block.

A stand growth model, Stand Projection System (SPS), was used to generate timber volumes by analysis area. These were then input into a database file and a GIS theme was created to produce the volume per hectare by analysis area by decade. This process was done for both the existing stands and for the projected regenerated stands. The block theme was overlaid onto the yield theme to generate the area and volume per hectare per decade for each block.

An small dBase program (Appendix 3, page 48) was then written that produced, for each block size, a table of existing and regenerated volumes per block per decade.

At the same time as the blocks were formulated, a road network was created (Appendix 4, page 52). As previously stated, the same main roads were used for all three block sizes. For each block, the road needed to log that particular block was tracked by noting the road links required to access that block. While the total length of the road network remained constant, shorter and more numerous road links were needed as the block size decreased. The number of road links required were 162, 185 and 292 for the 80, 40 and 20 ha. blocks respectively. For a limited number of blocks secondary roads were also required. Secondary roads are defined as roads that are needed only to harvest the block that contained them.

MCIP FORMULATION/INFORMATION REQUIRED

BLOCK DATA

For each block, the area, current age, zone and adjacent block numbers are required. The area is calculated from the GIS data processing. The existing stand age is an average, to the closest 10 years, of the stands that make up the block. The entire forest was broken into 8 zones to help the block selection process (Appendix 2, page 45). Each block, therefore, has a zone number. For each block the adjacent blocks were identified. An adjacent block is a cut block whose boundary touches the selected cut block, regardless of the length of common boundary. All the block data were incorporated into a file called blockdat.txt. For examples of all the input files used, see Appendix 5, page 56.

EXISTING AND REGENERATION BLOCK VOLUME TABLE

The total existing and regeneration volume per block per period was incorporated into files called exvoltab.txt and rgvoltab.txt, respectively.

EXISTING AND REGENERATION BLOCK REVENUE TABLE

For each block the estimated logging cost was calculated using the formula derived by Nawitka Resource Consultants (1987):

$$\text{LOGGING COST (\$/M3)} = -0.3316(\text{age}) + 43.482$$

The above formula was derived for a site of average logging difficulty. As the stands covered such a wide variety of topography, each block was assigned a logging difficulty of high, medium or low based on topography and expected logging difficulty. The logging cost formula was revised to reflect the high and low difficulty classes by adding or subtracting \$1/m³ respectively. The block revenue tables were incorporated into the files exrevtab.txt and rgrevtab.txt.

MAIN ROAD ACCESS

The road links required to access a particular block were incorporated into a file called mraccess.txt. Also included in this file was the transportation cost for each particular block. The transportation cost formula used was (Hackett, 1990):

$$\text{TRANS COST (\$/M3)} = 2.971 + .0575(\text{average weighted haul distance (one way)})$$

The haul distance for each block was calculated from the center of each block to the main access point of the valley.

ROAD LENGTHS AND CONSTRUCTION COSTS

For each main and secondary road to be constructed, the roads length and total construction cost were calculated and saved in a file called mrlinks.txt or srlinks.txt. Three categories of road construction cost were used, with the criteria being based on the topography. The estimated road costs were:

LOW \$ 30,000/km

MEDIUM \$ 45,000/km

HIGH \$ 60,000/km

Table 1: Summary of files used in the MCIP model

<u>File Name</u>	<u>Contents</u>
blockdat.txt	area, existing stand age, zone and adjacent block numbers of each block
constrai.txt	sets the parameters for use within the harvest planner (see below)
exrevtab.txt	estimated logging cost per block per period for existing stands
exvoltab.txt	existing volume per block per period
rgrevtab.txt	estimated logging cost per block per period for regenerated stands
rgvoltab.txt	regenerated volume per block per period
mraccess.txt	transportation costs and road links required to access each block
mrlinks.txt	main road lengths and construction costs
srlinks.txt	secondary road lengths and construction costs

CONSTRAINTS

The constraint file (constraints.txt) is the file that sets the parameters for use within the harvest planner. The information required to run the program is:

HABITAT DELAY - the number of periods (decades) required to pass before the adjacent block can be harvested.

MAXIMUM NUMBER OF BLOCKS - either 48, 96 or 192 depending on the block size being modelled.

MAXIMUM NUMBER OF PERIODS - the total length of the modelling period in decades (i.e. planning horizon).

MAXIMUM NUMBER OF ZONES - set at 8 for this model.

MINIMUM AGE - the minimum harvest age, in decades, for an existing or regeneration stand.

MAINTENANCE COST - the maintenance cost per meter for an previously constructed road. This was set at \$5100 per kilometer per year (Hackett, 1990) or \$51 per meter per period.

MAXIMUM ROAD - the maximum number of kilometers of road that can be constructed per period.

CUT 1 - the minimum volume that must be harvested in period 1.

CUT 2 - the minimum volume that must be harvested in period 2.

MINIMUM VOLUME - the minimum volume that must be harvested in any period other than periods 1 and 2.

MAXIMUM VOLUME - the maximum volume per period that can be harvested in any period.

MINIMUM COST - the minimum net revenue per period.

MAXIMUM COST - the maximum cost per period.

MAXIMUM GRADE COST - the maximum road construction and maintenance budget per period.

ZONE AND PERIOD OF FIRST ENTRY - the period of first entry into each particular zone. To prevent the early harvesting of cut-blocks at the far reaches of the road network, the forest was divided into 8 zones with each zone assigned a first entry time. (See appendix 2 ,page 45 for a map of the zones).

The objective function maximized the total volume produced over the planning horizon, subject to the constraints outlined above.

THE RANDOM SEARCH TECHNIQUE - BASIC STEPS

As the planning problems had approximately 2000 - 5000 integer variables, depending on block size, direct optimization techniques were impractical. As an alternative, a random search technique was used to find feasible solutions. This method is displayed on a flow chart in Appendix 6 ,page 70. The basic steps used by this technique are:

- 1) Set the planning period to 1.
- 2) All the blocks that are available for harvest in this period are assigned a random 0/1 variable. 1 identifies blocks that can be harvested in this period and 0 identifies blocks not to be harvested.
- 3) Let count be a variable to identify the number of attempts at finding a feasible solution for this period. Set count = count +1.
- 4) Queue the list of blocks available for harvest in step 2 according to decreasing accessibility. The blocks with the minimum number of main road links will be at the head of the queue.
- 5) Randomly select a block near the front of the queue and add its volume to the period harvest. The adjacent blocks to the selected block are made unavailable for harvest until the adjacency delay age has been met. The next available block is then selected and the process continues until the minimum harvest level in that period is met. If the minimum harvest level cannot be satisfied from the available blocks, return to the start of step 2 and try another random block selection. If the count variable exceeds 20, it is unlikely a solution exists and the harvest constraints must be reduced.
- 6) The road links needed to provide access to the harvested blocks are identified. The road links are checked to determine whether or not they have been constructed previously. If so they are assigned a maintenance cost. If not the road links are constructed, and the grade construction cost is calculated for this period.

- 7) The total cost for each period is calculated by summing the main and secondary road construction costs and the maintenance costs from each block harvested.
- 8) The constraints are checked. If the solution does not meet the constraints, return to step 3.
- 9) Adjust the block ages for harvest and growth and increment the planning period by 1. If the planning period is greater than the planning horizon, stop, otherwise go to step 2.

Preliminary program runs using the same constraints were executed to determine the suitable length of time for the program to complete a successful run. It was decided that runs should be kept to a maximum of 40, 60 and 90 seconds per solution for the 80, 40 and 20 hectare plans respectively. These were deemed reasonable times for a 80386 computer running at 20 Mhz. Refer to Appendix 7, page 72 for the main program file.

Many combinations of zones and entry times are possible and trial runs were carried out to provide a solution that seemed to maximize the annual allowable cut . Once identified, this zone/entry time frame was kept constant throughout the rest of the computer runs.

Table 2: Zone number and earliest possible entry

zone		entry time (period)
1		1
2		5
3		1
4		1
5		1
6		2
7		1
8		4

The minimum and maximum AAC were steadily increased until a set of 10 feasible solutions could not be found within the pre-determined time limits. Generally,

the minimum AAC was found by increasing the minimum AAC while keeping the maximum AAC large. When the minimum AAC was bounded, the maximum AAC was reduced in the direction of the minimum AAC until the solution was no longer feasible due to time constraints or no feasible solutions existed. See Appendix 8, page 94 for typical program solutions.

Feasible solutions were saved to files and their minimum and maximum AAC constraints, solution time, average AAC, AAC standard deviation and AAC range were recorded. The 'best' solution was determined as the solution with: 1) the smallest difference between the minimum and maximum AAC; 2) the largest average AAC and 3) the smallest standard deviation. Once the 'best' solution was identified, 50 solutions using the same constraints were generated. Finally, these 50 solutions were averaged for comparison purposes. Because MCIP is a random procedure, it was felt that the average of 50 solutions would provide a better bench mark than only the best solution. While the average is useful for evaluating trends, it does not represent a solution that is mapable. In hindsight, perhaps acceptance of the best solution is a preferred method. See Appendix 9, page 98 for typical block solutions by block size and delay period combinations.

Initially, no constraints were placed on the length of road that could be constructed during any one period. For comparison, the problem was re-solved with an additional constraint that limited road construction to 25 kilometers per period (ie 2.5 km. per year).

RESULTS AND DISCUSSION

PROGRESSIVE CLEAR CUT SOLUTION

To provide a datum, a progressive clear cut solution was generated using the 80 ha blocks. The blocks were progressively harvested from the beginning of the road

network without regard for adjacency constraint rules. For this particular block arrangement, the solution generated set the volume harvested per decade at 360 000 m³.

Figures 1 and 2 show the road construction and maintenance schedules for the progressive clear-cut solution.

Figure 1: Road construction schedule for the progressive clear-cut solution

Figure 2: Road maintenance schedules for the progressive clear-cut solution

UNCONSTRAINED ROAD SOLUTIONS

Table 3 summarizes the harvest volume by block size and exclusion period for the unconstrained road solutions.

Table 3: Average harvest volumes and percent reductions for the unconstrained road construction solutions

80 ha blocks				40 ha blocks				20 ha blocks			
delay period	harvest volume	% reduction from		harvest volume	% reduction from			harvest volume	% reduction from		
		clear cut	delay 1		clear cut	delay 1			clear cut	delay 1	
1	339910	5.5	0	332460	7.7	0		334673	7.0	0	
2	269963	25	21	264830	26	20		241223	33	28	
3	218621	39	36	192608	46	42		171497	52	49	

The percent reduction is calculated to :1) compare each solution to the progressive clear cut solution; and 2) compare the effect of increasing the exclusion period while maintaining the same block size.

Figures 3, 4 and 5 show the road construction schedules for each block size and exclusion period combination. Figures 6, 7 and 8 show the road maintenance schedules for each block size and exclusion period combination.

Figure 3: Road construction schedule for unconstrained road solution - 1 decade exclusion period

ROAD CONSTRUCTION NO LIMIT
2 DECADE EXCLUSION PERIOD

Figure 4: Road construction schedule for unconstrained road solution - 2 decade exclusion period

ROAD CONSTRUCTION NO LIMIT
3 DECADE EXCLUSION PERIOD

Figure 5: Road construction schedule for the unconstrained road solution - 3 decade exclusion period

Figure 6: Road maintenance schedule for the unconstrained road solution - 1 decade exclusion period

Figure 7: Road maintenance schedule for the unconstrained road solution - 2 decade exclusion period

Figure 8: Road maintenance schedule for the unconstrained road solution - 3 decade exclusion period

CONSTRAINED ROAD SOLUTIONS

Table 4 summarizes the harvest volume by block size and exclusion period for the constrained road solutions.

Table 4: Average harvest volumes and percent reductions for the constrained road construction solutions

delay period	80 ha blocks				40 ha blocks				20 ha blocks		
	harvest volume	% reduction from		harvest volume	clear cut	delay1	harvest volume	clear cut	delay 1		
		clear cut	delay 1								
1	340691	5.4	0	326556	9.3	0	321402	11	0		
2	276952	23	19	271697	24	17	255928	29	20		
3	221062	39	35	196498	45	40	179044	50	44		

The percent reduction is calculated to :1) compare each solution to the progressive clear cut solution; and 2)compare the effect of increasing the exclusion period while maintaining the same block size.

Figures 9, 10 and 11 show the road construction schedules for each block size and exclusion period combination. Figures 12, 13 and 14 show the road maintenance schedules for each block size and exclusion period combination.

Figure 9: Road construction schedule for the constrained road solution - 1 decade exclusion period

Figure 10: Road construction schedule for the constrained road solution - 2 decade exclusion period

ROAD CONSTRUCTION LIMITED
3 DECADE EXCLUSION PERIOD

Figure 11: Road construction schedule for the constrained road solution - 3 decade exclusion period

ROAD MAINTENANCE LIMITED
1 DECADE EXCLUSION PERIOD

Figure 12: Road maintenance schedule for the constrained road solution - 1 decade exclusion period

Figure 13: Road maintenance schedule for the constrained road solution - 2 decade exclusion period

Figure 14: Road maintenance schedule for the constrained road solution - 3 decade exclusion period

CONSTRAINED VERSUS UNCONSTRAINED SOLUTIONS

Comparison between Tables 3 and 4 shows that limiting the amount of road construction has very little effect on the overall periodic harvest volume. The difference between the volumes harvested was a low of 0.2 % (delay 1, 80 ha. blocks) to a high of 5.7 % (delay 2, 20 ha. blocks).

Table 5: Percent increase in harvest volume of the constrained road solutions (base volumes are unconstrained road solutions)

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	0.2	-1.8	-4.0
2	2.6	2.6	6.1
3	1.1	2.0	4.4

The reason for this low variation of harvest levels is that the 25 kilometer road construction constraint wasn't binding on volume. If the road constraint was lowered to 15 or 20 kilometers, one would expect the harvest volumes to decrease for the constrained road solutions. Though in our particular case the 25 kilometer road construction constraint wasn't binding on volume, the results in Table 5 show that with a road constraint, higher volume blocks may have been selected as a result of the road constraint. This in turn related to a small increase in harvest volumes of the constrained road solutions.

