

**Spatially Explicit
Timber Supply Modeling**

March 31, 2002

©Barrodale Computing Services Ltd.
Victoria, BC, Canada

Table of Contents

EXECUTIVE SUMMARY	1
INTRODUCTION	2
PROBLEM SCOPE AND ISSUES	2
SIMULATION OF HARVESTING	3
OPTIMIZATION OF HARVEST SCHEDULES	3
<i>Notation</i>	4
<i>Approaches to Optimization</i>	4
STATEMENT OF THE PROBLEM	6
HARVEST FLOW OPTIMIZATION.....	6
MANAGEMENT OBJECTIVES	7
<i>Environmental Constraints</i>	7
<i>Cutblock Size Distribution Constraints</i>	8
<i>Adjacent Cutblock Green-up Constraints</i>	9
<i>Cutblock Harvest Completion Constraints</i>	10
<i>Accessibility Constraints</i>	10
SIMPLIFICATIONS.....	11
SPATIALLY EXPLICIT SIMULATION	11
OVERVIEW	11
SIMULATION OBJECTIVES	12
SPATIAL HIERARCHY	14
MANAGEMENT ZONES AND RESERVES	16
SIMULATION PROCESS	16
SIMULATION STRUCTURES	17
<i>Spatial Units</i>	17
<i>Queues</i>	18
SPEED AND EFFICIENCY CONSIDERATIONS	19
OPPORTUNITIES FOR PARALLELISM	20
DATA I/O, STORAGE, AND ANALYSIS	21
CONSIDERATIONS FOR OPTIMIZATION	21
OPTIMIZATION: PROOF OF CONCEPT	23
SIMULATION PROCEDURE	23
<i>Overview</i>	23
<i>Implementation</i>	24
OPTIMIZATION APPROACH	25
RESULTS OF SIMULATION/OPTIMIZATION TESTS	26
DISCUSSION	30
BIBLIOGRAPHY.....	31
APPENDIX 1: ENVIRONMENTAL CONSTRAINTS	32
BIODIVERSITY CONSTRAINTS	32
<i>Formulation of Constraints</i>	32
<i>Associated Data</i>	34
VISUAL QUALITY CONSTRAINTS	35
<i>Formulation of Constraints</i>	35
<i>Associated Data</i>	35
COMMUNITY WATERSHED CONSTRAINTS	37

<i>Formulation of Constraints</i>	37
<i>Associated Data</i>	38
WILDLIFE HABITAT CONSTRAINTS.....	38
<i>Formulation of Constraints</i>	38
<i>Associated Data</i>	39
RIPARIAN MANAGEMENT CONSTRAINTS.....	39
<i>Formulation of Constraints</i>	39
<i>Associated Data</i>	40
GENERIC ENVIRONMENTAL CONSTRAINTS.....	41
<i>Formulation of Constraints</i>	41
<i>Associated Data</i>	42
COMPUTING CONTRIBUTIONS TO ENVIRONMENTAL CONSTRAINTS.....	42
<i>Computation of Weights</i>	42
<i>Simplification of Weighting for Efficient Modeling</i>	44
APPENDIX 2: CUTBLOCK CONSTRAINTS.....	44
CUTBLOCK DELINEATION AND SIZE CONSTRAINTS.....	45
GREEN-UP CONSTRAINT FOR HARVESTING ADJACENT CUTBLOCKS.....	45
TIMELY HARVEST COMPLETION CONSTRAINTS.....	46
FUTURE AUTOMATED CUTBLOCK GENERATION.....	46
APPENDIX 3: ACCESSIBILITY CONSTRAINTS	47
MODELING ACCESSIBILITY AT THE CUTBLOCK LEVEL.....	47
FUTURE ACCESSIBILITY MODELING.....	48
APPENDIX 4: ATTRIBUTES OF SPATIAL UNITS.....	49
FOREST MANAGEMENT UNIT ATTRIBUTES.....	49
LANDSCAPE UNIT ATTRIBUTES.....	50
DEVELOPMENT UNIT ATTRIBUTES.....	50
BLOCKING UNIT ATTRIBUTES.....	52
<i>BU Fixed Attributes</i>	52
<i>BU Variable Attributes</i>	53
CUTBLOCK ATTRIBUTES.....	53
<i>Cutblock Fixed Attributes</i>	53
<i>Cutblock Variable Attributes</i>	54
ATOM ATTRIBUTES.....	56
<i>Atom Fixed Attributes</i>	56
<i>Atom Variable Attributes</i>	58
APPENDIX 5: MANAGEMENT ZONE AND RESERVE ATTRIBUTES.....	61
MANAGEMENT ZONE ATTRIBUTES.....	61
RESERVE ATTRIBUTES.....	62
APPENDIX 6: ECONOMIC CONSIDERATIONS	63
APPENDIX 7: CUTTING TYPES.....	64
APPENDIX 8: TABLES	66

Executive Summary

Barrodale Computing Services Ltd. (BCS) has investigated issues involved in designing a spatially explicit timber supply model for the Timber Supply Branch of the British Columbia (BC) Ministry of Forests (MoF), and has prepared an overall design for this modeling. The purpose of this model is to predict sustainable timber harvest levels that are realizable in practice, and that satisfy the forest management (environmental) constraints as currently laid out in the BC Forest Practices Code. Realization of this model will ultimately provide detailed feasible solutions for harvesting that also ensure that the constraints are satisfied. The design includes both simulation and optimization components, and proposes a very effective approach for the latter.

This report contains a description of the issues taken into account in the design of a spatially explicit simulation model, outlines the major aspects of the design itself, and presents the formulation and a proof-of-concept of the proposed approach to optimization. The key contents and results of this report are as follows:

- A mathematical statement of the problem is presented.
- The general types of issues and objectives associated with the modeling are identified and discussed.
- The constraints for achieving environmental objectives in management zones for biodiversity, visual areas, community watersheds, wildlife habitat, and riparian areas are described and formulated mathematically. A computation scheme is also proposed for allowing the partial contribution of an area to a constraint.
- A spatial unit hierarchy with six levels is proposed, from the Forest Management Unit¹ (FMU) at the highest level to the resultant polygon (subdivision of a stand) at the lowest level. The attributes to be maintained for each of these spatial unit levels during the modeling are itemized and described.
- A general algorithm for the simulation process is presented. Possible bottlenecks, simplifications and opportunities for parallelism in this algorithm are discussed, and some key data structures to support it are defined.
- An approach to optimization based on target harvest flows is proposed that involves a very simple parameterization, and its feasibility is demonstrated in a prototype version using a simplified but representative simulation model.

The overall result of this work is a design for a spatially explicit simulation, and verification in principle of a proposed approach to harvest optimization. This verification is provided by numerical results obtained by applying an older general-purpose optimization algorithm (due to Nelder and Mead) to a parameterized formulation that dramatically reduces the computational complexity of this harvest optimization problem. It is our expectation that a more efficient optimization algorithm than Nelder-Mead could be devised that takes much greater advantage of the structure of the new parameterized formulation presented in this report.

¹ A Forest Management Unit is generally either a Timber Supply Area (TSA) or a Tree Farm License (TFL).

Introduction

In this project, BCS has worked under contract with the Timber Supply Branch (TSB) of the BC MoF to provide analysis and initial design specifications for a spatially explicit timber supply model. The aims of this project were to define the issues and scope of the simulation and optimization problems to be solved, to propose procedures and structures for the simulation, and to formulate approaches for optimization of harvest schedules subject to environmental and other types of constraints. This report describes the results of this joint work, and also demonstrates the feasibility of a novel BCS approach to handling the optimization problem that must be solved.

Problem Scope and Issues

The aim of the first stage of the project was to define the scope of the problem and the issues that must be addressed, and to formulate a mathematical statement of the problem. To this end, BCS and TSB participated in a series of meetings at which forest management issues were presented and discussed. As a result of this process, and additional sources of information (including the Forest Practices Code and technical papers – see the *Bibliography* section), the key characteristics of the required spatial modeling were identified, and some approaches to modeling were proposed. These included the definition of appropriate spatial subdivisions and the attributes for each, as well as the explicit formulation of environmental constraints. In addition, proposals were devised for handling cutblock-related issues and modeling the progression of road accessibility.

The major key modeling issues that were identified and are discussed in this report are:

- the spatial hierarchy for the simulation;
- the attributes to be maintained for the units at each spatial level during the spatially explicit modeling;
- explicit environmental constraints for achieving management objectives in biodiversity, visual areas, community watersheds, wildlife habitat, and riparian areas;
- considerations in cutblock definition and harvesting;
- issues in modeling road accessibility;
- the existence of various cutting types;
- some economic considerations, including various types of costs;
- the basic processes and structures that would be involved in realistic simulation;
- speed and efficiency considerations and opportunities for parallelism in simulation;
- the role of harvest schedules in simulation and optimization;
- the formulation of the parameter space and objective function in optimization.

Simulation of Harvesting

Forest management and timber supply analysis make increasing use of harvest scheduling to plan feasible and sustainable harvesting regimens that are consistent with environmental and economic objectives. In the past, much of this analysis has been strategic and has involved aggregating large numbers of similar individual units (e.g., forest stands) and then using the aggregated information to design overall schedules. Recently, the need has arisen to perform more realistic and detailed modeling in which the spatial units are treated individually rather than being aggregated.

During the course of this project, BCS has formulated a basic design for a simulation procedure and associated data that would support this spatially explicit modeling. One outcome of this process was the definition of a spatial unit hierarchy. In this hierarchy, resultant forest polygons (i.e., subdivisions of stands) constitute the atomic spatial units for the modeling. These atoms would be generated from forest cover polygons during a preprocessing stage. Subdivision of these polygons would be required in cases where riparian reserve zones or riparian management zones occur within a stand, or where the forest cover polygons are very large. Each of the resulting atoms must be associated with certain attributes that will be used in the modeling. In addition, these atoms constitute the basic units by which forest management constraints can be formulated and evaluated in the simulation. Hence, a substantial part of the analysis in support of the spatial model was the definition of the types of atom-related environmental constraints that must be satisfied and the identification of the necessary attributes of the atoms (and other spatial units) to be used in the simulation.

The overall procedure proposed for simulation is described in the *Spatially Explicit Simulation* section below, and the basic structures that would be involved in realizing the simulation, including queues and lists, are outlined. Several operations in the simulation that are likely to be most laborious are identified, and possible ways of improving the speed (including exploiting parallelism) are given. In addition, a proposal to link the simulation to an object-relational database is made, and a mechanism for combining simulation and optimization is presented. That section, together with Appendices 1 – 8, constitutes the basic design for the spatially explicit modeling

Optimization of Harvest Schedules

Recently, there has been an increased need for optimizing harvest schedules based on spatially explicit modeling. The aim here is to determine a near-optimal schedule for harvesting the individual forest units (perhaps several times over a period of decades) in such a way that spatially based constraints are satisfied for each time period over a long time horizon (typically several hundred years). These constraints can include:

- targets for the harvest flow (i.e., volume cut) for various spatial regions
- requirements on the age structure of certain spatial regions (e.g., watersheds, wildlife habitat)

- requirements that the sizes of the areas cut within a region (i.e., a cutblock) correspond closely to a predefined distribution for that region
- prohibition of the harvesting of cutblocks adjacent to those recently cut until a certain time has passed
- rapid completion of harvesting of the stands within a cutblock
- accessibility by road.

Harvest scheduling problems have been formulated in many ways, and at various levels of complexity. Two basic approaches have generally been taken in approaching the scheduling problem, as follows:

- **Exact solution** is possible when the problem formulation is relatively simple, the constraints are linear, and the size is not excessively large. In these cases, standard numerical optimization procedures such as linear programming and mixed integer programming have been used to compute a solution.
- **Approximate solution** has been used when the formulation is more complex, the constraints are nonlinear, and/or the problem size is very large. In such cases, simulation and heuristic iterative procedures (e.g., simulated annealing) have often been used to identify reasonable solutions.

Notation

Let the vector $\mathbf{x} = \{x_{it}\}$, $1 \leq i \leq I$, $1 \leq t \leq T$, denote the harvest schedule variables,

where:

x_{it} denotes the fraction of stand i that is cut during time period t ; for clear cutting, $x_{it} \in \{0,1\}$, while for partial cutting, we will assume that $L_i \leq x_{it} \leq U_i$ (this is actually a simplification, as there are various types of partial cutting regimes, for which the eligibility varies with i and t , and the priority varies with region);

I is the number of stands;

T is the number of time periods.

Note: There can be up to 10^7 stands and 500 time periods, for a total of $5 \cdot 10^9$ individual variables x_{it} .

Approaches to Optimization

In one type of approach to the optimization, the problem is to find $\mathbf{x}$ to maximize the objective function

$$\sum_t \sum_i \alpha_{it} x_{it} ,$$

where α_{it} is the net revenue resulting from completely cutting stand i during time period t , subject to environmental and other constraints. (Note: The coefficients α_{it} depend on the sequence of events in the simulation (e.g., the revenue depends on the time since the previous cut, and consequently on $\mathbf{x}$); hence, they cannot be defined explicitly in advance.)

An alternative approach to the above constraint-based formulation is a results-based procedure. Here, instead of maximizing revenue while requiring that all the constraints be satisfied, the problem is presented as one of minimizing a sum of penalty functions arising from deviations from the management objectives (still expressed as the above constraints). That is, the goal is to find $\mathbf{x}$ to minimize the objective function

$$\sum_t \sum_j p_{jt}(\mathbf{x}) ,$$

where p_{jt} is the penalty imposed for deviation from the j th management objective during time period t . These management objectives incorporate net revenue considerations as well as the constraint deviations. (Note: A substantial challenge here is to assign a functional form and relative weighting scheme to be applied to deviations for each of the types of constraints/objectives described in the sections below.)

During the course of defining the modeling problem, it became clear that the full MoF optimization problem is non-linear, of huge size, and dependent on data of questionable accuracy. The problem size, the nature of some of the constraint categories, and the desire by analysts to obtain numerical solutions in a matter of minutes, effectively prevents the use of conventional numerical optimization techniques, such as linear programming. Other approaches that also attempt to optimize $\mathbf{x}$ (e.g., simulated annealing) were considered likely to be of limited value here because of problem size and the large numbers of simulations that would be required to arrive at a near-optimal solution. Hence, we searched for other parameterizations and optimization techniques that could be applied to the harvesting problem.

The procedure we developed and propose here for optimization considerably simplifies the situation. Rather than viewing $\mathbf{x}$ as a parameter space and manipulating these parameters explicitly during the optimization, we use a simple harvest flow function defined by a very small number of parameters, which are then optimized in such a way that the constraints are respected. This allows the use of computationally efficient algorithms to solve the optimization problem. The suitability of this approach to optimization is demonstrated using a simple but representative example.

