A Linear Programming Framework for Models of Forest Management Strategy

by

Andrew B. Martin

Submitted in partial fulfillment of the requirements
for the degree of Master of Applied Science

at

Dalhousie University
Halifax, Nova Scotia
September 2013

Contents

List of Figures

List of Tables

Abstract

Results found in this thesis draw attention to limitations in the conventional approach to modelling forest management strategy, where models have insufficient spatial resolution and ignore industry. Addressing these limitations, a Model One linear programming framework was developed in which models built can model strategically relevant spatial resolution, and include industry representation. In a case-study on Nova Scotia's Crown Central Forest, models from this framework were compared with Woodstock™, a commercial modelling framework. When strategically relevant spatial resolution was modelled, these models found solutions in substantially less time than Woodstock. Of further interest, the framework's industry representation allows novel analysis to be performed. A comparison between a model that includes industry and a conventional model demonstrates that the conventional model schedules unprofitable stands for harvest. Then, models with industry representation are used to demonstrate industry based analysis, such as assessing the cost of a clearcut restriction policy and investigating the benefit of industrial expansion. Taken together, the results herein contained make an argument for modelling forest management strategy at strategically relevant spatial resolution, and including industry representation in modelling.

Acknowledgements

Eldon Gunn and Evelyn Richards made this thesis possible. I appreciate the time and energy they committed to this project; as well as the guidance and support with which they provided me.

Rob O'Keefe and the Nova Scotia Department of Natural Resources in Truro contributed significantly to this work through meetings and phone calls, and by providing us with forest data, without which the project would have been a shadow of what it is.

The NSERC VCO Network funded this research, and the efforts of Jean-Francois Audy, Mustapha Ouhimmou, Catherine Levesque, Barbara Mackenzie-Maisonneuve, and Julie Richard made being a part of the VCO Network a positive experience.

Thanks are due to Dick Sutherland and Corinne MacDonald for serving on my defense committee, and to John Blake for moderating.

Mom, Dad, Alex and Lisa took care of me and indulged me being detached and boring for the last year. Many other people, including Josh Cohen and Simon Berrebi, deserve thanks on that second point as well.

Simon Berrebi also deserves thanks for reading drafts of Chapters 1, 3 and 4, and making helpful comments.

Max at the Dalhousie GIS Center taught me how to use ArcGIS, and Andrew Palmer helped me get the prescription generator started.

Finally, I'd like to the acknowledge the IE grad students at Dal, particularly Sara Rezaee and Mahshid Jessri for distracting me and making me look forward to coming to school, and Sina Sadaatyar, Pegah Sohrabi, and Sharif Islam for being good role models.

Chapter 1

Introduction

1.1 Background

There is a strong case that forest management strategy in Canada should be modelled at the spatial resolution at which it applies. Nationally, the Canadian Council of Forest Minister's Criteria and Indicators (CCFM C&I) define the principles that a forest management strategy must address to be sustainable; many of these cannot be separated from spatial specifications. Then, provincially, the actual management strategies are determined based upon what the CCFM C&I mean in a policy context. Ecosystem Based Management (EBM) is widely used as the framework around which strategy is constructed so that it is regionally relevent and nationally coherent. EBM metrics address the C&I, and as such many are defined by spatial specifications. Managers employ models of forest management strategy to provide them with feedback on their strategies. However, commonly used models' solution time increases exponentially as spatial detail is modelled, suggesting they are not appropriate for modelling forest management strategy, and motivating an investigation into alternative models with which to evaluate strategy aimed at EBM and the CCFM C&I.

A strategic objective of forest managers is to determine management strategies that are sustainable and profitable. Strategies are developed using the managers' experience, principles of forestry, and decision support models. Experience and knowledge of forestry principles are crucial to this process because forest management strategy is defined regionally: a strategy that is reasonable for a Boreal forest in Northern Quebec will not necessarily be reasonable for an Acadian forest in Western Nova Scotia. Decision support models can compliment the experience and knowledge of managers by providing feedback as to how well their strategies might achieve the strategic objective of sustainability and profitability.

In Canada, the CCFM C&I [CCFM, 2003] comprise a national definition of sustainability in forest management. Canadian public forests are to be managed under the indicators that define six criteria: 1. Biological Diversity, 2. Ecosystem Condition and Productivity, 3. Soil and Water, 4. Role in Global Cycles, 5. Economic and Social Benefits, and 6. Society's Responsibility. An example of a characteristic indicator, from the Ecosystem Condition and Productivity criterion, is 'Area of forest disturbed by fire, insects, disease, and timber harvest'. This indicator is not prescriptive towards strategy; it is open to interpretation. Depending on the forest and prevailing economic conditions many different strategies can address it.

Observe that three of the six criteria 1, 2, and 3 are explicitly spatial. It is not possible to manage for the indicators of these criteria without having knowledge of, e.g., where animal habitat is (1), where old growth forest is (2), and the level of forest-cover in watersheds (3). The CCFM C&I are defined broadly at the ecozone level, but management that addresses them is applied provincially, at the ecoregion or ecodistrict level. In Nova Scotia, and other provinces, the means of managing for the CCFM C&I is EBM [Stewart and Neilly, 2008]:

> Ecosystem Based Management is a conservation planning approach that considers the composition, structure, functions, and processes of ecological communities occurring within a landscape. It uses a reference to the characteristics of ecological communities that occurred in an area historically, and strives to provide representation of these ecological communities while integrating human economic and social demands.

EBM in Nova Scotia means that provincial forests are to be managed so as to achieve certain ecosystem condition goals; these goals are based on metrics for ecosystem conditions of interest, such as marten habitat, measured by the area of old growth in a particular ecodistrict, and water quality, measured by the level of young forest cover in a particular watershed [Bowater, 2010]. Both marten habitat and water quality are measured using metrics that are defined by strata: old growth vegetation is measured in an ecodistrict, and young forest cover in a watershed. A strata is a layer of land division. For example a forest could be divided by ecodistricts—one strata—watersheds—another strata—and transportation costs—a third strata. Many ecosystem metrics are defined at one or more strata, so to determine a particular timber stand's contribution to one of these metrics it is thus necessary to know the ecodistrict, watershed, or other relevant strata to which it belongs.

In addition to Canada's commitment to the CCFM C&I, its three certifications of forest sustainability [NRCAN, 2013], and leading role in the Montreal Process [Montreal-Process, 1998] underscore the value Canadians place on sustainably managing forests. However, profitability of timber harvests cannot be separated from sustainability. Profitability is incorporated into the CCFM C&I as Criterion 5: Economic and Social Benefits. Profitable management strategy is spatially defined; for example, access and transportation costs depend on where wood is coming from and where it is going. High-grading, the act of debasing the future value of the forest by harvesting only the most valuable timber today, can result from a strategy that overlooks the spatial-economic component of forest management.

Presently, Model Two (M2) [Johnsson and Scheurmann, 1977] linear programming (LP) models are popular around the world for use in forest management strategy development. Unfortunately, these models become large when modelling many strata. Each additional strata included in an M2 LP causes the number of model variables and constraints to increase substantially. As a result, solution time increases exponentially. This means M2 is not well suited to modelling forest management strategy that is defined by strata, such as EBM.

If LP models are to provide feedback as to the sustainability and profitability of management strategy, it is desirable that they can model an environmentally and economically relevant level of spatial resolution, represented by strata. For this reason, despite the popularity of M2 strategic models, an alternative framework, known as Model One (M1), may be more suitable for modelling forest management strategy.

In an M1 framework model size, and solution time, does not increase exponentially as strata are modelled. Of further interest, it is not complicated to include industry representation, via transportation costs and product demand, in M1 models. In many forests there are stands with questionable economics of harvesting, because they have species for which there is little demand, or their wood is too expensive to harvest and ship. Models that do not take industry into account overlook that these stands are of little value, and in doing so may present an unrealistic representation of what might be sustainable harvest levels. By including industry representation, a model may present the user with more realistic feedback.

1.2 Thesis Overview

This thesis consists of five chapters in addition to this introduction.

Chapter 2 is a literature review that will provide the reader with an introduction to hierarchical linear programming models of forest management strategy. Particular attention is paid to past efforts to model spatial resolution and industry representation. The reader should come away with an understanding of what strategic level forest models are, how they fit into a hierarchical planning framework, and some of the advances in strategic level modelling made in the last twenty years.

Chapter 3 is entitled 'Modelling'. In this chapter Model One and Model Two are analyzed for modelling multiple strata. This analysis shows that Model Two model size, and as a result, solution time, increases exponentially as additional strata are modelled, while Model One model size and solution time increases linearly. In the second section of this chapter, the formulation is presented for a linear programming framework in which models built can include industry representation and are capable of modelling multiple strata.

In Chapter 4, a case-study on Nova Scotia's Crown Central Forest for the purpose of comparing Model One and Model Two is presented. The framework given in Chapter 3 was used to build Model One models, and Woodstock™was used to build Model Two models. It is found that when modelling multiple strata, using similar prescription sets, Model One obtains solutions in substantially less time. For example, the model with the highest spatial resolution is solved by Model One in 2574 seconds and Model Two in 20,480 seconds with objectives of 4.126×10^7 and 4.119×10^7 respectively.

In Chapter 5, a second case-study is presented on the Crown Central Forest. This time for the purpose of investigating including industry representation in models of forest management strategy. The study consisted of two parts. First, a model that includes industry was compared against a conventional model; it was found that the conventional model, by ignoring industry, scheduled unprofitable timber for harvest. In the second part, examples of analysis using models that include industry representation are provided. The first example attempts to assess the cost of instituting a clearcut restriction policy; the second introduces two profit-based harvest regulation strategies; the third investigates an industrial expansion scenario; and, the fourth allows harvested wood to be left in the forest instead of being shipped to a mill.

Chapter 6 reports on some conclusions. Limitations in the conventional approach to forest management strategy modelling are reviewed, findings based on the two case-studies are summarized, and ideas are put forth to further the research herein contained; for example, a suggested research project is to develop a simplex-based algorithm suited to these models' special structure.

Chapter 2

Literature Review

The following literature review will provide the reader with an introduction to hierarchical linear programming models of forest management strategy. Operational research applied to forestry is discussed, then the review focuses in on hierarchical forest management models. This leads into the third section where Model One, Two and Three are introduced. After a comparison of these three modelling frameworks, past efforts to model industry and environmental considerations in strategic level hierarchical models are presented.

2.1 Operational Research in Forestry

Forestland covers 54% of the Canadian landmass [NRCAN, 2012]; it forms the backbone of natural ecosystems, and forest industries provide the lifeblood for regional and national economies [Richards, 1997]. Forests also improve the quality of life of Canadians who use them for recreational purposes and appreciate their natural majesty. To maintain the benefits we currently derive from our forests, it is essential that they are judiciously managed.

Currently, operational research (OR) techniques are widely employed by forest managers in their pursuit of extracting the maximum economic potential of the forest within a context of achieving or maintaining natural ecosystems and aesthetic values [Weintraub and Bare, 1996]. This pursuit is a complex undertaking since its achievement is dependent on decisions made at several different levels of spatio-temporal detail. OR, the application of mathematics to decision making problems, compliments these decisions by providing a set of tools that can be used to help forest managers evaluate substantially more of the myriad factors that contribute to forest policy than they would otherwise be capable of considering. For instance, Linear Programming (LP), a technique commonly applied to model long-term forestry decisions, enables forest managers to consider the interactions between millions of variables and constraints. The application of LP and other OR techniques is a 20^{th} century development in forest managers' history of utilizing quantitative methods to accomplish management objectives.

Forests have been managed quantitatively since the early nineteenth century work on Forest Rent Theory by Hundeshagen, see Plochmann [1989]. In the mid-nineteenth century Faustmann determined optimal rotation timings for stand harvesting [Faustmann, 1849]. His computational method, which establishes the harvest age that maximizes the value of a stand, results in a sustainable harvesting regime over an infinite time horizon [Gunn, 2007]. In 1947, the discovery of the Simplex method [Dantzig, 1963] enabled operational researchers to efficiently solve Linear Programs. This advancement contributed to the development of quantitative forest management as it was soon observed that forest management decisions are amenable to modelling within an LP framework [Curtis, 1962], facilitating the modelling of decisions previously thought to be computationally intractable. The forest industry embraced LP modelling techniques in the 1970s by implementing systems such as TimberRAM [Navon, 1971] and MAXMillion [Ware and Clutter, 1971]. To this day, LP remains widely used as an aid to forest industry decision makers [Gunn, 2010]; though, not all experiences have been positive [Kent et al., 1991]. In addition to LP, other OR techniques, such as simulation [Robak and Richards, 2001; Baskent and Keles, 2005], and the Analytical Hierarchy Process (AHP) [Diaz-Balteiro and Romero, 2008; Ananda and Herath, 2009], have been applied to support forest management decisions. Appropriately, it has been remarked that few industries have adopted OR with the enthusiasm of the forest industry [Martell et al., 1998].

2.2 Hierarchical Planning

That the forest industry has sustained enthusiasm for OR for almost half a century can in part be attributed to the ease with which forest management decisions fit into a hierarchical planning framework. As mentioned earlier, forest management decisions take place at several levels of spatio-temporal resolution. For instance, periodic harvest volumes are often determined at an aggregated stand level of spatial detail, while bucking decisions are made at a sub-stand level of detail [Weintraub and Romero, 2006]. Similarly, planning horizons for rotation decisions can be longer than one hundred years, while those for harvest operation decisions may be shorter than six months [Ronnqvist, 2003]. This spatio-temporal stratification makes forest management decisions suitable to be modelled within a hierarchical planning framework [Weintraub and Davis, 1996]. Such a framework facilitates the division of spatially and temporally heterogeneous forest management decisions into spatially and temporally homogeneous levels, enabling the development of separate models that each model a particular spatial and temporal resolution [Weintraub and Davis, 1996]. The separate models can be connected with linkages that can be top-down [Weintraub and Cholaky, 1991], bottom-up [Gustafson et al., 2006], or a combination of the two [Kurttila et al., 2001]. Through these links, solutions from one model can be fed up or down, to constrain or inform, the model preceding or following it. In this way, the results of modelling are encouraged to be congruent at all spatio-temporal levels of decision-making. This review is concerned with top-down hierarchical models.

It is standard practice to denote the three levels of the planning hierarchy strategic, tactical, and operational [Weintraub and Bare, 1996]. Before moving to discuss strategic level models in detail, an overview of the operational and tactical levels of forest management decision-making and modelling is appropriate. Spatial and temporal resolution increases from strategic to tactical and again from tactical to operational levels. At the highest spatio-temporal resolution, the operational level is concerned with scheduling the day-to-day activities required to harvest a timber stand, such as felling and bucking. Spatial detail is at the sub-stand level and planning horizons are set at approximately a year [Ronnqvist, 2003]. Operational models can employ Linear Programming

[Epstein et al., 1999], Dynamic Programming, and Network Models [Marshall, 2007].

Operational decisions are preceded by tactical level decisions. These typically schedule the stands to harvest in each year in order to meet management objectives. Often the tactical level's unit of spatial resolution is the stand and its time horizon is less than 50 years [Richards and Gunn, 2000]. Many different approaches have been applied to modelling tactical level decisions: Heuristics [Richards and Gunn, 2000; Weintraub et al., 1994], Simulation [Gustafson et al., 2006; Covington et al., 1988], and Integer Programming [Constantino et al., 2008]. In tactical level models, spatially defined constraints such as maximum opening size, and green-up are taken into account [Weintraub and Bare, 1996]. Linked to the tactical level, concerned with the lowest spatio-temporal resolution, is the strategic level of decision-making.

At this level, schedules of interventions that maximize the Net Present Value (NPV) of the forest, while maintaining measures of sustainability, are determined [Gunn, 2010]. Often timber stands are aggregated into multi-stand management units, and time is aggregated into multi-year periods [Ronnqvist, 2003]. Traditionally, LP has dominated the modelling of this level [Gunn, 2007] though related formulations, such as Mixed Integer Linear Programs (MILP), are common [Weintraub and Navon, 1976; Cea and Jofre, 2000; Snyder and ReVelle, 1997]. A notable characteristic of strategic level LP models is that they are often aspatial [Martell et al., 1998].

An advantage of modelling within the planning hierarchy is that it facilitates the construction of models that yield comprehensible results. It provides a framework in which models can individually consider a particular spatio-temporal level of decision-making, while taking into account the decisions being made at the other levels of resolution [Martell et al., 1998]. The results of models concerned with a single level of spatio-temporal resolution are often easier to understand than those of models that attempt to tackle multiple levels of resolution simultaneously. The latter type of model has been given the sobriquet "Monolithic", reflecting how intimidating model size and results are to a user [Weintraub and Cholaky, 1991]. Modelling within a hierarchical planning framework has received considerable attention in the literature.

For over twenty years, models built within hierarchical frameworks have been employed to aid forest decision-makers manage forests. Vertinsky et al. [1994] coupled a strategic LP with a forest estate simulator and used a resource management GIS to visually evaluate harvest patterns. Weintraub and Cholaky [1991] linked an MILP strategic model that determines optimal harvest volumes for geo-zones to an MILP tactical model that disaggregates the strategic volumes and schedules harvests for each zone independently. Nelson et al. [1991] linked a strata-based LP with an area-based Monte Carlo Integer Program (MCIP) to generate solutions that are spatially and temporally coherent between models. Demonstrating the complementary nature of LP and simulation, the FOLPI system joined a strategic LP model with a forest estate simulator [Garcia, 1984]. Despite both being used to model strategic forest management decisions, Nelson et al's LP and that employed in FOLPI appear very different upon inspection. Indeed, Nelson et al's is modelled within a Model One framework and that of FOLPI within a Model Three framework.