Table 6: Total amount of road constructed (in kilometers) for block size and delay period combinations

80 HA BLOCKS			40 HA BLOCKS		20 HA BLOCKS	
ROAD CONSTRUCTED			ROAD CONSTRUCTED		ROAD CONSTRUCTED	
DELAY PERIOD	NO LIMIT	LIMITED	NO LIMIT	LIMITED	NO LIMIT	LIMITED
1	113.3	113.9	112.2	111.6	112.7	112.0
2	106.2	107.1	106.2	107.9	106.5	109.0
3	99.1	99.1	95.9	97.1	95.9	98.7

Table 6 shows the total amount of road constructed for each block size and delay period combination. It shows that the delay period reduces the amount of road constructed, which may limit the amount of volume eligible for harvest. Table 6 also shows that some roads are never constructed when long delays are specified.

Figures 3, 4 and 5 show that the unconstrained solutions tended to build large amounts of road during the first construction period (up to a high of 48.1 km of road constructed in period 1, 20 ha blocks, delay 1). After this initial peak, road construction fell below the maximum level set in the constrained model. After periods 4-5 the road construction schedules for both the constrained and unconstrained models were very similar for all combinations of block size and delay period.

Comparison between the road maintenance schedules show that except for period 1, there is very little difference between the maintenance schedules of the constrained and the unconstrained models. Both models display a relatively uniform schedule . The period 1 difference is caused by the lower road construction that takes place in the constrained model. There is a greater difference when the maintenance schedules for the two period delay models are compared. The maintenance schedule for the constrained model is still relatively uniform whereas the unconstrained model starts to display a two period cyclic pattern for all block sizes. The cyclic pattern in the

unconstrained model results from the combination of the large initial road construction during the first two periods, and the need to maintain rather than construct these roads as the model progresses through its two period adjacency delay constraint.

Comparison between the road maintenance schedules for the three period delay models shows that both the constrained and the unconstrained models show a cyclic pattern directly attributable to the three period adjacency delay constraint.

THE EFFECT ON THE AAC DUE TO BLOCK SIZE AND EXCLUSION PERIOD

A comparison can be done between the progressive clear cut solution and the constrained road construction solutions to determine the effect on the Annual Allowable Cut due to the reduction of block size and exclusion period. As shown above, due to the low variation in harvest levels, either the constrained or the unconstrained road construction data could be used for this comparison.

The results in table 4 show that the delay period had a greater effect on the AAC than did reducing the block size. The smallest reduction (5.4%) is observed between the progressive clear cut solution and the 80 ha, 1 period delay solution. As the progressive clear cut model is based on 80 ha blocks and the 1 period delay is not overly restrictive, this small difference is expected.

As previously discussed, the 40 ha and 20 ha blocks were created by dividing the original 80 ha blocks. Keeping the exclusion period constant, one would expect that there would be no reduction in harvest volumes as the block size decreased because even though the block size is smaller, more blocks are created and eligible for harvest. The results show that there was a reduction in harvest volumes as the block size decreased while keeping the delay period constant. The values ranged from a difference of 6.0 % (delay period 1) to 23.5 % (delay period 3). These volume reductions associated with block size were due to irregularly shaped blocks with a large number of adjacent blocks. Since an adjacent block was defined as a block that shared

any part of a common boundary with another block, situations arose with some blocks having from between 7, 8 or 9 adjacent blocks for the 80, 40 and 20 ha formulations respectively. As the block size decreased, adjacency constraints actually reduced the number of available blocks, thus reducing the harvest volume available.

As the delay or exclusion period is increased, adjacency constraints limit the number of blocks available for harvest and therefore reduce the harvest volume. Increasing the delay period from 1 to 2 and from 1 to 3 caused volumes to drop an average of 18.6 % and 40 % respectively. Simultaneously increasing the delay period and decreasing the block size results in large reductions in harvest volume of up to 50 % in the case of 20 ha blocks and delay period 3.

Another effect of increasing the delay period and decreasing the block size is that a large number of blocks may never be harvested. Values ranged from 6.8 % to 43.8 % of blocks not harvested for 20 ha delay period 1 and 20 ha delay period 3 respectively. (Table 7). This is due partially to adjacency constraints making these blocks ineligible for harvest during the length of the simulation. Also the unlogged blocks may require a large amount of road to be constructed to access a relatively low volume of timber. Though not directly investigated during the simulation, a maximum grade cost per period is built into the program. Chances are that a road with a large construction cost would be built if that road accessed a large amount of timber or accessed a large number of blocks, whereas a road that accessed little timber might not be built.

Table 7: Percentage of blocks not harvested during the planning period

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	8.3	7.3	6.8
2	18.8	19.8	24.4
3	33.3	36.5	43.8

Figures 1 and 2 show the road construction and maintenance schedules for the progressive clear cut solution. After the first rotation (decade 7), road construction dropped since most of the road network had been constructed. The construction peak in decade 2 was due to the development of branch roads. Road maintenance closely paralleled the progressive clear cut model with the amount of maintenance increasing as the cut progressed towards the back of the valley. Once the end of the first rotation was reached, maintenance decreases as the harvest again moves closer to the beginning of the road network. Road maintenance peaks were related to the maintenance and subsequent abandonment of branch roads.

Figures 9, 10 and 11 show that the smaller cut blocks require more road construction during the first 3 or 4 decades in order to develop enough volume to meet the allowable cut. As the simulation progressed beyond that period, the smaller cut blocks required slightly less construction as the majority of the road network had already been developed. Observation of Table 6 shows with the delay period held constant, the total amount of road constructed is fairly constant throughout the range of block sizes. As the delay period is increased, the total amount of road constructed decreases. These declines in road construction are directly attributable to the increase in blocks not eligible for harvest due to spatial restrictions.

Figures 12, 13 and 14 show a definite trend that the smaller cut blocks require a greater amount of road maintenance than did the larger cut blocks, even though the

volume of timber harvested was less for the smaller cut blocks. As the delay period increased, the total amount of road maintenance decreased, with the 20 ha blocks still requiring the greatest amount of road maintenance. As the delay period is increased, adjacency constraints decreases the number of available blocks for harvesting, thus the amount of road maintenance decreases.

The road maintenance schedules for the 1 and 2 period delay are fairly uniform with a large increase in maintenance during the first three decades, decreasing slightly or staying constant until around the seventh decade and increasing slowly beyond that point. This shows the need to maintain the large initial road construction phase and to re-access previously harvested blocks once the rotation age has passed.

The road maintenance schedule for the 3 period delay shows a definite cyclic pattern not displayed in the 1 and 2 period delay models. The cyclic pattern follows the 3 period delay constraint where the road maintenance schedule has peaks and valleys every three periods. This pattern is not evident in the 1 and 2 period delay models as the 1 and 2 period delays are not overly binding and do not force such a widely dispersed harvest.

The results show that current harvesting levels within a fixed area cannot be maintained if smaller cut blocks in combination with longer exclusion periods are used. Either additional forested land must be made available or harvest levels will drop. Other alternatives include designating land within a best-use policy whereby certain lands are set aside primarily as timber production areas or increasing timber production on a limited land base through intensive management.

ECONOMIC ANALYSIS

An economic analysis was done comparing the construction, hauling, road maintenance, tree to truck and delivered cost per cubic meter for the block size and delay period combinations. Costs were averaged across the decades. Periodic costs are addressed in Figures 15 through 23.

For each block size the typical pattern was for the road construction, hauling and maintenance (con/haul/main) cost to be the lowest for a one period delay and to increase for delay periods two and three. Values ranged from a high of \$16.01/m³ for 20 ha blocks delay period 3 to a low of \$10.74/m³ for 80 ha delay period 1. This trend was expected even though the actual cost of road construction, maintenance and hauling decreases as the delay period increases, due to the lower volumes harvested when the delay period increases.

Table 8: Average road construction, hauling and maintenance costs (\$/m³) harvested by block size and delay period

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	10.74	11.75	13.23
2	11.06	13.02	14.91
3	11.75	13.51	16.01

The tree to truck cost per cubic meter showed no trend. The values ranged from \$14.70/m³ for 20 ha delay period 2 to a high of \$15.34/m³ for 80 ha delay period 1. This was expected as the only variables that change the tree to truck cost are logging difficulty and harvest age, and there was no clearly identifiable pattern for the distribution of logging systems throughout the study area.

Table 9: Average tree to truck costs (\$/m3) by block size and delay period

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	14.35	15.07	15.00
2	14.57	14.72	14.70
3	14.80	14.84	14.72

Adding of the con/haul/main and tree to truck costs per cubic meter shows the total delivered cost per cubic meter. Following the trend of the con/haul/main costs, the total delivered cost was the lowest for delay period 1 and increased as the delay period increased. The values ranged from \$25.09/m3 for 80 ha delay period 1 to \$30.75/m3 for 20 ha delay period 3 (22.6% increase). Graphically, the total delivered costs are shown in Figures 15 through 23 .

Table 10: Average total delivered cost of wood (\$/m3) by block size and delay period

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	25.09	26.82	28.23
2	25.63	27.74	29.61
3	26.55	28.35	30.75

Figure 15: Total delivered cost by decade - 80 ha blocks, 1 decade exclusion period

Figure 16: Total delivered cost by decade - 80 ha blocks, 2 decade exclusion period

Figure 17: Total delivered cost by decade - 80 ha blocks, 3 decade exclusion period

40 HA
DELAY 1

Figure 18: Total delivered cost by decade - 40 ha blocks, 1 decade exclusion period

40 HA
DELAY 2

Figure 19: Total delivered cost by decade - 40 ha blocks, 2 decade exclusion period

40 HA
DELAY 3

Figure 20: Total delivered cost by decade - 40 ha blocks, 3 decade exclusion period

20 HA
DELAY 1

Figure 21: Total delivered cost by decade - 20 ha blocks, 1 decade exclusion period

Figure 22: Total delivered cost by decade - 20 ha blocks, 2 decade exclusion period

Figure 23: Total delivered cost by decade - 20 ha blocks, 3 decade exclusion period

IMPACT ON STUMPAGE

The average operating cost was calculated by dividing the total operating cost by the total harvest volume over the total simulation period. The operating cost/allowances include road construction, road maintenance, tree to truck, hauling,

overhead, crew transportation, crew accommodation, sort/scale and forestry costs.(Shaw, 1993). Average operating cost values ranged from \$41.07/m³ to \$45.50/m³ for the 80 ha, 1 delay period and 20 ha, 3 delay period respectively. (Table 11). For simplicity it was assumed that the calculated operating costs were equal to the operating allowances for use in the stumpage calculation.

<u>Table 11: Average operating cost (\$/m³) by block size and delay period</u>			
Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	41.07	41.96	43.23
2	41.22	42.35	44.17
3	41.42	43.48	45.50

The values in Table 11 are reasonable given the variation in road construction and road maintenance requirements for each combination. Also, the smaller cut-block solutions would be expected to have a higher operating cost due to the greater number of machine set-ups and moves required.

Stumpage revenue was calculated using the following formula (MOF, 1991):

$$IS = BR + ((SP - OC) - VI)$$

where:

IS = indicated stumpage

BR = base rate (assumed \$14.03/m³). The base rate is a quarterly adjusted value to ensure that the average of all coniferous sawlog rates is equal to the target rate set by the ministry for that quarter.

SP = stand as a whole selling price (assumed \$80/m³)

OC = operating allowance

VI = mean value index (assumed \$17.75/m³). The mean value index is a volume weighted average of the value indexes for all harvests that have occurred in the last twelve months. It provides a benchmark against which individual stands are compared.

The calculated indicated stumpage values (Table 12) ranged from \$35.21/m³ to \$30.78/m³ for the 80 ha, 1 delay period and 20 ha, 3 delay period respectively.

<u>Table 12: Average indicated stumpage values (\$/m³) by block size and delay period</u>			
Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	35.21	34.32	33.05
2	35.06	33.93	32.11
3	34.86	32.80	30.78

As stumpage is the amount of money that a logging company must pay to the province for the "right" to harvest timber on a tract of land, there is a drastic difference in the amount of money payable when comparing the different sized blocks and adjacency requirements (Table 13). Comparison between the 80 ha, 1 delay period and the 20 ha, 3 delay period shows the stumpage payable (indicated stumpage multiplied by the total harvest volume) of the 20 ha, 3 delay period is only 44% of the 80 ha, 1 delay period. This is a reduction of nearly \$67 million . These figures represent the dramatic economic impacts associated with block size and adjacency rules. It should also be noted that due to the high road construction costs in the first few decades, the short-term impacts on stumpage are probably greater than the average figures calculated.

Table 13: Average stumpage revenues (in millions of \$) by block size and delay period

Delay Period	80 ha blocks	40 ha blocks	20 ha blocks
1	119.7	114.1	110.6
2	94.6	89.9	77.5
3	76.2	63.2	52.8

CONCLUSION

An area-based scheduling model was developed to evaluate the impacts and costs associated with various cut-block sizes and adjacency constraints. Specifically, this paper: 1) determined the effect on the AAC and road network activity due to block size and the exclusion period for both unconstrained and constrained road budgets ; 2) quantified the delivered wood costs; and 3) forecasted the impacts on stumpage revenues.

Comparison between the constrained and unconstrained road budgets showed little effect on the overall periodic harvest volumes. The increase of the constrained harvest volume ranged from -4.0% to 6.1% of the unconstrained harvest volume. This difference is low because the road construction constraint of 25 kilometers was not particularly binding.

The effect on the AAC due to block size and exclusion period showed that the delay period had a greater effect on the AAC than did reducing the block size. As the block size decreased while keeping the delay period constant, the reduction in harvest volume ranged from 6.0% to 23.5%. As the block size decreased, adjacency constraints actually reduced the number of available blocks, thus reducing the volume available for harvest.