Statement of the Problem

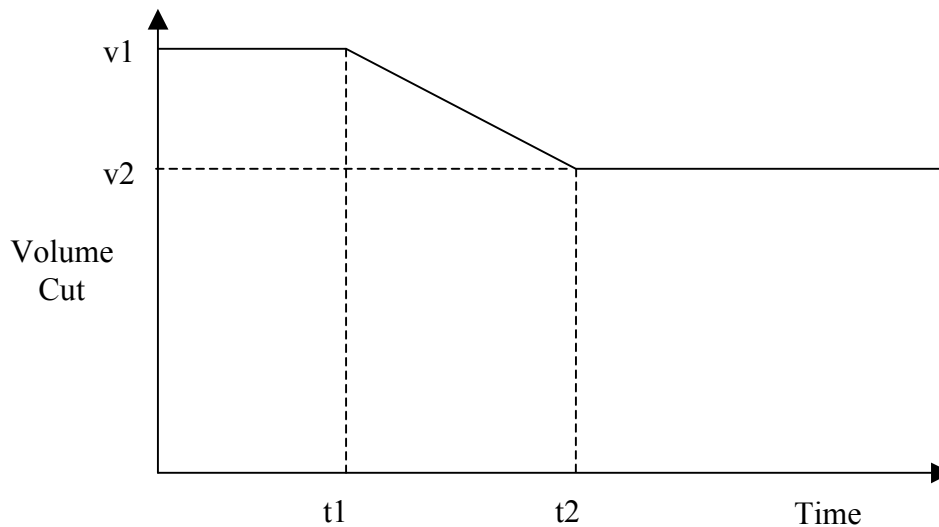
The primary objective for spatially explicit timber supply modeling by the MoF is to define a harvest schedule to maximize the volume or value of the timber cut within the region of interest (e.g., timber supply area) and that is consistent with:

- ***A desirable harvest flow pattern.*** Generally, the harvest flow (volume cut during each time period of the modeling) should start at the current level, maintain that level for a number of periods, and then decrease to a constant sustainable level. (The time periods involved and the sustainable level are not specified in advance.)
- ***Management objectives.*** These include environmental constraints, adjacency restrictions, accessibility requirements, and the like. They are typically handled by testing whether the harvesting of the stands within a cutblock would violate a constraint, or by removing particular cutblocks from consideration for harvesting during certain time periods.

Hence, the essential problem to be solved is to determine a maximal harvest flow profile that is of the desired functional form and that is also consistent with the specified set of initial conditions and constraints, using a spatially explicit harvest simulation.

Harvest Flow Optimization

This target harvest flow is often of the simple form shown in the following diagram (although other, more complex forms can be defined):



In this functional form, the harvest level is maintained at the initial level v_1 until time t_1 , then decreases linearly to level v_2 at time t_2 , and remains at this lower level until the end of the time horizon. This curve can be parameterized by the variables t_1 , t_2 , and v_2 .

In an optimization, each set of values for all these parameters would determine a possible candidate harvest profile. Simulations would then be run to determine whether the levels in this profile could be met for each time period while respecting the constraints, and each simulation run would provide a value for an objective function for that set of parameters. This function could consist of the total volume or value of all the timber harvested, decreased as necessary to penalize the situation where the actual levels harvested fell short of the targets for that simulation. It is reasonable to expect that a suitable approach could be implemented to define a near-optimal harvest profile with these characteristics (i.e., as high as possible while still consistent with the above functional form and the constraints).

The parameter space for such an approach to optimization consists of the variables that define the form of the harvest profile. As noted above, this profile could be simpler or more complicated than the one shown above, depending upon the requirements of the modeling.

The overall problem is to define a harvest flow profile of the desired form that maximizes the overall volume (or value) of the timber while respecting a set of constraints. These constraints are now presented in mathematical form (they are also described in more detail in Appendix 1).

Management Objectives

Management objectives are expressed mathematically in terms of constraints that must apply to certain spatial subsets of the region or in terms of restrictions that must be enforced at certain times as a result of events that occur during the simulation.

Environmental Constraints

The aim of these constraints is to ensure that certain environmental objectives are met for each of the regions in which they are meant to apply. These objectives are age- and area-based and require that the region contain at most a certain area with growth younger than a specified age, and at least a certain area with growth older than a specified age. These environmental constraints are of many individual types, but they may all be expressed in the form:

$$\frac{\sum_{i \in r_j} w_{it}(r_j^C) a_i(r_j^C)}{\sum_{i \in r_j} a_i(r_j^C)} \leq U(r_j^C) \quad ,$$

where:

$w_{it}(r_j^C)$ is the weight contribution of the i th stand within region r_j^C during time period t ;

r_j^C is the j th region within which the constraint type C must apply;

a_i is the area of the i th stand involved in constraint type C ;

L and U are lower and upper bounds for the constraint type C .

Note: The weights w_{it} depend on growth models and previous events (i.e., on $x_{i,1:t-1}$). They also depend on “utility functions”, the species composition and the age distribution of the trees in the stand, as follows:

$$w_{it}(r_j^C) = \sum_k f_{kit} \sum_n u(y_{nkit}) d(y_{nkit}),$$

where:

f_{kit} denotes the fraction of the area of the i th stand that is taken up by the k th species at time period t (note that $\sum_k f_{kit} = 1$);

y_{nkit} denotes the age range of the n th element of the age range distribution for the k th species for the i th stand at time period t ;

$u(y_n)$ is the value of the utility function for age range y_n ($0 \leq u(y_n) \leq 1$);

$d(y_n)$ is the fractional amount of the trees which are in age range y_n (note that $\sum_n d(y_n) = 1$).

See Appendix 1 for more details about the environmental constraints and the computation of the weights w_{it} .

Cutblock Size Distribution Constraints

Cutblocks are regions defined as contiguous groups of stands which are similar in characteristics and are to be treated as a unit (i.e., all stands in the cutblock are to be cut during the same time period).

The cutblocks might be delineated in advance, but if this is not the case, they must be formed as a part of the harvest scheduling process by appropriate grouping of stands. The requirement that a cutblock must be formed from stands which have similar

characteristics may be achieved by restricting the stands to be included in the cutblock to those uncut stands in the same predefined blocking unit as the stand to be cut. The aim of the size distribution constraints is to ensure that the size distribution of the cutblocks formed by grouping stands corresponds closely to the predefined patch size distribution for the region in question. These constraints are of the form:

$$L_n(r_j) \leq \frac{\sum_{k \in D_n(r_j)} a(B_k, r_j)}{\sum_k a(B_k, r_j)} \leq U_n(r_j),$$

where:

$a(B_k, r_j)$ is the area of the k th cutblock within the j th region in which patch size distribution constraints apply;

L_n and U_n are lower and upper bounds defining tolerances on the patch size distribution;

$D_n(r_j)$ is that set of k such that $a'_n(r_j) \leq a(B_k, r_j) \leq a'_{n+1}(r_j)$, where $a'_n(r_j)$ is the lower limit of the n th area bin of the histogram defining the area distribution of patch sizes which are to apply in the j th region.

Note: It simplifies the problem substantially if cutblocks are predefined, in view of the next constraint that the stands in all adjacent cutblocks must not be cut within a certain time after the first cutblock is cut.

Adjacent Cutblock Green-up Constraints

The aim of these constraints is to limit the size of the contiguous areas that have been recently cut, by “freezing” the cutting of cutblocks adjacent to the one just cut until a certain green-up period of time has passed.

For binary x_{ij} , these constraints are of the form:

$$x_{i \in B_i, t} + x_{j \in N(B_i), t: t+t_G(B_i)} \leq 1,$$

where:

B_i is the cutblock associated with stand i ;

$N(B_i)$ is the set of stands in all cutblocks that are neighbors of cutblock B_i ;

$t_G(B_i)$ is the green-up time for cutblock B_i .

For continuous-valued (or binary) x_{ij} , these constraints are of the form:

$$x_{i \in B_i, t} x_{j \in N(B_i), t: t+t_G(B_i)} = 0.$$

Cutblock Harvest Completion Constraints

The aim of these constraints is to ensure that if any stand i contained in cutblock B is cut during period t , then all the stands in cutblock B are also cut, i.e., $x_{it} > 0 \Rightarrow x_{j \in B, t} > 0$.

For binary x_{ij} , complete harvesting of the cutblock may be expressed by a constraint of the form:

$$x_{it}(1 - \prod_{j \in B} x_{jt}) = 0 \quad ,$$

where B is the set of stands in the cutblock containing the i th stand.

For continuous x_{ij} , a more realistic, less stringent, constraint would require that, if any cutting were done in the cutblock during a time period t , the extent of the cutting within the block would be at least some minimum value. This condition may be expressed as:

$$x_{it} > 0 \Rightarrow \frac{\sum_{j \in B} a_j x_{jt}}{\sum_{j \in B} a_j} \geq L \quad ,$$

where L is a lower bound on the fraction of the cutblock that must be cut in order to be considered completely harvested.

Accessibility Constraints

The aim of these constraints is to enforce the condition that a stand cannot be cut until it is accessible by road, i.e., $x_{it} = 1 \Rightarrow \rho_{it} = 1$, where $\rho_{it} = 1$ if there is road access into stand i at time period t , and $\rho_{it} = 0$ otherwise.

If the year in which a stand becomes accessible is predefined, these constraints are of the form:

$$x_{it}(1 - \rho_{it}) = 0 \quad .$$

If the year of road accessibility is not defined, but a road network and the costs of building each network segment is provided, the harvest scheduling process must include the cost of accessibility in the net revenue coefficients α_{it} , and include ρ_{it} as parameters to be estimated along with x_{it} .

If road network and segment costs are not provided, the scheduling process must still estimate ρ_{it} along with x_{it} , but do so on the basis of cutblock adjacency.

Simplifications

As noted above, the overall problem may be simplified in several ways to improve its tractability. Some simplifications and their implications are summarized in this section.

1. The route network connecting the stands could be predefined, but with no period of accessibility specified. This would remove the problem of having to determine the connectivity between the stands/cutblocks during the process of arriving at an optimal solution, but still leave the problem of determining the year of accessibility (i.e., the parameters ρ_{it}).
2. The route network connecting the stands could be predefined, with the period of accessibility explicitly specified. This would eliminate the need to determine both the connectivity and the year of accessibility.
3. The stands could be pre-assigned to invariant blocks. This would eliminate the need to group stands into cutblocks during the optimization process.
4. The utility function for contributing to the environmental constraints could be modified to be binary. This would simplify the computation used to evaluate the weights (although the role of the constraints in the optimization would remain in the same form).

Spatially Explicit Simulation

Overview

The modeling and optimization to be performed by MoF will require a realistic spatially explicit simulation to be developed to simulate the progression of selecting and cutting individual stands within the FMU while respecting various types of constraints. In this section, BCS outlines a basic approach to the simulation.

The simulation will proceed sequentially through a series of time periods. The basic unit for harvest scheduling will be the cutblock, but within each cutblock, the constraint checking, cutting, and growth modeling of individual atoms will be handled separately. These atoms (i.e., the fundamental spatial units of the simulation) will be generated by MoF during a preprocessing stage, in which large stands are subdivided into atoms of appropriate sizes, and stands are also subdivided to reflect riparian reserve/management areas, and areas associated with other types of environmental constraints. The outcome of this process will be a set of resultant polygons, each of which is an atom. A single stand may be divided into one or more atoms.

The MoF preprocessing and data formulation stage will also produce the following elements required for the simulation:

- the full set of atoms and other spatial units, with initial values of the attributes specified for each unit (this includes a delineation of the cutblocks and the atoms contained in each cutblock);
- applicable constraints and their bounds, and which atoms are associated with each constraint;
- growth and yield tables;
- cutting types and preferences;
- a directed road network containing all present and future roads, in which the future roads segments connect all the potentially accessible cutblocks;
- an algorithm for estimating the costs of building the road segments, based on the physical characteristics of the terrain.

Briefly, the process of the simulation is as follows. Following the initialization of the units and their attributes, the simulation proceeds to model the harvesting of atoms/cutblocks in each of the time periods. These units are queued based on some figure of merit (e.g., age or net value), and before any atom is cut, a check is made for whether cutting would violate a constraint. When an atom is cut, it is associated with a particular growth and yield curve and its age is reset to zero. Following cutting of each cutblock, adjacent cutblocks are prohibited from being cut until a green-up condition is met. Cutting proceeds more or less independently within the component units of the LU, based on harvest flow quotas for that unit, although provision is made for additional timber from one region being cut to make up for possible shortfalls in another. When the quota is reached, harvesting stops for that time period, the growth attributes of the atoms are updated, and new, previously unroaded, atoms are made accessible and added to the pool of potentially harvestable atoms. This process is repeated for each time period until the time horizon is reached.

The remainder of this section describes the objectives, spatial units, and general process proposed for the spatially explicit simulation. Additional, more detailed, information on other issues and aspects of the design, including detailed constraints and the attributes proposed for the various spatial units, is given in the appendices.

Simulation Objectives

The key objectives that will be satisfied by this design are as follows:

- ***Spatial hierarchy.*** The design provides for six hierarchical levels of spatial subdivision of the FMU, from the FMU itself to forest stands. The units at each spatial level are separate and completely contained within the next higher level (i.e., there is no overlap). The actual linework of each unit is not represented during the modeling, but rather the spatial information consists of the location of the unit and its relation to units in the same and other levels of the spatial hierarchy (i.e., its neighbors, parent and children). The units at each level of the hierarchy have a distinct set of attributes appropriate to that level.

- ***Spatial resolution.*** The design provides for the separate representation and modeling (harvesting and growth) of individual forest stands. Stands, or the resultant polygons created by subdividing stands during preprocessing, constitute the atomic spatial units of the modeling, and are termed “atoms”.
- ***Temporal progression and resolution.*** The simulation proceeds sequentially, with harvesting decisions made in one time period being influenced only by events in past periods; possible future events are not taken into consideration. The time period is typically one year. During each time period, the harvesting of the individual cutblocks and atoms also proceeds sequentially without considering the possible future implications of harvesting a particular unit.
- ***Compliance with harvest targets.*** The simulation harvesting scheme strives to reach specified FMU harvest targets (which may be parameterized in a simple manner, as above) during each time period. In this scheme, eligible atoms are examined and cut until the target is met or no further cutting can be done without violating constraints. It is possible to distribute the overall proportion of the harvest target among the units in the next level (i.e., the landscape units) and to have this distribution vary with the time period. It is also possible to allow for shortfalls in the target for one unit during a particular time period to be made up by additional cutting within other units during that time period, so that the overall target can be met.
- ***Compliance with environmental constraints in management zones.*** The decision on whether and/or how to cut an individual stand takes into account all applicable environmental constraints within management zones. These constraints are based on the area of the stand and the contribution of its structural composition (age or age-related, e.g., height) to the constraints. Separate, independent classes of constraints must be satisfied within management zones for biodiversity, visual area, community watershed, wildlife habitat, riparian, and similar, yet-to-be-defined types of zones.
- ***Compliance with green-up adjacency constraints.*** The simulation scheme provides for cutblocks that are adjacent to a recently harvested cutblock to be removed from consideration for harvesting for a period of time corresponding to the green-up of the harvested block.
- ***Ability to model reserves.*** It is possible to classify individual stands or blocks as belonging to a particular reserve category (e.g., wildlife or riparian). These stands are permanently off-limits to harvesting, but are separately modeled and taken into consideration when evaluating the constraints.
- ***Accessibility.*** The progression of harvesting in the simulation takes into account the accessibility of the cutblocks. Only those cutblocks that are accessible may be considered as candidates for harvesting (certain regions may be pre-designated as being permanently inaccessible by road). During each time period, the simulation design provides for the roading of previously unroaded cutblocks. The decision of which cutblocks to road is largely based on the projected cost and economic benefits of the roading. The design assumes that a directed road network is provided, in which all existing roads are represented, and all potentially accessible cutblocks have been connected to this existing road network and designated as “future roads”.
- ***Acceptable size distribution of cutblocks.*** The simulation uses predefined cutblocks that are assumed to have been generated with a size distribution that is compatible with the patch size distribution for the region. It also defines a separate blocking unit

level to facilitate the future development of an automated cutblock generation capability during simulation.