2.3 Model One, Model Two, Model Three

Strategic LP may be categorized as model types One, Two, or Three [Johnsson and Scheurmann, 1977; Gunn and Rai, 1987]. In the literature, Models One and Two have received the most attention [Martell et al., 1998]. Decision variables in Model One (M1) represent the number of hectares from each stand to allocate to each prescription under consideration [Davis et al., 2001]. A prescription is a schedule of forest interventions. Examples of M1 include Heureka [Wikström et al., 2011] and the MAXMillion software [Ware and Clutter, 1971]. There are two defining characteristics of M1 models. First, an M1 framework is stand-based; second, M1 is usually limited by the number of prescriptions it attempts model [Davis et al., 2001].

In order to consider a wider selection of prescriptions, Model Two (M2) formulations combine stands into a regeneration class upon harvesting, this is often referred to as stand aggregation. The decisions to be made in the model are in what period, b, will the stands regenerated in period, a, be harvested. As a result, stand identity is lost once a stand is harvested since all stands, of a given class, initially harvested in period b are combined [Davis et al., 2001]. Unlike M1, LP size grows linearly as prescriptions are added. Remsoft's Woodstock [Cogswell and Feunkes, 1997] is a popular commercial M2 system and ForPlan Version II included M2 modelling [Kent et al., 1991].

Similar to M2, the decision to be made in Model Three (M3) in each time period is to harvest and regenerate a given age class, or allow it to mature for another period. The difference between the two modelling frameworks lies in initial stand aggregation and the tracking of forest state. In M3 all stands of a given age class, sharing silvicultural attributes, are combined from the outset, and the state of forest is reported every period. The fact M3 reports on forest state every period makes it suitable for modelling forest disturbances such as fire [Boychuk and Martell, 1996]. FOLPI [Garcia, 1984], and SilviPlan [Davis and Martell, 1993] employ strategic M3 LP. In light of the preceding discussion, the level of aggregation and number of prescriptions evaluated can be viewed as the characteristics that define the three modelling frameworks.

That M1 does not combine stands upon harvest is significant because M2/M3 LP can become unwieldy when modelling spatial constraints, due to their stand aggregation. Spatial constraints are used to model ecosystem conditions that are often based on land stratification, e.g. age-class distributions in ecodistricts, or forest cover levels in watersheds. With each additional strata modelled, the number of rows in an M1 LP increases linearly, while the number of rows and columns grow exponentially in M2/M3 LP — M3 grows larger faster as a result of its greater stand aggregation. Similarly, with each additional prescription modelled, M1 size increases exponentially while M2/M3 size increases linearly. These two distinctions between model types, prescriptions and stand aggregation, underscore that despite being used by forest managers for the same purposes, the three LP strategic modelling frameworks approach determining schedules of sustainable, profit maximizing forest interventions, differently.

It must be noted, however, that the definition of sustainable forest interventions is a contentious issue. In North America, Non-Declining Yield (NDY) has been one of the forest industry's measures of sustainability since 1960 [U.S. Congress, 1960]. NDY means that the volume of timber harvested from a forest in the first year of management does not decrease in subsequent years.

Questions have been raised as to whether NDY is the best measure of sustainability [Gunn, 2010; Howard, 2001]; particularly, it has been noted that NDY harvest is not equivalent to economically sustainable harvest. Schedules based around NDY can include unprofitable timber on whose regeneration future years' harvests are dependent. Often, this unprofitable timber is not harvested, thus invalidating the NDY calculation [Paradis et al., 2013]. Constraining harvest volumes to fit NDY can also lead to under utilization of the timber resource. For example, profitable old growth may be left unharvested in initial years if the resulting volumes cannot be maintained in future years. Another common proxy for sustainability, Even-Flow (EF), is subject to a similar tendency towards resource under-utilization [McQuillan, 1986; Pickens et al., 1990]. An alternative to timber flow constrained management is Area Control, an easily understood method for forest managers to hedge against setting Annual Allowable Cuts that exhaust, or greatly under utilize the timber resource [Leak, 2011]. Using Area Control, with no flow constraints, allows old growth that would be left unharvested under NDY or EF, to be harvested while the forest is transitioning to a regulated state [Davis et al., 2001]. Once in a regulated state, the forest would produce a non-decreasing, sustainable, flow of timber [Howard, 2001].

2.4 Industrial and Environmental Modelling

The canonical LP model formulations include constraints to account for stand area, non-declining timber flow, and possibly measures of habitat or recreation conditions. However, often due to model size, industrial and environmental considerations must be omitted from strategic level models. In order to better reflect reality, there have been experiments to incorporate these considerations in strategic models.

Industrial considerations are relevant to strategic forest management decisions since the value of a particular forest product is dependent upon the demand for that product and the cost of getting it to that demand. To accurately represent the value of the resources being modelled, the models of Barros and Weintraub [1982] included industry representation, and Gunn and Rai [1987] considered product demand by approximating demand curves at demand centres; more recent examples of demand curve approximation in a forestry context are FPL-PELPS [Lebow et al., 2003], and the demand modelling in SPECTRUM [Greer and Meneghin, 2002]. In a divergent attempt to model forest decisions in a supply chain context, Cea and Jofre [2000] included the decision of whether to open, close, or expand industrial complexes in their MILP model. These examples demonstrate that it is feasible to include industrial considerations in strategic level LP. The other set of considerations often ignored, environmental considerations, have received greater attention, in large part due to public concern over forest ecosystem conditions.

Exemplified by the Montreal Process [Montreal-Process, 1998] a trend in the forest industry over the last two decades has been the shift from managing to achieve solely financial objectives, to managing to achieve financial and ecosystem objectives [Martell et al., 1998; Bettinger and Chung, 2004]. To keep pace with public demands, strategic forest models have been constructed to reflect an ecosystem based management style. The LP of Naesset [1997] and Vertinsky et al. [1994] both included ecosystem considerations by incorporating riparian buffer zones in their models. Ohman and Eriksson [1998] investigated wildlife habitat quality as measured by their 'core area' concept, and the optimal aggregation patterns of harvest blocks were examined in Ohman and Eriksson [2010].

Coinciding with the shift in the industry towards ecosystem based management has been increased attention directed towards strategic level spatial simulation models. The appeal of these models is that they acknowledge that the shape and distribution of forest habitats have an affect on the development of the forest. To approximate the future state of the forest, considerations such as non-fragmented old growth vegetation and vegetative corridors for wildlife are modelled [Baskent and Keles, 2005]. Examples of these models include Patchworks [Rouillard and Moore, 2008] and HARVEST [Gustafson and Crow, 1996].

Spatial simulation models provide decision makers with insight into the effects of a particular management regime by simulating the development of the forest in response to a given schedule of forest interventions [Baskent and Keles, 2005]. Herein lies the difference between strategic level

LP and simulation models. LP models are used to determine the optimal management decisions for given management objectives whereas simulation models are used to simulate the effects of proceeding with a particular set of management decisions. For this reason, the two types of strategic models can be seen as complimentary. The Swedish Heureka [Wikström et al., 2011] and Finnish Simo [Rasinmäki et al., 2009] demonstrate this relationship.

In 1994 the Montreal process convened, setting sustainability as a priority in forest management around the world. Given the resulting shift towards Ecosystem Base Management, it is likely researchers will be compelled to design new systems to keep pace with the increasingly complex objectives that accompany ecosystem based management. It seems that higher resolutions of spatial detail incorporated at the strategic level of hierarchical models will be necessary for these models to maintain relevance in the forest industry.

In closing, it is fitting to reflect on Gunn [2010] and emphasize that the purpose of these models is not to determine strategy but to explore the possible outcomes of pursuing a particular strategy. Davis and Martell [1993] take a similar view when they state that their model, SilviPlan, "provide[s] forest managers with insight into silvicultural decision making problems through experimentation and exploration." Models cannot supplant thinking managers. As demands on forest decision-makers become more complex, and reliance on models increases, it will be ever more important to remember this.

Chapter 3

Modelling

The purpose of this chapter is to demonstrate that currently used models may not be appropriate for modelling forest management strategy at strategically relevant spatial resolution, and then advance a model that is suited to this task. This chapter is divided into two sections. The first section discusses model size and solution time of Model One and Model Two when modelling strata. It is shown that M2 model size increases substantially, leading to exponentially increasing solution time, as the number of strata modelled increases. The second section presents an M1 formulation that includes industry representation and is capable of modelling multiple strata. This framework is used to build models throughout the rest of this thesis.

3.1 Modelling Framework Comparison

Linear Programming (LP) models are used to model forest management strategy to provide managers with feedback as to the sustainability and profitability of their strategies. It is desirable that these models can model a strategically relevant level of spatial resolution. Model Two (M2) model size, and thus solution time, increases exponentially—both the number of LP rows and columns increase substantially—as additional strata are modelled, while Model One (M1) size and solution time increases linearly—only the number of rows increases. For this reason an M1 framework may be more suitable for modelling forest management strategy than M2.

Before examining an M1 model, it will be helpful to understand why M2 model size and solution time increases exponentially when they model multiple strata. An M2 LP formulation is given in Figures 3.1 and 3.2.

Sets

I	Stands
K	Interventions
T	Periods
S	Spatial Strata

Parameters

$area_i$	$i \in I$	area of stand i
U		ending inventory condition
c_{iak}	$i \in I, a \in T, k \in K$	benefit of harvesting i in a using k
d_{abk}	$a \in T, b \in T, k \in K$	benefit of harvesting area regenerated in a in b using k
ζ_a	$a \in T$	ending inventory condition for stands harvested in a

Variables

y_{abk}	$a \in T, b \in T, k \in K$	Area regenerated in a and harvested in b using k
x_{iak}	$i \in I, a \in T, k \in K$	Area of i harvested in a using k
u_a	$a \in T$	area regenerated in a, not harvested again

Figure 3.1: Model Two LP Formulation: Sets, Parameters and Variables

Objective

$$\max \sum_{\substack{i \in I, a \in T \\ k \in K}} c_{iak} x_{iak} + \sum_{\substack{a \in T, b \in T \\ k \in K}} d_{abk} y_{abk} \tag{3.1}$$

Constraints

$$\sum_{i \in I, k \in K} x_{iak} + \sum_{\substack{b \in T, k \in K \\ b < a}} y_{bak} = \sum_{\substack{f \in T, k \in K \\ f > a}} y_{afk} + u_a \qquad a \in T \tag{3.2}$$

$$\sum_{a \in T} \zeta_a \cdot u_a >= U \tag{3.3}$$

$$\sum_{a \in T, k \in K} x_{iak} = area_i \qquad i \in I \tag{3.4}$$

Figure 3.2: Model Two LP Formulation: Objective and Constraints

The variables x_{iak} represent the area of stand i harvested in period a using intervention k. The variables y_{abk} represent the area regenerated in period a, and then harvested again in period b, using intervention k. The parameters c_{iak} and d_{abk} represent the benefit accruing from harvesting stand i in period a using intervention k, and harvesting a hectare in period a then again in period b using intervention k, respectively. The Network constraints (3.2) ensure the area harvested in period b is regenerated and harvested again in period f or allowed to remain unharvested and pass into u_a. The Ending inventory constraints (3.3) ensure a certain harvestable area remains at the end of the planning horizon. And, the Area constraints (3.4) ensure the area harvested from each stand is equal to the area covered by that stand. Together, these constraints define an acyclic network: in Figure 3.3 [Gunn, 2010] the x_{iak} and y_{abk} variables from Figure 3.2 correspond to the vertical and horizontal axes respectively.

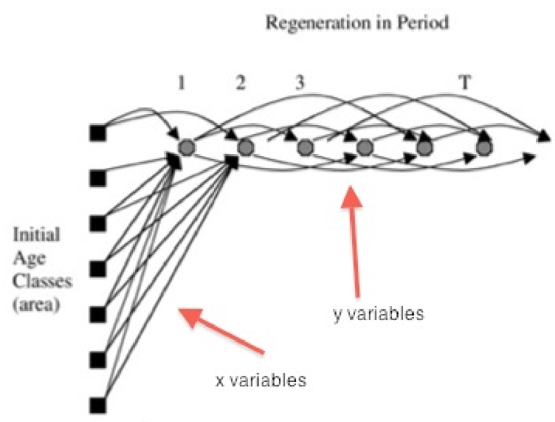

Figure 3.3: Model Two Acyclic Network

Notice that after first harvest stand identity is lost. In an M2 model, stands are combined by harvest period and intervention method. This makes dividing the forest by strata computationally expensive to model, since additional network constraints are needed to keep the area from separate strata from mixing together. An example will prove illustrative. Imagine a manager who wants to maintain 70% forest cover in each of the five watersheds that divide her forest. Using an M2 formulation she will need to introduce variables to the formulation given in Figure 3.2 that indicate the watershed to which a given stand or unit of area belongs; these variables will define a separate network for each watershed. For this reason, the new model, Figure 3.4, will be much larger and take longer to solve than the model in Figure 3.2.

$$\max \sum_{\substack{i\in I,a\in T\\k\in K,w\in S}} c_{iakw}x_{iakw} + \sum_{\substack{a\in T,b\in T\\k\in K,w\in S}} d_{abkw}y_{abkw} \tag{3.5}$$

subject to

$$\sum_{i\in I,k\in K} x_{iakw} + \sum_{\substack{b\in T,k\in K\\b<a}} y_{bakw} = \sum_{\substack{f\in T,k\in K\\f>a}} y_{afkw} + u_{aw} \qquad a\in T, w\in S \tag{3.6}$$

$$\sum_{a\in T} \zeta_{aw}\cdot u_{aw} >= U_w \qquad w\in S \tag{3.7}$$

$$\sum_{a\in T,k\in K} x_{iakw} = area_i \qquad i\in I, w\in S \tag{3.8}$$

$$\sum_{a\in T,k\in K} q_{abkw}y_{abkw} \geq 0.7Area_w \qquad b\in T, w\in S \tag{3.9}$$

Figure 3.4: Model Two LP Formulation with Watershed Constraints

There are two differences between the model in Figure 3.4 and that from Figure 3.2. First, the variables are now x_{iakw} and y_{abkw} where w indicates the watershed to which the particular stand or harvested area belongs. Realistically, area from one watershed cannot pass into another, so to reflect this in the model it is necessary to have five networks where previously there was only one. This means the watershed model will have five times as many transfer variables, y_{abkw}, and network constraints, set 3.6. Second, a new constraint set has been added (3.9): at least 70% of each watershed's area must qualify as cover condition—the q_{abkw} parameter represents area regenerated in period a then harvested in period b using intervention k belonging to watershed w that qualifies as forest cover. These constraints mean that the new model not only has more networks than the previous one, but these networks are more constrained as well.

A rough calculation will illustrate how the number of variables and constraints grow when additional strata are modelled. For a 30 period model, estimate that there are 100 feasible combinations of regeneration and harvest periods, and assume there are 5 different intervention methods, k, and one strata, w. Such a model would have 500 y_{abkw} variables and 30 network constraints. If the forest is divided into five watersheds then there are now 2500 y_{abkw} variables and 150 network constraints. If on top of that the forest is divided into eight ecodistricts, then, depending on the overlap between ecodistricts and watersheds, there could be up to 20,000 y_{abkw} variables and 1200 network constraints—where w now represents the watershed and ecodistrict to which the area belongs. Moreover, it is likely that the forest will also be divided by species, site-class, stocking level, age, and other considerations such as whether land belongs to a riparian zone, or is on a steep slope. Each of these divisions will further increase the number of y_{abkw} variables and the number of network constraints, leading to a model that could be intractably large.

This need for additional networks and variables for additional strata in M2 is a result of the model combining stands upon harvest. Encouragingly, M1 formulations do not combine stands, Figure 3.5.

Sets

I	Stands
P	Prescriptions
Y	Yields
T	Periods
W	Watersheds

Parameters

$area_i$	$i \in I$	area of stand i
y_{ijkt}	$i \in I, j \in P, k \in Y, t \in T$	yield of type k in t from i under j
c_{ij}	$i \in I, j \in P$	benefit of applying j to i

Variables

x_{ij}	$i \in I, j \in P$	Area of i assigned to j
Q_{kt}	$k \in Y, t \in T$	Yield of type k to be achieved in t

Objective

$$\max \sum_{i \in I, j \in P} c_{ij} x_{ij} \tag{3.10}$$

Constraints

$$\sum_{j \in P} x_{ij} = area_i \qquad i \in I \tag{3.11}$$

$$\sum_{i \in I, j \in P} y_{ijkt} x_{ij} \geq Q_{kt} \qquad k \in Y, t \in T \tag{3.12}$$

Figure 3.5: Model One LP Formulation

The variables x_{ij} represent the area of stand i managed under prescription j. The Area constraints (3.13) ensure the area of stand i given to all prescriptions does not exceed the area of stand i. The yield constraints (3.14) ensure the yield of type k, generated by applying prescription j to stand i, is at least Q_{kt} in each period t. Q_{kt} could be any variable quantity, such as last period's harvest volumes, total area harvested, or forested area in a particular strata. The yield parameters, y_{ijkt}, represent the yield of type k from stand i under prescription j in period t. These could be harvest volumes, development class, sawlog ratio, or any other yield value. For example, one way to model forest cover in a watershed is to define yield parameters, q_{ijwt}, such that q_{ijwt} is 1 if stand i, under prescription j satisfies cover conditions for watershed w in period t, and 0 otherwise. This can be incorporated into a constraint as follows:

$$\sum_{i \in I(w), j \in P} q_{ijwt} x_{ij} \geq 0.7 area_w \qquad w \in W, t \in T \qquad (3.13)$$

These constraints (3.15) ensure that the area of forest cover in each watershed, w, in each period, t, does not fall below 70% of the watershed's area. By defining appropriate yield parameters, the model can include constraints for strata based conditions without becoming unreasonably difficult to solve. This is because each strata based constraint only requires additional rows in an M1 framework, while in M2 they require additional rows and columns; as a result, in M1 model size and solution time only increase linearly as strata are added.