Increasing the delay period from 1 to 2 and from 1 to 3 caused volumes to drop an average of 18.6% and 40% respectively. Increasing the delay period and decreasing the block size simultaneously resulted in large reductions in harvest volume of up to 50%.

Another effect of increasing the delay period and decreasing the block size is that a large number of blocks are never harvested. Values ranged from 6.8% to 43.8% of blocks being ineligible for harvest.

Road construction schedules showed that the smaller cut-blocks required more road construction during the first 3 or 4 decades in order to develop enough volume to meet the allowable cut. Beyond that point, the smaller cut-blocks required less construction as the majority of the road network was already developed. The road maintenance schedules clearly show that the smaller cut-blocks require a greater amount of road maintenance than did the larger cut-blocks. This was due to the greater amount of road access required to service the dispersed harvest of small blocks.

The economic analysis showed that road construction, hauling and maintenance costs were the lowest for the 1 period delay and increase for delay periods 2 and 3.

A stumpage calculation showed that there is a drastic difference in the amount of stumpage payable when comparing the different sized blocks and adjacency options. Values ranged from \$119.7 million to \$52.8 million, a reduction of nearly \$67 million as the block size was decreased and the adjacency constraints increased.

As the block size decreased, the forest became more and more fragmented as groupings of different age classes began to emerge over time. While this may be more aesthetically pleasing than a large clear cut, a highly fragmented forest is not ideal in providing habitat for most species of wildlife. Most wildlife do require a specific age-class distribution throughout the forest in order to provide habitat for protection, food production and migration. These particular habitats may exist in the fragmented forest but they must be accessible to the animals and in sufficient quantities through time. Further research must be done to examine the benefits of maintaining a specific age

class distribution for wildlife habitat. For example, different adjacency rules could be set for different areas within a study area so that designated migration corridors would be maintained with a predefined age class distribution of timber.

If we are to follow other countries in the world, further research must be done to determine the impacts on harvest levels and road networks of going to cut blocks even smaller than 20 ha. Additional work is also required to investigate the use of selection systems and their economic, social and forest structure impacts.

BIBLIOGRAPHY

- Armel, N.B. 1986. Area analysis and Version II of FORPLAN. In Proceedings of the Workshop on Lessons From Using FORPLAN, Denver, CO, April 29 - May 1. USDA Forest Service, Land Management Planning Systems Section, Washington, D.C.
- Bare, B.B., Faaland, B.H., and Gupta, I. 1984. Timber harvest scheduling in the presence of spatial constraints. Paper presented at the Joint National Meeting of the Institute Of Management Sciences and the Operations Research Society of America, San Francisco, CA, May 14-16.
- Clements, S.E., Dallain, P.L., and Jamnick, M.S. 1990. An operational, spatially constrained harvest scheduling model. Can. J. For. Res. 20: 1438-1447.
- Hackett, J. MacMillan Bloedel Corporate Forestry, Vancouver, B.C. Personal Communication, 1990.
- Johnson, K.N., and Jones, D.B. 1991. Multiple Use Sustained Yield Calculation Timber Harvest Scheduling Model Users Guide. Pacific Forestry Center, Victoria, B.C.
- Johnson, K.N., Stuart, T.W., and Crim, S.A. 1986. FORPLAN Version 2 : An overview. USDA Forest Service, Land Management Planning Systems Section Washington, D.C.
- Ministry Of Forests, 1991. Stumpage Appraisal Information Paper No. 1 Comparative Value Timber Pricing, Valuation Branch, Victoria, B.C.
- Nawitka Resource Consultants. 1987. Impact Of Intensive Forestry Practices On Net Stand Values In British Columbia. Forestry Economic and Regional Development Agreement Report 014, Victoria, B.C.
- Nelson, J.D. 1988. Integrating Short-term Spatially Feasible Harvest Plans With Long-term Harvest Schedules Using Monte-Carlo Integer Programming And Linear Programming. Ph.D. dissertation, Oregon State University, Oregon.
- Nelson, J., and Brodie, J.D. 1990. Comparison of a random search algorithm and mixed integer programming for solving area-based forest plans. Can. J. For. Res. 20: 934-942.
- Nelson, J.D., and Finn, S.T. 1991. The influence of cut-block size and adjacency rules on harvest levels and road networks. Can. J. For. Res. 21: 595-600.
- O'Hara, A.J., Faaland, B.H., and Bare, B.B. 1989. Spatially constrained timber harvest scheduling. Can. J. For. Res. 19: 715-724.

Shaw, N. British Columbia Institute of Technology, Burnaby, B.C. Personal Communication, 1993.

APPENDIX 1

DESCRIPTION OF ANALYSIS AREAS

Description of the Analysis Areas (from Nelson, 1988)

ANALYSIS AREA	SITE/SPECIES/AGE	HA	VOL/HA
1	HCJ7	45	690
2	MCJ7	13	400
3	MFJ7	40	330
4	HHJ6	120	610
5	HCJ8	36	760
6	MFJ8	19	470
7	LCJ0	17	440
8	HHJ8	67	800
9	HHJ6	55	610
10	LFJ6	12	100
11	MFJ7	95	410
12	LFJ6	17	100
13	MFJ7	28	410
14	MFJ0	37	710
15	MHJ7	233	410
16	LHJ6	18	90
17	MFJ0	137	710
18	MFJC	21	620
19	LFJ7	45	200
20	MFJC	31	620
21	HHJB	71	970
22	MHJA	202	570
23	HFJ0	82	1170
24	HHJC	80	1010
25	HHJA	39	930
26	HHJB	29	970
27	HHJA	16	930
28	HHJC	28	1010
29	HCJ9	20	840
30	MHJA	44	570
32	HFJ1	368	0
34	HFJ7	15	680
35	HFJB	35	940
36	HFJC	12	990
37	HFJ0	92	1170
38	HHJ7	59	720
39	HHJ8	56	800
40	HHJB	70	800
41	HHJC	40	1010
43	HCJ4	26	330
44	MFJ1	148	0
45	MFJ2	64	0
46	MFJ3	295	14

47	MFJ7	65	410
48	MFJ8	27	470
49	MFJC	33	615
50	MFJ0	58	710
51	MHJ4	20	125
52	MHJ6	62	330
53	MHJ7	270	410
54	MHJ8	60	480
58	MCJ3	99	20
59	MCJ4	205	120
60	MCJ0	14	750
61	LFJ1	47	0
62	LFJ6	46	100

Codes Used To Define Site/Species/Age:

- the first character is the site class

H = High

M = Medium

L = Low

- the second character is the species group

F = Fir

H = Hemlock

C = Cedar

- the third and fourth characters are the 10 year age classes

J1 = 10

J2 = 20

J3 = 30

J4 = 40

J5 = 50

J6 = 60

J7 = 70

J8 = 80

J9 = 90

JA = 100

JB = 110

JC = 130

JO = 200+

APPENDIX 2

ANALYSIS AREA MAP / ZONE MAP

ANALYSIS AREA MAP

ZONE MAP

APPENDIX 3

DBASE FILE TO MAKE TABLES OF BLOCK VOLUMES

* MAKETAB
* CREATES A TABLE OF VOLUMES PER BLOCK PER PERIOD TO BE USED BY
* MCIP (NELSON)
*

* WRITTEN BY STEVE FINN AND IAN THOMAS
* MARCH 29, 1990
*

* FOR INTERNAL USE ONLY
*

*****INSTRUCTIONS FOR USE*****

*
* THE FIELD NAMES AND LENGTHS SHOULD CORRESPOND TO THOSE IN THIS
* FILE.
*

* USE C:\TERRA\USER\STILLWTR.DB_
*

* TO RUN THE PROGRAM TYPE "DO MAKETAB" AT THE DOT PROMPT IN DBASE
* THE VOLUME TABLE WILL BE WRITTEN TO THE FILE "SUMS".
*

SORT ON POLY_LABEL TO C:\TMP\TENT.DBF
CLOSE ALL
USE C:\TMP\TENT.DBF
GO TOP

SELECT B
USE C:\TMP\SUMS
ZAP

SELECT A
DO WHILE .NOT. EOF()

SUM01 = 0
SUM02 = 0
SUM03 = 0
SUM04 = 0
SUM05 = 0
SUM06 = 0
SUM07 = 0
SUM08 = 0
SUM09 = 0
SUM10 = 0
SUM11 = 0
SUM12 = 0
SUM13 = 0
SUM14 = 0
SUM15 = 0


```
SUM16 = 0
SUM17 = 0
SUM18 = 0
SUM19 = 0
SUM20 = 0
SUM21 = 0
SUM22 = 0
SUM23 = 0
SUM24 = 0
SUM25 = 0
SUM26 = 0
SUM27 = 0
SUM28 = 0
SUM29 = 0
SUM30 = 0
CURRENT_BLK = POLY_LABEL
DO WHILE POLY_LABEL = CURRENT_BLK
    SUM01 = (POLY_AREA * EX_YLD01) + SUM01
    SUM02 = (POLY_AREA * EX_YLD02) + SUM02
    SUM03 = (POLY_AREA * EX_YLD03) + SUM03
    SUM04 = (POLY_AREA * EX_YLD04) + SUM04
    SUM05 = (POLY_AREA * EX_YLD05) + SUM05
    SUM06 = (POLY_AREA * EX_YLD06) + SUM06
    SUM07 = (POLY_AREA * EX_YLD07) + SUM07
    SUM08 = (POLY_AREA * EX_YLD08) + SUM08
    SUM09 = (POLY_AREA * EX_YLD09) + SUM09
    SUM10 = (POLY_AREA * EX_YLD10) + SUM10
    SUM11 = (POLY_AREA * EX_YLD11) + SUM11
    SUM12 = (POLY_AREA * EX_YLD12) + SUM12
    SUM13 = (POLY_AREA * EX_YLD13) + SUM13
    SUM14 = (POLY_AREA * EX_YLD14) + SUM14
    SUM15 = (POLY_AREA * EX_YLD15) + SUM15
    SUM16 = (POLY_AREA * REG_YLD01) + SUM16
    SUM17 = (POLY_AREA * REG_YLD02) + SUM17
    SUM18 = (POLY_AREA * REG_YLD03) + SUM18
    SUM19 = (POLY_AREA * REG_YLD04) + SUM19
    SUM20 = (POLY_AREA * REG_YLD05) + SUM20
    SUM21 = (POLY_AREA * REG_YLD06) + SUM21
    SUM22 = (POLY_AREA * REG_YLD07) + SUM22
    SUM23 = (POLY_AREA * REG_YLD08) + SUM23
    SUM24 = (POLY_AREA * REG_YLD09) + SUM24
    SUM25 = (POLY_AREA * REG_YLD10) + SUM25
    SUM26 = (POLY_AREA * REG_YLD11) + SUM26
    SUM27 = (POLY_AREA * REG_YLD12) + SUM27
    SUM28 = (POLY_AREA * REG_YLD13) + SUM28
    SUM29 = (POLY_AREA * REG_YLD14) + SUM29
    SUM30 = (POLY_AREA * REG_YLD15) + SUM30
SKIP
```