- **Consideration of economic factors.** The simulation takes key economic factors into account in the choices of which atoms to harvest and which roads to build. These include stand value, costs for road, site development, and harvesting, and the yearly budgets for each of the various cost categories.
- **High level harvesting control.** It is possible to encourage or discourage harvesting in certain sub-regions of the FMU, or to concentrate the activity in particular areas during certain time frames. This is done by providing the ability to specify the distribution of the overall harvest targets throughout the sub-regions of the FMU.
- **Low-level harvesting control.** The design provides for different options for determining the order in which the cutblocks are to be considered for harvesting. These include value, age (several variants) or height, current or recent harvesting activity in the immediate region, and environmental considerations.
- **Alternative harvesting regimes.** The simulation allows for several different types of harvesting regime to be applied. Options here include clear-cut, partial harvesting, single-species removal, and commercial thinning.
- **Effective silviculture and growth-and-yield modeling.** For each stand, the simulation models growth and yield in a realistic and effective way, and provides for prescription of and response to silvicultural treatments. This must include the ability to effect a transition from one growth curve to another when a silvicultural treatment is applied.
- **Natural disturbance modeling.** The simulation is capable of allowing the modeling of random natural disturbances such as wildfire, infestation and blowdown.
- **Efficient application to large problems.** The simulations are intended to be run for FMUs with up to 10,000,000 atoms, and for a simulated time of up to 500 years, with a time step (period) of one year. Since it is essential that these computer runs can be completed in a reasonable elapsed time (i.e., a matter of minutes), the detailed simulation design will have efficiency as a key consideration. The design will also identify independent processes at various levels that could be run in parallel.
- **Extensibility to optimization.** The simulation, though able to be run as a stand-alone system, is also a component of an optimization system. The overall design of the integrated optimization/simulation system allows the parameters to be passed from the optimization procedure to a function that runs the simulation, and to return a value for an objective function. The simulation provides for flexibility in the choice of terms and weights for this objective function.

Spatial Hierarchy

Six² levels of areal unit subdivision are represented in the modeling, in a strict spatial hierarchy. That is, each unit at one level completely contains an integral number of units in the next lower level, units at the same level may not overlap, and no unit can be contained by more than one unit at the level above.

² The number of levels is currently six, but this could be made more flexible to allow for greater control of the modeling.

These spatial units are as follows, from top to bottom of the hierarchy:

1. **Forest Management Unit (FMU).** This is the highest-level unit of the hierarchy. There are 71 FMUs (37 Timber Supply Areas and 34 Tree Farm Licenses) in the Province, and they range in size from about 15,000 ha to about 15,000,000 ha.
2. **Landscape Unit (LU).** This is a planning area based on topographic or geographic features that is appropriately sized (up to 100,000 ha) and designed for application of landscape-level biodiversity objectives. Landscape units completely span the FMU.
3. **Development Unit (DU).** This is an economic unit defined for the purposes of the current modeling. Each development unit will have one or more destination points assigned, to which the timber cut in that unit is to be transported, and for which hauling costs can be modeled. Development units completely span the landscape unit space.
4. **Blocking Unit (BU).** This is also a unit defined for the purposes of the current modeling. Each blocking unit defines a region within which the forest cover is relatively uniform and which can be subdivided into cutblocks solely on the basis of geometry, possibly by an automated process. Blocking units are only needed where harvestable stands/atoms have not yet been combined into cutblocks, and hence need not span the development unit space. However, an atom cannot be harvested until it is contained in a cutblock, so blocking units should be provided for all regions of a development unit that contain harvestable atoms if automated blocking is to be used.
5. **Cutblock.** Cutblocks are contiguous groups of stands/atoms that are candidates for harvesting, and they constitute the control unit for harvesting decisions. They may be predefined or generated dynamically during the simulation. Atoms that are not part of a cutblock cannot be harvested until they have been blocked. Since there may be regions of the blocking unit that are not contained in a cutblock, the cutblocks do not necessarily span the blocking unit.
6. **Atom.** Atoms are the most basic units of the simulations. They correspond to stands or resultant polygons, i.e., subdivisions of stands generated during preprocessing to accommodate cutblock and other types of boundaries (e.g., management zones and reserves – see below) and/or to generate harvest units of suitable size. The terms “stand” and “atom” are used synonymously in this report. In many cases, atoms will correspond to the inventory forest cover polygons. Atoms may be potentially harvestable or may be excluded from harvesting (e.g., part of a reserve). In order to be considered for harvesting, an atom must have been included in a cutblock, but it is not necessary that all atoms be included in cutblocks.

With the exception of cutblocks, the spatial extents of the units do not change during the modeling.

The attributes proposed for each spatial level are described in detail in the section *Appendix 4: Attributes of Spatial Units*.

Management Zones and Reserves

Management Zones (MZs) are regions within which one or more categories of management objectives (i.e., environmental constraints) must be met. These include MZs for biodiversity, visual quality, wildlife habitat, community watersheds, and riparian management. Reserves are regions within which no harvesting activity is permitted. Examples include parks, Federal and privately owned land, and riparian and wildlife habitat reserves.

These regions are independent of the spatial hierarchy described in the above section, and may overlap a number of spatial units of any level. In addition, an atom may be included in several different MZs and/or reserves. It may also be the case that some of the zones/MZs are not contiguous, so that a single MZ could consist of several separate units.

For modeling these reserves and MZs, each is represented as a separate entity, with description of the nature of the zone/reserve, and a list of the atoms that are included in it; MZs also have information about the constraints. Conversely, each atom has a list of any MZs and reserves of which it is a part.

The attributes proposed for the management zones and reserves are described in detail in the *Appendix 5: Management Zone and Reserve Attributes*.

Simulation Process

The following pseudocode summarizes the overall process proposed for the simulation of harvesting within an FMU.

Generate a target harvest flow for all periods.

Generate a distribution of this harvest flow among the LUs for all periods.

For each time period:

While harvest target is not met and spare harvest capacity exists in some LUs:

Define those LUs to harvest and increase targets as required.

For each LU to be harvested:

While this harvest quota is not filled and harvest capacity still exists:

Obtain the next cutblock from the active list.

For each atom in the cutblock:

For each cutting type:

Evaluate all constraints in which that atom is involved.

If no constraints violated:

Cut by cutting type and update atom attributes.

Update constraint sums.

Next atom (break to End for(1)).

End if

End for

End for (1)

Update cutblock attributes and transfer cutblock to wait queue.

Update adjacent cutblock attributes and transfer to wait queue.

End while

End for

End while

Update growth characteristics and composite values for active and unroaded queues.

Model natural disturbances and modify the attributes of affected cutblocks and atoms (all queues).

Import released cutblocks from wait queue.

Import cutblocks from unroaded queue, updating and sorting this queue as each cutblock is transferred.

Sort active queue.

Update environmental constraint sums.

End for

Simulation Structures

Spatial Units

Each spatial unit in each of the six lower levels of the spatial hierarchy will be represented by a structure. The attributes of the unit (which are described in *Appendix 4 : Attributes of Spatial Units*) will be stored in this structure, and the various units within a level will be represented by an array of structures. We note here that the geometry (i.e.,

boundary linework) of the above units will not be stored, but key spatial information (e.g., location, area, adjacent units) will be maintained as attributes in the structure.

In a similar way, each MZ and reserve will be represented by a structure. The attributes of these entities are described in *Appendix 5: Management Zone and Reserve Attributes*.

Queues

There are three queues containing cutblocks in the simulation. These are:

- the active queue
- the wait queue
- the unroaded queue.

All three queues are populated during initialization and are then sorted after this initial population. They are now described in more detail.

Active Queue

- **Contents.** This queue contains all currently harvestable cutblocks that are accessible (i.e., not on the unroaded queue) and not subject to time restrictions (i.e., not on the wait queue). This includes cutblocks that are not immediately harvestable due to potential violation of environmental constraints. It also includes cutblocks that are accessible only from the air (the costs of this are reflected in the net value).
- **Ordering.** The queue is sorted by one or more of the following properties:
 - absolute age
 - relative age
 - volume
 - volume/ha
 - value
 - value/ha
 - proximity to recent activity (promote or discourage).
- **Transfer and Update.** As each cutblock is harvested, it is transferred to the wait queue, with the growth characteristics and composite value being set to apply to the date of release from that queue. After each cutblock is harvested, any adjacent cutblocks in the active or unroaded queues are also identified. Those in the active queue are transferred to the wait queue for a specified green-up period, and the growth characteristics and composite values are updated to correspond to that release date. Those in the unroaded queue are kept in that queue and are treated as described below. Those already in the wait queue have their growth characteristics and composite values updated, if necessary, to correspond to the later release date. At the end of each period, following the import of cutblocks from the unroaded queue and the wait queue, the active queue is sorted.

Wait Queue

- **Contents:** This queue contains all cutblocks that cannot be harvested until a particular future date is reached.
- **Ordering:** The queue is sorted by release date.
- **Transfer and Update.** During the harvesting, the harvested or adjacent cutblocks are inserted into an appropriate position in the wait queue. At the end of each period, all cutblocks whose release dates correspond to the next period are then transferred to the active or unroaded queues.

Unroaded Queue

- **Contents.** This queue contains all potentially harvestable cutblocks which do not yet have road access.
- **Ordering:** The queue is sorted by value and/or value/ha.
- **Transfer and Update.** The growth and composite value for all cutblocks in the unroaded queue are updated at the end of each period of the simulation. Then, based on the annual budget available for road construction, cutblocks are transferred from the unroaded queue to the active queue until the budget is exhausted. After transfer of each cutblock to the active queue, any affected cutblocks are updated and the unroaded queue is also re-sorted after each such update. This allows successive roading of several cutblocks within a region during the same period, if this is economically justified. If a cutblock in the unroaded queue is identified as being adjacent to a cutblock that has just been harvested (see above), the former is flagged as being unavailable for harvest until the release date, and its growth characteristics and composite value are set to correspond to that date. It is not transferred to the wait queue, but remains on the unroaded queue. If it is roaded before the release date, it is then transferred to the wait queue.

Speed and Efficiency Considerations

It is expected that the most time-consuming processes in the simulation (outside of the initial data loading) would be the following:

- **Updating growth and sorting queues.** In order to keep the order of the items in the active and unroaded queues current, the volumes/values of the timber in each stand should be updated at the end of each time period, and the queues reordered. This is also necessary to ensure items from the wait and unroaded queues are inserted into the correct positions in the active queue as they are brought into that queue. A possible method for speeding up this stage would be to only update the volumes/values every several time periods (e.g., every five years). A related approach would be to only update the top portion of the queue during each time period, and update the remainder every several periods.
- **Updating the unroaded queue.** In order to ensure that the effects of just-roaded blocks are propagated to adjacent blocks, the unroaded queue should be updated and

sorted after each individual block is roaded and placed on the active queue. A possible method for speeding up this stage would be to only update and sort the unroaded queue after all the blocks to be roaded are brought in for that period.

- ***Evaluating constraints.*** As the simulation proceeds, it may be necessary to examine and skip over a large number of the entries in the active queue because cutting them would violate a constraint. Hence, there could be many fruitless tries before a harvestable cutblock is found. It would be desirable to identify those atoms/blocks for which cutting would violate a constraint and shift them down in the queue. The times required to perform the computation involved in evaluating the constraints could also be reduced by using binary utility functions (see Appendix 1).

Opportunities for Parallelism

There are several processing levels at which parallel processing could be employed to increase the speed of the overall simulation. These are:

- ***Parallel landscape unit harvesting.*** The harvest quotas can be apportioned to each landscape unit independently for the purposes of the simulation, so the simulation can proceed in parallel for the various landscape units during each time period. The exception to this is a provision for there to be more than one pass through a particular landscape unit during a single time period, if the capacity of some other landscape units in the forest management unit is not sufficient to meet the overall harvest flow constraint.
- ***Parallel constraint evaluation.*** Since the constraints are independent, each could be evaluated for an atom in a separate parallel process. If one of the processes found that the constraint would be violated, it could report this and stop the evaluation of the others.
- ***Parallel updating of growth characteristics.*** Growth within each atom is independent and so this updating at the end of each time period could be done in parallel.
- ***Parallel updating of constraint sums.*** The constraints are independent, and so their re-evaluation at the end of each time period could be done in parallel.

In addition to these areas, opportunities for lower-level parallelism may be present. These could exploit the capabilities of parallel hardware architectures to perform the computations in a distributed manner. An effective approach here would be to implement the design for the simulation in prototype form and to identify the bottlenecks in the processing. After an initial stage of algorithm revision and refinement (or simplification) to address these bottlenecks where possible, the application of particular hardware to improve the speed or parallelism of these operations would be assessed.

Data I/O, Storage, and Analysis

The input data for the simulations will normally be in the form of files containing lists of spatial units and attributes. A user interface will also be required to provide an environment for specifying the detailed conditions for the simulation and for selecting the input data. It is proposed to use an object-relational database to store the initial run conditions and data files and the intermediate and final results of the simulation. This would allow the user to conveniently access and query these results when the simulation is complete, and would avoid the production of large report files during the simulation itself. The information that would be written to this database would include:

- the names of input data files involved
- the harvest flow targets for each period and LU
- the detailed harvest schedule – i.e., which cutblocks and atoms were cut during each period for each LU
- the constraint status for each period and LU.

A facility should also be provided to download the results from the database and convert them into a format where they can be visualized and analyzed. This could include the generation of comma-separated lists for import to Access, and the generation of shapefiles for input to a GIS.

Ideally the object-relational database would also be used to store the geographic objects (as points, lines, polygons, and coverages) from which the input files are generated. This would allow the user to better visualize and control how the atoms are created. It would also provide context for visualizing the simulation / optimization results

Considerations for Optimization

In the first part of this report, it was proposed that the problem of defining a harvest flow consistent with constraints could be effectively addressed by optimization of the parameters of a simple functional form for the harvest flow targets. Referring to the notation in the *Harvest Flow Optimization* section above and the *Optimization Approach* section below, some candidate functional forms and parameterizations could be as in the following table:

Fixed Parameters	Optimizable Parameters	Functional Form
None	v1	Constant level v1.
v1	t2, v2	Immediate slope from fixed level v1 to level v2 at time t2.
None	t2, v1, v2	Immediate slope from level v1 to level v2 at time t2.
v1	t1, t2, v2	Fixed level v1 until time t1, then sloping to level v2 at time t2.
None	t1, t2, v1, v2	Constant level v1 until time t1, then sloping to level v2 at time t2.
v1	t2, t3, t4, v2, v3	Immediate slope from fixed level v1 to level v2 at time t2, constant until time t3, then sloping to level v3 at time t4, and constant thereafter.
None	t2, t3, t4, t1, v2, v3	Immediate slope from level v1 to level v2 at time t2, constant until time t3, then sloping to level v3 at time t4, and constant thereafter.
v1	t1, t2, t3, t4, v2, v3	Fixed level v1 until time t1, then sloping to level v2 at time t2, the constant until time t3, then sloping to level v3 at time t4, and constant thereafter.
None	t1, t2, t3, t4, v1, v2, v3	Constant level v1 until time t1, then sloping to level v2 at time t2, the constant until time t3, then sloping to level v3 at time t4, and constant thereafter.

Each set of values for any of the above parameterizations defines a particular target harvest flow function for the simulation. The simulation process can be called by an optimization routine, with the target function for the harvest flow as defined by the parameters. During the simulation, harvesting within each region continues until the target is reached, and any cases where the target cannot be reached are stored. The actual amounts cut (volumes and/or value), along with other results of the simulation, can be used to evaluate an objective function with positive terms (e.g., total value of timber cut) and negative (penalty) terms (e.g., shortfalls with respect to the target). The optimization routine uses the values of the objective functions returned by the simulation in its computations to determine an optimal solution.

Other considerations could also be included in the objective function. For example, deviations from certain management objectives could be included as additional penalty terms. Alternatively, the simulation could be designed to proceed in a different way, perhaps by allowing the violation of constraints to force achievement of a certain target harvest flow and including the violations as penalty function terms. The objective function could also be altered in form to have quadratic rather than linear terms for the penalty function.

Optimization: Proof of Concept

In the proposed approach to optimization described in this section, the harvest flow function forms the basis for both a simple parameter space and an objective function. The harvest flow function is parameterized by a small number of variables (say, 2 – 6), with each set of parameters defining a particular target harvest flow function for the volumes to be cut. For each set of parameters, a simulation is run that attempts to reach (but exceed by as little as possible) the target harvest flow levels for each period while respecting all the constraints. The simulation also computes an objective function that reflects the overall volume cut, and is penalized for any shortfalls. An optimization process then uses a sequence of the simulations to determine the optimal values for the parameters that maximize the objective function. This new approach provides a mechanism for realizing the ultimate aim of defining a maximal target harvest flow that can be satisfied without violating the constraints.