The greater modelling flexibility and theoretically faster solution times associated with M1 come at a cost however. Increasing the number of prescriptions under evaluation makes M1 models more difficult to solve. A prescription is a series of interventions spanning the planning horizon. In M1 adding a new prescription necessitates adding a new x_{ij} variable for each stand to which it applies. To demonstrate how model size grows with prescriptions, consider the following. It would not be unusual for a forest being modelled to contain 100,000 stands. If each of the stands had 10 prescriptions, then the model would have at least 1,000,000 variables; if each of the stands had 20 prescriptions, then the model would have at least 2,000,000 variables.

It is worth noting that M1 and M2 do not model prescriptions the same way. All M1 prescriptions are user defined prior to LP generation. M2 models individual interventions, and constructs prescriptions from these interventions while solving the LP. Like M1, M2 size also increases when additional interventions are modelled, particularly when multiple strata are also modelled. However, very few interventions need to be included in an M2 model to produce a comprehensive set of prescriptions, because, due to model structure, every combination of interventions is available as a prescription. If a model contains more prescriptions then there are more ways to satisfy constraints, so typically M2 models find higher objective function values than similar M1 models.

This difference in prescription modelling makes M2 more susceptible to prescription errors than M1. With M1, the user has to decide which combinations of interventions are reasonable as prescriptions; with M2, the user only has to consider which interventions are reasonable independent of how they fit together as prescriptions. Thus, prescriptions such as "Shelterwood in periods 1 and 3, pre-commercial thin in period 5, commercial thin in period 9, clearcut in period 12, and clearcut in period 20" can arise. That is, prescriptions where each intervention is reasonable, but the combination of interventions does not seem like a prescription a forester would assign to the stand. Another example of an unrealistic M2 prescription is letting stands that start the planning horizon at an advanced age grow for many periods before cutting them, e.g. a stand that starts the planning horizon at 100 years of age is not felled until 10 periods into the model. These unrealistic prescriptions constitute errors in the model. Notably, with M1 such errors are errors of commission, i.e. the user chooses to include an unrealistic prescription, while with M2 such errors are errors of omission, i.e. the user has not programmed constraints to disallow unrealistic prescriptions. The latter error is not difficult to make, while the former is almost impossible to make unconsciously. Further, in M2 it is difficult to determine the proportion of the forest assigned to unrealistic prescriptions.

When reviewing the solution of an M1 model it is possible to look at the x_{ij} variables to see the prescriptions assigned to each stand. When reviewing the solution of an M2 model, it is more difficult to determine the prescriptions assigned to each stand, since area from each stand is combined into y_{abkw} area variables after initial harvest. This compounds the problem of M2 constructing unrealistic prescriptions because it is not possible to review the solution and determine the proportion composed of these prescriptions. For these reasons, M1 is more robust to avoiding prescription errors than M2.

Spatial resolution and prescriptions define the trade-off between M1 and M2. M1 can become large when many prescriptions are added, but its size is relatively unaffected when multiple strata are modelled. On the other hand, M2 can model hundreds of prescriptions, but when strata are modelled it becomes very large, and some of its prescriptions may be unrealistic. For the purposes of modelling forest management strategy, it seems that hundreds of prescriptions may not be necessary; a reasonable set of prescriptions should be all that is needed to give management useful feedback on the sustainability and profitability of their strategy. For instance, a practicing forester when assessing a stand would probably not consider hundreds of possible prescriptions, but rather relatively few based on some simple principles. This suggests that M1 prescription limitations may not hinder its ability to provide feedback useful for evaluating the sustainability and profitability of management strategy. Together with the fact its size and solution time do not increase exponentially as additional strata are modelled, this makes M1 appear to be a more appropriate framework than M2 for modelling forest management strategy.

3.2 A Framework for Models of Forest Management Strategy

The previous section discussed that due to its ability to model multiple strata without exponentially increasing model size and solution times, M1 appears to be a more suitable environment than M2 in which to model forest management strategy. In this section the formulation of an M1 modelling framework is presented in which models built are capable of modelling strategically relevant spatial resolution and include industry representation through modelling shipping costs and product demand.

3.2.1 Framework

The modelling framework shown in Figures 3.6 and 3.7 is based on a model proposed in [Gunn, 2010], and shares its Model One structure with the Swedish system, Heureka [Wikström et al., 2011]; though, Heureka does not include industry representation. The framework as presented here is general; specific instances of it are used in Chapters 4 and 5.

Sets

I	Stands
P_i	Prescriptions for stand i
T	Periods
T_v	Periods in which demand is modelled
Y	Yields
$Y_w \in Y$	Timber yields
$Y_e \in Y$	Non-timber yields
S	All combinations of strata
$I(s)$	Stands Belonging to $s \in S$
R	Timbersheds
$I(r)$	Stands Belonging to $r \in R$
M	Demand Centres

Parameters

y_{ijkt}	$i \in I, j \in J, k \in Y, t \in T$	Yield of type k from i in t under j
Lq_{skt}	$s \in S, k \in Y, t \in T$	Lower bound on k from s and t
Uq_{skt}	$s \in S, k \in Y, t \in T$	Upper bound on k from s and t
dc_{mk}	$m \in M, y \in Y_w$	Conversion of m^3 to $
$area_i$	$i \in I$	Area of stand i
dem_m	$m \in M$	Minimum demand to m
cap_m	$m \in M$	Capacity of m

Variables

x_{ij}	$i \in I, j \in P_i$	Area of i assigned to j
q_{skt}	$s \in S, k \in Y, t \in T$	Total yield of k from s in t
z_{smkt}	$s \in S, m \in M, k \in Y_w, t \in T_v$	Volume of k from s to m in t
p_{mnkt}	$m \in M, n \in M, k \in Y_w, t \in T_v$	Volume of k from m to n in t
d_{mt}	$m \in M, t \in T_v$	Demand supplied to m in t

Figure 3.6: Modelling Framework Formulation: Sets, Parameters and Variables

Constraints

$$\sum_{j \in P_i} x_{ij} = area_i \qquad i \in I \qquad (3.1)$$

$$\sum_{\substack{i \in I(s) \\ j \in P_i}} y_{ijkt} \cdot x_{ij} = q_{skt} \qquad s \in S, k \in Y, t \in T \qquad (3.2)$$

$$Lq_{skt} \leq q_{skt} \leq Uq_{skt} \qquad s \in S, k \in Y, t \in T \qquad (3.3)$$

Shipping Network

$$\sum_{m \in M} z_{rmkt} = \sum_{\substack{i \in I(r) \\ j \in P_i}} y_{ijkt} \cdot x_{ij} \qquad r \in R, k \in Y_w, t \in T_v \qquad (3.4)$$

$$\sum_{n \in M} p_{mnkt} \leq \gamma \cdot \sum_{r \in R} z_{rmkt} \qquad m \in M, k \in Y_w, t \in T_v \qquad (3.5)$$

$$\sum_{\substack{r \in R \\ k \in Y_w}} dc_{mk} \cdot z_{rmkt} + \sum_{\substack{n \in M \\ k \in Y_w}} dc_{mk} \cdot p_{nmkt} = d_{mt} \qquad m \in M, t \in T_v \qquad (3.6)$$

$$\sum_{\substack{r \in R \\ k \in Y_w}} z_{rmkt} + \sum_{\substack{n \in M \\ k \in Y_w}} p_{nmkt} \leq cap_m \qquad m \in M, t \in T_v \qquad (3.7)$$

$$d_{mt} \geq dem_m \qquad m \in M, t \in T_v \qquad (3.8)$$

Figure 3.7: Modelling Framework Formulation: Constraints

The primary decision variables are x_{ij}: the number of hectares of stand i to manage under prescription j. z_{rmkt} represent the volume of wood of type k shipped from timbershed r to demand centre m in period t, and p_{mnkt} represents the volume of secondary product, e.g. chips from sawlogs, of type k shipped from demand centre m to demand centre n in period t. S, contains all combinations of strata being modelled, such as ecodistricts, management units, watersheds, ownerships, riparian buffer zones, etc. Each $s \in S$ represents one combination of strata. Constraints 3.1 are known as area accounting constraints; they ensure that no more than the area of each stand is assigned to prescriptions. Constraint set 3.2 is actually a definition for generalized yield variables, q_{skt}. These variables can represent any quantity of interest, such as spruce-fir volume harvested in each management unit in a given period, or old-growth forest cover in each ecodistrict in each period. Constraints 3.3 describe a general form of yield constraints where the yield value of type k from s in t is lower bounded by Lq_{skt} and upper bounded by Uq_{skt}.

Industrial representation is integrated into the model using the shipping network defined by 3.4-3.8. Demand centres, M, could be any location that exchanges money for wood, such as sawmills, pulpmills, or biofuel refineries. If shipments from each stand to each demand centre were modelled, this would require a large number of variables, so to facilitate modelling, a strata, called timbersheds, has been defined. All stands belonging to a particular timbershed are assumed to have the same transportation costs to all demand centres. Timbersheds are represented here as R, and, though they comprise a subset of S, stating them explicitly facilitates describing the shipping network. Constraint 3.4 states the total wood harvested in each timbershed, r, is equal to all the wood shipped from r to demand centres, m. At demand centre m, wood of type k is converted from cubic meters to dollars using the d_{mk} parameter, constraint set 3.6. Demand centres have fixed capacities based on volume, constraint set 3.7, and must achieve minimum demand levels in dollars, constraint 3.8. The transshipment of secondary products from demand centres to other demand centres is modelled by constraint set 3.5, where γ indicates the proportion of primary wood products that can be shipped for secondary processing.

3.2.2 Comments

Before modelling, stands can be aggregated based on spatial and silvicultural attributes. If two stands belong to the same combination of strata, and share silvicultural characteristics, for strategic purposes they are the same stand. Models built in this framework are linear, so stands can be assigned multiple prescriptions. This is not a concern since if a solution were to be implemented, stands would be split up and assigned different interventions in a higher resolution, tactical or operational, model. In the general formulation provided above, an objective function has not been specified. Many are possible; in Chapters 4 and 5 two examples will be shown: maximizing spruce-fir harvest volume while minimizing deviations to ecosystem constraints, and maximize system wide mill profit while minimizing deviations to ecosystem constraints.

Observe that models built in this framework have a large number of area accounting constraints (3.2). These give LP a predominantly Generalized Upper Bound (GUB) structure. A constraint set is said to be GUB if each model variable is found in one and only one of the constraints [Dantzig and Van Slyke, 1967]. Note that the more stands, or prescriptions, being modelled, the more prominent the GUB matrix in the LP. This creates the potential for models of very large forests to be solved using algorithms that take advantage of GUB structure, e.g. see Yang [2008].

In constraint 3.6 the implication of the equality sign is all harvested wood must be shipped to a demand centre. Similarly, the implication of the inequality in constraint 3.7 is that all secondary products do not need to be shipped. This observation draws attention to the fact that industry representation in this framework is limited and parts of the supply-chain are not considered. In reality, something would always be done with the secondary products that would either incur a cost or generate a profit, thus warranting modelling. Determining how to model the handling of all secondary products would be a useful extension of this thesis.

A final observation. As defined in Figure 3.6, the dc_{mk} parameters are the net of wood revenue minus harvesting cost and transportation costs. Note also that the dc_{mt} parameters allow demand centres to value wood of different types differently, for instance spruce-fir sawlogs might be worth $\$35/m^3$ and hemlock sawlogs might be worth $\$20/m^3$ at the same demand centre.

Chapter 4

Case-Study:

Model One and Model Two

4.1 Introduction

Chapter 3.1 offered a theoretical comparison between M1 and M2 which suggested M1 as a potentially better framework in which to model forest management strategy than M2 due to the fact that as additional strata are modelled, M2 model size increases exponentially. The case-study presented in this chapter is the empirical complement to the theoretical comparison performed in that chapter. Using data for Nova Scotia's Crown Central Forest, an M1 model—based on the model presented in Chapter 3.2—and an M2 model—built using Woodstock™[Cogswell and Feunkes, 1997], a commercially available, widely used, matrix generating interface, were compared for the purpose of modelling forest management strategy at multiple strata. The comparison was done in two phases. In the first phase, restrictions were placed on the M2 model so that both M1 and M2 models had near identical prescription sets. These models were run with four different constraint sets, each set having a different level of spatial resolution. Then, in the second phase, restrictions on M2 prescription generation were removed, and an M1 model with a prescription set approximating that of the new M2 model was constructed. These new models were then run with each of the four constraint sets used in the first phase.

Two results were found. First, as the spatial resolution of the constraints increased, M2 solution time increased exponentially while M1 solution time grew linearly. Second, for these models, the hundred or more prescriptions available per stand in M2 did not enable it to find better solutions than an M1 model with a set of 15-25 prescriptions per stand.

This case-study was made possible through collaboration with the Nova Scotia Department of Natural Resources (NSDNR); they provided the stand table, yield data, and a copy of their Woodstock model. The rest of this chapter will proceed in four parts. First the study forest will be introduced; second, the Model One formulation will be provided, and the study described, third, the results will be presented, and fourth, this chapter will conclude with a discussion of the results.

4.2 The Crown Central Forest

The Crown Central forest covers 379,000 ha, divided among, 3 ownerships, 22 Ecodistricts (Figure 4.2), 24 watersheds (Figure 4.3), and covers 5 counties: Halifax, Hants, Colchester, Cumberland, and Pictou. Figure 4.1 is a map of the forested area of the Crown Central Forest. The forest has a somewhat Gaussian age distribution with many young stands (Appendix A.1). It is part of the Acadian ecozone [Webb and Marshall, 1999], and, in NSDNR strategic modelling, 16 species associations are represented. Most stands are under natural even aged management, but there are managed softwood plantations and uneven age managed stands as well. Crown land is either unlicensed or assigned to the Northern Pulp or Port Hawkesbury Paper licenses. Softwood, specifically Spruce-Fir, makes up the majority of harvests [NSDNR, 2013]. Examples of environmental policies on Crown land in NS include having representative species mixes and age class distributions for each natural disturbance regime [Neilly et al., 2007], limiting harvests in riparian buffer zones, reducing clearcuts to less than 50% of harvests by area [NSDNR, 2011], and selecting 12% of high conservation quality land for protection [NSENV, 2012].

It is worth noting that the crown central forest comprises less than a quarter of the entire central Nova Scotian forest. In 2012, in each of the five counties contained in central Nova Scotia, no more than 40% of total harvest, and more often less than 10%, came off crown land [NSDNR, 2013]. This situation is disadvantageous for modelling forest management strategy defined by strata, such as ecodistricts and watersheds, because in most ecodistricts or watersheds crown land does not make up the majority; so, models for the crown central forest that are able to satisfy management objectives do not tell us if it is possible to achieve these objectives over the entire central forest. This dataset was used because it was available and of interest to the NSDNR.