ENDDO

SELECT B

APPEND BLANK

REPLACE BLOCK_NUM WITH CURRENT_BLK

REPLACE EX_VOL01 WITH SUM01

REPLACE EX_VOL02 WITH SUM02

REPLACE EX_VOL03 WITH SUM03

REPLACE EX_VOL04 WITH SUM04

REPLACE EX_VOL05 WITH SUM05

REPLACE EX_VOL06 WITH SUM06

REPLACE EX_VOL07 WITH SUM07

REPLACE EX_VOL08 WITH SUM08

REPLACE EX_VOL09 WITH SUM09

REPLACE EX_VOL10 WITH SUM10

REPLACE EX_VOL11 WITH SUM11

REPLACE EX_VOL12 WITH SUM12

REPLACE EX_VOL13 WITH SUM13

REPLACE EX_VOL14 WITH SUM14

REPLACE EX_VOL15 WITH SUM15

REPLACE REG_VOL01 WITH SUM16

REPLACE REG_VOL02 WITH SUM17

REPLACE REG_VOL03 WITH SUM18

REPLACE REG_VOL04 WITH SUM19

REPLACE REG_VOL05 WITH SUM20

REPLACE REG_VOL06 WITH SUM21

REPLACE REG_VOL07 WITH SUM22

REPLACE REG_VOL08 WITH SUM23

REPLACE REG_VOL09 WITH SUM24

REPLACE REG_VOL10 WITH SUM25

REPLACE REG_VOL11 WITH SUM26

REPLACE REG_VOL12 WITH SUM27

REPLACE REG_VOL13 WITH SUM28

REPLACE REG_VOL14 WITH SUM29

REPLACE REG_VOL15 WITH SUM30

SELECT A

ENDDO

*****END OF FILE*****

APPENDIX 4

BLOCK LAYOUT AND ROAD NETWORKS BY BLOCK SIZE

80 HECTARE BLOCKS AND ROAD NETWORK

40 HECTARE BLOCKS AND ROAD NETWORK

20 HECTARE BLOCKS AND ROAD NETWORKS

APPENDIX 5
EXAMPLES OF INPUT FILES

BLOCKDAT.TXT

block	ha	age	zone	adjacent	blocks
1	78	7	3	2	4 5 0 0 0 0 0
2	86	8	3	1	5 6 3 0 0 0 0
3	93	7	3	2	6 7 0 0 0 0 0
4	83	7	1	1	5 8 9 0 0 0 0
5	98	7	2	1	2 4 6 8 9 10 0
6	74	7	2	2	3 7 11 10 5 0 0
7	66	7	3	3	6 11 16 0 0 0 0
8	79	8	1	4	5 9 12 0 0 0 0
9	77	7	2	4	5 8 10 12 13 0 0
10	84	7	2	5	6 11 14 13 9 0 0
11	90	7	2	6	7 16 15 14 13 10 0
12	77	8	1	8	9 13 20 19 0 0 0
13	79	8	1	9	10 11 14 21 20 12 0
14	98	7	4	10	11 15 13 21 0 0 0
15	89	20	4	11	16 17 23 22 14 0 0
16	67	20	3	7	11 15 22 23 17 0 0
17	71	7	3	16	15 22 23 18 0 0 0
18	77	8	4	17	23 0 0 0 0 0 0
19	84	7	1	12	20 0 0 0 0 0 0
20	77	7	1	12	19 13 21 0 0 0 0
21	78	7	6	20	13 14 24 0 0 0 0
22	89	7	4	15	16 17 23 31 30 25 0
23	80	7	4	18	17 16 15 22 31 0 0
24	65	7	6	21	25 26 0 0 0 0 0
25	84	7	5	24	22 30 32 0 0 0 0
26	83	7	6	24	27 28 0 0 0 0 0
27	100	7	6	26	28 29 0 0 0 0 0
28	81	7	6	26	27 29 0 0 0 0 0
29	81	7	6	27	28 0 0 0 0 0 0
30	77	7	5	25	22 31 35 34 33 32 0
31	81	7	4	23	22 30 35 36 0 0 0
32	87	7	5	25	30 33 38 37 0 0 0
33	72	20	7	32	30 34 38 37 0 0 0
34	78	12	7	33	30 35 39 38 0 0 0
35	79	11	8	34	30 31 36 40 39 0 0
36	81	11	8	31	35 40 0 0 0 0 0
37	86	7	5	32	33 38 42 41 0 0 0
38	73	7	7	37	32 33 34 39 42 0 0
39	79	20	7	38	34 35 40 43 45 42 0
40	87	12	8	39	35 36 43 0 0 0 0
41	84	7	5	37	42 0 0 0 0 0 0
42	86	20	7	41	37 38 39 45 44 0 0
43	71	10	8	39	40 46 45 0 0 0 0
44	81	20	7	42	39 45 47 0 0 0 0
45	73	10	8	44	42 39 43 46 47 0 0

46	69	11	8	45	43	47	48	0	0	0	0
47	74	10	8	44	45	46	48	0	0	0	0
48	82	20	8	46	47	0	0	0	0	0	0

CONSTRA1.TXT

HAB_DELAY	MAX_BLOCKS	MAX_PERIODS	MAX_ZONES	MIN_AGE	MAX_ADJAC
2	48	10	8	6	8
MAX_MRD	MAX_ACCESS	COST_MAINT	MAX_ROAD	CUT_1	CUT_2
162	51	51	25	210000.	210000.
DISC	MINVOL	MAXVOL	MINCOST	MAXCOST	MAXGRADE COST
1.	265000.	275000.	0.	50000000.	50000000.
ZONE	FIRST_ENTRY				
1	1				
2	5				
3	1				
4	1				
5	1				
6	2				
7	1				
8	4				

[illegible]

EXVOLTAB.TXT

1	26968	30580	34285	36950	40374	41129	42436	43270	44577	45410
2	35161	38696	42838	45470	49268	49825	51380	52722	53799	55140
3	37989	44259	47976	52457	55207	57911	59745	61575	63360	64322
4	42415	47372	52588	56550	59298	62067	64089	65811	67345	68185
5	41578	46832	52077	56290	58809	62054	63971	65367	67263	68651
6	37262	41293	45086	47722	50043	52323	53909	54947	56238	57020
7	29032	33010	35620	38450	40263	42076	43314	44332	45481	46057
8	50175	55695	60451	64031	66918	69671	71902	73429	74874	75917
9	48546	53895	58805	62785	65591	68393	70995	72941	74359	75700
10	49578	55678	60617	65080	68042	71011	73672	75689	77323	78652
11	33990	37858	41531	44401	46581	48760	50162	51128	52408	53186
12	52837	57779	61803	64635	67359	69713	71594	73076	74167	75100
13	40982	45681	48901	52133	54399	56618	58408	59804	61097	61824
14	37570	44144	47855	52432	55296	58052	59961	61888	63690	64645
15	52664	55061	56427	58674	59713	60753	61470	62680	63323	63645
16	43807	44349	44692	45631	45923	46215	46479	47149	47181	47209
17	28994	33888	36685	40186	42283	44381	45779	47182	48581	49280
18	53060	58321	62482	65662	68512	71301	73321	74859	76336	77105
19	33548	39341	43035	47173	49656	52139	54178	55833	57488	58316
20	32982	38320	42193	45944	48294	50644	52930	54455	55980	56743
21	29846	35217	38731	42498	44823	47094	49113	50663	52158	52933
22	45753	51901	56753	61144	64116	67089	69446	71165	72946	74113
23	51161	56635	61825	65964	68936	71694	73971	76146	77527	78642
24	25934	30472	33706	36947	38892	40837	42775	44071	45368	46016
25	34226	40066	43406	47578	50079	52583	54253	55920	57589	58425
26	33359	38244	42312	45570	48011	50453	52083	52900	54527	55341
27	40401	46314	51241	55182	58139	61095	63066	64051	66022	67007
28	32853	37661	41668	44873	47277	49681	51283	52085	53687	54488
29	31950	37445	41428	45330	47720	50110	52418	53930	55523	56320
30	51925	58799	64134	69459	71781	75578	77868	79398	81684	83195
31	48490	54449	59185	63534	65886	69008	71042	72340	74303	75463
32	58404	66113	72107	78102	80673	84955	87524	89238	91808	93521
33	79287	79530	79748	79963	80134	80191	80363	80391	80968	80989
34	78729	81824	84208	85892	87494	88345	89884	91354	92211	92346
35	75263	78496	81541	84205	86159	87665	88968	90153	91602	92626
36	61058	63846	66430	68285	69974	71391	72479	74020	74533	75213
37	51244	58131	63565	68769	71299	75047	77343	78798	81094	82547
38	40313	45627	49884	53759	55911	58726	60505	61575	63351	64398
39	87122	87457	87773	87990	88194	88384	88503	88695	89424	89509
40	84534	88287	91050	93561	95220	96656	98347	99682	100500	100693
41	36115	41288	45566	49076	51579	54193	55919	56815	58535	59423
42	62043	62824	63310	64442	64855	65275	65630	66452	66563	66627
43	56604	59131	61571	63310	64730	66076	66864	68272	68987	69620
44	56950	57126	57302	58175	58292	58410	58469	59341	59341	59400
45	43692	45929	48165	49675	51135	52595	53325	54789	54882	55612
46	63456	65860	68264	70060	71354	72649	73257	74552	75554	76162
47	50713	53134	55554	57237	58716	60194	60934	62414	62822	63561
48	89907	91004	92100	92967	93603	94238	94469	95105	95977	96208

MRACCESS.TXT

[illegible]

36 3.49 140 139 103 102 75 101 100 74 61 58 59 60 10 9 8 7 6 5 4 3 2 1 0 0 0 0 0 0 0
0
37 3.6 112 109 108 71 68 66 62 52 51 50 45 44 42 34 23 22 21 20 16 15 14 122 121 120 119 118 117
116 107 106 105 104 75 74 61 58 59 60 10 9 8 7 6 5 4
38 3.57 120 119 118 117 132 129 130 128 127 126 125 124 116 107 106 105 104 75 74 61 58 59 60 10 9 8
7 6 5 4 3 2 1 0
39 3.59 115 114 113 107 106 105 104 75 74 61 58 59 60 10 9 8 7 6 5 4 3 2 1 0 0 0 0 0 0 0
0
40 3.55 144 142 141 140 139 103 102 75 74 61 58 59 60 10 9 8 7 6 5 4 3 2 1 0 0 0 0 0 0 0
0
41 3.66 111 110 109 108 71 68 66 62 52 51 50 45 44 42 34 23 22 21 20 16 15 14 123 122 121 120 119
118 117 116 107 106 105 104 75 74 61 58 59 60 10 9 8 7 6
42 3.64 134 135 133 132 127 126 125 124 116 107 106 105 104 75 74 61 58 59 60 10 9 8 7 6 5 4 3
2 1 0
43 3.6 146 144 145 147 143 142 141 140 139 103 102 75 74 61 58 59 60 10 9 8 7 6 5 4 3 2 1 0
0
44 3.69 137 136 135 133 132 127 126 125 124 116 107 106 105 104 75 74 61 58 59 60 10 9 8 7 6 5 4
3 2 1 0
45 3.68 153 152 150 151 146 144 141 140 139 103 102 75 74 61 58 59 60 10 9 8 7 6 5 4 3 2 1 0
0
46 3.68 155 158 156 154 152 146 144 141 140 139 103 102 75 74 61 58 59 60 10 9 8 7 6 5 4 3 2
1 0
47 3.71 159 157 158 156 154 152 146 144 141 140 139 103 102 75 74 61 58 59 60 10 9 8 7 6 5 4 3
2 1 0
48 3.72 160 159 157 156 154 152 146 144 149 148 147 143 142 141 140 139 103 102 75 74 61 58 59 60 10
9 8 7 6 5 4 3 2 1 0

MRLINKS.TXT

1	245	7350
2	550	16500
3	225	6750
4	480	14400
5	635	19050
6	820	24600
7	470	14100
8	710	21300
9	200	6000
10	465	13950
11	1050	31500
12	335	10050
13	1425	42750
14	600	18000
15	60	1800
16	340	10200
17	420	12600
18	1480	44400
19	1240	37200
20	140	4200
21	430	12900
22	145	4350
23	255	7650
24	510	15300
25	930	27900
26	960	28800
27	990	29700
28	440	13200
29	915	27450
30	965	28950
31	1315	39450
32	500	22500
33	1770	79650
34	270	12150
35	170	7650
36	430	19350
37	1125	50625
38	305	13725
39	310	13950
40	260	15600
41	810	48600
42	785	35325
43	1530	68850
44	1420	63900
45	185	8325
46	380	17100
47	300	13500
48	240	10800
49	705	31725
50	700	31500
51	275	12375
52	915	27450
53	615	18450
54	380	11400
55	360	10800
56	1080	32400
57	280	8400

58	290	8700
59	225	6750
60	700	21000
61	1045	31350
62	930	27900
63	385	23100
64	270	16200
65	860	51600
66	975	29250
67	220	6600
68	205	9225
69	1120	33600
70	285	8550
71	825	37125
72	825	37125
73	1340	40200
74	175	5250
75	165	4950
76	640	38400
77	190	11400
78	920	55200
79	605	18150
80	410	24600
81	680	20400
82	380	22800
83	425	25500
84	150	9000
85	505	30300
86	1690	101400
87	405	24300
88	795	47700
89	810	48600
90	1350	81000
91	1755	105300
92	175	10500
93	535	32100
94	460	27600
95	265	15900
96	1055	63300
97	1290	77400
98	475	28500
99	470	28200
100	975	29250
101	950	28500
102	755	22650
103	100	3000
104	530	15900
105	180	5400
106	300	9000
107	160	4800
108	555	24975
109	1210	54450
110	1250	56250
111	200	9000
112	1600	72000
113	1180	35400
114	160	4800
115	945	28350
116	210	6300

117	205	6150
118	360	10800
119	575	17250
120	655	19650
121	215	6450
122	245	7350
123	975	43875
124	435	13050
125	500	15000
126	150	4500
127	315	9450
128	130	3900
129	170	5100
130	495	14850
131	545	16350
132	250	7500
133	485	14550
134	870	26100
135	390	11700
136	255	11475
137	1385	62325
138	1225	36750
139	660	19800
140	510	15300
141	265	7950
142	785	23550
143	125	3750
144	675	20250
145	710	21300
146	940	28200
147	360	10800
148	630	18900
149	610	18300
150	540	16200
151	790	23700
152	270	8100
153	225	6750
154	140	4200
155	995	29850
156	240	7200
157	370	11100
158	895	26850
159	710	21300
160	1105	33150
161	1085	32550
162	830	24900

RGREVTAB.TXT

[illegible]