In this section, the simulation and optimization procedures that were set up to test the feasibility of this new approach are outlined, and the results of these procedures applied to a synthetic test problem are presented. Our view is that these results provide a proof of concept for the proposed approach.

Simulation Procedure

Overview

A simulation procedure for spatially explicit harvest modeling was written to implement the harvest flow parameterization and provide the capability for computing objective function values. This procedure was written in IDL (Interactive Data Language, Research Systems, Inc. – see www.rsinc.com). This simulation procedure was a simplified version of a more realistic model, in that roading issues were ignored, and the basic spatial units were taken to be cutblocks (which were assumed to be predefined).

The following pseudocode summarizes the overall harvest simulation process used in these studies.

Generate harvest quota for all periods, given the parameters.

For each time period:

While harvest quota for this period not filled and there are more cutblocks to consider:

For the next cutblock in the active list.

Evaluate all constraints in which that cutblock is involved.

If no constraints are violated:

Proceed with cutting: i.e.:

- Update constraint sums.**
- Update cutblock attributes and transfer cutblock to wait queue.**
- Update adjacent cutblock attributes and transfer to wait queue.**

End if

End for

End while

Increment time period and update growth characteristics for all cutblocks.

Import cutblocks released from wait queue.

Sort active queue in decreasing order of volume.

Update constraint sums.

End for

Compute objective function based on the target quota and the actual volumes cut.

Implementation

The strategy used to set up the initial conditions involved generating synthetic cutblocks, populating their volumes on the basis of area and age, setting up a simple function to model growth, and establishing a mechanism for specifying and evaluating the constraints.

Synthetic cutblocks were created as follows. The cutblock centers were generated as realizations of a uniform distribution within a square region. These points were triangulated and the Voronoi diagram was computed. The Voronoi polygons were taken to be the cutblocks. Using these polygons and the triangulation, the area and a list of adjacent cutblocks was obtained for each cutblock.

The ages of the cutblocks were assigned based on a uniform distribution, and the volume of the timber in each cutblock was calculated based on the area (assuming direct proportionality) and the age of the cutblock. For computing the age, the relative timber volume of the cutblocks as a function of time was modeled as a sigmoidal function with the following form:

$$g(t) = \begin{cases} \frac{1}{2}(1 - \cos(\pi \frac{t}{T})), & 0 \leq t \leq T \\ 1, & t > T \end{cases},$$

where $g(t)$ is the relative timber volume ($0 \leq g(t) \leq 1$), t is the age of the stand and T is the time to full maturity.

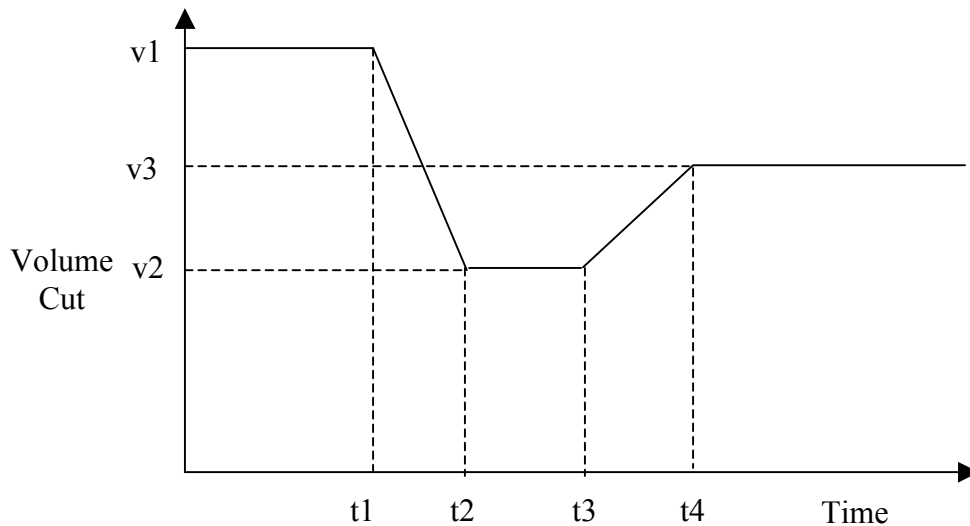
Constraints were implemented in upper bound form, i.e., as an upper limit on the maximum amount of area within the region affected by the constraint that could be less than a certain age. These constraints are of the form

$$\frac{\sum_j w_j A_j(s)}{\sum_j A_j(s)} \leq U(s)$$

where $A_j(s)$ is the area of the j th cutblock for the region s in which the constraint applies, and $w_j = 1$ if the age of the j th cutblock is less than the age limit specified by the constraint, and $w_j = 0$ otherwise.

Optimization Approach

Two parameterizations of the harvest flow function were examined for optimization. The first was the simple three-parameter model (t_1 , t_2 , v_2) given in the *Harvest Flow Optimization* section above. The second was an extension of this functional form to include three constant and two sloping regions, as in the following diagram.



In both cases, the objective function was of the form

$$\sum_{i=1}^N \min(T_i, C_i) - P \sum_{i \in S} \max((T_i - C_i), 0),$$

where N is the number of periods, T_i is the target harvest volume for the i th period, C_i is the volume actually cut for the i th period, P is a penalty factor for shortfalls, and S is the set of periods for which shortfalls were observed during the simulation.

The Nelder-Mead simplex optimization method was used to perform the optimization calculations. This method requires only function values (derivatives are not available in the current situation), and is part of the suite of mathematical routines provided in IDL.

Results of Simulation/Optimization Tests

The general conditions used for the simulations are given in Table 1.

Table 1. General conditions used in the simulations.

Quantity	Value
Number of periods	200
Wait time following cut (periods)	50
Wait time for green-up (periods)	20
Age for maximum volume	100
Maximum volume/ha	100
Number of cutblocks	5,000
Region size (ha)	10000
Maximum age for initial state	300
Value for penalty factor P	10

Three constraints were implemented, with the values as listed in Table 2.

Table 2. Values used for the constraints in the simulations.

Number of Blocks Involved in Constraint	Cutoff Age (periods)	Maximum Percent Area
5000	80	80%
5000	30	50%
845	20	10%

For example, the first constraint requires that at most 80% of the area be occupied by cutblocks with an age less than 80 (old growth constraint). The third constraint requires that at most 10% of the area be occupied by cutblocks with an age less than 20 periods (young growth constraint).

Table 3. Results of six optimizations of harvest flow parameters.

Optimization Number	Conditions	Number of Function Evaluations	Initial Parameter Values	Final Parameter Values	Final Objective Function
1	<ul style="list-style-type: none"> • 3 parameters • initial solution resulted in shortfalls • no scaling • $t1 < 0$ permitted 	Not recorded	$t1=20$ $t2=50$ $(v1=9000)$ $v2=8000$	$t1=21.71$ $t2=50.16$ $v2=7999.3$	1,341,973
2	<ul style="list-style-type: none"> • 3 parameters • initial solution did not result in shortfalls • no scaling • $t1 < 0$ permitted 	265	$t1=5$ $t2=20$ $(v1=9000)$ $v2=1000$	$t1=85.00$ $t2=85.00$ $v2=959.3$	762,801
3	<ul style="list-style-type: none"> • 3 parameters • initial solution did not result in shortfalls • scaling • $t1 < 0$ permitted 	174	$t1=5$ $t2=20$ $(v1=9000)$ $v2=1000$	$t1=-25.31$ $t2=28.48$ $v2=7742.4$	1,558,884
4	<ul style="list-style-type: none"> • 3 parameters • initial solution did not result in shortfalls • scaling • $t1 < 0$ not permitted 	188	$t1=5$ $t2=20$ $(v1=9000)$ $v2=1000$	$t1=0$ $t2=41.60$ $v2=7751.5$	1,576,900
5	<ul style="list-style-type: none"> • 3 parameters • initial solution did not result in shortfalls • scaling • $t1 < 0$ not permitted 	230	$t1=10$ $t2=50$ $(v1=9000)$ $v2=3000$	$t1=0$ $t2=56.60$ $v2=7766.6$	1,578,870
6	<ul style="list-style-type: none"> • 6 parameters • initial solution did not result in shortfalls • scaling • $t1 < 0$ not permitted 	402	$t1=5$ $t2=20$ $t3=35$ $t4=50$ $(v1=9000)$ $v2=1000$ $v3=2000$	$t1=0.74$ $t2=17.35$ $t3=23.99$ $t4=28.85$ $v2=3574.2$ $v3=7900.6$	1,515,392

Six optimizations were performed, the first five with the three-parameter model and the sixth with the six-parameter model. In all cases, the (fixed) value for $v1$ was 9000. For optimization 1, the scaling option of the IDL Nelder-Mead algorithm was not used, while for optimizations 2-6, the scaling factors were set to 10 for the parameters $t1$, $t2$, $t3$, and $t4$, and to 1000 for the parameters $v2$ and $v3$. The results of these optimization runs are shown in Table 3.

The harvest flow results for three of these optimizations are also shown in Figures 1 – 3. In these figures the target function is plotted as a solid line, and the actual volumes cut in each time period are displayed in two ways: as single points when the volume cut during the period reached the target, and as boxes when there was a shortfall.

Figure 1 shows the harvest flow results for optimization number 1 (using the three-parameter model), where the initial solution resulted in shortfalls and scaling was not used. The optimized result in this case also gave rise to harvest flow shortfalls (i.e., many of the actual volumes cut were less than the target) and this final solution was almost the same as the initial solution.

Figure 2 shows the harvest flow result for optimization number 5 (also using the three-parameter model), where the initial solution did not result in shortfalls and scaling was used. The optimized result did not give rise to shortfalls, and was very different from the initial solution, immediately starting with a downward slope. This optimized harvest flow, and the final parameters and objective function value, were all close to those of optimization number 4, where the conditions were the same except for the initial solution.

Figure 3 shows the harvest flow results for optimization number 6 (using the six-parameter model), with an initial solution that did not result in shortfalls and using scaling. The optimized result did not give rise to shortfalls, and reached a final plateau slightly higher than that of the three-parameter results, but displayed a large initial dip.

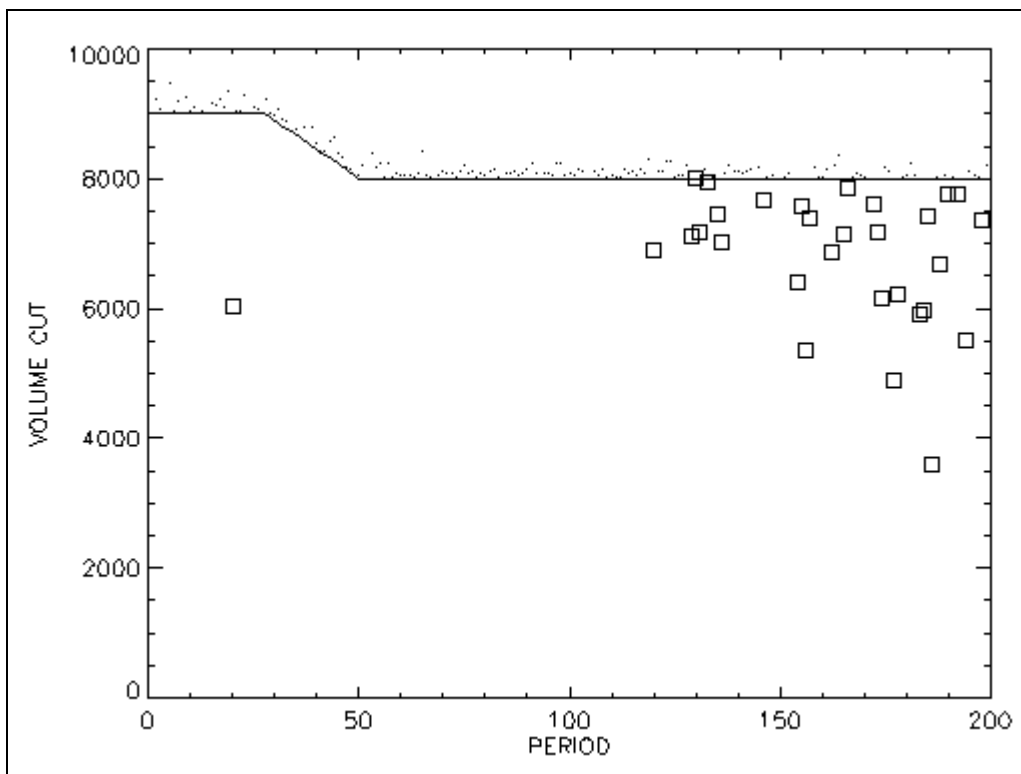


Figure 1. Target and actual harvest profiles for optimization number 1.

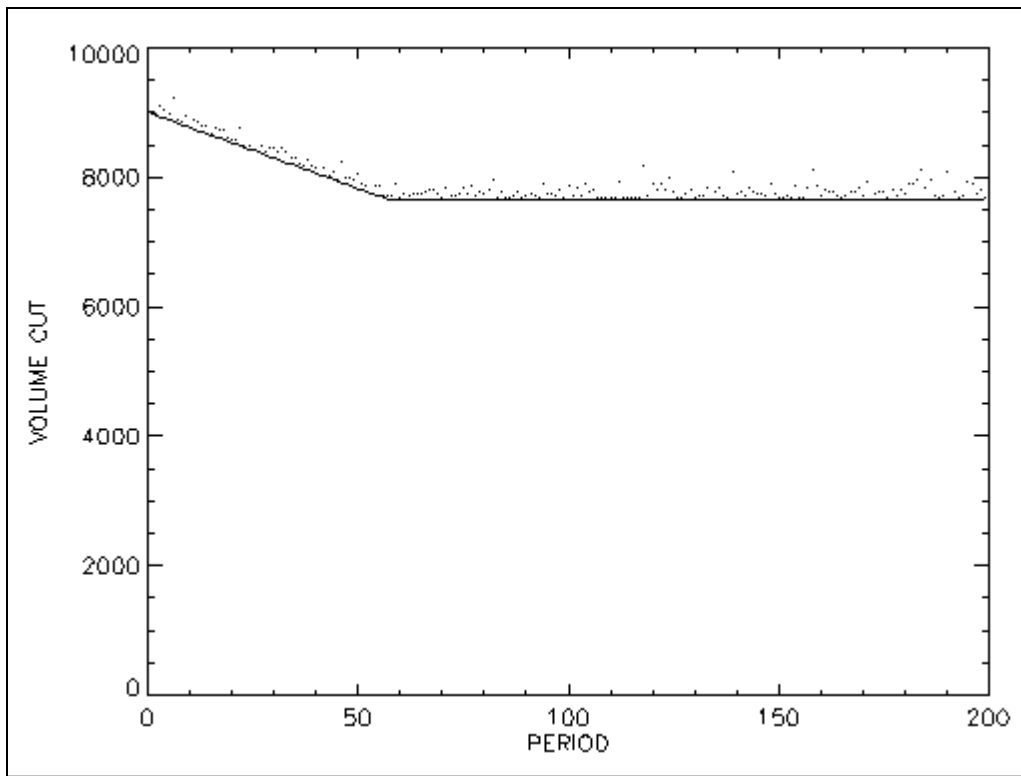


Figure 2. Target and actual harvest profiles for optimization number 5.