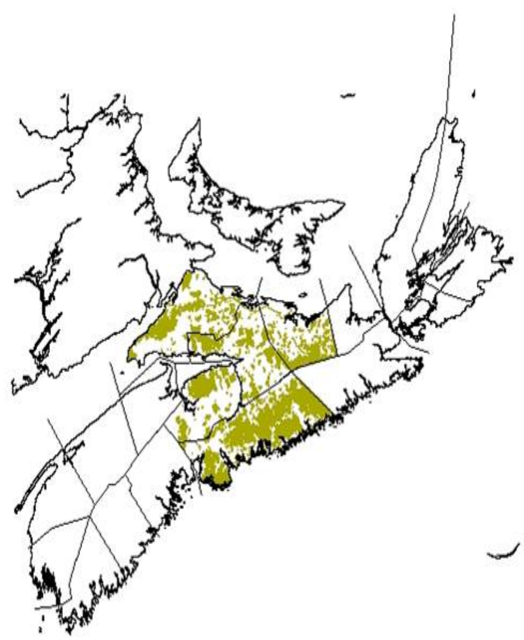

Figure 4.1: The Crown Central Forest Divided by County

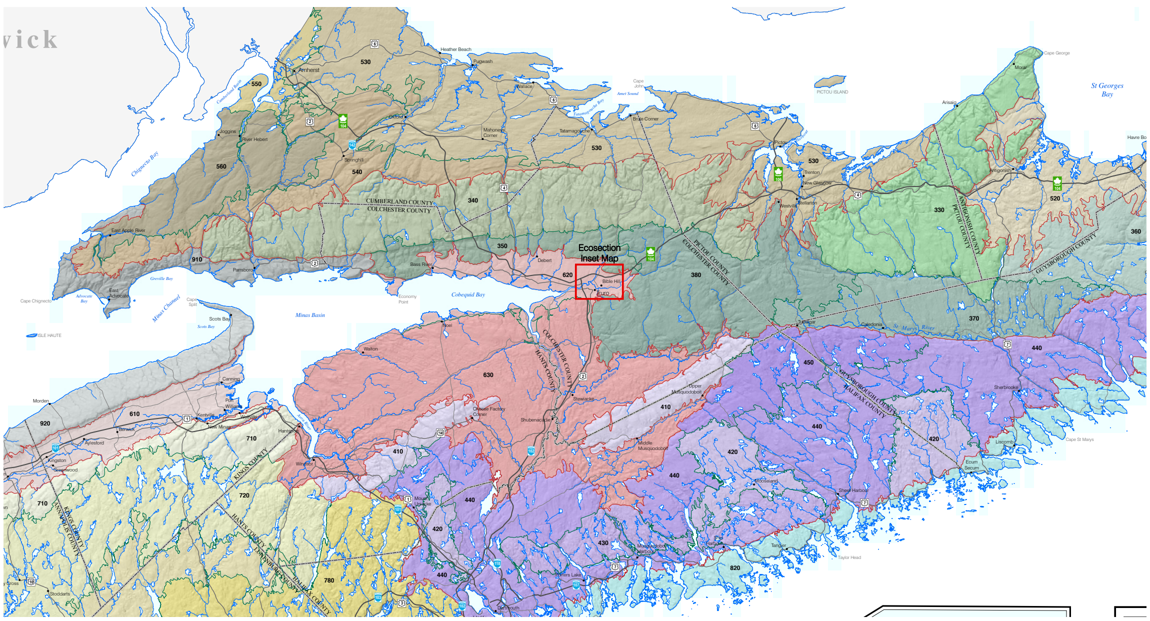

Figure 4.2: Central Nova Scotia's Ecodistricts [NSDNR, 2007]

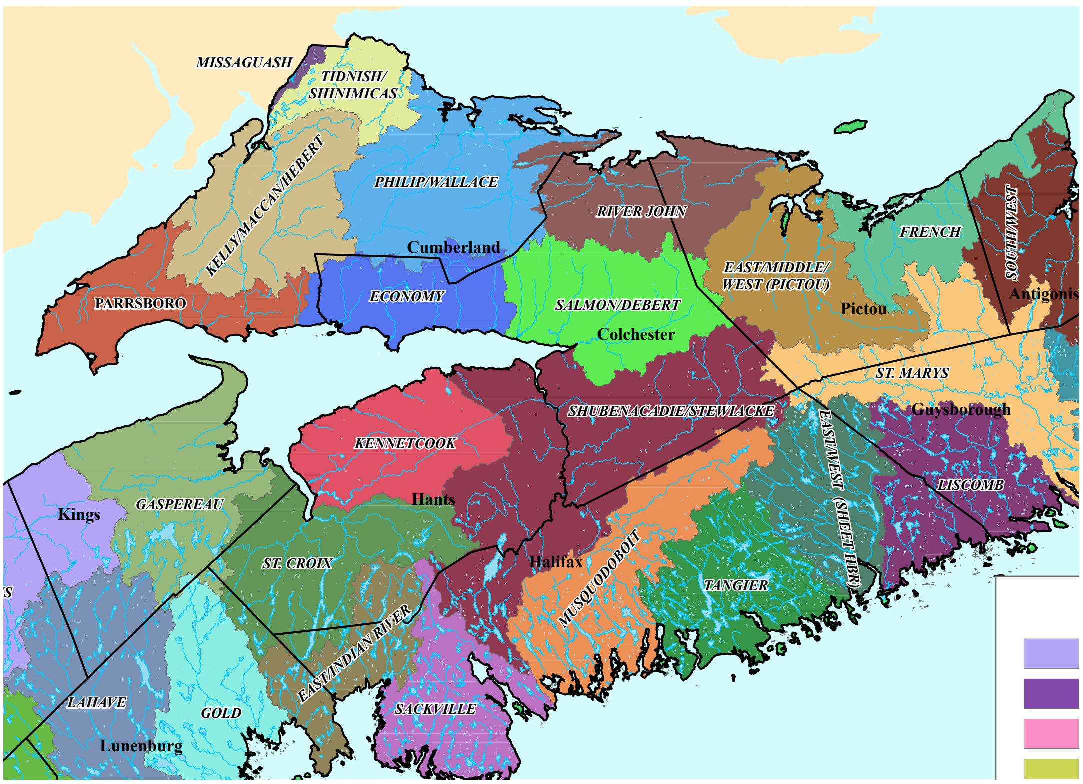

Figure 4.3: Central Nova Scotia's Watersheds [NSENV, 2011]

4.3 Model Formulation

Models are based on an NSDNR Woodstock model; all constraints and parameters come from this model. In this section the M1 model formulation will be presented and the study described. The M1 formulation is given in Figures 4.4, and 4.5. It was written in AMPL [Fourer et al., 1993]; source code for the model can be found in the appendix (A.3).

The M2 model is analogously defined; copies of the files that describe its structure can be found in the appendix (A.4, A.5). All constraint references refer to Figures 4.5. Constraints 4.2 are area accounting constraints. 4.3 states spruce-fir harvest volumes in each ownership, u, must not decrease period on period. Low-value species—as defined by the NSDNR for Central Nova Scotia as intolerant hardwoods, beech, red oak, pine, eastern hemlock, and tamarack larch—are limited to less than 25% of total harvest in constraint set 4.4. Constraint set 4.5 indicates that in the last 19 periods, operable spruce-fir inventory must not decrease. Note that Constraint set 4.5, as well as 4.6, 4.7 and 4.8 have end written in parentheses in Figure 4.5; this indicates that these values are computed at the end of the period, as opposed to harvest volumes which are computed at the beginning of the period.

Constraint sets 4.6–4.8 comprise the ecosystem constraints. Observe that the parameters for these constraints are defined as "0-1", meaning the parameter equals 1 if stand i under j in t satisfies appropriate development class, seral stage or forest cover conditions, and 0 if it does not. 4.6 states that in the last 20 periods the area of forest in each development class d and natural disturbance regime [NSDNR, 2008], n, and ecodistrict, e, should be A_{dn} percent of total area in that ecodistrict and natural disturbance regime. Violations to these constraints are recorded in the J_{dent} variables, which are penalized in the objective function at 120 m^3 per hectare. Constraints 4.7 are similar to 4.6 except with seral stage [Stewart and Neilly, 2008] instead of development class. The G_{cent} variables record violations to these constraints and are penalized in the same way as the J variables. 4.6 and 4.7 are goals; they are not strict constraints. The reason for this is that due to the structure of the forest they are not feasible in every period. Penalty weights, and A and B parameter values,

Sets

I	Stands
P_i	Prescriptions for stand i
$T := 30$	Periods
Y	Yields
$Y_t \in Y$	Timber yields
$Y_e \in Y$	Non-timber yields
U	Ownerships
E	Ecodistricts
N	Natural Disturbance Regimes
W	Watersheds
$I(u)$	Stands Belonging to $u \in U$
$I(n, e)$	Stands Belonging to $n \in N and e \in E$
$I(w)$	Stands Belonging to $w \in W$
D	Development Classes
C	Seral Classes

Parameters

y_{ijkt}	$k \in Y, t \in T$	Yield of type k from i in t under j
$spbf_{ijt}$	$i \in I, j \in P_i, t \in T$	Spruce-fir volume harvested per hectare from i under j in t
$spbfinv_{ijt}$	$i \in I, j \in P_i, t \in T$	Spruce-fir standing inventory per hectare on i under j in t
$other_{ijt}$	$i \in I, j \in P_i, t \in T$	Low-value volume harvested per hectare on i under j in t
$total_{ijt}$	$i \in I, j \in P_i, t \in T$	Total volume harvested per hectare from i under j in t
dev_{dijt}	$d \in D, i \in I, j \in P_i, t \in T$	0-1 i under j is in d in t
ser_{cijt}	$c \in C, i \in I, j \in P_i, t \in T$	0-1 i under j is in c in t
$cover_{ijt}$	$i \in I, j \in P_i, t \in T$	0-1 i under j is in cover condition in t
$area_i$	$i \in I$	Area of i
$area_{en}$	$e \in E, n \in N$	Area of e in n
$area_w$	$w \in W$	Area of w
A_{dn}	$d \in D, n \in N$	Target percentage of forest area in d, and n
B_{cn}	$c \in C, n \in N$	Target percentage of forest area in c, and n

Variables

x_{ij}	$i \in I, j \in P_i$	Area of i assigned to j
J_{dent}	$d \in D, e \in E, n \in N, t \in T$	Hectares of violation to constraint 4.6
G_{cent}	$c \in C, e \in E, n \in N, t \in T$	Hectares of violation constraint 4.7

Figure 4.4: Model One Formulation: Sets, Parameters and Variables

Objective

$$\max \sum_{\substack{i\in I(u)\\ j\in P_i\\ t\in T}} spbf_{ijt}\cdot x_{ij} - 120\cdot \sum_{\substack{d\in D,e\in E\\ n\in N,t\in T}} J_{dent} - 120\cdot \sum_{\substack{c\in C,e\in E\\ n\in N,t\in T}} G_{cent} \tag{4.1}$$

Constraints

$$\sum_{j\in P_i} x_{ij} = area_i \qquad i\in I \tag{4.2}$$

Timber Constraints

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} spbf_{ijt}\cdot x_{ij} \le \sum_{\substack{i\in I(u)\\ j\in P_i}} spbf_{ijt+1}\cdot x_{ij} \qquad u\in U, t\in T \tag{4.3}$$

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} other_{ijt}\cdot x_{ij} \le 0.25 \sum_{\substack{i\in I(u)\\ j\in P_i}} total_{ijt}\cdot x_{ij} \qquad u\in U, t\in T \tag{4.4}$$

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} spbfinv_{ijt}\cdot x_{ij} \le \sum_{\substack{i\in I(u)\\ j\in P_i}} spbfinv_{ijt+1}\cdot x_{ij} \qquad u\in U, t\ge 11(end) \tag{4.5}$$

Ecosystem Constraints

$$\sum_{\substack{i\in I(n,e)\\ j\in P_i}} dev_{dijt}\cdot x_{ij} + J_{dent} \ge A_{dn}\cdot area_{en} \qquad d\in D, e\in E, n\in N, t\ge 10(end) \tag{4.6}$$

$$\sum_{\substack{i\in I(n,e)\\ j\in P_i}} ser_{cijt}\cdot x_{ij} + G_{cent} \ge B_{cn}\cdot area_{en} \qquad c\in C, e\in E, n\in N, t\ge 10(end) \tag{4.7}$$

$$\sum_{\substack{i\in I(w)\\ j\in P_i}} cover_{ijt}\cdot x_{ij} \ge 0.6\cdot area_w \qquad w\in W, t\ge 5(end) \tag{4.8}$$

Figure 4.5: Model One Formulation: Objective and Constraints

come from the original Woodstock model supplied by the NSDNR. A and B values can be found in the AMPL code or the Woodstock optimize file, appendices A.3 and A.4.2 respectively. Constraints 4.8 are the sole case of an element being introduced to this study that was not in the original NSDNR Woodstock model. They state that in the last 25 periods at least 60% of the forest in each watershed, w, must qualify as suitable watershed forest cover, i.e. not be in an establishment development class. The NSDNR is interested in modelling watersheds, but for computational reasons have not yet included them in their models. The number 60% was chosen because when higher numbers were tested, the models were infeasible.

Yield parameter data was the same for both M1 and M2, and came from the NS Growth and Yield model for even-aged stands and from Permanent Sample Plot (PSP) data for uneven aged stands [O'Keefe and McGrath, 2006]. This data was stored in a database, called the Yield Table. Woodstock generates a Yield Table from DLL functions every time a model is run. For use in the M1 model, a Yield Table containing all possible yield entries was extracted from Woodstock, and a yield file was compiled to supply parameters to the M1 LP. The yield file was recompiled for each new M1 model and would contain all relevant yield values from the Yield Table for the model being generated. This method was faster and less computationally demanding than the M2 method of calling DLL functions.

The Yield Table consisted of 125 columns and over 1,000,000 rows. Table A.4 displays the headings of the 126 columns. The Yield Table can be thought of as having two sections. The stand description section, and the yields section. The first 9 columns form the stand description section; they specify the possible combinations of natural disturbance regime, ownership, buffer status, species association, site-class, stocking level, forest state, exclusion status, and age a stand can have. These are the values yield calculations are based on. The next 140 columns comprise the yields section. They contain per hectare timber and ecosystem yields that a stand corresponding to values in the first 9 columns would generate if harvested, for timber yields, and standing for ecosystem yields. Examples of timber yields are spruce-fir pulp volume, and total softwood volume. Examples of ecosystem yields are development class, and seral class.

The objective (4.1) of the model is to maximize spruce-fir harvest volume over the first 20 periods of modelling, and minimize violations to constraints 4.6 and 4.7 over the 30 period planning horizon. This was chosen because it was the objective of the original NSDNR Woodstock model. A period is 5 years. Time horizon effects, where solutions contain prescriptions that have been chosen or constructed to suit the planning horizon, are reduced if models for the purpose of planning 20 periods are run for 30.

Time horizon effects are more prominent in M2 than M1. In M1, each prescription has been defined by the user, so even if they are combined in a way to suit to the planning horizon, each prescription is reasonable at the stand level. In M2, however, prescriptions such as 'Clearcut in period 1, then again in period 12, then again in period 20' are constructed in a 20 period model where 8 periods (40 years) is the earliest clearcut age. This prescription has been constructed to harvest the most volume from the stand over a 20 period horizon without consideration for the state of the forest after the 20 periods. If the planning horizon is extended to 30 periods but maximization of harvest volume remains over the first 20 periods, these prescriptions will be replaced with 'Clearcut in 1, then again in 15, then again in 30' or similar. In a 30 period model, forest state and harvest levels must be maintained for 10 periods beyond the 20 period volume maximizing horizon. The models used in this case-study were programmed to have all constraints shown in Figure 4.5 apply to harvests for 30 periods, and harvests only contribute to the objective for the first 20 periods. It was still possible for M2 to construct prescriptions specifically for this planning horizon and constraint set, but the results are less extreme than if the constraints only applied for 20 periods and the objective was to maximize harvest volume over 20 periods.

For this case-study, the same interventions were defined in both models: clearcut, pre-commercial thin, commercial thin, shelterwood harvest, selection harvest, and buffer harvest. In the M1 model, prescriptions have been abstracted into a set of five lists that detail when interventions take place, how the state of the stand responds to each intervention, how the age of the stand changes as a result of each intervention, the method of each intervention, and criteria for a stand to be eligible for the prescription. Before generating the LP the stand table is passed through a routine that first generates a prescription set based on user supplied parameters and then constructs tables that simulate the age and state progression of each stand as it follows each prescription. These tables are used to generate the M1 LP. Developing the prescription generator was a non-trivial task; "A Note on Model One Prescription Modelling" and "A Note on Modelling Prescriptions for Old Stands" in appendices A.2.1 and A.2.2 describe this process in detail, and the Python source code can be found in A.2.5. A listing of the M1 prescription set can be found in the appendix (A.2.3, A.2.4). For the M2 model, prescription definitions are shown in the Woodstock action, regimes, and transitions files also found in the appendix (A.4.1, A.5.1).

The case-study consisted of two phases. 68,346 stands were modelled in both phases. The NSDNR stand table contained 176,480 forested stands, but stands were aggregated if they shared the same ecodistrict, natural disturbance regime, watershed, county, species association, forest state, stocking level, site-class, riparian status, exclusion status, ownership, and age. In the first phase, M1 and M2 models with almost identical prescription sets were run with four different constraint sets, each constraint set dealing with a different level of spatial resolution. Table 4.1 describes the spatial resolution, and constraints, referring to Figure 4.5, that apply for each of the four scenarios. Note that it is possible for scenario 1 to have no constraints but ownership spatial resolution because spruce-fir harvest volumes were computed by ownership. There were 3 ownerships: unlicensed Crown land, Northern Pulp licensed Crown land, and Port Hawkesbury Paper licensed Crown land; 22 ecodistricts, see Figure 4.2; and, 24 watersheds, see Figure 4.3.

Getting the M1 and M2 models to have nearly identical prescription sets, for phase 1, required restricting the M2 models' prescription generation so that interventions could not be mixed freely: second and third interventions were defined so as to be determined by the initial intervention. For example, a stand that was clearcut as a first intervention could only be clearcut at a fixed age for the second and third interventions, or a stand that was commercially thinned as a first intervention would receive a commercial thin again after its initial final-felling. Note that despite efforts to make the models identical, there were prescriptions available to M2 that weren't in M1 due to differences in how prescriptions are defined. An example of one of these prescriptions is: if a stand was 14 periods old in period 1, it could receive a shelterwood first entry immediately and second entry in period 3, then be placed on a selection harvesting regime starting in period 20; there were not many such prescriptions.