RGVOLTAB.TXT

1	0.0	204.0	881.3	6073.3	12935.8	19986.9	26968.3	30579.7	34285.4	36950.2
2	0.0	22.9	1210.5	8134.9	16935.6	24565.4	31814.5	36779.4	40446.5	44034.8
3	0.0	21.7	1870.2	11055.1	22055.9	30304.9	37639.8	44051.3	47724.9	52305.0
4	0.0	2804.7	6396.8	16596.0	26572.5	34794.8	42415.4	47371.9	52588.4	56550.2
5	0.0	1667.5	3857.7	12758.8	22417.5	32022.0	41567.7	46826.2	52069.3	56285.4
6	0.0	1754.2	4148.8	12344.2	20906.3	28057.8	35000.6	39400.0	43400.4	46611.8
7	0.0	0.0	1016.5	7638.5	15589.4	21867.6	27180.2	31510.7	34473.4	37479.5
8	0.0	3638.6	8078.6	20428.7	31757.8	40155.2	47796.1	53364.0	58566.7	62774.7
9	0.0	4294.2	9244.3	21322.5	31982.2	39948.6	48157.9	53541.1	58508.1	62587.2
10	0.0	3629.3	8245.6	21076.8	32942.7	41450.0	49528.7	55634.9	60579.3	65054.9
11	0.0	0.0	689.9	6969.9	15336.6	23493.0	31286.0	35669.7	39855.4	42983.1
12	0.0	3735.9	8212.7	20129.8	31011.8	39052.2	46599.2	52087.3	57165.6	61294.1
13	0.0	1796.9	4583.7	13127.0	22295.4	29246.0	36051.2	41544.9	45440.8	49310.5
14	0.0	0.0	1714.0	10377.0	21199.4	29548.2	37012.1	43697.8	47501.7	52134.7
15	0.0	10.5	1227.5	10102.5	20742.6	29286.9	36380.3	42019.8	46135.6	50004.0
16	0.0	0.0	646.4	7064.4	14835.6	21411.4	26777.8	30679.6	33976.8	36595.0
17	0.0	0.0	1403.0	8441.0	16886.6	23225.4	28855.8	33777.8	36597.6	40112.0
18	0.0	3801.0	8536.3	21111.3	32508.9	40574.4	48353.8	54218.1	58982.6	63309.3
19	0.0	0.0	1655.2	9931.2	19862.4	27310.8	33548.0	39341.2	43035.2	47173.2
20	0.0	433.3	2329.9	10513.0	19911.8	26899.0	32486.3	37886.4	41821.7	45696.6
21	0.0	0.0	1441.8	8650.8	17518.0	24276.6	29845.6	35216.5	38731.1	42497.9
22	0.0	2100.0	5318.3	16902.6	28768.7	37750.3	45752.8	51900.9	56752.7	61143.7
23	0.0	4049.5	8901.7	21827.8	33457.6	42012.5	49871.4	55560.2	60964.5	65318.5
24	0.0	0.0	1296.4	7778.4	15556.8	21390.6	25934.2	30471.6	33706.4	36947.4
25	0.0	6.0	1678.1	10029.6	20042.7	27552.9	34226.3	40065.9	43405.6	47577.6
26	0.0	0.0	816.8	8953.8	18718.2	26852.1	33358.6	38243.9	42312.4	45570.3
27	0.0	0.0	985.4	10839.4	22664.2	32518.2	40401.4	46313.8	51240.8	55182.4
28	0.0	0.0	801.3	8814.3	18429.9	26442.9	32853.3	37661.1	41667.6	44872.8
29	0.0	0.0	1511.4	9478.4	19038.8	26291.1	31950.0	37444.9	41428.4	45329.9
30	0.0	2979.3	6746.0	21178.5	34187.9	44146.8	51834.9	58721.7	64069.3	69406.1
31	0.0	2168.5	5142.9	17601.1	29644.8	39028.0	46523.9	52754.6	57761.7	62380.6
32	0.0	3427.6	7715.1	24011.2	38596.5	49749.7	58330.7	66053.3	72057.6	78063.4
33	0.0	2762.5	6213.3	19291.0	30989.0	39930.7	46816.8	53008.4	57826.5	62641.5
34	0.0	5413.4	11610.9	26495.3	39056.5	47723.3	56322.3	62613.7	68099.4	72822.6
35	0.0	5513.1	11816.4	26802.4	39423.9	48128.3	56786.6	63086.5	68645.7	73393.0
36	0.0	2523.5	6304.7	17640.2	28791.4	36794.3	44348.6	50372.8	54690.7	59096.7
37	0.0	2438.0	5722.2	19669.7	32871.6	43162.1	51150.7	58056.4	63506.4	68719.7
38	0.0	1455.2	3640.4	14141.7	24669.4	33014.9	39577.1	45013.4	49374.8	53351.1
39	0.0	2900.3	6662.2	20941.2	33937.9	43844.0	51595.3	58511.3	63803.1	69141.3
40	0.0	5986.4	12849.9	29093.5	42787.9	52330.5	61679.8	68459.7	74667.3	79872.5
41	0.0	222.8	1285.5	10185.8	20543.1	29109.2	35939.8	41146.3	45457.2	48983.9
42	0.0	396.0	1706.2	11198.2	22014.4	30858.3	37937.1	43448.8	47893.2	51684.7
43	0.0	2763.4	6551.1	17211.8	27315.5	34505.7	41366.3	46730.7	50766.0	54716.2
44	0.0	0.0	872.5	9009.5	18773.9	26852.1	33361.7	38302.7	42312.4	45626.0
45	0.0	324.8	2066.5	9816.1	18800.1	25494.8	31499.6	36675.3	39750.4	43459.7
46	0.0	3824.0	8441.5	20530.4	31171.3	38670.5	45826.3	51292.9	55780.4	59873.4
47	0.0	1430.6	4141.8	13432.1	23184.8	30299.0	36864.2	42278.8	45873.0	49799.8
48	0.0	3958.7	8734.2	24253.4	37908.5	48065.9	56464.4	63585.1	69302.7	74789.8

SRLINKS.TXT

1	625	18750
2	0	0
3	1365	40950
4	230	6900
5	0	0
6	0	0
7	1220	36600
8	245	7350
9	0	0
10	0	0
11	100	3000
12	335	15075
13	0	0
14	1395	62775
15	400	12000
16	895	26850
17	360	10800
18	2130	63900
19	425	19125
20	825	37125
21	0	0
22	0	0
23	0	0
24	140	8400
25	0	0
26	0	0
27	0	0
28	0	0
29	0	0
30	340	10200
31	0	0
32	0	0
33	650	19500
34	505	15150
35	0	0
36	290	8700
37	215	9675
38	0	0
39	555	16650
40	843	25290
41	0	0
42	415	12450
43	60	1800
44	400	18000
45	315	9450
46	0	0
47	555	16650
48	165	4950

APPENDIX 6
MCIP FLOW CHART

MCIP BLOCK SCHEDULER FLOW CHART

APPENDIX 7

MAIN PROGRAM FILE

```

(* Monte Carlo Integer Program *)
(* trans.PAS MCIP WITH ROADS minvol as constraint,transportation *)
(* DATE: February 1993 *)
(* For MB -Stillwater *)
PROGRAM MCIP (INPUT, OUTPUT,DISKFILE);
USES
  DOS, CRT, PRINTER;
CONST
  MOSTBLOCK =230;           { max no of blocks }
  MOSTZONES =10;
  MOSTPERIOD =10;           { max time periods }
  MOSTACCESS =100;          { MAX NUMBER OF ROADS TO ACCESS BLOCK }
  MOSTADJAC =10;             { MAX NUMBER OF ADJACENT BLOCKS }
  MOSTMRD =300;              { MAXIMUM NUMBER OF MAIN ROADS }
  MAXREPS =100;             { MAXIMUM NUMBER OF REPETITIONS }

TYPE
  ZONEARRAY =ARRAY[1..MOSTZONES] OF INTEGER;
  ZCOUNTERPTR=^ZONEARRAY;
  FIRSTENTRYPTR=^ZONEARRAY;
  INTZONEBLOCKARRAY=ARRAY[1..MOSTZONES,1..MOSTBLOCK] OF INTEGER;
  BLOCKINZONEPTR=^INTZONEBLOCKARRAY;
  ZONEPERIODARRAY=ARRAY[1..MOSTZONES,1..MOSTPERIOD] OF REAL;
  ZVOLPTR=^ZONEPERIODARRAY;

  VOLREVARARRAY =ARRAY[1..MOSTBLOCK,1..MOSTPERIOD,1..2] OF REAL; {vol and rev tables}
  volumePtr =^volrevararray;
  revenuePtr =^volrevararray;
  REALARRAY =ARRAY[0..MOSTBLOCK,1..MOSTPERIOD] OF REAL;
  PERIODARRAY =ARRAY[1..MOSTPERIOD] OF real;
  BLOCKARRAY =ARRAY[1..MOSTBLOCK] OF REAL;
  areaPtr =^blockarray;
  blockPtr =^blockarray;
  MRLENARRAY =ARRAY[1..MOSTMRD] OF REAL;
  MAINROADARRAY=ARRAY[1..MOSTMRD,1..MOSTPERIOD] OF REAL;
  mrPtr =^mainroadarray;
  mrmaintPtr =^mainroadarray;
  DNRARRAY =ARRAY[1..maxreps] OF REAL; {number of solutions}
  ADJACARRAY =ARRAY[1..MOSTBLOCK,0..MOSTADJAC] OF INTEGER;{integers needed for adjac
procedure}
  adjacPtr =^adjacarray;
  ACCESSARRAY =ARRAY[1..MOSTBLOCK,1..MOSTACCESS] OF INTEGER; {access roads for each
block}
  accessPtr =^accessarray;
  INTBLOCKARRAY=ARRAY[1..MOSTBLOCK] OF INTEGER; {for vtable and r table}
  startagePtr =^intblockarray;
  vtablePtr =^intblockarray;
  rtablePtr =^intblockarray;
  ZONEPTR =^INTBLOCKARRAY;
  INTARRAY =ARRAY[1..MOSTBLOCK,0..MOSTPERIOD] OF INTEGER;
  agePtr =^intarray;

VAR
  ZVOL :ZVOLPTR;
  BLOCKINZONE:BLOCKINZONEPTR;
  ZCOUNTER :ZCOUNTERPTR;
  ZONE :ZONEPTR;
  FIRSTENTRY:FIRSTENTRYPTR;

```


COSTMAINT :REAL; {MAINTENANCE COST/KM/YR}
MAXZONE :INTEGER;
MAXBLOCK, {max no of blocks}
MAXPERIOD, { max time periods }
MAXACCESS, { MAX NUMBER OF ROADS TO ACCESS BLOCK}
MAXADJAC, { MAX NUMBER OF ADJACENT BLOCKS}
MAXMRD, { MAXIMUM NUMBER OF MAIN ROADS }
MINAGE :INTEGER; { MINIMUM HARVEST AGE }
HABDELAY,CUT1,CUT2 :REAL;
DISKFILE :TEXT;
AGE :agePtr;
STARTAGE :startagePtr;
ACCESS :accessPtr;
ADJ :adjacPtr;
SRCOST :BLOCKARRAY;
BLOCK :blockPtr;
AREA :areaPtr;
VOLUME :volumePtr;
NETVAL :revenuePtr;
VTABLE :vtablePtr;
RTABLE :rtablePtr;
S,
SR :REALARRAY;
SRLENGTH :BLOCKARRAY;
SRBUILT :BLOCKARRAY;
SECLNGTH :PERIODARRAY;
SRMAINT :REALARRAY;
SECMaint :BLOCKARRAY;
TRANS, LOGCOST, BLKVOL :blockarray;
mr :mrPtr;
mrmaint :mrmaintPtr;
MRLENGTH :MRLENARRAY;
MRBUILT :MRLENARRAY;
MAINLENGTH :PERIODARRAY;
MAINMAINT :MRLENARRAY;
MRCOST :MRLENARRAY;
TOTALLENGTH :PERIODARRAY;
TOTALMAINT :PERIODARRAY;
GRADECOST :PERIODARRAY;
MAXGRADECOST :PERIODARRAY;
DISC,
MAINRDS, SECRDS,
VOL, HARVCOST,
TAREA,
MINVOL, MAXVOL, MAXROAD,
MINREV, MAXREV,NETREV,
NET,ROADMAINT,
MARGIN, HAUL :PERIODARRAY;

CURSORX,CURSORY,
CODE,
REP,
REPSDESIRED,
J,R :INTEGER;
SEED :longint;
RANDSEEDS,
DNR :DNRARRAY;
HIGHSOFARDNR,
HIGHESTDNR :REAL;

```
ASKPRINT          :CHAR;
A,B,C,D,E,X,Y,Z  :TEXT;
OUTFILE           :STRING;
HOUR1,MIN1,SEC1,HUNDRETH1 :WORD;
Resp,passroad,
Test, passvol,passrev,passgrade :Boolean;
zx,SEEDY          :double;
```

```
(*=====SCREEN DISPLAY=====*)
```

```
PROCEDURE DRAWBOX; { Draws a header for the screen }
```

```
var
```

```
W : Integer;
```

```
BEGIN
```

```
CLRSCR;
```

```
GOTOXY(1,1);
```

```
WRITE (#201);
```

```
FOR W:= 1 TO 78 DO
```

```
WRITE (#205);
```

```
WRITE (#187);
```

```
WRITE (#186,'          MCIP   Block   Scheduler');
```

```
WRITE ('          %186);
```

```
WRITE (#200);
```

```
FOR W:= 1 TO 78 DO
```

```
WRITE (#205);
```

```
WRITE (#188);
```

```
WRITELN;
```

```
END;
```

```
(*-----screen display-----*)
```

```
(*=====introduction=====*)
```

```
PROCEDURE INTRODUCTION;
```

```
var
```

```
PAUSE           :STRING;
```

```
PROCEDURE COLLECT;
```

```
var
```

```
RESP, TESTSEED, TESTREP : BOOLEAN;
```

```
SEEDRESPONSE           : CHAR;
```

```
BEGIN
```

```
DRAWBOX;
```

```
TESTREP:=FALSE;
```

```
WRITE ('PLEASE TYPE IN THE DESIRED NUMBER OF REPETITIONS OF THE PROGRAM: ');
```

```
REPEAT { TESTREP }
```

```
READLN (REPSDESIRED);
```

```
IF REPSDESIRED > 0 THEN
```

```
BEGIN
```

```
  If repsdesired <= maxreps then BEGIN
```

```
    TESTREP:=TRUE;
```

```
  END
```

```
  ELSE
```

```
    WRITE ('PLEASE ENTER A NUMBER BETWEEN 1 AND', MAXREPS:5,' : ');
```

```
  END
```

```
  ELSE
```

```
    WRITE ('PLEASE ENTER A NUMBER BETWEEN 1 AND', MAXREPS:5,' : ');
```

```
  UNTIL TESTREP;
```

```
    WRITE ('PLEASE ENTER SEED: ');
```

READLN(zx); {seed for random numer generator}

END;

(*-----end introduction-----*)

(*=====clear result files=====*)

PROCEDURE CLEARFILES; { EMPTIES RESULT FILES }

BEGIN

ASSIGN (X,'SEEDDNR.txt');

RESET (X);

REWRITE (X);

CLOSE (X);

END;

(*-----end result files-----*)

(*=====read input files constraints=====*)

PROCEDURE READCONSTRAINTS; { READS IN VOLUME REV CONSTRAINT DATA FILE }

VAR

I :INTEGER;

TITLE1,TITLE2,TITLE3,TITLE4 :STRING;

ZONENUM :ARRAY[1..MOSTZONES] OF INTEGER;

BEGIN

ASSIGN (E, 'CONSTRAINTS.txt');