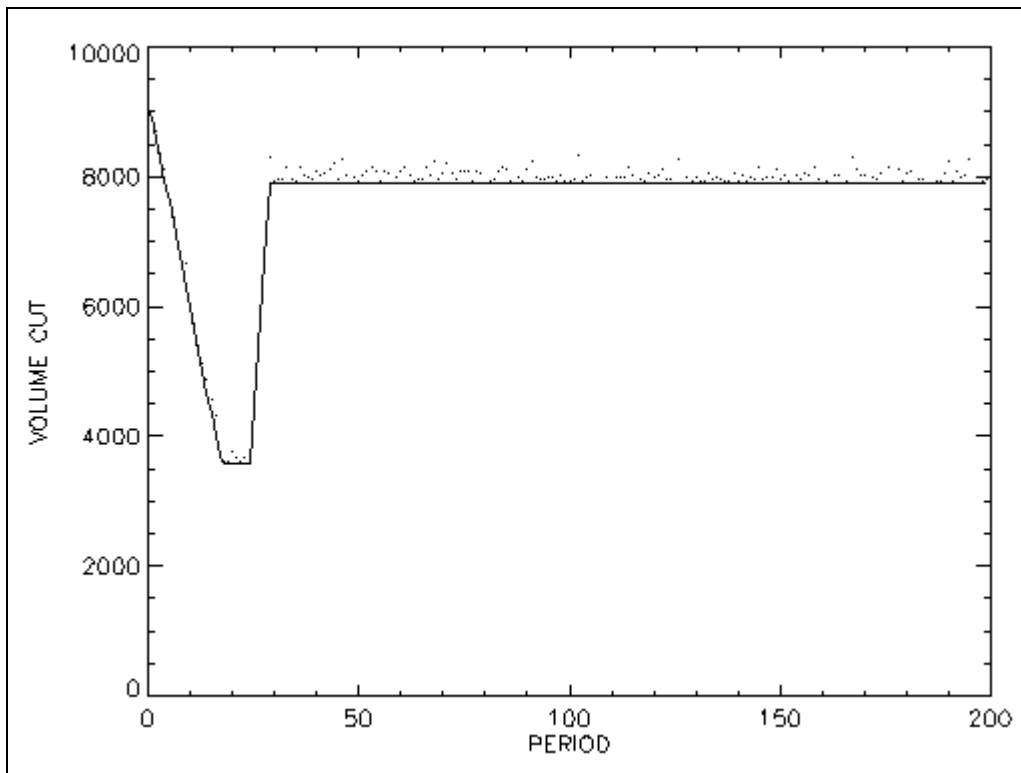


Figure 3. Target and actual harvest profiles for optimization number 6.

Discussion

These initial results are promising and indicate that harvest flows can indeed be automatically defined that conform to the desired shape and are consistent with the constraints. The Nelder-Mead simplex method (which operates with function values and does not require derivatives) is also confirmed as a suitable choice for the optimization procedure in this application.

For successful results, it is necessary to start with a solution that does not give rise to shortfalls in the initial constant region of the target harvest flow. For example, if a shortfall occurs before time t_1 , then changing t_1 by small amounts will not eliminate the effect of this shortfall on the objective function, and so it is unlikely that the overall optimum will be found. In practice, an initial solution that does not result in shortfalls will be easy to define.

It is also necessary to scale the parameters appropriately. If the scaling is not suitable, there is a risk that the algorithm cannot escape from a sub-optimal solution near the initial value, particularly if the initial solution gives rise to shortfalls (e.g., optimization 1).

Increasing the complexity of the function (and the number of parameters) does not necessarily result in an improved optimization. This is apparent in the results from optimization 6, where the three-level, six-parameter target function was used. The objective function for this case was actually greater than that for the three-parameter model (optimizations 4 and 5). This failure to identify a better solution could result from the presence of multiple local optima due to the discrete nature of the problem, and may also indicate the presence of ill-conditioning (where widely different solutions can yield nearly the same values for the objective function). In practice, it will likely be desirable to keep the functional form simple.

It is also noted that the choice of the exact form chosen for the objective function and the penalty factor P will affect the solution path and final solution obtained, even for identical initial conditions for the optimization.

One of the interesting aspects of this problem is that the Nelder-Mead algorithm can be applied even though the function is quantized, i.e., the optimization involves binary decisions on discrete units (cutblocks) of different sizes. The objective function will be discontinuous whenever a slight adjustment in one of the parameters results in a different choice of cutblocks for the harvest schedule. Despite this, the Nelder-Mead optimization method gives closely similar (though likely sub-optimal) results when starting with different initial solutions (e.g., optimizations 4 and 5). Hence, it is not necessary to use computationally more intensive methods such as simulated annealing or genetic algorithms to achieve an acceptable near-optimal harvest schedule. The low number of function evaluations for convergence (about 200, stabilizing after 100) is also encouraging, given the long times that will be required for function evaluations in any realistic simulation.

In summary, the results of these preliminary studies suggest that the method proposed in this report is likely to be of considerable value in optimizing forest harvest flow targets in the presence of environmental constraints.

Bibliography

1. BC Ministry of Forests (1995) Biodiversity guidebook (Forest Practices Code of BC).
2. BC Ministry of Forests (1995) Visual impact assessment guidebook (Forest Practices Code of BC).
3. BC Ministry of Forests (1998) Procedures for factoring visual resources into timber supply analyses.
4. BC Ministry of Forests (2001) Timber Supply Review: Merritt Timber Supply Area Analysis Report (Timber Supply Branch).
5. BC Ministry of Forests (2001) Timber Supply Review: Kootenay Lake Timber Supply Area Analysis Report (Timber Supply Branch).
6. Clark, M. M., Meller, R. D., and McDonald, T. P. (2000) A three-stage heuristic for harvest scheduling with access road network development. *Forest Science* **46**(2), 204-218.
7. Clark, M. M., Meller, R. D., McDonald, T. P., and Chao, C. T. (1997) A new harvest operation cost model to evaluate forest harvest layout alternatives. In "Forest operations for sustainable forests and healthy economies"; Proceedings, Council on Forest Engineering, 20th annual meeting; 1997 July 28-31; Rapid City, SD, pp. 42-47.
8. Clements, S. E., Dallain, P. L., and Jamnick, M. S. (1990) An operational, spatially constrained harvest scheduling model. *Can. For. Res.* **20**, 1438-1447.
9. Dean, D. J. (1997) Finding optimal routes for networks of harvest site access roads using GIS-based techniques. *Can. J. For. Res.* **27**, 11-22.
10. Hoganson, H. M. and Borges, J. G. (1998) Using dynamic programming and overlapping subproblems to address adjacency in large harvest scheduling problems. *For. Sci.* **44**, 526-538.
11. Howard, A. F. and Nelson, J. D. (1993) Area-based harvest scheduling and allocation of forest land using methods for multiple-criteria decision making. *Can. J. For. Res.* **23**, 151-158.
12. Kurtilla, M. (2001) The spatial structure of forests in the optimization calculations of forest planning – A landscape ecological perspective. *For. Ecol. Man.* **142**, 129-142.
13. Liu, G., Nelson, J. D., and Wardman, C. W. (2000) A target-oriented approach to forest ecosystem design – changing the rules of forest planning. *Eco. Model.* **127**, 269-281.
14. Liu, K and Sessions, J. (1993) Preliminary planning of road systems using digital terrain models. *J. For. Eng.* **4**, 27-32.
15. Lockwood, C and Moore, T. (1992) Harvest scheduling with spatial constraints: A simulated annealing approach. *Can J. For Res.* **23**, 468-478.
16. Murray, A. T. (1998) Route planning for harvest site access. *Can. J. For. Res.* **28**, 1084-1087.

17. Murray, A. T. (1999) Spatial restrictions in harvest scheduling. *For. Sci.* **45**, 45-52.
18. Murray, A. T. and Church, R. (1995) Heuristic solution approaches to operational forest planning problems. *OR Spektrum* **17**, 193-203.
19. 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.
20. Nelson, J. D. and Errico, D. (1993) Multiple-pass harvesting and spatial constraints: An old technique applied to a new problem. *For. Sci.* **39**, 137-151.
21. 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.
22. Nelson, J. and Liu, G. (1994) Scheduling cutblocks with simulated annealing. *Proceedings of the International Seminar on Forest Operations under Mountainous Conditions*. (John Sessions, ed.) Harbin, PR of China.
23. O'Hara, A. J., Faaland, B. H. and Bare, B. B. (1989) Spatially constrained timber harvest scheduling. *Can. J. For. Res.* **19**, 715-724.
24. Reed, W. J. and Errico, D. (1989) A new look at whole-forest modeling. *Nat. Res. Model.* **3**, 399-427.
25. Sessions, J. (1987) A heuristic algorithm for the solution of the variable and fixed cost transportation problem. In *Proceedings, The 1985 Symposium on Systems Analysis in Forest Resources*, University of Georgia, Athens GA, pp. 324-336.
26. Snyder, S. and ReVelle, C. (1995) Dynamic selection of harvests with adjacency restrictions: The SHARe model. *For. Sci.* **43**, 213-222.
27. Snyder, S. and ReVelle, C. (1996) Temporal and spatial harvesting of irregular systems of parcels. *Can. J. For. Res.* **26**, 1079-1088.

Appendix 1: Environmental Constraints

Biodiversity Constraints

Biodiversity is defined as the overall diversity of plants, animals and other living organisms in all their forms and levels of organization, and includes the diversity of genes, species and ecosystems as well as the evolutionary and functional processes that link them.³ In biodiversity management, the central principle is that biodiversity is most likely to be achieved when anthropogenic disturbances mimic natural disturbances as closely as possible. In practical terms, this translates into the objective of maintaining a distribution of seral stages that is similar to that which existed before human intervention.

Formulation of Constraints

Separate sets of constraints, which are based on age, may be needed, depending on whether the constraints are to be applied to the total land base or the timber harvest land

³ Biodiversity Guidebook (Forest Practices Code of British Columbia), p. 2.

base, or both. For each land base b , landscape unit l , natural disturbance type n , biogeoclimatic ecosystem classification c , variant v , and biodiversity emphasis option o , there are generally three constraints, as follows:

Early seral constraint:

$$\frac{100 \times \sum_j w_j^E(b, l, n, c, v, o) A_j(b, l, n, c, v, o)}{\sum_j A_j(b, l, n, c, v, o)} \leq U^E(b, l, n, c, v, o),$$

Mature+Old seral constraint:

$$\frac{100 \times [\sum_j w_j^M(b, l, n, c, v, o) A_j(b, l, n, c, v, o) + \sum_j w_j^O(b, l, n, c, v, o) A_j(b, l, n, c, v, o)]}{\sum_j A_j(b, l, n, c, v, o)} \geq L^{M+O}(b, l, n, c, v, o),$$

Old seral constraint:

$$\frac{100 \times \sum_j w_j^O(b, l, n, c, v, o) A_j(b, l, n, c, v, o)}{\sum_j A_j(b, l, n, c, v, o)} \geq L^O(b, l, n, c, v, o),$$

where

$A_j(b, l, n, c, v, o)$ is the area of the j th atom in the set of atoms in the category defined by land base b , landscape unit l , natural disturbance type n , biogeoclimatic ecosystem classification c , variant v , and biodiversity emphasis option o ;

$w_j^E(b, l, n, c, v, o)$, $w_j^M(b, l, n, c, v, o)$, and $w_j^O(b, l, n, c, v, o)$ are the fractional amounts (weights) of the j th atom in category (b, l, n, c, v, o) in the early (E), mature (M), and old (O) seral stage age ranges, respectively (see below for a description of the significance and computation of these weights, which are based on utility functions and distribution functions);

$U^E(b, l, n, c, v, o)$ is an upper bound on the percent area that may be occupied by forest in the early seral stage in category (b, l, n, c, v, o) as above;

$L^{M+O}(b, l, n, c, v, o)$ is a lower bound on the percent area that may be occupied by forest in either the mature or the old seral stage in category (b, l, n, c, v, o) as above;

$L^O(b, l, n, c, v, o)$ is a lower bound on the percent area that may be occupied by forest in the old seral stage in category (b, l, n, c, v, o) as above.

The following points should be noted with respect to the biodiversity constraints (and, in some cases, the other types of constraints considered below):

- The upper bound for the early seral stage is sometimes unconstrained (e.g., for NDT1 and BEO category “Low”).
- The finer the subdivision, the greater the likelihood that there may be only a very few atoms in a category, so sometimes it may not be possible to satisfy the constraints, even at the outset. A suitable rule for handling this situation (e.g., by placing some of the atoms in a reserve) must be defined.
- The impact of violating a hard constraint could be mitigated by softening the constraints (i.e., adopting continuous utility curves – see *Computing Contributions to Environmental Constraints* section below).

Associated Data

Each atom to which biodiversity constraints are to apply must be associated with each of the categories indicated in the following table:

Category	Source of Data
Land Base (i.e., timber harvest or total)	Depends on particular FMU.
Landscape Unit (LU)	MoF district non-standard overlay.
Natural Disturbance Type (NDT)	Derived from BEC and does not need to be mapped.
Biogeoclimatic Ecological Classification (BEC) and Variant	MoF Research Branch.
Biodiversity Emphasis Option (BEO)	Landscape unit planning (district level), but not always available.
Age Range for Seral Stage	Derived from BEC.

Each atom will be assigned its corresponding LU, NDT, BEC and variant, and BEO during a preprocessing stage. The age ranges for seral stages will be determined from the BEC during this preprocessing stage.

The tables containing values for the lower and upper bounds for the constraints are contained in the *Biodiversity Guidebook*.

Visual Quality Constraints

Visual quality objective (VQO) management starts with the definition of boundaries of discrete visual areas within an FMU. Each of these areas has associated with it a visual quality class (VQC, which is one of the following: preservation (P), retention (R), partial retention (PR), modification (M), and maximum modification (MM)). VQO constraints apply to each visual area on an individual, not a composite, basis.

Formulation of Constraints

It is assumed here that each visual area has been assigned a single VQC, and a single visual absorption capability (VAC) of low (L), medium (M) or high (H). The constraints for the area then limit the percent of the recently denuded area, i.e., the area that is covered with trees of height less than the green-up height. This green-up height is slope-dependent (increasing with slope). For the purposes of simulation modeling, the constraints are usually expressed in terms of the green-up age, which is obtained by using the appropriate age-height curves. The form of the constraints, which must apply to each separate visual area, is as follows:

$$\frac{100 \times \sum_j w_j^V(k) A_j(k)}{\sum_j A_j(k)} \leq U^V(k),$$

where

$A_j(k)$ is the area of the j th atom (all ages) in visual area k ;

$w_j^V(k)$ is the fractional amount (weight) of the j th atom in visual area k which has been denuded, and which is younger than the visual green-up age (see Appendix A for computation of these weights);

$U^V(k)$ is an upper bound on the percent denudation for visual area k ; this bound depends on the VQC and VAC specified for this visual area. Values for these bounds are given in Table 4 of the document *Procedures for Factoring Visual Resources into Timber Supply Analysis* (Province of B.C., Ministry of Forests, March 17, 1998).

It is noted here that selective cutting is treated as a partial cutting type. That is, rather than trying to satisfy separate constraints for selective cutting (e.g., for visual quality), selective cuts are simply modeled as an equivalent clear-cut of a portion of the stand.

Associated Data

Each atom to which visual quality constraints are to apply will be assigned the following attributes during preprocessing, from the sources of data indicated in the table below. The VQO and VAC are constant for all atoms in the visual unit, while the green-up age may vary with location within the unit.

Attribute	Source of Data
VQO	District overlays
VAC	District overlays
Green-up age	Age-height curves

The values for the upper bounds for the constraints are contained in Table 4 of the document *Procedures for Factoring Visual Resources into Timber Supply Analysis*.

Community Watershed Constraints

To preserve the quality of the water within community watersheds, restrictions on the extent of the disturbance within the watershed have been defined. The extent of permissible disturbances depends on the region, and the class of the community watershed.