Scenario	Constraints	Spatial Resolution
1	none	Ownership
2	4.3, 4.4, 4.4	Ownership
3	scenario 2 and 4.6, 4.7	Ownership and Ecodistrict
4	scenario 3 and 4.8	Ownership, Ecodistrict and Watershed

Table 4.1: Phase 1 and Phase 2 Constraints and Spatial Resolution

In the second phase, restrictions were removed from M2 so that the first entry did not determine future entries. Commercial thinning, shelterwood, selection, and clearcut interventions could be combined within operability limits. Prescriptions were added to the M1 model to approximate M2's expanded prescription set; the M2 prescription set consists of thousands of prescriptions, matching all them in an M1 model is not feasible. These models were run through the same four constraint sets as the models from the first phase. The phase 1 models are meant to calibrate the comparison, to show that with almost the same prescriptions available to them M1 and M2 will find similar solutions; the phase 2 models were more realistic since the M2 model was allowed to model prescriptions as it does when used in practice.

M1 and M2 models were run on different computers. M1 was run on 64-bit Windows 7 with 8Gb of Ram and a 2.53Ghz processor, and M2 was run on 64-bit Windows 7 with 8Gb of Ram and a 3.00Ghz processor. Both computers were network computers so a controlled computational environment was not possible; nonetheless, every attempt was made to keep the model runs as undisturbed as possible. M1 models were generated using AMPL and M2 models were generated within Woodstock. All model generation times were in the order of 20 minutes. The concurrent optimizer in Gurobi 5.1.1 [Gurobi Optimization, 2013] was used for both M1 and M2 in all scenarios.

4.4 Results

Figure 4.6 shows linear growth in solution time for M1 and inconsistent growth in solution time for M2 in phase 1. Figure 4.7 illustrates linear growth in solution time of M1 and exponential growth in solution time of M2 in phase 2 models. Table 4.2 shows that with restricted and similar prescription sets, M1 achieves about 5% lower objective values than M2. Table 4.3 shows that with a realistic prescription set M1 achieves almost identical objective values as M2 and solves these models up to 7x faster. Table 4.5 shows M2 matrix size in phase 2 more than doubles along both row and column dimensions between scenario 2 and 3, and increases by more than 50% in both these dimensions again between scenarios 3 and 4, while M1 matrix size increases marginally, and steadily from scenario 1 through to 4. Model sizes for phase 1 show similar, though less pronounced, growth behaviour, Table 4.4.

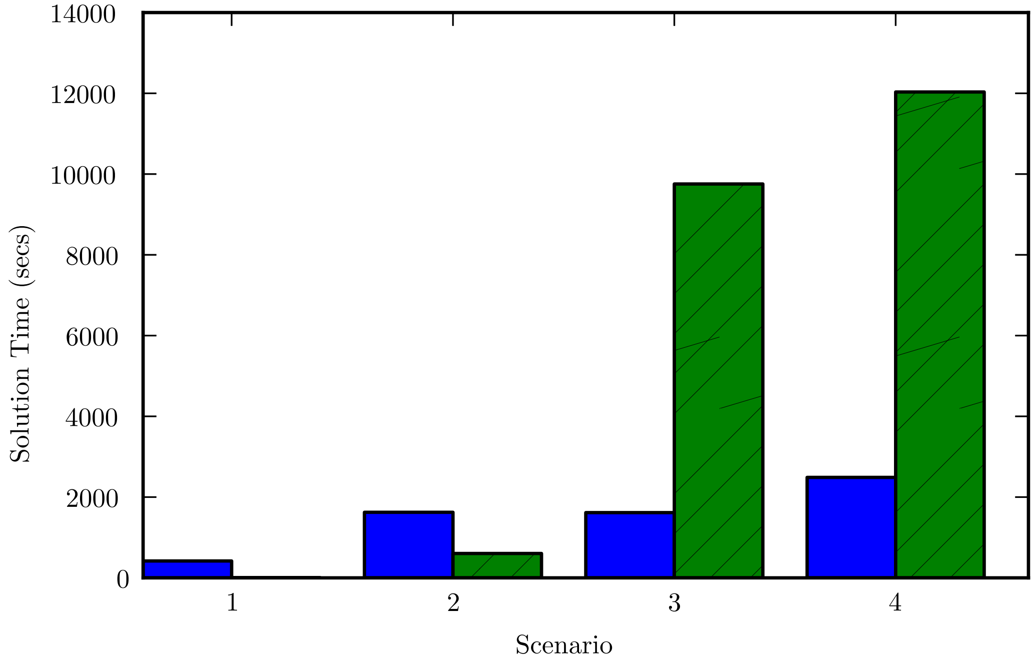

Figure 4.6: Model One (solid) and Model Two (hatched) Phase 1 Solution Times

	M1		M2	
Scenario	Solution Time (secs)	Objective (10^7)	Solution Time (secs)	Objective (10^7)
1	417.81	4.715	4.41	4.790
2	1623.81	3.788	603.44	3.869
3	1615.75	3.574	9754.32	3.744
4	2488.03	3.573	12,032.67	3.744

Table 4.2: Phase 1 Model One, Model Two Comparison Results

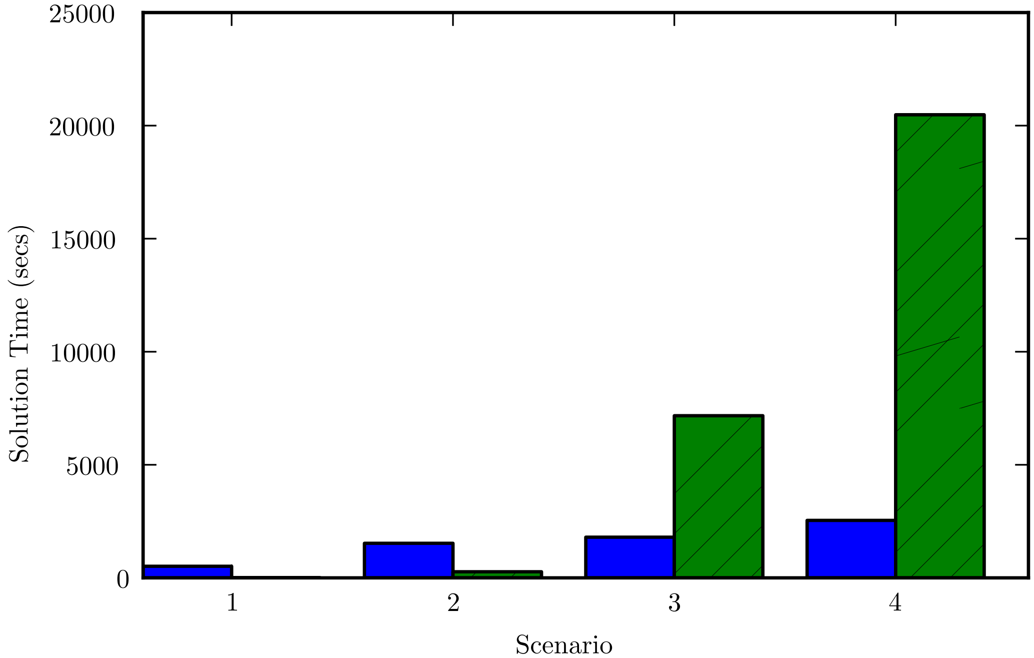

Figure 4.7: Model One (solid) and Model Two (hatched) Phase 2 Solution Times

	M1		M2	
Scenario	Solution Time (secs)	Objective (10^7)	Solution Time (secs)	Objective (10^7)
1	512.14	5.296	5.27	5.380
2	1528.38	4.294	269.44	4.191
3	1798.42	4.126	7170.34	4.119
4	2541.74	4.126	20,480.18	4.119

Table 4.3: Phase 2 Model One, Model Two Comparison Results

	Model One			Model Two		
Scenario	Rows	Columns	Non-Zeroes	Rows	Columns	Non-Zeroes
1	100,679	665,381	60,435,407	260,296	787,506	1,700,820
2	100,910	665,381	60,435,959	262,723	788,952	8,927,070
3	106,190	665,381	60,446,519	511,513	1,360,729	60,268,649
4	106,790	665,381	60,447,119	722,002	1,823,163	100,002,631

Table 4.4: Phase 1: Model One and Model Two Matrix Sizes

	Model One			Model Two		
Scenario	Rows	Columns	Non-Zeroes	Rows	Columns	Non-Zeroes
1	100,679	768,427	71,227,495	322,852	821,549	1,646,491
2	100,910	768,427	71,228,047	325,279	822,995	6,999,298
3	106,190	768,427	71,238,607	738,450	1,855,004	59,330,090
4	106,790	768,427	71,239,207	1,156,015	2,900,105	120,417,153

Table 4.5: Phase 2: Model One and Model Two Matrix Sizes

4.5 Discussion

This case-study shows that M1 matrix size grows moderately compared to M2 as additional strata are added to the models, resulting in M1 solving the models in substantially less time than M2 once multiple strata are modelled in scenarios 3 and 4. Further, M1 and M2 find similar objective function values. This is notable in phase 2 when M2 has the ability to generate more prescriptions than M1 has available.

The results illustrate what was hypothesized in Chapter 3.1. In the models from phase 2, where M2 is used as it is in practice, as constraints that apply to different strata are added to the models, M2 solution time increases exponentially, while M1 solution time increases linearly, Figure 4.7. At low spatial resolution M2 finds solutions much faster than M1. For the phase 2 models, scenario 2 is solved by M2 more than 5x faster than by M1, but after adding the ecodistrict strata in scenario 3 it takes M2 more than 4x longer than M1 to find an optimal solution, Table 4.3. In the phase 1 models M1 shows linearly increasing solution times, while M2 shows inconsistently increasing solution times, Figure 4.6.

The purpose of the phase 1 models was to show that with similar prescription sets the models found similar objective function values. Table 4.2 shows these were within 5% of each other. M2 found higher objective values because the models did not have identical prescription sets, and some of the prescriptions M2 had that M1 did not were beneficial. It is also worth noting that the modelling of interventions in M1 was based on the modelling of interventions in M2; however, there was no access to M2 source code to see exactly how interventions were modelled, so all M1 interventions were calibrated by comparing the results of M1 and M2 models containing only one intervention. These comparisons all agreed to within a fraction of a percent. Nonetheless, this approach does not guarantee that all interventions are modelled identically, explaining how M1 finds slightly higher objective values in phase 2 models, Table 4.3.

Table 4.5 shows that M2 size increases substantially as strata are added, leading to the exponentially increasing solution times shown in Figure 4.7. Between scenarios 2 and 3 in phase 2 both the number of rows and the columns more than double, then between scenarios 3 and 4 they each increase by more than 50% again. Comparing phase 1 and 2 models, this trend for M2 is more pronounced in phase 2, where more prescriptions are modelled. Also notable is that for scenarios 1-3 the phase 1 M2 model has more non-zeroes than the phase 2 model. This is unexpected since phase 2 models have more rows and columns, due to having more prescriptions available, than phase 1 models. This suggests that restricting the number of prescriptions does not introduce new constraints, but rather introduces new non-zeroes to signal that certain interventions cannot be combined into prescriptions.

Observe that M1 matrix size only increases marginally from scenario 1-4 in both phases. This is partly due to the fact that by defining the number of stands and prescriptions available to each, most of the matrix is determined. Additionally, all scenario models had the same inventory constraints and variables. So, for example, the scenario 1 models had watershed inventory variables and constraints. These variables and constraints only track quantities, and did not contribute to the models that did not utilize them in a constraint. For this reason they would have likely been removed during presolve, potentially skewing M1 solution times to be slightly higher than they would be in a model without them, and making the M1 matrices appear larger than they actually are. This could be particularly significant for the scenario 1 and 2 models where M1 takes much longer than M2 to find solutions. If these inventory variables were not modelled, M1 size would still increase only marginally between scenarios 1 and 4 for both phases, from about 87,000 rows to 106,000 rows.

In all phases and relevant scenarios, penalties were slightly higher in M1 models than M2 models, though not significant in either case; values can be found in the appendix (A.6). With the phase 2 models both M1 and M2 obtain about 15% higher objective function values than the phase 1 models, reflecting the fact that the models in phase 1 had restricted prescription sets. Further, in phase 2, despite having more prescriptions available to it, M2 found solutions with very similar objective values to M1.

Stands in the phase 2 M1 models had 15-25 prescriptions available to them; that M1 found very similar objective values as M2 suggests 15-25 user defined prescriptions per stand can perform as well as hundreds of computer generated ones. The implication of this is that many of the computer generated prescriptions are not beneficial. Note that non-exclusion stands had 15-25 prescriptions available to them; a substantial portion of the forest was in riparian buffer zones or had harvest exclusion status. In both M1 and M2, for both phases, these stands only had 1-3 prescriptions available to them.

Finally, in Chapter 3, it was noted that adding prescriptions could cause M1 solution times to increase exponentially, since each prescription requires additional x_{ij} variables; however, on comparing the solution times of the phase 1 and phase 2 models, M1 solution times increases marginally despite the phase 2 model having approximately 15% more prescriptions than the phase 1 model. This is encouraging for the application of M1 to modelling larger forests, since additional stands and additional prescriptions effect model size and solution behaviour similarly.

Taken together, these results suggest that M1 may be a more suitable framework in which to model forest management strategy at strategically relevant spatial resolution than M2. With a reasonable prescription set, it can find solutions comparable to M2, and, when multiple strata are modelled, M1 finds these solutions substantially faster.

Chapter 5

Case-Study: Modelling Industry

5.1 Introduction

Traditionally, models of forest management strategy do not consider industry. This is unfortunate since it is difficult to separate strategy from the industry in which it is to be implemented; the prevailing industry determines demand for wood products as well as transportation costs. To include industry in Model Two models requires an additional strata; this leads to the exponential growth in model size and solution time observed in Chapter 4. In this chapter, a second case-study on the Crown Central Forest is presented where including industry representation, through modelling of shipping costs and product demand, was investigated.

The case-study consisted of two parts. First, a conventional model, one that does not include industry, was compared with a model that does include industry. It was found that harvest levels differed significantly between models, due to the fact that the conventional model schedules substantial volumes of timber for harvest that are unprofitable once shipping costs and product demand are considered.

In the second part, examples of the kind of analysis that can be performed using models that include industry are provided: an assessment of the cost of a clearcut restriction policy, a comparison of profit-based and non-declining yield harvest regulations, an assessment of industrial expansion, and an assessment of allowing harvested wood to be left in the forest instead of being shipped to a mill. These examples demonstrate analysis that is not possible using models that ignore industry. Before presenting the case-study, the model formulation from which all models in this chapter were built is presented.

5.2 Model Formulation

The formulation given in Figures 5.1, 5.2, 5.3, and 5.4 expands on the Model One formulation in chapter 4 by including a shipping network to represent industry. Variables and constraints that were defined in the formulation from chapter 4—see Figures 4.4 and 4.5, and Chapter 4.3—are not redefined here. The AMPL code these models were generated from is the same code that Chapter 4 models were generated from; the only difference is constraints and variables were removed for the Chapter 4 models and the objective function changed between chapters. The code can be found in the appendix (A.3).

This model is a direct implementation of the general framework presented in Chapter 3. Notably, demand centres from chapter 3 are realized as pulp mills, saw mills and a biorefinery; and, secondary forest products are realized as wood chips that can be shipped from saw mills to pulp mills or the biorefinery. The set of timbersheds, R, is central to how industry is modelled. A timbershed is a region to which all stands belonging are assumed to have the same transportation costs to each mill. In this implementation, timbersheds have been defined along county lines, with Halifax county divided into two timbersheds, see Figure 5.5.

The planning horizon was again set at 30 periods. Notice that the shipping network was not modelled for the entire planning horizon. The set of periods in which the shipping network is modelled, T_v, was defined to be five for these experiments. The shipping network was not modelled

Sets

	From Chapter 4 Models
I	Stands
P_i	Prescriptions for stand i
$T := 30$	Periods
Y	Yields
$Y_w \in Y$	Timber yields
$Y_e \in Y$	Non-timber yields
U	Ownerships
E	Ecodistricts
N	Natural Disturbance Regimes
$I(u)$	Stands Belonging to $u \in U$
$I(n, e)$	Stands Belonging to $n \in N, e \in E$
D	Development Classes
C	Seral Classes
	New for Chapter 5
$I(u, r)$	Stands belonging to $u \in U$ and $r \in R$
$I(e)$	Stands belonging to $e \in E$
$P(cc)_{it}$	Prescriptions for $i \in I$ with clearcut entries in $t \in T$
T_v	Periods in which shipping network is modelled
Y_{wl}	Low-value species timber yields
R	Timbersheds
M	Mills
$M_p \in M$	Pulp Mills
$M_s \in M$	Saw Mills

Parameters

	From Chapter 4 models	
$spbf_{ijt}$	$i \in I, j \in P_i, t \in T$	Spruce-fir volume harvested per hectare from i under j in t
$spbfinv_{ijt}$	$i \in I, j \in P_i, t \in T$	Spruce-fir standing inventory per hectare on i under j in t
$other_{ijt}$	$i \in I, j \in P_i, t \in T$	Low-value volume harvested per hectare on i under j in t
$total_{ijt}$	$i \in I, j \in P_i, t \in T$	Total volume harvested per hectare from i under j in t
dev_{dijt}	$d \in D, i \in I, j \in P_i, t \in T$	0-1 i under j is in d in t
ser_{cijt}	$c \in C, i \in I, j \in P_i, t \in T$	0-1 i under j is in c in t
$area_i$	$i \in I$	Area of i
$area_{en}$	$e \in E, n \in N$	Area of e in n
A_{dn}	$d \in D, n \in N$	Target percentage area in d, and n
B_{cn}	$c \in C, n \in N$	Target percentage area in c, and n

Figure 5.1: Industry Model Formulation: Sets and Parameters

Parameters Cont.