RESET(E);

READLN(E,TITLE1);

READLN (E, HABDELAY,MAXBLOCK,MAXPERIOD,MAXZONE,MINAGE,MAXADJAC);

READLN(E,TITLE2);

READLN (E, MAXMRD,MAXACCESS,COSTMAINT,MAXROAD[1],CUT1,CUT2);

READLN(E,TITLE3);

READLN (E, DISC[1],MINVOL[1],MAXVOL[1],MINREV[1],MAXREV[1],MAXGRADEECOST[1]);

READLN (E,TITLE4);

FOR I:= 1 TO MAXZONE DO

BEGIN

READ(E,ZONENUM[I],FIRSTENTRY^[I])

END;

CLOSE (E);

FOR I:=1 TO MAXPERIOD DO

BEGIN

DISC[I]:=DISC[1];

MINVOL[I]:=MINVOL[1];

MAXVOL[I]:=MAXVOL[1];

MINREV[I]:=MINREV[1];

MAXREV[I]:=MAXREV[1];

MAXGRADEECOST[I]:=MAXGRADEECOST[1];

MAXROAD[I]:=MAXROAD[1];

END;

MINVOL[1]:=CUT1;

MINVOL[2]:=CUT2;

END;

(*-----end read constraints-----*)

```

begin
  READCONSTRAINTS;
  DRAWBOX;
  WRITELN;
  WRITELN (TO PROMPT THE PROGRAM TO CONTINUE JUST PRESS THE "RETURN" KEY.);
  WRITELN;
  WRITELN (THE CONSTRAINTS USED IN THE MODEL ARE:);
  WRITELN;
  WRITELN (' PLANNING HORIZON:= ',MAXPERIOD:8,' { MAX TIME PERIODS}');
  WRITELN (' NUMBER OF BLOCKS:= ',MAXBLOCK:8,' { MAX NO. BLOCKS }');
  WRITELN (' NUMBER OF ZONES:= ',MAXZONE:8,' { NO. ZONES }');
  WRITELN (' NUMBER OF MAIN ROADS:= ',MAXMRD:4,' { MAX NO. ROADS }');
  WRITELN (' MIN HARVEST AGE := ',MINAGE:8,' { MINIMUM HARVEST AGE }');
  WRITELN (' MAX ACCESS LINKS := ',MAXACCESS:8,' { ACCESS LINKS PER BLOCK }');
  WRITELN (' ADJAC DELAY AGE := ',HABDELAY:8:0,' { MINIMUM AGE OF ADJAC BLOCK }');
  WRITELN (' MINIMUM VOLUME := ',MINVOL[3]:8:0,' { MIN VOLUME HARVESTED PER PERIOD
});
  WRITELN (' MAXIMUM VOLUME := ',MAXVOL[1]:8:0,' { MAX VOLUME HARVESTED PER PERIOD
});
  WRITELN (' MAXIMUM COST := ',MAXREV[1]:8:0,' { MAX COST PER PERIOD }');
  WRITELN (' MAX GRADE COST := ',MAXGRADE[1]:8:0,' { MAX GRADE BUDGET PER
PERIOD}');
  WRITELN (' MAINTENANCE COST/KM := ',COSTMAINT:8:0,' {MAINT. COST/M}');
  WRITELN (' MAX ROAD COSTRUCTION:= ',MAXROAD[1]:8:0,' {MAX KM. ROAD CONST. PER
PERIOD }');
  WRITELN (' MIN VOL PERIOD1 := ',CUT1:8:0);
  WRITELN (' MIN VOL PERIOD2 := ',CUT2:8:0);
  pause:=READKEY;
  COLLECT;
  CLEARFILES;
END;
(*-----end read constraints-----*)

```

```

(*=====read in block data=====*)
PROCEDURE READBLOCKDATA; { READS IN BLOCK,AREA,AGE AND ADJACENT BLOCKS }
VAR
  I,K :INTEGER;
  TITLE1 :STRING;
BEGIN
  ASSIGN (A, 'BLOCKDAT.txt');
  RESET(A);
  FOR I:=1 TO MAXZONE DO
  BEGIN
    ZCOUNTER^[I]:=0;
  END;
  readln(A,TITLE1);
  FOR I := 1 TO MAXBLOCK DO
  BEGIN
    READ (A, BLOCK^[I],AREA^[I],STARTAGE^[I],ZONE^[I]);
    ZCOUNTER^[ZONE^[I]]:=ZCOUNTER^[ZONE^[I]]+1;
    BLOCKINZONE^[ZONE^[I],ZCOUNTER^[ZONE^[I]]]:=I;
    FOR K:=1 TO MAXADJAC DO
    BEGIN

```

```
      READ (A, ADJ^[I,K]);  
      END;  
    END;  
  CLOSE (A);  
END;  
(*-----end block data-----*)
```

```
(*=====read in existing block volume tables=====*)  
PROCEDURE EXISTVOLUME;  
VAR  
  I,K :INTEGER;  
BEGIN  
  ASSIGN (A, 'EXVOLTAB.TXT');  
  RESET(A);  
  FOR I := 1 TO MAXBLOCK DO  
    BEGIN  
      READ (A, BLOCK^[I]);  
      FOR K:=1 TO MAXPERIOD DO  
        BEGIN  
          READ (A, VOLUME^[I,K,1]);  
          END;  
          readln(a);  
        END;  
      CLOSE (A);  
    END;  
  END;  
(*-----end existing volumes-----*)
```

```
(*=====read in regen block volume tables=====*)  
PROCEDURE REGENVOLUME;  
VAR  
  I,K :INTEGER;  
BEGIN  
  ASSIGN (A, 'RGVOLTAB.TXT');  
  RESET(A);  
  FOR I := 1 TO MAXBLOCK DO  
    BEGIN  
      READ (A, BLOCK^[I]);  
      FOR K:=1 TO MAXPERIOD DO  
        BEGIN  
          READ (A, VOLUME^[I,K,2]);  
          END;  
          readln(a);  
        END;  
      CLOSE (A);  
    END;  
  END;  
(*-----end regen volumes-----*)
```

```
(*=====read in existing block revenue tables=====*)
PROCEDURE EXISTREVENUE;
VAR
  I,K :INTEGER;
BEGIN
  ASSIGN (A, 'EXREVTAB.TXT');
  RESET(A);
  FOR I := 1 TO MAXBLOCK DO
    BEGIN
      READ (A, BLOCK^[I]);
      FOR K:=1 TO MAXPERIOD DO
        BEGIN
          READ (A, NETVAL^[I,K,1]);
          END;
          readln(a);
        END;
      CLOSE (A);
    END;
  (*-----end existing revenues-----*)
```

```
(*=====read in regen block revenues tables=====*)
PROCEDURE REGENREVENUE;
VAR
  I,K :INTEGER;
BEGIN
  ASSIGN (A, 'RGREVTAB.TXT');
  RESET(A);
  FOR I := 1 TO MAXBLOCK DO
    BEGIN
      READ (A, BLOCK^[I]);
      BEGIN
        FOR K:=1 TO MAXPERIOD DO
          READ (A, NETVAL^[I,K,2]);
          END;
          readln(a);
        END;
      CLOSE (A);
    END;
  (*-----end regen volumes-----*)
```

```
(*=====read in main roads to access blocks=====*)
PROCEDURE ACCESSROADS;
VAR
  I,K,MARKER :INTEGER;
BEGIN
  ASSIGN (E, 'MRACCESS.TXT');
  RESET (E);
  FOR I:=1 TO MAXBLOCK DO
    BEGIN
      READ (E, BLOCK^[I], TRANS^[I]);
      K:=0;
      WHILE NOT EOLN(E) DO
        BEGIN
```

```
        K:=K+1;
        READ (E, ACCESS^[I,K]);
        END;
    END;
    READLN(E);
    MARKER:=K;
    WHILE NOT EOF(E) DO
    BEGIN
        FOR I:=1 TO MAXBLOCK DO
        BEGIN
            K:=MARKER;
            WHILE NOT EOLN(E) DO
            BEGIN
                K:=K+1;
                READ (E, ACCESS^[I,K]);
                END;
            READLN(E);
            END;
        END;
    END;
    CLOSE (E);
END;

(*-----end read road access-----*)
```

```
(*=====read main road lengths and const'n costs=====*)
PROCEDURE READMRCOST;
VAR
    M :INTEGER;
BEGIN
    ASSIGN (B, 'MRLINKS.TXT');
    RESET (B);
    FOR M := 1 TO MAXMRD DO
    BEGIN
        READ (B, MR^[M,J], MRLLENGTH[M], MRCOST[M]);
        END;
    CLOSE (B);
END;

(*-----end main road const'n costs-----*)
```

```
(*=====read secondary road lengths and const'n costs=====*)
PROCEDURE READSRCOST;
VAR
    I :INTEGER;
BEGIN
    ASSIGN (B, 'SRLINKS.TXT');
    RESET (B);
    FOR I := 1 TO MAXBLOCK DO
    BEGIN
        READ (B, BLOCK^[I], SRLLENGTH[I], SRCOST[I]);
        END;
    CLOSE (B);
END;

(*-----end secondary road const'n costs-----*)
```

```
(*=====random number generator=====*)
PROCEDURE RANNUM;
VAR
  I,k      :INTEGER;
  XETA     :REAL;      {X=DECIMAL, S=O or }
  KX,ALPHA,MU,MM,ZZX :DOUBLE;
BEGIN
  ALPHA:=EXP(LN(7)*5);      {alpha,mu,mm for random generator}
  MU:=EXP(LN(2)*31);
  MM:=EXP(LN(2)*31)-1;
                        {ZX= seed}
  FOR I:=1 TO MAXBLOCK DO
  BEGIN

    ZZX:=INT(ALPHA*ZX/MU);      {whole number from division}
    ZZX:=(ALPHA*ZX)-(ZZX*MU);    {modulus arithmetic}
    KX:=INT(ALPHA*ZX/MU);      {whole number}
    IF (ZZX+KX) < MM THEN ZX:=ZZX+KX
    ELSE
      ZX:=ZZX+KX-MM;
    XETA:=ZX/MU;

    S[I,J]:= INT(XETA+0.5);
    IF AGE^[I,J] < MINAGE THEN S[I,J]:=0; {if unit immature..set to zero}
    END;

    {set zones zero if before first entry}
    FOR I:=1 TO MAXZONE DO
      if j < FIRSTENTRY^[I] then
        begin
          for K:=1 to ZCOUNTER^[I] do
            S[BLOCKINZONE^[I,K],j]:=0
          END;
        END;
  END;
END;
(*-----end random number generator-----*)
```

```
(*=====set blocks to zero=====*)
PROCEDURE SETTOZERO;
VAR
  I,K      :INTEGER;
BEGIN
  VOL[J]:=0;
  NET[J]:=0;
  FOR K:=1 TO MAXPERIOD DO DNR[K]:=0;
  FOR I:=1 TO MAXBLOCK DO S[I,J] :=0;
END;
(*-----end blocks to zero-----*)
```



```
(*=====set roads to zero=====*)
PROCEDURE INITROADS;
VAR
  M,K :INTEGER;
BEGIN
  FOR M:=1 TO MAXMRD DO      (*main roads to zero*)
    BEGIN
      MR^[M,J] :=0;
      MRMAINT^[M,J]:=0;
    END;
  FOR K:=1 TO MAXBLOCK DO
    BEGIN
      SR[K,J] := 0;      (* initialize secondary roads *)
      SRMAINT[K,J]:=0;
    END;
  END;
(*-----end roads to zero-----*)
```

```
(*=====assign roads=====*)
(* assign roads for each period *)
PROCEDURE ASSIGNROADS;
VAR
  I,K :INTEGER;
BEGIN
  {main roads}
  FOR I:=1 TO MAXBLOCK DO
    BEGIN
      IF S[I,J] = 1 THEN
        BEGIN
          {should use repeat until k=maxaccess or access[i,k]=0}
          FOR K:=1 TO MAXACCESS DO
            BEGIN
              IF ACCESS^[I,K] > 0 THEN
                BEGIN
                  MR^[ACCESS^[I,K],J]:=1;
                  MRMAINT^[ACCESS^[I,K],J]:=1;
                END;
            END;
          END;
        END;
      END;
    END;
  {secondary roads}
  FOR I:= 1 TO MAXBLOCK DO
    BEGIN
      IF S[I,J] =1 THEN
        BEGIN
          SR[I,J]:=1;
          SRMAINT[I,J]:=1;
        END;
      END;
    END;
  END;
(*-----end assign roads-----*)
```

```
(*=====check redundancies in road assignments=====*)
PROCEDURE REDUNDANTCHECK;
```

```
VAR
  I,K,M  :INTEGER;
BEGIN
  IF J > 1 THEN
    BEGIN
      FOR M:=1 TO MAXMRD DO
        BEGIN
          FOR K:=1 TO J-1 DO
            BEGIN
              IF MR^[M,K] =1 THEN MR^[M,J]:=0;
            END;
          END;
        END;

      (* check secondary roads for redundancy and correct *)
      FOR I:=1 TO MAXBLOCK DO
        BEGIN
          FOR K:=1 TO J-1 DO
            BEGIN
              IF SR[I,K] =1 THEN SR[I,J]:=0;
            END;
          END;
        END;
      END;
    END;
  END;
  (*-----end road redundancy check-----*)
```

(*=====check for adjacent blocks and set to zero=====*)

```
PROCEDURE ADJAC;
VAR
  I,K  :INTEGER;
  ADJACDELAY :BOOLEAN;
BEGIN
  FOR I:=1 TO MAXBLOCK DO
    BEGIN
      IF S[I,J] = 1 THEN
        begin
          k:=0;
          ADJACDELAY:=TRUE;
          repeat
            k:=k+1;
            IF (ADJ^[I,K] > 0) AND (AGE^[ADJ^[I,K],J] < HABDELAY) THEN
              BEGIN
                ADJACDELAY:= FALSE;
                S[I,J]:=0;
              END;
            UNTIL (ADJACDELAY=FALSE) OR (ADJ^[I,K]=0) OR (K=MAXADJAC);
          END;
        END;
      END;

      FOR I:=1 TO MAXBLOCK DO
        BEGIN
          IF S[I,J] = 1 THEN
            FOR K:=1 TO MAXADJAC DO
              BEGIN
                IF ADJ^[I,K] >0 THEN S[ADJ^[I,K],J]:=0;
              END;
```