Formulation of Constraints

Currently, hard constraints are specified for cutting within community watersheds. These take the form that a maximum specified percentage of the area within a community watershed of a certain watershed class may be less than the green-up height. In practice, this translates into age constraints, similar to those for visual areas. These constraints limit the percent of the area that is covered with forest younger than the green-up age, as follows:

$$\frac{100 \times \sum_j w_j^w(k) A_j(k)}{\sum_j A_j(k)} \leq U^w(k),$$

where

$A_j(k)$ is the area of the j th atom (all ages) in the k th community watershed;

$w_j^w(k)$ is the fractional amount (weight) of the j th atom in the k th community watershed which has been denuded and which is younger than the green-up age (see Appendix A for computation of these weights);

$U^w(k)$ is an upper bound on the percent area of the k th community watershed (in class C_k) that may be covered in forest younger than the green-up age.

Associated Data

Each atom in a community watershed will be attributed with the code of that watershed during preprocessing. This attribution will be based on the boundaries of each community watershed and its classification, which will be obtained from the Water Management Branch, (former) Ministry of Environment, Lands and Parks.

Wildlife Habitat Constraints

In order to ensure the existence of sufficient habitat for one or more species, there are restrictions on the extent of the disturbance within specified wildlife habitat regions. These can include both wildlife reserve areas for critical habitat, with a total restriction on cutting activity, and wildlife management areas, in which restrictions on the percent disturbance within the region apply. It is noted that these areas may overlap with other types of reserves and management areas, and hence may also combine with and contribute to satisfying other types of constraints (e.g., biodiversity, visual and community watershed).

Formulation of Constraints

Within reserves, constraints involving the total restriction on harvesting are satisfied by not allowing these atoms to be harvested.

Within habitat management zones, constraints are again based on age and area. They establish an upper bound on the percent area that may be less than a certain early age, and require that at least a certain percent area be above a specified older age. These constraints may be expressed as follows:

$$\frac{100 \times \sum_j w_j^E(k) A_j(k)}{\sum_j A_j(k)} \leq U^E(k),$$
$$\frac{100 \times \sum_j w_j^O(k) A_j(k)}{\sum_j A_j(k)} \geq L^O(k),$$

where

$A_j(k)$ is the area of the j th atom (all ages) in the k th habitat management zone;

$w_j^E(k)$ is the fractional amount (weight) of the j th atom in the k th habitat management zone which is less than the specified early age (see Appendix A for computation of these weights);

$w_j^O(k)$ is the fractional amount (weight) of the j th atom in the k th habitat management zone which is greater than the specified older age (see Appendix A for computation of these weights);

$U^E(k)$ is an upper bound on the percent area of the k th habitat management zone that may be covered in forest younger than the specified early age;

$L^O(k)$ is a lower bound on the percent area of the k th habitat management zone that must be covered in forest older than the specified older age.

Associated Data

Each atom that falls within a habitat reserve or management zone will be designated as such during preprocessing.

The boundaries of the wildlife reserve and management zones, and the values for the lower and upper bounds for the constraints will be obtained from LMRPs, landscape unit and higher level plans, and other MoF sources.

Riparian Management Constraints

Both riparian reserve zones (RRZs) and riparian management zones (RMZs) are to be explicitly delineated and treated as separate atomic units. It is assumed here that this subdivision of atoms into areas corresponding to RRZs and RMZs has been done during preprocessing, and that the resulting atoms have been attributed appropriately.

Formulation of Constraints

The forest in an RRZ cannot be disturbed, and so this constraint is easily satisfied by never admitting these atoms (once they are spatially defined) into the list of candidates for harvesting.

For RMZs, the situation is rather different. The *Riparian Management Area Guidebook* (Forest Practices Code) summarizes the anticipated *maximum* overall levels of *retention* within the RMZs for each class of stream, wetland and lake. These are described as representing the average levels of retention for all the RMZs for each riparian class in the cutblocks of a plan. It is also stated that these retention levels are not intended to be applied on a cutblock-by-cutblock basis and should not be interpreted as being the required or target level for every RMZ.

This introduces some ambiguity in how the constraints are to be formulated. A specified maximum level of retention implies that no retention at all (i.e., no constraints) is within

the scope of the guidelines, and the implied correspondence between the maximum and the overall average is somewhat confusing (since this situation can only occur when all the values are equal).

It is proposed here that the constraint should in fact take the form of an upper bound on the percent area of disturbance, since, if this bound were reached, it would result in the retention being close to the average retention set down in the above Guidebook. Note that this is a non-age-based area constraint and is different from the age-based area constraints used above for biodiversity, visual quality and community watersheds.

If it is assumed here that the constraints should apply at the landscape unit and riparian class subdivision levels, then these constraints would take the form:

$$\frac{100 \times \sum_j w_j^R(l, r) A_j(l, r)}{\sum_j A_j(l, r)} \leq U^R(l, r),$$

where

$w_j^R(l, r)$ is the fractional amount (weight) of the area of the j th atom in landscape unit l and riparian class r which has been disturbed/cut (see Appendix A for computation of these weights);

$A_j(l, r)$ is the overall area of the j th atom in landscape unit l and riparian class r ;

$U^R(l, r)$ is an upper bound on the percent disturbance for landscape unit l and riparian class r .

Alternatively, constraints of the form given above for biodiversity could be used to limit the cutting within RMZs. These would involve upper bounds on the relative area with trees younger than a certain limit, and lower bounds on the area that must be occupied by trees older than a specified limit.

Associated Data

Each atom in an RRZ or an RMZ will be attributed as such during a preprocessing stage, using TRIM data, FIP files, and the buffer widths and overall retentions specified in the *Riparian Management Guidebook* (FPC). This preprocessing will involve splitting of existing stands to generate RRZ and RMZ atoms for areas bordering streams, wetlands and lakes.

Values for the upper bounds on the percent disturbances are given in Table 4 of the above Guidebook.

Generic Environmental Constraints

From time to time, new types of constraints are expected to arise as a result of evolving issues in forest management, and they must be taken into account in the modeling. Consequently, there must be a provision for adding one or more types of constraints, associating them with the appropriate subdivision categories, and providing lower and upper bounds.

Formulation of Constraints

It is assumed that all types of generic constraints will be based on age (or height) and area. That is, there will be an upper bound on the percent area that may be less than a certain early age, and a lower bound on the percent area that is above a specified older age. For each type of generic management objective, the constraints may be expressed as follows:

$$\frac{100 \times \sum_j w_j^E(\mathbf{s}) A_j(\mathbf{s})}{\sum_j A_j(\mathbf{s})} \leq U^E(\mathbf{s}),$$

$$\frac{100 \times \sum_j w_j^O(\mathbf{s}) A_j(\mathbf{s})}{\sum_j A_j(\mathbf{s})} \geq L^O(\mathbf{s}),$$

where

$\mathbf{s}$ is a vector defining the particular subdivision for which the constraint applies. This will depend on the type of constraint and the degree to which the overall FMU is subdivided for applying the constraints. As an example, for the riparian management constraints above, $\mathbf{s} = (l, r)$.

$A_j(\mathbf{s})$ is the area of the j th atom (all ages) in subdivision $\mathbf{s}$ corresponding to the constraint;

$w_j^E(\mathbf{s})$ is the fractional amount (weight) of the j th atom in subdivision $\mathbf{s}$ which is less than the specified early age (see Appendix A for computation of these weights);

$w_j^O(\mathbf{s})$ is the fractional amount (weight) of the j th atom in subdivision $\mathbf{s}$ which is greater than the specified older age (see Appendix A for computation of these weights);

$U^E(\mathbf{s})$ is an upper bound on the percent area of subdivision $\mathbf{s}$ that may be covered in forest younger than the specified early age;

$L^o(\mathbf{s})$ is a lower bound on the percent area of subdivision $\mathbf{s}$ that must be covered in forest older than the specified older age.

Associated Data

The atoms to which each generic constraint should apply must be attributed as such during a preprocessing stage.

Computing Contributions to Environmental Constraints

The purpose of this section is to describe a way of generalizing the contribution that the attributes of a particular atom may make to a constraint so that this contribution may be partial rather than all-or-none (as at present). The desirability for partial contribution to a constraint is common and occurs whenever:

1. the utility curve for the attribute (i.e., the contribution-vs.-attribute curve) has a form other than a binary step function, and/or
2. the distribution of attribute values is not concentrated at a single value.

Methods based on the approach in this section provide for the atom to make a partial contribution to the constraints. That is, the entire atom may contribute fully or partially, or a portion of the atom may contribute fully or partially. We first present the procedure for the computation of weights to allow computation of partial contributions to constraints. Then a major simplification is provided that will probably be necessary in practice to achieve reasonable computation times.

Computation of Weights

In the spatial modeling problem of interest here, the constraints for the attributes are of the form:

$$\frac{\sum_j w_j A_j}{\sum_j A_j} \leq U \quad \text{or} \quad \frac{\sum_j w_j A_j}{\sum_j A_j} \geq L,$$

where

w_j is the fractional amount (weight) of the j th atom that contributes to the constraint ($0 \leq w_j \leq 1$);

A_j is the area of the atom;

L and U are lower and upper bounds, respectively.

Note that the above constraints are presently, and will continue to be, “hard”, in that the bounds are fixed and each constraint is either satisfied or violated. However, the currently “hard” (i.e., all-or-none, either $w_j = 0$ or $w_j = 1$) contributions of an atom to a constraint are “softened” by the procedure below for computing continuous values for w_j .

It is assumed here that the weight w_j will be dependent on the utility curve for the attribute (usually age), the distribution of the attribute values, and the species composition. It is also assumed here that the attribute of interest is age, but the analysis could be applied to other attributes (e.g., height) as well. Considering only the j th atom, and assuming that the age distribution of each of $k=1, \dots, K$ species can be represented by a set of N_k (age, fractional component) values, the weight w_j can be computed using the expression

$$w_j = \sum_{k=1}^K w_{jk} f_{jk}, \quad \text{where}$$

$$w_{jk} = \sum_{i=1}^{N_k} u_k(x_{ik}) d_{jk}(x_{ik}),$$

and

w_{jk} denotes the weight contribution of the k th species in the j th atom to the constraint;

f_{jk} denotes the fraction of the area of the j th atom that is taken up by the k th species (note that $\sum_{k=1}^K f_{jk} = 1$);

x_{ik} denotes the age of the i th element of the age distribution for the k th species;

$u_k(x_{ik})$ is the value of the utility function for age x_{ik} ($0 \leq u_k(x_{ik}) \leq 1$);

$d_{jk}(x_{ik})$ is the fractional component of the trees of the k th species in the j th atom which are of age x_{ik} (note that $\sum_{i=1}^{N_k} d_{jk}(x_{ik}) = 1$).

This approach to computing w_j can accommodate situations where:

- The contributions to the constraints by single-valued attributes (a “spike” distribution) are all-or-none (i.e., binary utility curves). Example: an even-aged stand that, if less than age 20, contributes fully to the constraint, but otherwise does not contribute at all.
- The contributions to the constraints by single-valued attributes are partial (i.e., continuous utility curves). Example: an even-aged stand that contributes partially to

the constraint for a watershed area if the age is between 5 and 20, fully if the age is 20 or more, and not at all if the age is less than 5.

- All of the atom contributes to the constraints (i.e., the entire distribution falls within a region of the curve where $u(x) = 1$). Example: a mixed-age stand with three ages (10, 20, and 40 years) where the binary constraint is that the age be less than 50.
- None of the atom contributes to the constraints (i.e., the entire distribution falls within a region of the curve where $u(x) = 0$). Example: a mixed-age stand with three ages (10, 20, and 40 years) where the binary constraint is that the age be greater than 50.
- Only some portion of the atom contributes to the constraints (the distribution spans a non-unity region of the utility curve). Example: a mixed-age stand with three ages (10, 20, and 40 years) where the binary constraint is that the age be greater than 30.

Simplification of Weighting for Efficient Modeling

While the above approach allows atoms to make a partial contribution to the constraints, it is expected that actually generating the histograms and performing the summations at each stage might be prohibitively time-consuming at present. Hence, it may be necessary to simplify the mechanism for computing the contribution of an atom to a constraint, effectively reducing it to an all-or-none contribution. This may be done by assuming that:

1. The various species may be combined into a single “effective” species.
2. The attribute distribution can be represented by a single “averaged” value (e.g., a distribution of ages can be reduced to an average age).
3. The utility curve is binary, and defines a domain for the attribute value within which its area contributes fully to the constraint, and outside of which it does not contribute at all.

The weights may then be defined as:

$$w_j = \begin{cases} 0, & \text{if the "averaged" attribute value is outside the domain;} \\ 1, & \text{if the "averaged" attribute value is inside the domain.} \end{cases}$$

This greatly simplifies the computation load of constraint evaluation, but at the cost of not allowing the partial contribution of an atom to a constraint.

Appendix 2: Cutblock Constraints

Harvesting is controlled at the cutblock level, and before any atom can be cut, it must previously have been included in a cutblock. Cutblocks can range in size from a few hectares to a few thousand hectares. There are several issues that apply to cutblock definition and management during FMU modeling. First, their size distribution is ecologically important, and should conform to specified distributions. Secondly, when a

cutblock is harvested, there is a prohibition of harvesting within adjacent cutblocks until a green-up period is reached. Third, there is an urgency to complete the cutting of the block once it has been initiated. These issues are described in more detail in this section.

Cutblock Delineation and Size Constraints

In considering cutblock delineation, we must first consider the topic of *patches*. Patches are regions of the forest in which the trees are more or less similar in age within the region, but significantly different in age (e.g., by at least 20 years) from adjoining regions. In the absence of human activity, patches are produced by natural disturbances such as wildfire, blowdown, or infestation. They represent an area where the forest was disturbed and has since regenerated. The distribution of the sizes of these patches in nature is of ecological importance and depends on the BEC and NDT.

Using the principle (as noted in the *Biodiversity Constraints* section above) that anthropogenic disturbances should mimic natural disturbances as closely as possible, it is desirable to maintain a size distribution of patches resulting from harvesting that is similar to that arising naturally. This can be done by defining individual cutblocks so that their size distribution corresponds closely to the natural (known or predefined) patch size distribution. Hence, the requirement to preserve a natural patch size distribution translates to imposing size constraints on cutblocks.

For the initial version of the modeling, MoF will provide an initial set of predefined cutblocks with size distributions that conform to the desired target size distributions. (MoF will also predefine the road connections between cutblocks and costs as noted below.) This will simplify and increase the speed of the modeling process.

Green-up Constraint for Harvesting Adjacent Cutblocks

There is a general constraint that, as soon as harvesting has been initiated within a cutblock, harvesting of adjacent cutblocks must be “frozen” until the first cutblock has reached the green-up height (or age). To satisfy this constraint, lists of adjacent cutblocks should be kept for each cutblock. As soon as harvesting activity begins in one cutblock, the adjacent cutblocks must then be prohibited from being harvested until green-up is achieved. An exception to this rule occurs for commercial thinning, where a certain volume may be taken from cutblocks adjacent to recently harvested cutblocks while the constraint is still considered to be satisfied.

It is possible that this process may lead to sealing off access to otherwise harvestable stands bordering the frozen cutblocks. To handle this situation, a process must be provided for identifying cutblocks outside the frozen region and making them accessible by traversing one of the frozen cutblocks. Hence, it will be necessary to dynamically estimate the cost of connecting a currently inaccessible cutblock to a roaded (or otherwise accessible) cutblock. This would allow the least costly route to be identified.

Timely Harvest Completion Constraints

Once harvesting has started within a cutblock, it is desirable that it should proceed to completion as rapidly as possible, for the following reasons:

- Cutblocks would then correspond more closely to patches, i.e., age-homogeneous areas distinct from adjacent areas, thus mimicking the likely impact of a local natural disturbance.
- Potentially harvestable timber is not tied up. If the cutting took place over many years, then cutting of adjacent cutblocks could not proceed during both this time and the time taken for green-up, resulting in a delay in availability of the timber in these adjacent blocks.
- It is operationally more realistic, since equipment and facilities would already be in the area and not have to be moved long distances.