	New to Chapter 5 models	
$harvarea_{ijt}$	$i \in I, j \in P_i, t \in T$	Area of i under j harvested in t
cap_m	$m \in M$	Capacity of m in m^3
dem_m	$m \in M$	Minimum demand of m in \$
sc_{rm}	$r \in R, m \in M$	Shipping cost from $r \in R$ to $m \in M$
sc_{mn}	$m \in M, n \in M$	Shipping cost from $m \in M$ to $n \in M$

Variables

x_{ij}	$i \in I, j \in P_i$	Area of i assigned to j
J_{dent}	$d \in D, e \in E, n \in N, t \in T$	Violation to constraint 6
G_{cent}	$c \in C, e \in E, n \in N, t \in T$	Violation to constraint 7
Shipping Variables		
z_{urmyt}	$u \in U, r \in R, m \in M, y \in Y_w, t \in T_v$	Shipments of y from u and r to m in t
p_{umnyt}	$u \in U, m \in M_s, n \in M_p, y \in Y_w, t \in T_v$	Shipments of y from u and m to n in t
vol_{mkt}	$m \in M, k \in Y_w, t \in T_v$	Total volume of k shipped to m in t
REV_{mt}	$m \in M, t \in T_v$	Revenue of m in t
$TRANS_{mt}$	$m \in M, t \in T_v$	Transportation costs to m in t
$PROF_{mt}$	$m \in M, t \in T_v$	Profit at m in t

Figure 5.2: Industry Model Formulation: Variables

Objective

$$\max \sum_{\substack{m\in M\\ t\in T_v}} 0.95^t \cdot PROF_{mt} - 3000 \cdot \sum_{\substack{d\in D,e\in E\\ n\in N,t\in T}} J_{dent} - 3000 \cdot \sum_{\substack{c\in C,e\in E\\ n\in N,t\in T}} G_{cent} \tag{5.1}$$

Constraints

$$\sum_{j\in P_i} x_{ij} = area_i \qquad i \in I \tag{5.2}$$

Timber Constraints

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} spbf_{ijt} \cdot x_{ij} \le \sum_{\substack{i\in I(u)\\ j\in P_i}} spbf_{ijt+1} \cdot x_{ij} \qquad u \in U, t \in T \tag{5.3}$$

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} other_{ijt} \cdot x_{ij} \le 0.25 \sum_{\substack{i\in I(u)\\ j\in P_i}} total_{ijt} \cdot x_{ij} \qquad u \in U, t \in T \tag{5.4}$$

$$\sum_{\substack{i\in I(u)\\ j\in P_i}} spbfinv_{ijt} \cdot x_{ij} \le \sum_{\substack{i\in I(u)\\ j\in P_i}} spbfinv_{ijt+1} \cdot x_{ij} \qquad u \in U, t \ge 11(end) \tag{5.5}$$

Ecosystem Constraints

$$\sum_{\substack{i\in I(n,e)\\ j\in P_i}} dev_{dijt} \cdot x_{ij} + J_{dent} \ge A_{dn} \cdot area_{en} \qquad d \in D, e \in E, n \in N, t \ge 10(end) \tag{5.6}$$

$$\sum_{\substack{i\in I(n,e)\\ j\in P_i}} ser_{cijt} \cdot x_{ij} + G_{cent} \ge B_{cn} \cdot area_{en} \qquad c \in C, e \in E, n \in N, t \ge 10(end) \tag{5.7}$$

$$\sum_{\substack{i\in I(e)\\ j\in P(cc)_{it}}} x_{ij} \le 0.5 \cdot harvarea_{ijt} \cdot x_{ij} \qquad e \in E, t \in T \tag{5.8}$$

Figure 5.3: Industry Model Formulation: Objective and Constraints

Shipping Network

$$\sum_{m\in M} z_{urmkt} = \sum_{\substack{i\in I(u,r)\\ j\in P_i}} y_{ijkt}\cdot x_{ij} \qquad u\in U, r\in R, k\in Y_w, t\in T_v \tag{5.9}$$

$$\sum_{n\in M} p_{umnkt} \leq 0.5\cdot \sum_{r\in R} z_{urmkt} \qquad u\in U, m\in M, k\in Y_w, t\in T_v \tag{5.10}$$

$$vol_{mkt} = \sum_{\substack{u\in U\\ r\in R}} z_{urmkt} + \sum_{\substack{u\in U\\ n\in M_s}} p_{unmkt} \qquad m\in M, k\in Y_w, t\in T_v \tag{5.11}$$

$$\sum_{\substack{u\in U\\ r\in R\\ k\in Y_w}} dc_{mk}\cdot z_{urmkt} + \sum_{\substack{n\in M\\ u\in U\\ k\in Y_w}} dc_{mk}\cdot p_{unmkt} = d_{mt} \qquad m\in M, t\in T_v \tag{5.12}$$

$$\sum_{\substack{u\in U\\ r\in R\\ k\in Y_w}} z_{urmkt} + \sum_{\substack{n\in M\\ u\in U\\ k\in Y_w}} p_{unmkt} \leq cap_m \qquad m\in M, t\in T_v \tag{5.13}$$

$$d_{mt} \geq dem_m \qquad m\in M, t\in T_v \tag{5.14}$$

$$\sum_{k\in Y_{wl}} vol_{mkt} \leq 0.1 \sum_{k\in Y_w} vol_{mkt} \qquad m\in M, t\in T_v \tag{5.15}$$

$$TRANS_{mt} = \sum_{\substack{u\in U\\ r\in R\\ k\in Y_w}} sc_{rm}\cdot z_{urmkt} + \sum_{\substack{u\in U\\ n\in M_s\\ k\in Y_w}} sc_{nm}\cdot p_{unmkt} \qquad m\in M, t\in T_v \tag{5.16}$$

$$REV_{mt} = d_{mt} + \sum_{\substack{u\in U\\ n\in M_p\\ k\in Y_w}} 20\cdot p_{umnkt} \qquad m\in M, t\in T_v \tag{5.17}$$

$$PROF_{mt} = REV_{mt} - TRANS_{mt} \qquad m\in M, t\in T_v \tag{5.18}$$

Figure 5.4: Industry Model Formulation: Shipping Network

for the entire planning horizon, because modelling 5 periods achieves a similar effect as modelling 20 or 30 periods and is less computationally demanding.

Constraints 5.9 – 5.18 define the shipping network. Constraint 5.9 states that the volume of wood of type k shipped from each ownership, u, and timbershed, r, is equal to the volume of wood of type k harvested in u and r. Constraints 5.10 state that up to 50% of sawlog volume can be shipped as chips. Constraint set 5.11 defines variables that inventory the volume of timber of type k sent to mill m in period t. These are used in constraint set 5.15. Constraints 5.12 convert cubic meters of wood to dollars of revenue at each mill $m \in M$. Constraints 5.13 and 5.14 state mill m cannot receive more than cap_m cubic meters of wood, and must receive at minimum dem_m dollars of wood, in each period respectively. Constraints 5.15 state that no more than 10% of mill feedstock can come from low-value species.

Constraints 5.16 define transportation costs; the mill receiving the wood pays the transporation costs. Constraints 5.17 define mill revenue as value of wood received plus \$20/m^3 for chips shipped. Constraints 5.18 define profit at each mill as the net of revenue and transportation costs. The objective function is to maximize mill profit and minimize deviations from constraints 5.6 and 5.7.

Constraint set 5.8 was not included in the Chapter 4 model. These constraints state that no more than 50% of harvest, by area, can be by clearcuts in each ecodistrict, e, and period, t. These constraints only appear in Section 5.4.1, for the clearcut policy assessment.

For this case study, 7 mills, 6 timbersheds, and 3 ownerships were modelled. Mills, see Figure 5.5, are approximations of what currently exist in Nova Scotia's central region, see Table 5.1; their minimum demand and capacity levels are enumerated in Table B.1. Mill 7 is not actually a mill, but a biofuel refinery. Along with mill 8, it is one of the two demand centers not based on existing infrastructure. There is no industrial market for certain species in Nova Scotia, so the biorefinery was modelled to accept species that cannot be sent to any of the mills. In reality, the species that have no industrial market are sold as firewood, or sent to export markets. The biorefinery takes low-value softwood, and all hardwood, logs and pulp, and paid the same price \$10/m^3 for everything. Mill 8 is only included in this chapter's industrial expansion subsection of the Example Analysis Section (5.4.3).

The Crown Central Forest is divided amongst three licenses, modelled as follows:

- Unlicensed (UNL) - wood off this license can go to any mill
- Northern Pulp License (NPL) - softwood pulp, and chips off softwood logs goes to mill 6; softwood logs, and hardwood logs and pulp can go to any mill
- Port Hawkesbury Paper License (PHPL) - same as Northern Pulp except mill 2 instead of mill 6.

Mill 6 is a softwood pulp mill and the only mill owned by the NPL license, so all softwood pulp from NPL land is modelled as getting shipped to mill 6. The same situation applies to mill 2 and the PHPL license.

Mill	Accepts
1	all softwood sawlogs
2	all softwood pulp
3	valuable hardwood sawlogs
4	all softwood sawlogs
5	all softwood sawlogs
6	all softwood pulp
7	low-value softwood and all hardwood
8	all softwood sawlogs

Table 5.1: Mills and the Wood Types they Accept

Eight timber yield types were modelled; the price they fetch at the mills, independent of transportation costs, are listed in Table 5.2. The values shown have already had $\$30/m^3$ harvest costs subtracted from them. Not listed in the table is that logs and pulp of all hardwood species, and low-value softwood species, could be sent to the biorefinery, mill 7, and fetch a price of $\$10/m^3$. In NSDNR strategic modelling, softwood and hardwood are each split into two species groups. Softwood is either Spruce-Fir (SPBF), or Pine/Eastern Hemlock/Tamarack Larch (PIEHTL); and hardwood is either Sugar-Maple/Yellow Birch (SMYB), or Intolerant Hardwood/Red Oak/Beech (IHROBE). SPBF and SMYB are termed 'valuable' softwood and hardwood because there is a market for these species in Nova Scotia. Constraints 5.15 limit the amount of low-value, PIEHTL and IHROBE, logs and pulp mills could receive to less than 10% of their total feedstock. Representative of Central Nova Scotia, there was no mill that pays for hardwood pulp of any species, the only option for this wood was to get sent to the biorefinery.

Species	Log Value ($\$/m^3$)	Pulp Value ($\$/m^3$)
Sugar-Maple/Yellow Birch	35.00	10.00
Intolerant Hardwood/Red Oak/Beech	10.00	10.00
Spruce-Fir	35.00	15.00
Pine/Eastern Hemlock/Tamarack Larch	20.00	15.00

Table 5.2: Species Log and Pulp Values

Transportation costs were modelled as $\$6.50/m^3$ plus $\$0.07$/km/$m^3$ in constraints 5.16 (Appendices B.1.2, and B.1.3 list per cubic meter timbershed to mill costs and mill to mill costs, respectively); $\$20/m^3$ was paid to sawmills shipping chips to account for a portion of the harvest costs and to provide incentive in the system to ship chips. Distances were computed as the centre of timbersheds to mills.

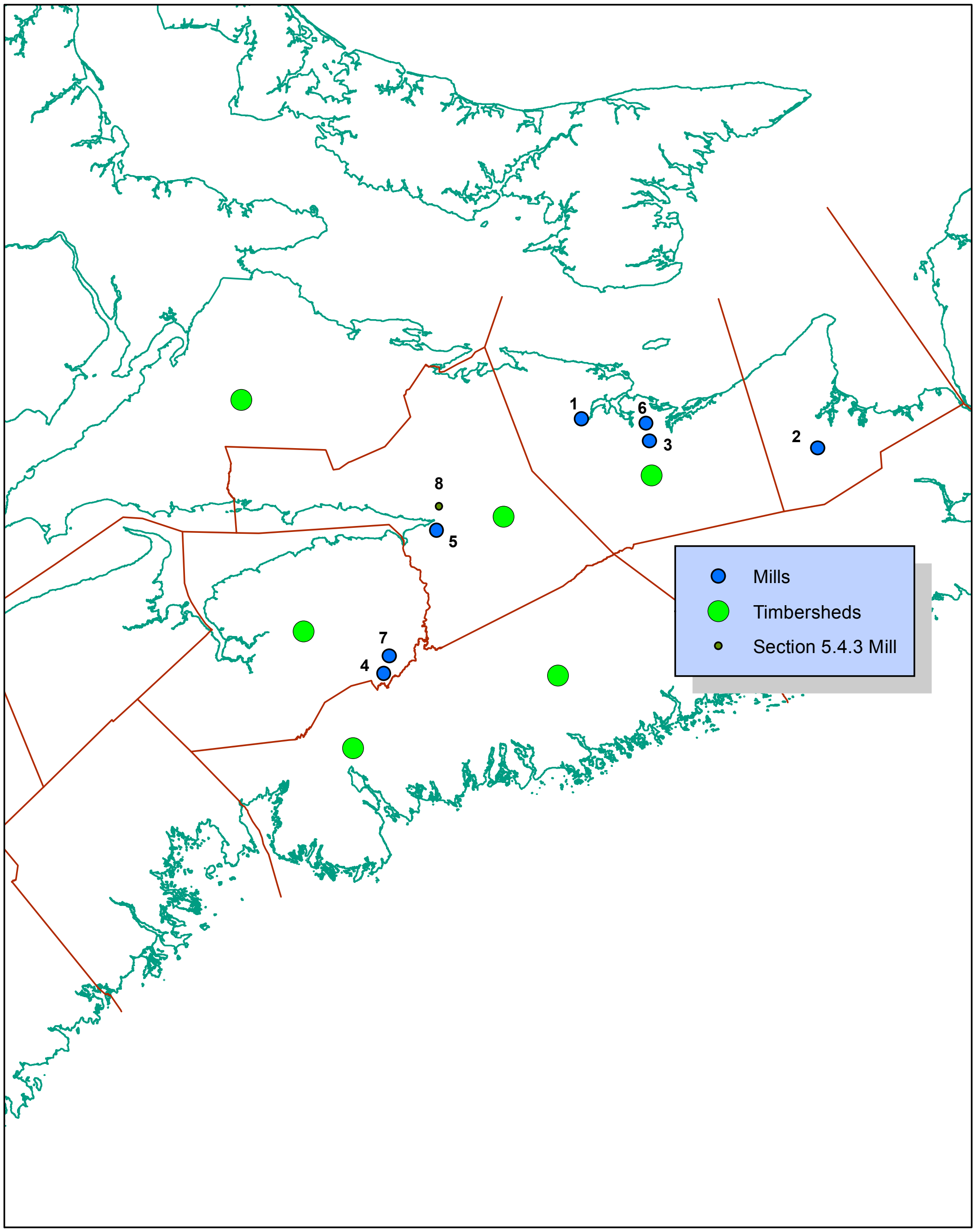

Figure 5.5: Mills and Timbersheds

Penalties on the J and G variables, equation 5.1, were set at \$3000/ha of violation, from assuming that each hectare of violation had a $120m^3$ penalty, as in Chapter 4, and assigning those cubic meters the average of Spruce-Fir Log and Pulp prices, $\$25/m^3$. A 5% per period discount rate was applied to profit, but not to penalties. The penalty values, though substantial in some periods, differ insignificantly across models. Penalty values for the base model can be found in the appendix (A.6).

It is worth emphasizing that the industry modelling in this case-study is demonstrative. The comparisons in Section 5.4 should not be interpreted as analysis of actual policy. Wood prices, transportation costs, minimum demand levels at mills and mill capacities were chosen to be illustrative. The specific way industry was modelled is not the only way to model it, and no claims are made that the way chosen is the best. Determining good ways to model shipping costs and product demand would be a productive way to expand on this work.

As in chapter 4, the study forest is too small to provide solid ground on which to make policy decisions, for instance, in reality, in addition to crown land wood, Central Nova Scotian mills receive a substantial amount of wood from large and small private land owners, thus making it difficult to determine mill capacity and minimum demand levels based only on crown land.

5.3 Industry Model vs. Conventional Model

A model of forest management strategy that does not model industry assumes either there is infinite demand for wood and that transportation costs are negligible, or it assumes these considerations are irrelevant to modelling. To investigate how model solutions change once industry is modelled, the phase two scenario 3 model from chapter 4, Table 4.1, was compared with an equivalent model based on this chapter's industry formulation. The only differences being that the industry model included the shipping network from Figure 5.4 while the model without industry, the conventional model, did not, and the industry model sought to maximize profit while the conventional model sought to maximize spruce-fir harvest volume. Both models sought to minimize violations to constraints

5.6 and 5.7, penalized as described in Chapter 4.3 and Section 5.2. Clearcut restriction constraints (5.8) were not included in this model. Figure 5.6 shows that spruce-fir harvest levels are more than 33% less in the industry model than in the conventional model.