END;

END;

(*-----end adjacency check-----*)

(*=====check net revenue of period solution=====*)

PROCEDURE REVENUE; (* first the undiscounted main road costs*)

VAR

I,M,PERIOD :INTEGER;

BEGIN

MAINRDS[J] :=0;

MAINLENGTH[J]:=0;

MAINMAINT[J]:=0;

FOR M:=1 TO MAXMRD DO

BEGIN

MAINRDS[J]:=MAINRDS[J] + MRCOST[M] * MR^[M,J];

MAINLENGTH[J]:=MAINLENGTH[J] + MRLLENGTH[M] * MR^[M,J];

MAINMAINT[J]:=MAINMAINT[J] + MRLLENGTH[M]*MRMAINT^[M,J];

END;

(* then the undiscounted secondary road costs in each period *)

SECRDS[J]:=0;

SECLENGTH[J]:=0;

SECMAINT[J]:=0;

FOR I:=1 TO MAXBLOCK DO

BEGIN

SECRDS[J]:=SECRDS[J] + SRCOST[I] * SR[I,J];

SECLENGTH[J]:=SECLENGTH[J] + SRLENGTH[I] * SR[I,J];

SECMAINT[J]:=SECMAINT[J]+SRLENGTH[I] *SRMAINT[I,J];

END;

(* calc total length (and costs) of roads constructed and maintained*)

TOTALLENGTH[J]:=MAINLENGTH[J] + SECLENGTH[J];

TOTALMAINT[J]:= MAINMAINT[J] + SECMAINT[J];

GRADEECOST[J]:=MAINRDS[J] + SECRDS[J];

(* calculate the area harvested *)

TAREA[J]:=0;

FOR I:=1 TO MAXBLOCK DO

TAREA[J]:=TAREA[J]+(S[I,J]*AREA^[I]);

(* now calculate the gross margin from each sale *)

MARGIN[J] :=0;

HAUL[J]:=0;

FOR I:=1 TO MAXBLOCK DO

BEGIN

IF RTABLE^[I] = 1 THEN PERIOD:=AGE^[I,J]-STARTAGE^[I]+1;

IF RTABLE^[I] = 2 THEN PERIOD:=AGE^[I,J];

MARGIN[J]:=

MARGIN[J]+(S[I,J]*VOLUME^[I,PERIOD,VTABLE^[I]]*(NETVAL^[I,PERIOD,RTABLE^[I]]
+TRANS[I]));

HAUL[J]:= HAUL[J]+(S[I,J]*VOLUME^[I,PERIOD,VTABLE^[I]]*TRANS[I]);

END;

(* CALCULATE THE ROAD MAINTENANCE COSTS PER PERIOD *)

ROADMAINT[J]:=TOTALMAINT[J]*COSTMAINT;

```
(* CALCULATE THE TOTAL COST FOR THE PERIOD *)
NET[J]:=0;
NET[J]:=MARGIN[J]+MAINRDS[J]+SECRDS[J]+ROADMAINT[J];
END;
(*-----end revenue -----*)
```

```
(*=====calculate DNR=====*)
PROCEDURE SUMSALES; (* sum discounted roads and sales to get DNR *)
VAR
  J :INTEGER;
BEGIN
  DNR[R]:=0;
  FOR J:=1 TO MAXPERIOD DO
    DNR[R]:=DNR[R]+NET[J]*DISC[J];
  END;
(*-----end DNR-----*)
```

```
(*=====screen display of results=====*)
PROCEDURE DISPRESULTS; (* display solutions to screen *)
VAR
  I,J :INTEGER;
BEGIN
  { clrscr;
  TEXTBACKGROUND(blue);
  WINDOW(1,1,80,25);
  drawbox;
  GOTOXY(1,12);
  WRITELN ('REP: ',R-1:3,' COST: ',DNR[R]:8:0);
  WRITELN ('BLOCKS HARVESTED BY PERIOD');
  FOR J :=1 TO MAXPERIOD DO
    BEGIN
      WRITE ('PERIOD ',J:2,'|');
      FOR I:=1 TO MAXBLOCK DO
        BEGIN
          IF S[I,J]=1 THEN WRITE ( I:4);
        END;
        Write(' ');
      WRITELN;
    END;
    WRITELN;
    WRITELN(' PERIOD VOLUME CONST_N LEN MAINT LEN CONST_N COST MAINT COST
TOTAL COST');
    FOR J:=1 TO MAXPERIOD DO
      BEGIN
        WRITELN(J:5,VOL[J]:10:0,TOTALLENGTH[J]/1000:10:1,TOTALMAINT[J]/1000:11:1,
        GRADECOST[J]/vol[j]:13:2,ROADMAINT[J]/vol[j]:13:2,NET[J]/vol[j]:13:2);
      END;
    END;
  END;
(*-----end display-----*)
```

```
(*=====signoff screen display=====*)
PROCEDURE SIGNOFF;
BEGIN
```

```
CLRSCR;
DRAWBOX;
WRITELN (' The program MCIP.exe has completed its run. ');
WRITELN (' The results have been written to the following files:');
```

```
Readln;
exit;
END;
(*-----end signoff-----*)
```

```
(*=====print results=====*)
PROCEDURE PRINT;
VAR
  I,J,L :INTEGER;
Procedure Configurepage;
VAR
  I,J,L :INTEGER;
  ANS :STRING;
Begin { configurepage }
  WRITELN (LST, '***** FEASIBLE SOLUTION *****');
  writeln(lst);
  WRITE (LST, 'COST=', DNR[R]:10:2);
  WRITELN (LST, '          SEED= ', SEEDY:15:0);
  WRITELN (LST);
  FOR J:=1 TO MAXPERIOD DO
    BEGIN
      Write (LST, 'Period ', J:2, ' ');
      FOR I:=1 TO MAXBLOCK DO
        IF S[I,J]=1 THEN WRITE (LST, ' ', I);
      Writeln(lst);
    END;
  WRITELN (LST);
  WRITELN (LST, 'HECTARES VOLUME SEC ROADS MAIN ROADS TOTAL COST SEC LEN
MAIN LEN TOTLEN');
  FOR J:=1 TO MAXPERIOD DO
    BEGIN
      WRITELN(LST, TAREA[J]:8:0, VOL[J]:8:0, SECRDS[J]:10:0, MAINRDS[J]:12:0, NET[J]:12:0,
        SECLENGTH[J]/1000:10:1, MAINLENGTH[J]/1000:10:1, TOTALLENGTH[J]/1000:10:1);
    END;

  WRITELN (LST);
  WRITELN (LST, 'SEC MAINT MAIN MAINT TOT MAINT GRADE COST RD MAINT COST
TRANSCOST');
  FOR J:=1 TO MAXPERIOD DO
    BEGIN
      WRITELN (LST, SECMAINT[J]/1000:8:1, MAINMAINT[J]/1000:10:1,
        TOTALMAINT[J]/1000:12:1, GRADECOST[J]:14:1,
        ROADMAINT[J]:14:1, HAUL[J]:14:1);
    END;
  (*prints out road links constructed and maintained over time*)
  WRITELN;
  WRITELN('DO YOU WANT THE ROADS PRINTED IN DETAIL ? ANSWER Y');
  READLN(ANS);
  IF (ANS='Y') OR (ANS='y') THEN
    BEGIN
      WRITELN (LST);
      WRITELN (LST, '          ROADS TO BE CONSTRUCTED');
      WRITELN (LST);
```

```

WRITELN (LST,' 1 2 3 4 5 6 7 8 9 10 ');
WRITELN (LST,'-----');
FOR L :=1 TO MAXMRD DO
  BEGIN
    FOR J:=1 TO MAXPERIOD DO
      BEGIN
        IF (MR^[L,J] =1) AND (J=1) THEN WRITELN(LST,L:5);
        IF (MR^[L,J] =1) AND (J=2) THEN WRITELN(LST,L:10);
        IF (MR^[L,J] =1) AND (J=3) THEN WRITELN(LST,L:15);
        IF (MR^[L,J] =1) AND (J=4) THEN WRITELN(LST,L:20);
        IF (MR^[L,J] =1) AND (J=5) THEN WRITELN(LST,L:25);
        IF (MR^[L,J] =1) AND (J=6) THEN WRITELN(LST,L:30);
        IF (MR^[L,J] =1) AND (J=7) THEN WRITELN(LST,L:35);
        IF (MR^[L,J] =1) AND (J=8) THEN WRITELN(LST,L:40);
        IF (MR^[L,J] =1) AND (J=9) THEN WRITELN(LST,L:45);
        IF (MR^[L,J] =1) AND (J=10) THEN WRITELN(LST,L:50);
      END;
    END;
    WRITELN (LST);
    WRITELN (LST,'*****');

    WRITELN (LST);
    WRITELN (LST,'          ROADS TO BE MAINTAINED');
    WRITELN (LST);
    WRITELN (LST,' 1 2 3 4 5 6 7 8 9 10 ');
    WRITELN (LST,'-----');
    FOR L :=1 TO MAXMRD DO
      BEGIN
        FOR J:=1 TO MAXPERIOD DO
          BEGIN
            IF (MRMAINT^[L,J] =1) AND (J=1) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=1) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=2) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=2) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=3) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=3) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=4) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=4) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=5) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=5) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=6) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=6) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=7) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=7) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=8) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=8) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=9) THEN WRITE(LST,L:5);
            if (mrmaint^[L,j] =0) and (j=9) then write(lst,' ');
            IF (MRMAINT^[L,J] =1) AND (J=10) THEN WRITE(LST,L:5);
            {if (mrmaint^[L,j] =0) and (j=10) then write(lst,' ');}
          END;
        END;
        WRITELN(LST);
      END;
    END;
    WRITELN (LST);
    WRITELN (LST,'*****');
  END;
END;

END; { configurepage }

```

Procedure Askforprint;

var

resp :boolean;

Begin { askforprint }

Resp:=false;

DRAWBOX;

WRITELN;

If repsdesired = 1 then

begin

WRITE ('DO YOU WANT A PRINTOUT OF THE FEASIBLE SOLUTION ? : ');

Repeat

ASKPRINT:= READKEY;

If askprint in [#89,#121,#78,#110] then Resp:= true; { yes or no }

Until resp =true;

end;

END; { askforprint }

BEGIN { procedure print main }

Askforprint;

CASE ASKPRINT OF

#89,#121:begin

Configurepage; { Y OR y }

end;

#78,#110: exit;

end; { case }

end; { print }

(*-----end print results-----*)

(*=====write DNR and seed to file=====*)

PROCEDURE WRITESEEDTOFILE;

VAR

I,J :INTEGER;

TOTALVOL,LENGTH :REAL;

BEGIN

TOTALVOL:=0;

LENGTH:=0;

FOR J:=1 TO MAXPERIOD DO

BEGIN

TOTALVOL:=TOTALVOL+VOL[J];

LENGTH:=LENGTH+TOTALLENGTH[J]/1000;

END;

ASSIGN (X,'SEEDDNR.txt');

APPEND (X);

RANDSEEDS[R] := RANDSEED;

BEGIN

WRITE (X,SEEDY:12:0,DNR[R]:10:0,TOTALVOL:10:0,LENGTH:10:1);

FOR I:=1 TO MAXPERIOD DO

WRITE(X,TOTALLENGTH[I]/1000:5:1);

FOR I:= 1 TO MAXPERIOD DO

WRITE(X,TOTALMAINT[I]/1000:5:1);

WRITELN(X);

CLOSE (X);

END;

END;

(*-----end DNR and seed to file-----*)

(*=====calc run time=====*)

Procedure CalcElapsedTime;

var

ELAPSEDHOUR, ELAPSEDMIN,
ELAPSEDSEC, ELAPSEDTIME : Word;
HOUR2,MIN2,SEC2,HUNDRETH2 :WORD;

begin

GETTIME(HOUR2,MIN2,SEC2,HUNDRETH2);

IF SEC2 < SEC1 THEN

BEGIN

ELAPSEDSEC := 60 + SEC2 -SEC1;

MIN2 :=MIN2 - 1;

END

ELSE

ELAPSEDSEC :=SEC2-SEC1;

IF MIN2 < MIN1 THEN

BEGIN

ELAPSEDMIN := 60 + MIN2 -MIN1;

HOUR2 := HOUR2-1;

END

ELSE

ELAPSEDMIN :=MIN2-MIN1;

ELAPSEDHOUR:=HOUR2-HOUR1;

ELAPSEDTIME:=(ELAPSEDHOUR*3600+ELAPSEDMIN*60+ELAPSEDSEC);

WRITELN ('LOWEST COST IS:',HIGHESTDNR:10:0);

WRITELN ('ELAPSED TIME:');

WRITELN (ELAPSEDTIME:8,' sec');

WRITELN(ELAPSEDHOUR:3,' hr',ELAPSEDMIN:3,' min',ELAPSEDSEC:3,' s');

end;

(*-----end run time-----*)

(*=====initialize vol & rev tables and flags for constructed roads=====*)

PROCEDURE INITTABLES;

VAR

I :INTEGER;

BEGIN

FOR I:=1 TO MAXBLOCK DO

BEGIN

AGE^[I,1]:=STARTAGE^[I];

VTABLE^[I]:=1;

RTABLE^[I]:=1;

END;

END;

(*-----end initialize tables and const'n flags-----*)

(*=====INCREMENT BLOCK AGES=====*)