The aim during simulation would be to cut all the atoms in a cutblock during the same period. If this is not possible, for example, due to environmental constraints, then one option would be to close the cutblock and leave it incompletely harvested. An alternative strategy would be to give it top priority for harvesting consideration during the next period. If it happens that cutting of all the atoms in a cutblock cannot be completed after several consecutive periods, then the block should probably be closed to allow the green-up interval to proceed. A further option in this event would be to designate the unharvested atoms as reserves for the duration of the simulation, rather than keep on trying to harvest them.

Future Automated Cutblock Generation

In the future, as computing power increases, it will be desirable for cutblock generation (and road access) to be handled dynamically during the modeling. Hence, the design of the modeling system should also provide for its eventual extension to incorporate these capabilities, and include a provision for a future automated process that would allow the automated delineation of cutblocks, perhaps using some initial set of MoF-provided cutblocks as a starting point.

The use of the *blocking unit* spatial level – intermediate between a management unit and a cutblock (see the *Spatial Hierarchy* section above) – would simplify the development of automated cutblock generation procedures. Cutblocks must be made up of regions containing atoms with a similar species composition and age range, and these regions would constitute the blocking units (to be predefined by MoF). These blocking units would provide a spatial unit within which cutblocks could be defined on the basis of geometry alone, and would also provide an appropriate level for specifying cutting types and harvest/silviculture regimes.

Automated cutblock generation would be performed within these blocking units, either in a preprocessing stage or during simulation. This process would dynamically group atoms within a blocking unit into cutblocks on the basis of geometry. During this process, the target size of each individual cutblock would be based on a random realization from the patch size distribution.

The following additional points with respect to future automated cutblock generation should be noted:

- If the automated generation was done in a preprocessing stage, the resulting cutblocks could be used (after adjustment or “tuning”, if required) as input to subsequent simulations. This would save the time required for dynamic cutblock delineation during these runs and ensure the consistency of the spatial units between runs.
- The assignment of cutblock sizes should take into account that, in some regions, the scope for large cutblocks may be restricted, while in others, large tracts may be available. Using a single histogram for all regions in the FMU would often not be appropriate, and several region-specific distributions may have to be used.
- It may be desirable to be able to keep track of the order in which the cutblocks are harvested, to result in a “pseudo-road” network or progression.
- In the cutblock delineation process, it will also be necessary to prevent connections between certain cutblocks, e.g., those on either side of a steep ridge or a wide river.

Appendix 3: Accessibility Constraints

An area can generally only be harvested if there is some means of road access, although there are some exceptions (e.g., coastline-based operations and helicopter logging). Hence, there is usually a spatial constraint that a cutblock can only be logged if it is penetrated by a road or can be accessed at reasonable cost, and this should be taken into account in the simulation.

Modeling Accessibility at the Cutblock Level

The basic level of roading is the cutblock, and the road accessibility status of each cutblock will be maintained as a variable attribute. For the initial modeling, MoF will not only predefine the cutblocks (see the *Cutblock Constraints* section above) but will also provide a directed “future road network” in which every potentially accessible cutblock in the FMU is connected. In this preprocessing, the centers of the cutblocks will be connected in a physically reasonable way (or, if appropriate, the blocks will be designated as inaccessible) and the outward, or “downstream”, direction of the connecting segments will be indicated. Each cutblock will store the ID of the adjacent blocks to which it was connected by road, and the associated access costs. The trade-off of this strategy is a reduced computational effort in the modeling stage at the expense of a much more intensive (but one-time-only) preprocessing stage.

With this predefined network approach, each separate road segment (i.e., path between branches) could be stored, along with the IDs of the blocks along the segment (in order), and the cost of the individual connections between the blocks. The blocks could also store the segment IDs of their internal roads. Each segment could also store the ID of the next “downstream” segment, i.e., the one which would provide the preferred route for reaching the outside world. The existence of such network-based functionality would provide considerable flexibility and realism in the simulation.

There are three categories of accessibility that are of interest here, namely:

- presently roaded (already built at start of the current time period)
- roadable (not yet built, but on the future road network – see below)
- inaccessible.

At the start of the modeling, each cutblock would be initialized according to its accessibility category and projected roading cost. During the modeling, only those cutblocks that are roaded at the beginning of a time period would be considered for harvesting during that period. At the end of each time period, the costs of roading, the value of the cutblocks, and the roading budget would be taken into consideration in deciding which roads to build. The accessibility category and costs of the cutblocks affected by the newly-built roads would then be updated. The criteria for deciding which roads to build in which time period would be expected to have a substantial effect on the overall simulation, since this would determine the evolution of the supply of new cutblocks.

It is noted here that roading further constrains harvesting by reducing the area of an unroaded cutblock by a specified percentage the first time the cutblock is roaded, since this area will not be available for growing trees after the initial harvest.

Future Accessibility Modeling

In the future, it may be desirable to handle the need to model road accessibility automatically, and it is proposed to provide an option for a progressive assignment of accessibility based on adjacency. This scheme would designate as roaded (and use as a starting point) those cutblocks that are currently road-accessible or have been recently cut (i.e., are younger than some specified age). During each time period, all cutblocks adjacent to these cutblocks, and which have not been identified as being permanently inaccessible, would then be designated as potential candidates for being roaded in that period. The actual assignment of which cutblocks are to be roaded would depend on the difficulty of constructing a road (which could be estimated by the slope and cross-slope) and the associated cost. Budgetary considerations would then limit the number of candidate cutblocks that could actually be roaded.

Also in the future, MoF will probably require more realistic procedures than simple adjacency for automated roading during the preprocessing stage, to provide more realistic input for the modeling. These algorithms should also take topographic and geographic information into account, including slope, terrain stability, hydrography, and soil type. A number of different approaches to automated roading have been proposed (see Clark et al. (1997, 2000), Dean (1997), Liu and Sessions (1993), Murray (1998), Sessions (1987)), and may be useful in developing procedures for roading the cutblocks in the current application.

Appendix 4: Attributes of Spatial Units

Each spatial unit has one or more attributes associated with it, depending on the role it plays in the simulation. An initial set of attributes is given in the current section, and this set may be modified or expanded as required. In this attribution scheme, the attributes have been assigned to spatial units as high as possible in the spatial hierarchy, to avoid redundancy of attribution. Some adjustment here may be necessary as the design of the simulation evolves.

Some of these attributes will be fixed during the entire simulation, while others will vary. During the simulation, the variable attributes will be updated as appropriate, and these attributes will constitute the *state* of the unit. The state for each unit could be saved at the end of each modeling period, or it could be deduced from an initial state followed by a list of changes to the state. (In the latter case, it will be necessary to determine an efficient way of doing this, using initial conditions and stand history.)

It is desirable, for computational efficiency, that attributes be expressed in the programs as numeric quantities (preferably integers) rather than strings, whenever possible. It is intended that an interface would be developed to allow the convenient viewing and/or specification of these attributes before or during the simulation.

Forest Management Unit Attributes

The following are the attributes for the FMU; they are all fixed during processing.

Name

Definition: The name of the FMU.

Source: Assigned during preprocessing.

Representation: String.

Constituent landscape units

Definition: The ID numbers of the LUs that the FMU contains.

Source: Assigned during preprocessing.

Representation: List of integer values.

Landscape Unit Attributes

The following are the attributes for the LUs; they are all fixed during processing.

ID number

Definition: The unique identifier for the LU.

Source: Assigned during preprocessing.

Representation: Integer.

Constituent development units

Definition: The ID numbers of the DUs that the LU contains.

Source: Assigned during preprocessing.

Representation: List of integer values.

Location

Definition: The centroid of the LU, or alternatively, the center of the bounding box for the LU.

Source: Computed from the polygon points defining the spatial extent of the LU during preprocessing.

Representation: A pair of integer values ([x,y] expressed as meters, in the BC Albers projection).

Area

Definition: The planimetric area of the LU.

Source: Computed from polygon points defining the spatial extent of the LU during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Development Unit Attributes

The following are the attributes for the DUs; they are all fixed during processing.

ID number

Definition: The unique identifier for the DU.

Source: Assigned during preprocessing.

Representation: Integer.

Parent landscape unit

Definition: The ID number for the LU containing this DU.

Source: Assigned during preprocessing.

Representation: Integer.

Constituent blocking units

Definition: The ID numbers of the BUs that the DU contains.

Source: Assigned during preprocessing.

Representation: List of integer values.

Location

Definition: The centroid of the DU, or alternatively, the center of the bounding box for the DU.

Source: Computed from the polygon points defining the spatial extent of the DU during preprocessing.

Representation: A pair of integer values ([x,y] expressed as meters, in the BC Albers projection).

Area

Definition: The planimetric area of the DU.

Source: Computed from polygon points defining the spatial extent of the DU during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Development cost

Definition: All costs associated with cruising, forest planning (involvement in higher level and strategic planning and preparation of operational plans), forest protection, supervision of road construction and logging, road and cutblock layout, divisional due diligence and the divisional/operation share of head office forest management expense.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Hauling cost

Definition: All costs associated with hauling logs from the logging site to the mill log yard or an intermediate dump site such as a rail reload point or a water dump site for intermediate water transport by boom or barge.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Dump, sort, boom and rehaul

Definition: All costs of intermediate rail or water transport, including log dewatering costs for intermediate water transport but excluding log yard dumping or sorting costs.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Crew transportation costs

Definition: The cost of transportation for the logging crews within the DU.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Crew accommodation costs

Definition: The cost of housing the logging crews within the DU.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Overhead costs

Definition: All remaining camp overhead, including administrative support wages/salaries and benefits, scaling expense, residue and waste expense, vehicle expense for logging administrative staff, licenses, fees, insurance, taxes, leases annual rent and depreciation of administrative buildings.

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Blocking Unit Attributes

Blocking units have both fixed and variable attributes, as described below.

BU Fixed Attributes

ID number

Definition: The unique identifier for the BU.

Source: Assigned during preprocessing.

Representation: Integer.

Parent management zone

Definition: The ID number for the DU containing this BU.

Source: Assigned during preprocessing.

Representation: Integer.

Location

Definition: The centroid of the BU, or alternatively, the center of the bounding box for the BU.

Source: Computed from the polygon points defining the spatial extent of the BU during preprocessing.

Representation: A pair of integer values ([x,y] expressed as meters, in the BC Albers projection).

Area

Definition: The planimetric area of the BU.

Source: Computed from polygon points defining the spatial extent of the BU during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Cutting type options

Definition: The type of harvesting strategies that can be applied within the BU. Options include clear-cut, commercial thinning, single-species removal, partial cutting, and variable retention.

Source: Preprocessing.

Representation: List of integer codes.

Harvesting and silviculture regime

Definition: The harvesting, regeneration, and silvicultural strategies to be applied to the stands in the blocking unit. This includes frequency of harvest, type of harvest (clear-cut, partial harvesting, single-species removal, commercial thinning) and the intervals and extents associated with partial cutting, as well as the silvicultural treatments to be applied during the growth/harvest cycle and their timing.

Source: Preprocessing.

Representation: To be specified by MoF. (See *Cutting Types* section below.)

BU Variable Attributes

Constituent cutblocks

Definition: The ID numbers of the cutblocks that the BU contains.

Source: Assigned during preprocessing.

Representation: List of integer values.

Update: During blocking stage of the simulation, if new cutblocks are created.

Cutblock Attributes

Cutblocks have both fixed and variable attributes, as described below.

Cutblock Fixed Attributes

ID number

Definition: The unique identifier for the cutblock.

Source: Assigned during preprocessing.

Representation: Integer.

Parent blocking unit

Definition: The ID number for the BU containing this cutblock.

Source: Assigned during preprocessing.

Representation: Integer.

Harvesting system

Definition: The type of harvesting process that will be used to cut the atom: either conventional (ground-based, high lead or grapple) or non-conventional (helicopter, long-line).

Source: Preprocessing.

Representation: Integer code.

Site access

Definition: The type of access that is required to harvest the site: either road, aerial, water or inaccessible.

Source: Preprocessing.

Representation: Integer code.

Harvest development road cost

Definition: All costs related to road and bridge construction, maintenance and de-activation for both expensed and capital roads, but excluding road design and layout costs and supervision, which should be included in Development costs above.

Source: Preprocessing – determined on the basis of area, slope, and empirical cost data.

Representation: Expressed as dollars/ha (integer).

Road maintenance cost

Definition: The yearly cost of maintaining the roads within the atom.

Source: Preprocessing.

Representation: Expressed as dollars/km (integer).

Tree-to-truck cost

Definition: All costs associated with falling, bucking, yarding, and loading timber from the point immediately after the pre-work conference to immediately before the logs are hauled. This category does not include the cost of salaried logging supervisors which is included within Development cost (see *Development Unit* section above).

Source: Preprocessing.

Representation: Expressed as dollars/m³.

Forest regeneration (silviculture) cost

Definition: All costs associated with bringing a logged site back to free growing status, including site preparation, planting, brushing and weeding, fill planting and surveys (together, "basic silviculture"). This category does not include "intensive" or "incremental" silviculture techniques applied to free growing stands.

Source: Preprocessing.

Representation: Expressed as dollars/ha.

Cutblock Variable Attributes

Constituent atoms

Definition: The ID numbers of the atoms that the cutblock contains.

Source: Initialized during preprocessing.

Representation: List of integer values.

Update: During blocking stage of the simulation, if new cutblocks are created.

Location

Definition: The centroid of the cutblock, or alternatively, the center of the bounding box for the cutblock.

Source: Initialized from the polygon points defining the spatial extent of the cutblock during preprocessing.

Representation: A pair of integer values ($[x,y]$ expressed as meters, in the BC Albers projection).

Update: During blocking stage of the simulation, if new cutblocks are created.

Area

Definition: The planimetric area of the cutblock.

Source: Initialized from polygon points defining the spatial extent of the cutblock during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Update: During blocking stage of the simulation, if new cutblocks are created.

Adjacent cutblocks

Definition: A list of other cutblocks sharing a boundary with this cutblock, and the lengths of these boundary segments. (An option here is to include only those adjacent cutblocks that share a boundary greater than a certain minimum length.)

Source: Initialized during preprocessing.

Representation: List of integer IDs and lengths (meters).

Update: During blocking stage of the simulation, if new cutblocks are created.

Road accessibility status

Definition: The status of the current accessibility of the cutblock, i.e., presently roaded, future roaded, or inaccessible (see *Accessibility Constraints* section above).

Source: Preprocessing.

Representation: Integer code.

Update: When the cutblock becomes roaded and/or during blocking stage of the simulation, if new cutblocks are created.

Road access cost

Definition: The cost of road-building to connect this cutblock to the presently roaded portion of the road network.

Source: Preprocessing.

Representation: Expressed as dollars (integer).

Update: When the cutblock becomes presently roaded, or any road is built which affects the access to this cutblock. Also, during blocking stage of the simulation, if new cutblocks are created.

Adjacent cutblocks in road network

Definition: A list of other cutblocks to which this cutblock is connected in the road network.

Source: Preprocessing.

Representation: List of integer IDs.

Update: During blocking stage of the simulation, if new cutblocks are created.

Road segment IDs

Definition: The identifiers of any present or future road segments that are contained within the cutblock.

Source: Preprocessing.

Representation: List of integer codes.

Update: During blocking stage of the simulation, if new cutblocks are created.

Atom Attributes

Atoms have both fixed and variable attributes, and the number of these attributes is greater than for any other type of spatial unit.

Atom Fixed Attributes

The following attributes of an atom are constant with time, and are initialized once, at the start of the modeling.

ID number

Definition: The unique identifier for the atom.

Source: Assigned during preprocessing.

Representation: Integer.

Parent cutblock

Definition: The ID number for the cutblock containing this atom.