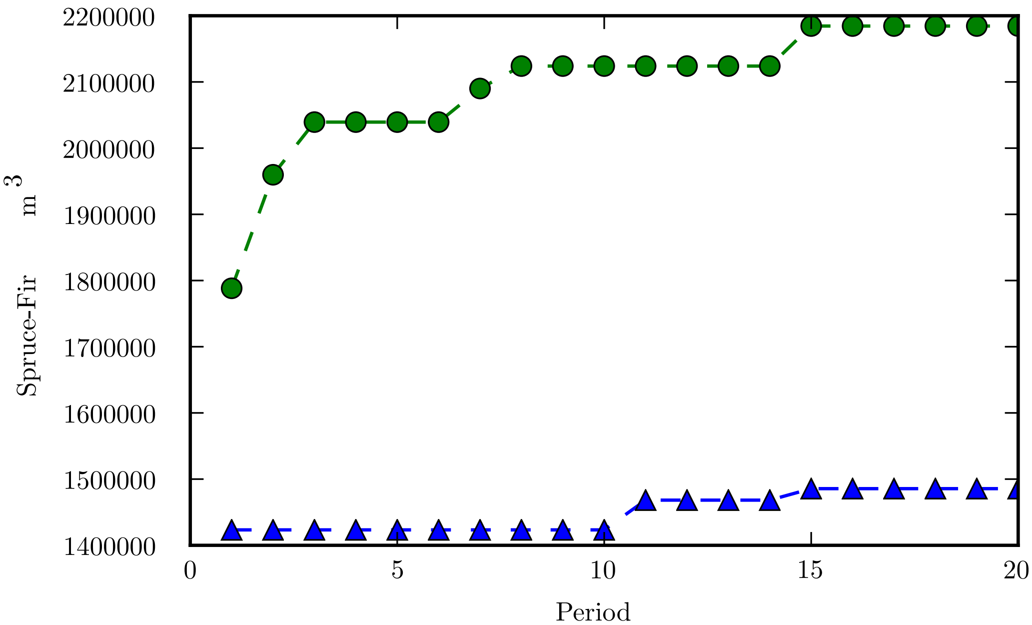

Figure 5.6: Industry Model vs. Conventional Model: Spruce-Fir Harvests - Industry Model (triangles), Conventional Model (circles)

5.3.1 Discussion

For all industry models in this chapter, tables showing model size, solution times, objective function values and periodic breakdowns of profit and spruce-fir volume can be found in appendix B.2. Model size and solution time for the conventional model can be found in Tables 4.5 and 4.3. Figure 5.6 shows that the industry model determines harvest volumes that are more than 33% less than those determined by the conventional model. Note that there is sufficient capacity at the mills and biorefinery in the industry model to accommodate the entire harvest obtained in the conventional model. Penalty values in the industry model were about half of those in the conventional model, but in neither case very substantial; they can be found in Table A.6.

The difference in spruce-fir harvests shown in Figure 5.6 is a result of the conventional model maximizing spruce-fir harvests and the industry model maximizing profit. Much of the wood harvested in the conventional model is not profitable to harvest once harvests must be justified by wood value and transportation costs. In this way, models that do not include industry may misrepresent what sustainable harvest levels might be.

5.4 Example Analysis

In this section, examples of analysis that can be performed using models that include industry are provided. By including industry in modelling, model scope increases and it becomes possible to analyze the cost of return of policies instead of only their effect on harvest levels, which, as was demonstrated in section 5.3, may not be a good reflection of harvest economics. Example analysis will be demonstrated through comparisons between a base model—the industry model of Section 5.3—and models that have been modified for each scenario. Graphs for profit levels over the first 5 periods, and spruce-fir harvest levels over the first 20 periods are displayed with each comparison, and tables describing the models can be found in the appendix (B.2). All objective functions are to maximize discounted profit over the first five periods, and minimize deviations to constraints 5.6 and 5.7, penalized as described in Section 5.2, over the 30 period horizon. The first example assesses the cost of instituting a restriction on clearcuts to less than 50% of harvest by area. In the second example, profit-based harvest regulation strategies are compared with non-declining yield. The third example investigates an industrial expansion scenario. And, the final example looks at allowing harvested wood to be left in the forest instead of shipped to a mill.

5.4.1 Clearcutting Restriction

Constraints 5.8 were added to the base model to investigate the impact on profit and spruce-fir harvests of instituting a 50% restriction on clearcuts by area in each ecodistrict. This clearcut policy is based on policy that is to be implemented in Nova Scotia [NSDNR, 2011]. Figures 5.7 and 5.8 show there is less than 1% difference in total profit and spruce-fir harvests, respectively, as a result of the clearcut restriction.

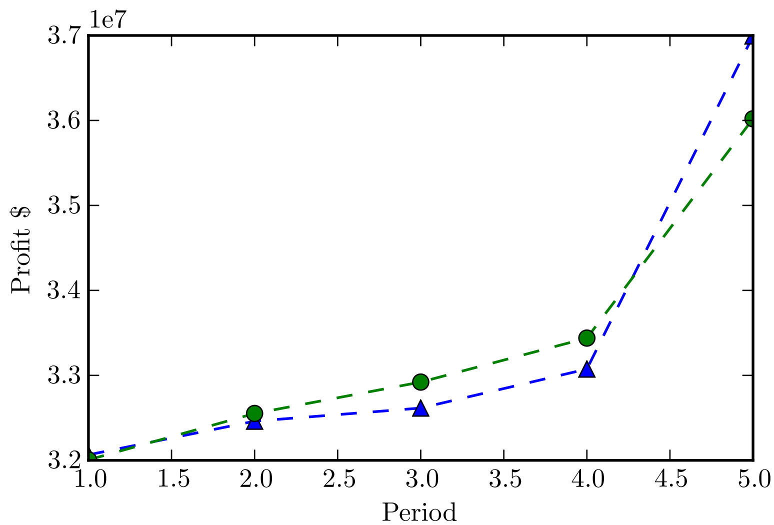

Figure 5.7: Clearcut Restriction: Profit - Base (triangles), Clearcut Restricted (circles)

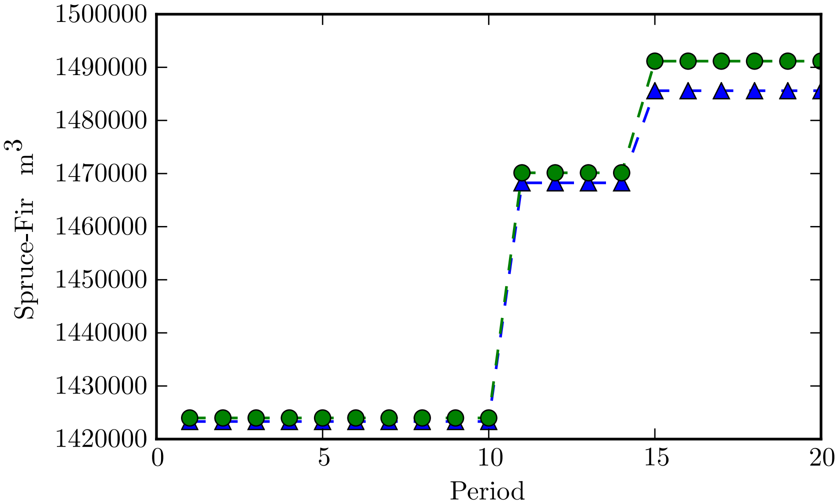

Figure 5.8: Clearcut Restriction: Spruce-Fir Harvests - Base (triangles), Clearcut Restricted (circles)

5.4.2 Profit-Based Harvest Regulation

Non-Declining Yield (NDY) has been a standard regulation strategy to manage for sustainable harvests. In Ch 2.3, criticisms against its use as the sole method of harvest regulation were summarized. Contrasting with NDY based conventional models, modelling industry creates the opportunity for alternative regulation strategies that are flexible with respect to industry. In this section, two new regulation strategies are presented. The first, called Mill Regulation, regulates harvests in the first 5 periods on the condition that each mill must receive a non-declining profit in each period. It is modelled by removing NDY constraints, i.e. constraint set 5.2, in the first four periods—5.2 still applies from period five to thirty to ensure harvests do not drop off after period 5—and adding constraint set 5.19:

$$PROF_{mt} \leq PROF_{mt+1} \qquad m \in M, t \in T_v \tag{5.19}$$

The second regulation strategy, called Mean Regulation, allows harvest volumes in any period to decrease by up to 5% of the mean harvest volume for the planning horizon—Mean Regulation is inspired by a control heuristic to prevent bus bunching [Simon Berrebi, personal communication Aug. 2013]. It is modelled by removing constraint set 5.2 in all periods, and adding constraint set 5.20:

$$\sum_{\substack{i \in I(u) \\ j \in P_i}} spbf_{ijt} \cdot x_{ij} \geq (0.95/30) \sum_{\substack{i \in I(u) \\ j \in P_i \\ q \in T}} spbf_{ijq} \cdot x_{ij} \qquad u \in U, t \in T \tag{5.20}$$

The appeal of both of these new regulation strategies is that harvests are allowed to fluctuate based on profitability. Fluctuating harvests are obviously not allowed under NDY, so for example, valuable forest that is profitable to harvest in early periods my be left standing because in later periods it is not possible to equal or exceed early period harvest levels. This means flexible regualtion strategies may be more profitable than NDY. Using models that include industry it is possible to investigate this. The base model from the previous sections has NDY Regulation; here it is compared against Mill Regulation and Mean Regulation models. Figure 5.9 shows profit between Mean Regulation and NDY is similar, while Mill Regulation is about 25% less. Figure 5.10 shows that spruce-fir harvests fluctuate according to profit for Mean and Mill Regulation, and do not for NDY.

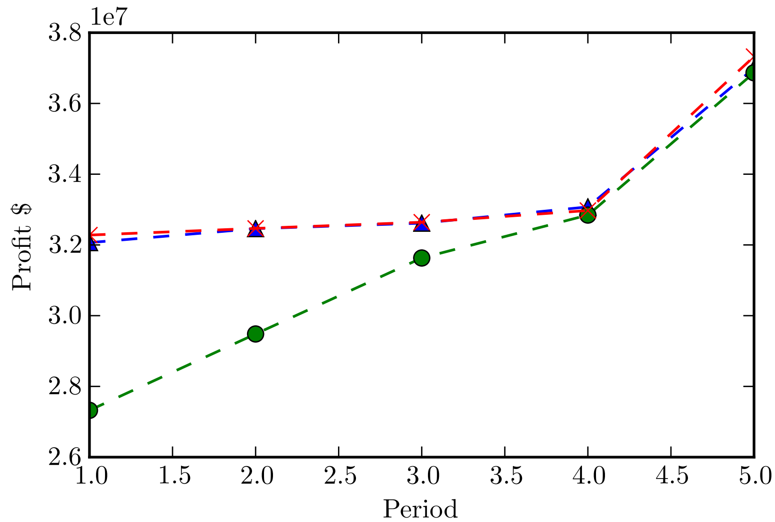

Figure 5.9: Profit-Based Regulation Strategies to Non-Declining Yield: Profit - NDY (triangles), Mill Regulation (circles), Mean Regulation (x)

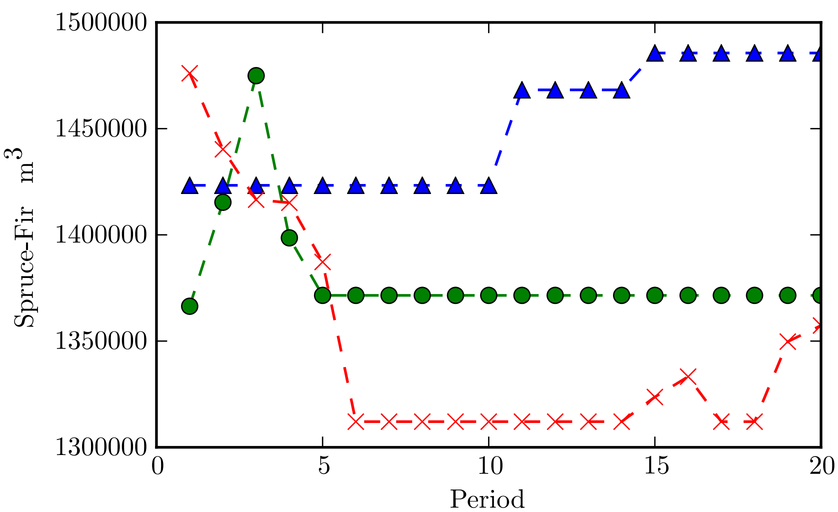

Figure 5.10: Profit-Based Regulation Strategies to Non-Declining Yield: Spruce-Fir Harvests - NDY (triangles), Mill Regulation (circles), Mean Regulation (x)

5.4.3 Industrial Expansion

This comparison explores the introduction of a new mill to Central Nova Scotia's industry. The mill is a softwood sawmill owned by Northern Pulp, located just outside Truro. It is mill 8, referenced in Table 5.1 and Figure 5.5. This mill can receive softwood logs from any license, but softwood logs harvested from the Northern Pulp license can only be shipped to it. Figures 5.11 and 5.12 show profit and spruce-fir harvest volume increase substantially as a result of adding the new mill to the industry.

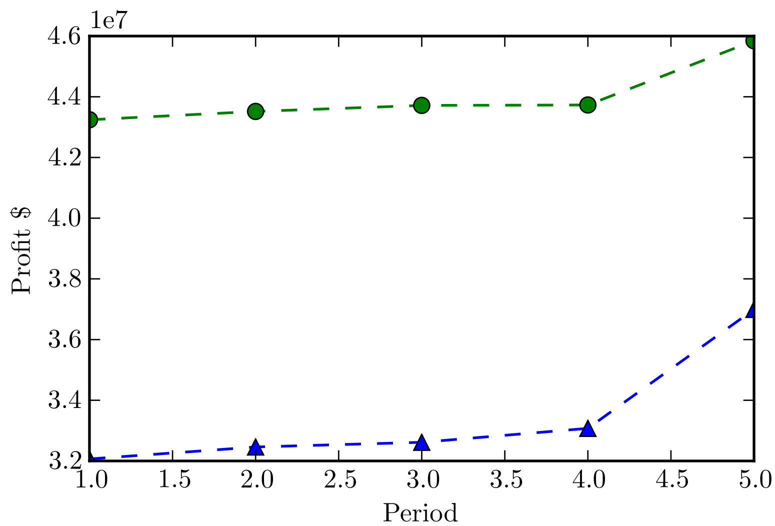

Figure 5.11: Industrial Expansion: Profit - Base (triangles), Expanded Industry (circles)

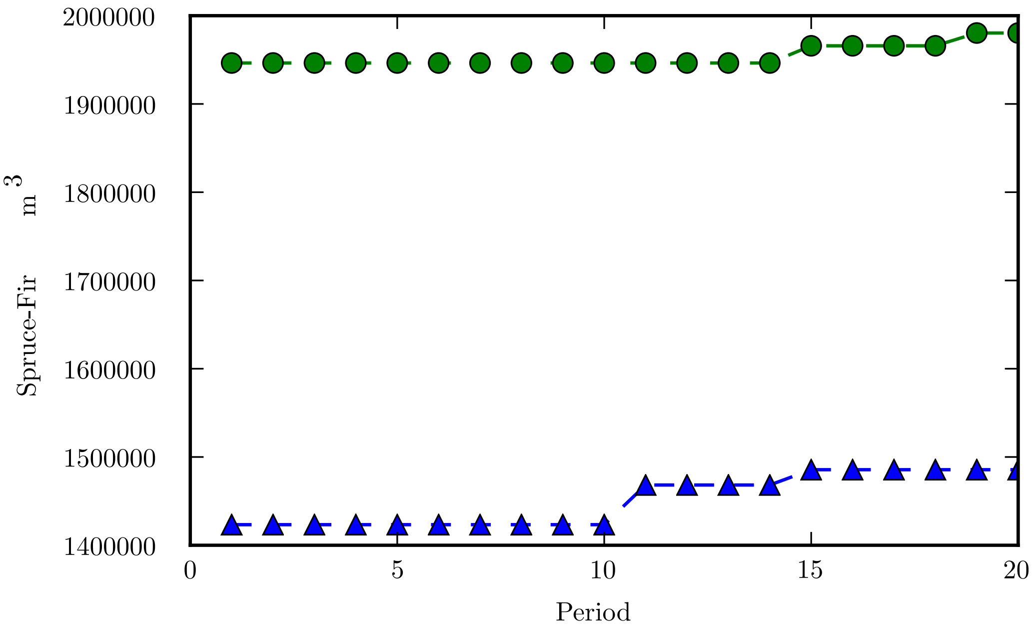

Figure 5.12: Industrial Expansion: Spruce-Fir Harvests - Base (triangles), Expanded Industry (circles)

5.4.4 Leaving Wood in the Forest

In all the models so far in this chapter it has been assumed that all wood harvested must be shipped to a demand centre. In reality this is often not the case; low-value wood gets left on the side of the road or isn't harvested in the first place. To give the model the option to leave wood in the forest constraint set 5.15 was changed from an equality to a less than or equal inequality. Figures 5.13 and 5.14 show that as a result of allowing wood to be left in the forest profit increases by about 5% and spruce-fir harvests increase more, about 30%.