PROCEDURE INCREMENTAGE;

VAR

I :INTEGER;

BEGIN

IF J > 1 THEN

BEGIN

FOR I:=1 TO MAXBLOCK DO


```

    AGE^[I,J]:=AGE^[I,J-1] + 1;
  END;
END;
(*-----end increment block ages-----*)

```

```

(*=====set age of blocks cut to zero=====*)
PROCEDURE ADJUSTAGE;
VAR
  I :INTEGER;
BEGIN
  FOR I:=1 TO MAXBLOCK DO
    IF S[I,J] =1 THEN
      BEGIN
        STARTAGE^[I]:=0;
        AGE^[I,J]:=0;
        VTABLE^[I]:=2;
        RTABLE^[I]:=2;
      END;
    END;
  END;
  (*-----end setting ages to zero-----*)

```

```

(*=====sum volumes for period j=====*)
PROCEDURE SUMVOL;
VAR
  I,PERIOD,K :INTEGER;
BEGIN
  FOR I:=1 TO MAXZONE DO
    BEGIN
      FOR K:=1 TO ZCOUNTER^[I] DO
        BEGIN
          IF VTABLE^[BLOCKINZONE^[I,K]]=1 THEN PERIOD:=AGE^[BLOCKINZONE^[I,K],J]
            -STARTAGE^[BLOCKINZONE^[I,K]]+1;
          IF VTABLE^[BLOCKINZONE^[I,K]]=2 THEN PERIOD:=AGE^[BLOCKINZONE^[I,K],J];

```

{check if volume is above lower bound, if so,
 {set the rest of the blocks in the zone(s) to zero}

```

          IF VOL[J] >= minvol[j] THEN
            BEGIN
              {current zone}
              FOR K:=K TO ZCOUNTER^[I] DO
                S[BLOCKINZONE^[I,K],J]:=0;
              END;
              BLKVOL[BLOCKINZONE^[I,K]]:=
                VOLUME^[BLOCKINZONE^[I,K],PERIOD,VTABLE^[BLOCKINZONE^[I,K]]]
                * S[BLOCKINZONE^[I,K],J];

              VOL[J]:=VOL[J] + BLKVOL[BLOCKINZONE^[I,K]];
            END;
          END;
        END;
      END;
    END;
  END;
  (*-----end sum volumes-----*)

```

```
(*=====start check the constraints=====*)
PROCEDURE CHECKCONSTRAINTS;
BEGIN
  IF (TOTALLENGTH[J]/1000 <= MAXROAD[J] ) THEN
    PASSROAD:=TRUE;
  IF (VOL[J] >= MINVOL[J]) AND (VOL[J] <= MAXVOL[J]) THEN
    passvol:=true;
  IF (NET[J] >= MINREV[J]) AND (NET[J] <= MAXREV[J]) THEN
    passrev:=true;
  IF (GRADEECOST[J] < MAXGRADEECOST[J]) THEN
    passgrade:=true;
  writeln('  vol ',vol[j]:10:0,'  cost ',net[j]:10:0,'  grade ',gradecost[j]:10:0,'  road
  ',totallength[j]/1000:2:0);
END;
(*-----end check constraints-----*)
```

```
(*=====beep=====*)
PROCEDURE BEEP;
BEGIN
  SOUND (500);
  DELAY (75);
  NOSOUND;
END;
(*-----end beep-----*)
```

```
(*=====start period by period loop=====*)
PROCEDURE PERIOD2;
  Var
    ExitProcPrem2 : Boolean;
    counter       : array[1..50] of integer;
BEGIN
  J:=0;
  INITTABLES;
  REPEAT
    BEGIN
      J:=J+1;
      COUNTER[J] :=0;
      INCREMENTAGE;
      ExitProcPrem2 := False;

      REPEAT
        initroads;
        settozero;
        passvol:= false;
        passrev:= false;
        passgrade:=false;
        passroad:=FALSE;
        COUNTER[J] :=COUNTER[J]+1;
        IF COUNTER[J] > 30 THEN
          BEGIN
            Test := False;
```

```

        ExitProcPrem2 := True;
        EXIT;
        END;
        RANNUM;
        ADJAC;
        SUMVOL;
        ASSIGNROADS;
        REDUNDANTCHECK;
        REVENUE;
        WINDOW(1,1,80,25);
        TEXTBACKGROUND(red);
        gotoxy(45,5);
        write('ITERATIONS PERIOD : ',J,COUNTER[J]:10);
        CHECKCONSTRAINTS;
        UNTIL (passvol AND passrev AND passgrade AND passroad);
        ADJUSTAGE;
        END;
    UNTIL J >= MAXPERIOD;
    END;

```

(*-----end period loop-----*)

(*=====start the main program=====*)

```

BEGIN
    (* MAIN PROGRAM *)
    J:=0;
    new(volume);
    new(netval);
    new(startage);
    new(age);
    new(adj);
    new(access);
    new(vtable);
    new(rtable);
    new(mr);
    new(mrmaint);
    new(area);
    new(block);
    new(zcounter);
    new(zone);
    new(zvol);
    new(firstentry);
    new(blockinzone);
    INTRODUCTION;
    GETTIME(HOUR1,MIN1,SEC1,HUNDRETH1);
    EXISTVOLUME;
    REGENVOLUME;
    EXISTREVENUE;
    REGENREVENUE;
    ACCESSROADS;
    READMRCOST;
    READSRCOST;
    HIGHSOFARDNR :=9999999999.;
    DRAWBOX;
    FOR R:=2 TO REPSDESIRED+1 DO
        BEGIN
            WINDOW(2,2,76,24);

```

```
TEXTBACKGROUND(red);
GOTOXY(1,1);
WRITELN('REP: ',R-1:5);
REPEAT
  Test:= True;
  SEEDY:=ZX;
  READBLOCKDATA;
  PERIOD2;
  IF test=TRUE THEN
    BEGIN
      SUMSALES;
      WRITESEEDTOFILE;
    END;
  UNTIL test;
  IF DNR[R] < highsofardnr THEN
    HIGHSOFARDNR:=DNR[R];
  HIGHESTDNR:=HIGHSOFARDNR;
  DISPRESULTS;
END;
CALCELAPSEDTIME;
BEEP;
READLN;
PRINT;
{signoff;}
END.
```

(*-----end of main program-----*)□

APPENDIX 8

EXAMPLES OF OUTPUT FILES

FEASIBLE SOLUTION

COST = 83366009.86

SEED = 143601313

PERIOD 1 : 1 7 8 13 15 19
 PERIOD 2 : 2 4 12 17 30
 PERIOD 3 : 14 16 18 20 24 31 41
 PERIOD 4 : 22 27 32 34 42
 PERIOD 5 : 3 9 21 25 28 33
 PERIOD 6 : 5 29 37 40 44
 PERIOD 7 : 6 8 15 19 26 36 38
 PERIOD 8 : 2 4 7 12 17 30 45 48
 PERIOD 9 : 1 10 16 20 24 35 46
 PERIOD 10 : 13 18 31 32 34 41 43

HECTARES	VOLUME SEC ROADS		MAIN ROADS	TOTAL COST	SEC LEN MAIN LEN		TOTLEN
475	233369	93825	720300	6619883	2.9	19.5	22.4
394	236534	42975	335250	5992715	1.3	10.9	12.1
549	335679	199050	715050	8780604	5.4	16.6	22.0
440	344762	27600	625425	8513922	0.9	13.1	14.0
485	343111	60450	321750	8231911	2.0	7.5	9.5
433	342277	52965	453150	8928417	1.5	11.1	12.5
563	335729	8700	147450	8736617	0.3	4.0	4.2
615	342859	14400	211650	9527481	0.5	7.1	7.5
519	344760	0	122400	8424133	0.0	4.1	4.1
557	343351	1800	21300	9610326	0.1	0.7	0.8

SEC MAINT	MAIN MAINT	TOT MAINT	GRADE COST	RD MAINT COST	TRANS COST
2.9	19.5	22.4	814125.0	1142145.0	740996.8
1.3	24.5	25.7	378225.0	1312995.0	760174.1
5.4	37.9	43.3	914100.0	2208300.0	1131632.2
0.9	38.6	39.5	653025.0	2015265.0	1216631.7
2.0	37.0	39.0	382200.0	1990785.0	1155298.3
1.5	49.0	50.4	506115.0	2572848.0	1207249.2
1.4	39.7	41.0	156150.0	2092020.0	1128142.6
3.0	35.3	38.3	226050.0	1950750.0	1176189.2
2.5	34.4	36.9	122400.0	1880370.0	1164064.4
2.7	38.8	41.5	23100.0	2116500.0	1190865.7

TYPICAL SOLUTION FOR A 80 HECTARE DELAY PERIOD 1

FEASIBLE SOLUTION

COST = 67810904.24

SEED = 1

PERIOD 1 : 3 8 13 16 31
 PERIOD 2 : 18 19 24 28 39
 PERIOD 3 : 2 14 22 33 41
 PERIOD 4 : 4 12 27 40
 PERIOD 5 : 7 23 25 38 44
 PERIOD 6 : 5 20 35 46
 PERIOD 7 : 3 15 26 29 37
 PERIOD 8 : 1 9 19 36 45
 PERIOD 9 : 6 16 18 24 30 41
 PERIOD 10 : 4 13 28 39 48

HECTARES	VOLUME SEC ROADS	MAIN ROADS	TOTAL COST	SEC LEN	MAIN LEN	TOTLEN	
399	221443	75150	660375	6313897	2.5	20.0	22.5
386	253252	108075	865875	7332786	3.3	17.7	20.9
429	272760	82275	537750	6908205	2.0	14.7	16.8
347	269928	47265	321450	6448749	1.4	7.2	8.6
384	273481	54600	237150	6314494	1.6	6.2	7.8
323	273012	37125	263850	5932608	0.8	7.9	8.8
432	273619	21675	303000	7667755	0.6	5.9	6.5
393	272331	36900	130500	6129366	1.2	4.4	5.6
444	271999	10200	70800	7548646	0.3	2.4	2.7
404	272616	4950	117300	7214399	0.2	3.9	4.1

SEC MAINT	MAIN MAINT	TOT MAINT	GRADE COST	RD MAINT COST	TRANS COST
2.5	20.0	22.5	735525.0	1148265.0	713764.8
3.3	30.2	33.4	973950.0	1704165.0	881859.0
2.0	32.3	34.4	620025.0	1753635.0	927975.6
1.4	28.5	29.9	368715.0	1524033.0	904328.9
1.6	25.8	27.4	291750.0	1397400.0	944458.8
0.8	21.6	22.4	300975.0	1141125.0	923469.8
2.0	43.0	44.9	324675.0	2291430.0	957742.8
1.7	23.2	24.8	167400.0	1266330.0	905348.6
3.5	38.3	41.8	81000.0	2133330.0	919287.8
1.0	35.8	36.8	122250.0	1876290.0	958412.5

TYPICAL SOLUTION FOR A 80 HECTARE DELAY PERIOD 2

FEASIBLE SOLUTION

COST = 58274230.45

SEED = 810616507

PERIOD 1 : 1 3 13 15 18
 PERIOD 2 : 8 25 26 29 31
 PERIOD 3 : 19 33 39
 PERIOD 4 : 2 7 14 17 41
 PERIOD 5 : 24 28 36 46
 PERIOD 6 : 4 12 34
 PERIOD 7 : 15 18 32 42
 PERIOD 8 : 10 21 40
 PERIOD 9 : 1 3 19 31 47
 PERIOD 10 : 8 25 29 37

HECTARES	VOLUME SEC ROADS		MAIN ROADS	TOTAL COST	SEC LEN MAIN LEN		TOTLEN
416	211663	135600	711300	6266347	4.5	20.3	24.8
408	225899	7350	968550	7007142	0.2	19.1	19.3
235	210556	55275	258450	4770470	1.6	7.5	9.1
405	225614	110175	382800	6029787	3.0	10.2	13.1
296	227497	17100	313050	5893739	0.4	8.8	9.3
238	220125	37125	44100	4551109	1.1	1.2	2.2
339	223015	12450	155175	6179895	0.4	4.8	5.2
249	226034	25290	270750	5597294	0.8	7.1	7.9
410	211288	16650	32400	5640978	0.6	1.1	1.6
330	213422	9675	72000	6337470	0.2	1.6	1.8

SEC MAINT	MAIN MAINT	TOT MAINT	GRADE COST	RD MAINT COST	TRANS COST
4.5	20.3	24.8	846900.0	1266585.0	687567.5
0.2	31.0	31.2	975900.0	1591965.0	767151.9
1.6	17.4	19.0	313725.0	969510.0	732239.0
3.0	31.9	34.9	492975.0	1779390.0	750005.8
0.4	31.8	32.2	330150.0	1643220.0	817869.2
1.1	15.6	16.7	81225.0	849660.0	715300.6
2.9	28.8	31.8	167625.0	1619505.0	779601.2
0.8	29.2	30.0	296040.0	1529643.0	758966.6
3.0	23.6	26.5	49050.0	1353540.0	711642.6
0.5	41.2	41.6	81675.0	2123385.0	737639.1

TYPICAL SOLUTION FOR A 80 HECTARE DELAY PERIOD 3

APPENDIX 9
TYPICAL BLOCK SOLUTIONS

A 80 HECTARE, DELAY PERIOD 1 SOLUTION

A 80 HECTARE, DELAY PERIOD 2 SOLUTION

A 80 HECTARE, DELAY PERIOD 3 SOLUTION

A 40 HECTARE, DELAY PERIOD 1 SOLUTION

A 40 HECTARE, DELAY PERIOD 2 SOLUTION

A 40 HECTARE, DELAY PERIOD 3 SOLUTION

A 20 HECTARE, DELAY PERIOD 1 SOLUTION

A 20 HECTARE, DELAY PERIOD 2 SOLUTION

A 20 HECTARE, DELAY PERIOD 3 SOLUTION