Source: Assigned during preprocessing.

Representation: Integer.

Location

Definition: The centroid of the atom, or alternatively, the center of the bounding box for the atom.

Source: Computed from the polygon points defining the spatial extent of the atom during preprocessing.

Representation: A pair of integer values ($[x,y]$ expressed as meters, in the BC Albers projection).

Area

Definition: The planimetric area of the atom.

Source: Computed from polygon points defining the spatial extent of the atom during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Slope

Definition: The gradient at a point on a surface in the direction of maximum descent (averaged over a number of points distributed throughout the atom).

Source: Analysis of TRIM DEM data: triangulation, sampling and spatial averaging over atom area (will be computed during preprocessing).

Representation: Expressed as percent, i.e., change in elevation per 100 m of horizontal extent (integer).

Aspect

Definition: The direction of the maximum downward gradient of a point on a surface (averaged over a number of points distributed throughout the atom).

Source: Analysis of TRIM DEM data: triangulation, sampling and spatial averaging over atom area. (will be computed during preprocessing).

Representation: Expressed as degrees, rounded to the nearest 10 degrees (integer).

Elevation

Definition: The height above sea level of a point on a surface (averaged over a number of points distributed throughout the atom).

Source: Analysis of TRIM DEM data: triangulation, sampling and spatial averaging over atom area (will be computed during preprocessing).

Representation: Expressed as meters (integer).

Adjacent atoms⁴

Definition: A list of other atoms sharing a boundary with this atom, and the lengths of these boundary segments. (An option here is to include only those adjacent atoms that share a boundary greater than a certain minimum length.)

Source: Preprocessing.

Representation: List of integer IDs and lengths (meters).

Site index

Definition: A measure of site productivity, taken as the average height (m) that the tallest trees within an atom are expected to achieve in 50 years.

Source: Preprocessing (derived from inventory data).

Representation: Expressed as meters (integer).

Wildlife tree patch

Definition: A flag denoting whether the atom contains wildlife tree patches, and the percent area of this atom that is to be occupied by these patches.

Source: Preprocessing.

Representation: Integer flag, integer percent area.

Management Zones

Definition: A list of the Management Zones of which the atom is a part.

⁴ An atom adjacency list is required for dynamic cutblock generation, and may not be required if the cutblocks are predetermined.

Source: Preprocessing.
Representation: Set of integer IDs.

Reserves

Definition: A list of the reserves of which the atom is a part.
Source: Preprocessing.
Representation: Set of integer IDs.

Deferral status

Definition: An indication that this atom is subject to special considerations which are time-constrained, i.e., a reversion date is defined. These categories include:

- Areas involved in land claims
- Environmentally sensitive areas

Source: Preprocessing.
Representation: A list of integer (code, date) pairs.

Atom Variable Attributes

The following attributes of an atom vary as the modeling proceeds. They are initialized at the start of the modeling and updated as required at the end of each period.

Parent cutblock

Definition: The ID number for the cutblock containing this atom.
Source: Assigned during preprocessing.
Representation: Integer.
Update: When an atom is incorporated into, or removed from, a cutblock.

Land categorization

Definition: One or more codes indicating whether this atom falls into a particular land category (as distinct from the reserve category above). These classifications are:

- Forested
 - productive
 - non-productive
- Non-vegetated (examples: water, ice, rock)
- Vegetated- non-forest (examples: brush, meadow, agricultural land)

Source: Preprocessing.
Representation: An integer code denoting one of the above categorizations.
Update: When an atom is transferred from one category to another.

History

Definition: Any event that alters the growth characteristics of the atom.
Source: MoF records, model harvesting events, silviculture prescriptions.
Representation: A chronological list of (year, event description) records. Events include:

- Full harvest
- Partial harvest
- Fire
- Infestation
- Planting
- Silviculture treatment, including:
 - rehabilitation
 - fertilization
 - pruning
 - spacing
 - pre-commercial thinning
 - commercial thinning

Expressed as a set of (integer year, integer treatment code) pairs.

Update: Whenever a harvest event or other disturbance occurs, and at prescribed times after the event, depending on the silviculture regime. Note that following a treatment, the growth and yield curve may need to be changed.

Density

Definition: The number of stems/ha.

Source: Preprocessing of forest inventory files, regeneration during modeling.

Representation: Integer value.

Update: Whenever a harvest event occurs.

Basal area (site occupancy)

Definition: The cross-sectional area of the trees in the atom compared to the total area of the atom.

Source: Preprocessing of forest inventory files, regeneration during modeling.

Representation: Expressed as m²/ha (integer).

Update: Whenever a harvest event occurs.

Structure

Definition: The species, age, density and volume composition of the atom.

Source: Preprocessing of forest inventory files, regeneration during modeling.

Representation: Each of up to four species is represented separately with the following information:

- *Species:* The species of tree for which this data applies. Expressed as an integer code.
- *Percent:* The percent of the stand that is taken up by trees of this species. Expressed as an integer value.
- *Age:* The age distribution of the species, in the form of pairs of (age, percent) values. Other values, such as height and diameter, can be derived from this data if required. Expressed as a set of integer pairs.
- *Volume:* The total volume of timber of this species. Expressed as m³ (integer).

Update:

- *No harvest:* At the end of each period, or following a silviculture treatment, the ages are updated, and the volumes are recomputed based on growth-and-yield curves for the species.
- *Full harvest:* If the entire atom is cut, all stand structure parameters are re-initialized. The species and percent area are set on the basis of prescribed values, the age is set to zero and the volumes are recomputed.
- *Partial harvest:* The update depends on the nature of the partial harvest. In general, the percent area and age composition for each species are redefined based on the characteristics of the partial harvesting. The species volumes are then recomputed accordingly.

(Note: For a stand of known age and composition, the stand characteristics are currently computed and assigned on the basis of growth-and-yield curves, and any actual inventory information is ignored. This approach is to be changed to one which starts with the inventory information and applies a trend to make it coincide with the growth-and-yield curve after a certain time.)

Growth and yield table

Definition: The current growth and yield table that represents the growth of the trees within the stand.

Source: Assigned on the basis of site index, history and structure.

Representation: Integer code.

Update: Whenever the atom is harvested (clear or partial cut) or a silviculture treatment is applied.

Current volume

Definition: The current total volume of the timber in the stand.

Source: Derived from stand structure.

Representation: Expressed as m³ (integer).

Update: At the end of each period.

Timber value by species

Definition: The timber value of all the trees of each species of tree in the atom.

Source: Derived from the stand structure information for the atom and empirical information.

Representation: Expressed as dollars (integer).

Update: Whenever the stand structure is updated.

Timber value – total

Definition: The total timber value of all the trees in the atom.

Source: Derived from the stand structure information for the atom and empirical information.

Representation: Expressed as dollars (integer).

Update: Whenever the stand structure is updated.

Timber value - net

Definition: The net value of all the trees in the atom: stand value less road costs (first time harvest only) and harvesting cost.

Source: Derived from the stand structure information for the atom and the fixed stand attributes.

Representation: Expressed as dollars (integer).

Update: Whenever the stand structure is updated.

Appendix 5: Management Zone and Reserve Attributes

Management Zone Attributes

The following are the attributes for the Management Zones.

ID number

Definition: The unique identifier for the MZ.

Source: Assigned during preprocessing.

Representation: Integer.

Management Zone type

Definition: The type of management objective that is to be satisfied for this MZ. The objectives are each associated with an environmental constraint, e.g.:

- biodiversity constraints:
 - land base (total or timber harvest)
 - natural disturbance type (NDT)
 - biogeoclimatic ecosystem classification (BEC) and variant
 - biodiversity emphasis option (BEO)
- visual quality constraints:
 - visual quality class
- community watershed constraints:
 - community watershed class
- wildlife habitat management constraints:
 - wildlife habitat class
- riparian management constraints:
 - riparian class
- generic constraints:
 - user-defined classes

Source: Assigned during preprocessing.

Representation: Integer code corresponding to a particular environmental management objective.

Atoms included

Definition: A list of atoms that are contained in this MZ, i.e, are subject to this constraint.

Source: Assigned during preprocessing.

Representation: Set of integer IDs.

Age/height bound

Definition: The cutoff age or height that defines whether the atom contributes toward the constraint (assumes binary utility curve).

Source: Assigned during preprocessing.

Representation: Integer.

Area bound

Definition: The cutoff percent area applying to the constraint.

Source: Assigned during preprocessing.

Representation: Integer.

Total area

Definition: The total area of the atoms involved in the constraint.

Source: Assigned during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Contributing area

Definition: The combined area of those atoms contributing to the constraint.

Source: Assigned during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Percent area

Definition: The percent area of atoms contributing to the constraint.

Source: Assigned during preprocessing.

Representation: Integer (multiple of 0.01).

Reserve Attributes

The following are the attributes for the reserves.

ID number

Definition: The unique identifier for the reserve.

Source: Assigned during preprocessing.

Representation: Integer.

Reserve type

Definition: The type of reserve. This could be one of:

- Old growth management area
- Park
- Critical wildlife habitat
- Riparian reserve zone (RRZ)
- Wildlife tree reserve

- Private
 - TFL
 - Federal Land
- Source:* Preprocessing.

Representation: Integer code.

Atoms included

Definition: A list of atoms that are contained in this reserve.

Source: Assigned during preprocessing.

Representation: Set of integer IDs.

Total area

Definition: The total area of the atoms in the reserve.

Source: Assigned during preprocessing.

Representation: Expressed as hectares (integer multiple of 0.1 ha).

Appendix 6: Economic Considerations

Most, if not all, of the decisions made about whether, how, and when to harvest are at least partly based on economic considerations. At the most basic level, this involves taking into consideration both the worth of the timber and the costs associated with access and harvesting. The former can be modeled using tables that provide the worth/m³, in terms of species and age (or size). Decisions on harvesting are made at the cutblock level, and so the worth/m³ of the timber in the cutblock must be determined. This simply requires the summing of the volume-weighted stand value, i.e.,

$$W(c_j) = \frac{\sum_i W(s_{ij})V(s_{ij})}{\sum_i V(s_{ij})},$$

where

$W(c_j)$ is the worth/m³ of the j th cutblock;

$W(s_{ij})$ is the worth/m³ of the i th stand within the j th cutblock;

$V(s_{ij})$ is the volume (m³) of timber within the i th stand of the j th cutblock.

Costs are more complicated to model than value, but in general, these can be broken down into a number of categories, with associated units, time frames and budgets, as in the following table.

Cost	Associated Unit	Timing
Development (\$/ha)	Development Unit	Once-only, before initial harvest
Hauling (\$/m ³)	Development Unit	During harvest
Dump, sort, boom and rehaul (\$/m ³)	Development Unit	During harvest
Crew transportation (\$/m ³)	Development Unit	During harvest
Crew accommodation (\$/m ³)	Development Unit	During harvest
Overhead(\$/m ³)	Development Unit	During harvest
Traversal road (fixed path) (\$/km)	Stand	Once-only
Harvest development road (\$/ha)	Stand	During initial harvest
Road maintenance (\$/km)	Stand	Continuous, following initial harvest
Tree-to-Truck (\$/m ³)	Stand	During harvesting
Forest regeneration (silviculture) (\$/ha)	Stand	Following each harvest

In computing the costs associated with harvesting a particular cutblock during a specific period, the following considerations should be taken into account:

- Costs may be provided in terms of timber volume, land area or road length, depending on the nature of the activity.
- Once-only costs could be spread out over a number of years, and over all the cutblocks that are affected by the actions incurring the cost.
- Other types of costs are applied during the period in which the activity occurs.
- Stand-level costs can be summed (with appropriate weighting) to determine their contribution to the cutblock-level cost.
- Development unit-level costs can be subdivided (with appropriate weighting) to estimate the cost at the cutblock level.
- The cutblock-level costs will change with time, as development and road access proceed.

Appendix 7: Cutting Types

There are five different cutting types, i.e., approaches that may be taken to harvesting within a cutblock, and these should be included in the modeling. In addition, the modeling should provide for controlling the types that could be used in any given situation. This section discusses the role and control of cutting types in modeling.

The five cutting types are as follows:

- ***Clear-cutting.*** This is the complete removal of all standing timber. It is the most straightforward of the types to model. The age of the stand is reset to zero following a clear-cut, and an appropriate growth-and-yield table is assigned to the stand.
- ***Commercial thinning.*** This is the removal of a volume dispersed throughout the stand (either pure or mixed species), without targeting a particular species. In order to be eligible for commercial thinning, either the stand timber volume/ha must exceed a certain minimum volume, or the tree diameter must be greater than a specified diameter. Also, the age of the stand must be less than a certain maximum age, so that there is effectively a window of opportunity for this cutting type. Commercial thinning may be used to remove a volume from stands adjacent to a recently harvested cutblock, but at levels that do not violate the adjacency green-up constraints. It allows some of the volume in adjacent cutblocks to be removed before the green-up period has passed; the remainder of the stand is generally clear-cut some years later, at the end of this period.
- ***Single-species removal.*** This allows the removal of a single species from a mixed species stand. It is similar to commercial thinning, in that the volume removed is dispersed throughout the stand, but here a single species is targeted. As with commercial thinning, the remainder of the stand is usually clear-cut at some point after single-species removal.
- ***Partial cutting.*** This allows a percent removal of timber within a pure or mixed species stand without targeting a particular species. It does not necessarily assume a later clear-cut, and requires that the multi-age structure of the stand be explicitly tracked on an ongoing basis. For the stand to be eligible for partial cutting, the age of its trees must exceed a certain minimum harvest age, or the stand must contain a specified minimum residual volume/ha. The harvest is controlled by specifying a set of percent removals (or recovery volumes) and re-entry periods.
- ***Variable retention.*** This allows a once-only percent volume reduction without targeting individual species. The area that is cut is replanted and then left alone, so that the volume of the older part of the stand is effectively reserved and can contribute to biodiversity and other constraints. Both the younger and older components of the stand must be separately tracked in this harvest type.

The harvesting model will require the specification of the cutting types that are available for each stand, and also the preferences for the type that should be used in a particular situation. In addition, for each cutting type, the harvesting/silviculture regimen that is to be applied must be specified. Currently, most harvesting is done using clear-cutting, but other cutting types are also used, and these can also help to make up shortfalls in achieving the harvest targets while keeping within the limits of the constraints. For different regions (spatial units), it may be appropriate to consider only certain of these cutting types, and the preference for the order in which these various cutting types should be considered may also differ between regions. Moreover, the type of harvest/silviculture regime will also vary between different regions. Hence, mechanisms for the selection, control, and treatment regime must be supplied at the appropriate level. To this end, it is proposed that:

- The cutting type options should be specified at the *blocking unit* level. That is, each blocking unit will have been assigned a set of cutting types which could, in principle, be used to harvest the cutblocks within that blocking unit.
- The harvest/silviculture regime for each cutting type should also be specified at the *blocking unit* level. The cutblocks within that unit would then be harvested and treated according to the regime specified in the harvest/silviculture table for that blocking unit.
- The order of preference for which cutting types should be considered should be specified at the *landscape unit* level. For example, if the clear-cut option is exhausted within a particular LU, then the commercial harvesting option may be the next preference to use within that LU, followed by variable retention. For another LU, management considerations may make partial cutting the first preference, then variable retention.

These cutting type-related specifications will be performed by MoF during a preprocessing stage, and serve as input to the model.

When any type of cutting is performed, the stand structure, volume curves, and other data must be reset or adjusted appropriately, depending on the cutting type, cutblock location, etc. The detailed specifications for this updating will be provided by MoF.

Appendix 8: Tables

Several additional types of input data will be required, in the form of tables. These include:

- Growth and yield tables
- Harvest/silviculture specification tables, including treatment costs
- Green-up age tables
- Log price tables
- Product price tables
- Succession tables for species colonization and establishment following a disturbance.