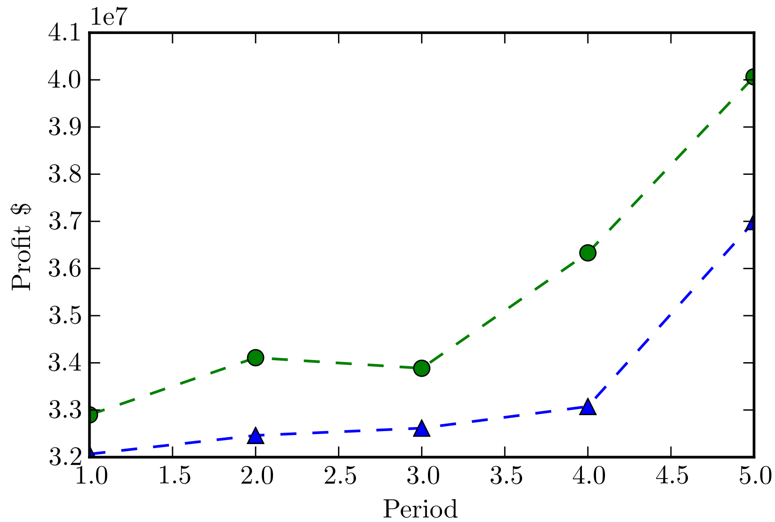

Figure 5.13: Leaving Wood in the Forest: Profit - Ship Everything (triangles), Ship Selectively (circles)

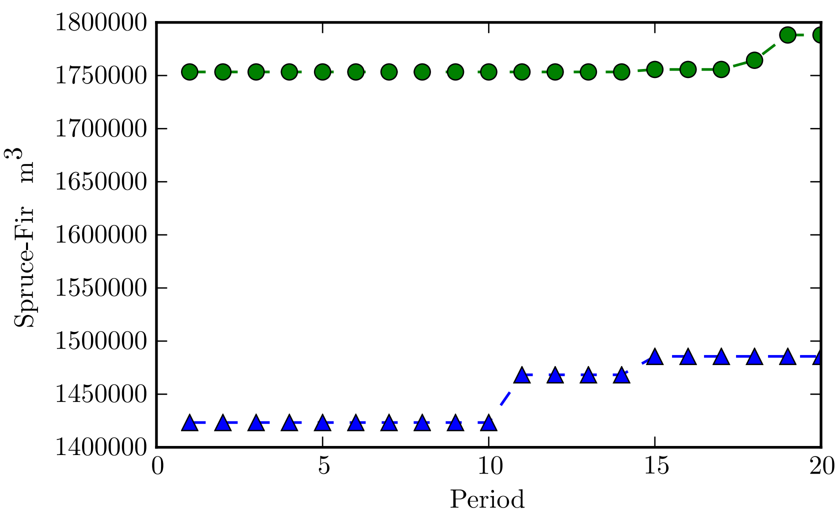

Figure 5.14: Leaving Wood in the Forest: Spruce-Fir Harvests - Ship Everything (triangles), Ship Selectively (circles)

5.4.5 Discussion

The comparisons will be discussed in the order they were presented. Figures 5.7 and 5.8 show that the clearcut restriction has virtually no impact on harvest levels or profit. Though Figure 5.7 does not make it clear, the unrestricted, base, model achieves marginally greater profit over the first 5 periods than the clearcut restricted model, see Table B.9. There are a two reasons that the clearctu restriction policy has so little effect. Foremost among them is that prescription costs were not modelled, so applying a three-entry thinning prescription was assumed to cost the same as a single clearcut entry. Thus, the base model was already harvesting less than 50% of area by clearcuts in most ecodistricts and periods, so the policy had little effect. The second reason is that imposing the clearcut restriction constraint causes the area harvested to increase, because more partial harvesting systems are being employed. This lessens the effect of the constraint since if more area is harvested then more area can be clearcut harvested, so it is possible that the constraint could actually increase the area clearcut harvested every period as long as the area partial harvested increased commensurately. A final note is that spruce-fir harvests are marginally higher in later periods in the clearcut restricted model, see Figure 5.8. This is because in the restricted model slightly more partial harvesting prescriptions are applied, leading to better stand regeneration in later periods.

The profit-based regulation models present a number of results worth commenting on. The NDY model harvests the most wood of the three—about 7% more than Mill Regulation and about 10% more than Mean Regulation. This does not translate into the NDY model generating the highest profit of the three; in fact, Mean Regulation is marginally more profitable than NDY, while Mill Regulation is about 25% less profitable. As expected, both profit-based regulation strategies produced fluctuating harvest levels. Mean Regulation produced a harvest profile that corresponds to harvesting valuable standing timber in the first five periods and then settling into a sustained harvest about 10% lower than the first period's harvest. Mill Regulation harvest volumes fluctuate by about 8% in the first five periods. The purpose of this comparison was to show that since industry is being modelled, regulation that allows harvests to fluctuate based on industry is possible. Though the results of this comparison do not allow any conclusions to be drawn, they provide examples of flexible regulation strategies that could be built upon in further investigations of alternatives to NDY regulation.

Figure 5.11 suggests building the new mill could substantially increase the profit of the crown central forest and allow for a profitable increase in spruce-fir harvests levels, Figure 5.12. The reason for this is that the new mill's location is favourable with respect to spruce-fir stands and access to the Northern Pulp pulp mill, so sawlogs get shipped to the new mill at low cost, and it is profitable for it to ship residues from these sawlogs as chips. This kind of industrial expansion analysis could be usefully employed by government or industry to help them identify how effectively they are using their resource base.

Allowing wood to be left in the forest caused scheduled harvests to increase by over 20% but profit only increased by around 5%. This is possibly because volume was being scheduled for harvest that was only marginally profitable. In the model where all harvested wood must be shipped, spruce-fir from a mixedwood stand or hardwood stand might not be scheduled for harvest because it is not profitable to deal with the hardwood species that would accompany the spruce-fir. In the leaving wood in the forest model, this spruce-fir can be harvested—even if it brings in a small profit—and the low-value hardwood species left in the forest. The total wood left in the forest over the planning horizon was 4,753,839 m^3. This suggests industrial demand is not well-aligned with wood supply.

Chapter 6

Conclusion

6.1 Summary

The modelling approach taken in this thesis differs from the conventional modelling of forest management strategy, where Model Two models that represent insufficient spatial resolution and ignore industry are used to evaluate strategy. Throughout this thesis limitations of this approach were identified. This chapter will start by recording these limitations together.

As discussed in Chapter 1, forest management strategy in Canada applies at strata, such as ecodistricts and watersheds; to provide managers with feedback on the possible effects of their strategies, models should be capable of modelling strategy at the strata at which it applies. Currently used models are not well suited to this. In Chapter 3, it was shown that M2 model size, and hence, solution times, increase exponentially as additional strata are modelled; this is why they are unsuitable for modelling strategy at multiple strata. Attention was also drawn in Chapter 3 to the fact that M2 models are subject to prescription errors caused by interventions being combined into unrealistic prescriptions.

Then, in Chapter 5, it was shown that by ignoring industry, models can schedule unprofitable stands for harvest. This may have the effect of overestimating available wood-supply.

Addressing these limitations, a Model One modelling framework was proposed. Within it, it is possible to build models that include industry representation, and can model multiple strata. In a Model One, Model Two comparison, models from this framework were compared with Woodstock™, and it was shown they can obtain solutions substantially faster than Woodstock when modelling strategically relevant spatial resolution. Further, despite the fact that Woodstock is able to generate hundreds of prescriptions, it was shown that the proposed model, with a reasonable set of prescriptions, was able to find solutions as good as Woodstock's. This result suggests Model One may be more suitable for modelling forest management strategy than Model Two.

Strategic level models with industry representation are not unknown in the literature of forest management strategy, but they are not commonly used in practice. For instance, when clearcut restrictions were being assessed by the Nova Scotia provincial government, industry was not included in their modelling.

In Chapter 5, using models with industry representation, novel analysis was demonstrated. First, the cost of instituting a clearcut restriction policy was assessed. It was found that the policy had little effect because most stands were already scheduled to receive partial harvests. Second, two profit-based harvest regulation strategies were compared with non-declining yield to provide examples of harvest regulation that is flexible with respect to industry. Third, the effect on profit and harvest volumes of introducing a new mill near Truro was investigated. It was found that the new mill led to a profitable increase in harvest volume. Fourth, harvested wood was allowed to be left in the forest instead of being shipped to a mill; it was found that spruce-fir harvests increased substantially while profit only increased marginally, illustrating misalignment in the industry.

Models from this framework were tested on a section of a forest, but the intention is to scale up and model full forests. More work needs to be done to investigate the robustness of these models to changes in the number of stands and prescriptions. Results from Chapter 4, that show, when the number of decision variables increases by 15%, model solution time only increases marginally, are encouraging.

6.2 Directions for Further Research

Further research could be directed at developing an algorithm to take advantage this modelling framework's special structure, modelling a larger forest, and further investigating the differences between including and ignoring industry in strategic level models.

Solution Algorithm

Models built within the proposed modelling framework have a predominantly Generalized Upper Bound (GUB) structure, with an embedded network due to the shipping network. Yang [2008] shows models of forest management strategy with similar GUB structure to those herein presented can be solved with a SprintGUB algorithm that can perform significantly better than Dual or Primal Simplex methods. Adapting a SprintGUB algorithm to the proposed modelling framework's GUB and network structure could lead to similar improvements in solution time for these models.

Modelling a Larger Forest

The size of the study forest was repeatedly mentioned as a limitation of the case-studies presented in this thesis. A natural extension of this work would be to model a full forest, and to compare the modelled forest with how the forest actually operates. This would be particularly interesting for identifying mismatches in wood-supply and industrial demand.

Modelling Industry

In Chapter 5 it was pointed out that the manner in which industry was modelled in this thesis is not the only way to model industry. In particular, that not all secondary products had to be shipped or processed, and that mills were paid \$20/m^3 for shipping chips to mills, may not be the best way to model secondary products. Investigating this further and identifying other ways to model the shipping of products and handling of transaction costs would help the modelling of industry in models of forest management strategy mature.

The industrial expansion model, Chapter 5.4.3, could inspire research into including the decision of whether to open or close an industrial facility in models of forest management strategy. This would lead to much larger models, with binary variables, so these models might be more suited to heuristic solution procedures than exact optimization. Another area within industry modelling of interest is the difference between industry models and conventional models. It could be instructive to compare industry and conventional models on other forests. It would be interesting to recruit policy-makers to employ models that include industry to complement their current modelling procedures. It is possible they would find results similar to those found here.

References

Ananda, J. and Herath, G. (2009). A critical review of multi-criteria decision making methods with special reference to forest management and planning. *Ecological Economics*, 68(10).

Barros, O. and Weintraub, A. (1982). Planning for a vertically integrated forest industry. *Operations Research*, 30(6):1168–1182.

Baskent, E. Z. and Keles, S. (2005). Spatial forest planning: A review. *Ecological Modelling*, 188(2):145–173.

Bettinger, P. and Chung, W. (2004). The key literature of, and trends in, forest-level management planning in north america, 1950-2001. *International Forestry Review*, 6(1):40–50.

Bowater (2010). Bowater Mersey Company Limited. High conservation value forest assessment for the medway district. Retrieved from http://novascotia.ca/natr/forestry/certification/pdf/High-Conservation-Value-Forest.pdf.

Boychuk, D. and Martell, D. L. (1996). A multistage stochastic programming model for sustainable forest-level timber supply under risk of fire. *Forest Science*, 42(1):10–26.

CCFM (2003). Canadian council of forest ministers: Defining sustainable management in canada. Retrieved from http://www.ccfm.org/pdf/CI_Booklet_e.pdf.

Cea, C. and Jofre, A. (2000). Linking strategic and tactical forestry planning decisions. *Annals of Operations Research*, 95(1):131–158.

Cogswell, A. and Feunkes, U. (1997). A hierarchical approach to spatial forest planning. In *Society of American Foresters, International Symposium on System Analysis and Management Decisions in Forestry, Traverse City, Michigan, May 28-June 1*.

Constantino, M., Martins, I., and Borges, J. (2008). A new mixed-integer programming model for harvest scheduling subject to maximum area restrictions. *Operations Research*, 56(3):542–551.

Covington, W. W., Wood, D. B., Young, D. L., Dykstra, D. P., and Garrett (1988). L. d. teams: a decision support system for multi resource management. *Journal of Forestry*, 86(8):25–33.

Curtis, F. (1962). Linear programming the management of a forest property. *Journal of Forestry*.

Dantzig, G. (1963). *Linear Programming and Extensions*. Princeton University Press, New Jersey.

Dantzig, G. B. and Van Slyke, R. M. (1967). Generalized upper bounding techniques. *Journal of Computer and System Sciences*, 1(3):213–226.

Davis, L. S., Johnson, K. N., Bettinger, P. S., and Howard, T. E. F. M. (2001). *Forest Management*. McGraw-Hill, New York, fourth edition.

Davis, R. G. and Martell, D. L. (1993). A decision support system that links short-term silvicultural operating plans with long-term forest-level strategic plans. *Canadian Journal of Forest Research*, 23(6):1078–1095.

Diaz-Balteiro, L. and Romero, C. (2008). Making forestry decisions with multiple criteria: A review and an assessment. *Forest Ecology and Management*, 255:3222–3241.

Epstein, R., Morales, R., Sern, J., and Weintraub, A. (1999). Use of or systems in the chilean forest industries. *Interfaces*, 29(1):7–29.

Faustmann, M. (1849). Berechnung des wertes welchen waldboden sowie noch nicht haubare holzbestnde fr die waldwirtschaft besitzen. *Allgemeine Forst- und Jagdzeitung*, (15). Republished with permission from Commonwealth Forestry Association as "Calculation of the Value Which Forest Land and Immature Stands Possess for Forestry" in Journal of Forest Economics 1(1) 1995.

Fourer, R., Gay, D. M., and Kernighan, B. W. (1993). *Ampl*. Boyd and Fraser.

Garcia, O. (1984). Folpi, a forestry-oriented linear programming interpreter. In Nagumo, H. e. a., editor, *Proceedings IUFRO Symposium on Forest Management Planning and Managerial Economics*, pages 293–305. the Organizing Committee of the IUFRO Symposium on Forest Management Planning and Managerial Economics, University of Tokyo.

Greer, K. and Meneghin, B. (2002). Spectrum: an analytical tool for building natural resource management models. Retrieved from http://www.ncrs.fs.fed.us/pubs/gtr/other/gtr-nc205/pdffiles/p53.pdf.

Gunn, E. (2007). Models for strategic forest management. In *Handbook of Operations Research in Natural Resources*, pages 317–341. Springer US. International Series in Operations Research & Management Science.

Gunn, E. (2010). Some perspectives on strategic forest management models and the forest products supply chain. *INFOR: Information Systems and Operational Research*, 47(3):261–272.

Gunn, E. A. and Rai, A. K. (1987). Modelling and decomposition for planning long-term forest harvesting in an integrated industry structure. *Canadian Journal of Forest Research*, 17(12):1507–1518.

Gurobi Optimization, I. (2013). Gurobi optimizer reference manual. Retrieved from http://www.gurobi.com.

Gustafson, E. J. and Crow, T. R. (1996). Simulating the effects of alternative forest management strategies on landscape structure. *Journal of Environmental Management*, 46:77–94.

Gustafson, E. J., Roberts, L. J., and Leefers, L. A. (2006). Linking linear programming and spatial simulation models to predict landscape effects of forest management alternatives. *Journal of Environmental Management*, 81(4):339–350.

Howard, T. E. (2001). The forester's dilemma: Paradoxes in the criteria and indicators for sustainable forestry. *Bois Et Forets Des Tropiques*, 270(4):75–84.

Johnsson, K. N. and Scheurmann, H. L. (1977). Techniques for prescribing optimal timber harvest and investment under different objectives. discussion and synthesis. *Forest Science*, 18:1–30.

Kent, B., Bare, B. B., Field, R. C., and Bradley, G. A. (1991). Natural resource land management planning using large-scale linear programs: the usda forest service experience with forplan. *Operations Research*, 39(1):13–27.

Kurttila, M., Pukkala, T., and Kangas, J. (2001). Composing landscape level forest plans for forest areas under multiple private ownership. *Boreal Environment Research*, 6(4):285–296.

Leak, W. B. (2011). Estimating allowable-cut by area-scheduling. Res. Note NRS-115.

Lebow, P., Spelter, H., and Ince, P. (2003). Fpl-pelps, a price endogenous linear programing system for economic modelling; supplement to pelps iii, v 1.1. Research paper, Forest Products Laboratory. FPL-RP-614, USDA Forest Service, Madison, WI.

Marshall, H. (2007). Log merchandizing model used in mechanical harvesting. In A. Weintraub, C. Romero, T. B. and R. Epstein, e., editors, *Handbook on Operations Research in Natural Resources*, pages 379–389. Springer, New York.

Martell, D. L., Gunn, E. A., and Weintraub, A. (1998). Forest management challenges for operational researchers. *European Journal of Operational Research*, 104(1):1–17.

McQuillan, A. G. (1986). The declining even flow effect–non sequitur of national forest planning. *Forest Science*, 32(4):960–972.

Montreal-Process (1998). The montreal process. Retrieved from http://www.mpci.org/whatis/criteria.html.

Naesset, E. (1997). A spatial decision support system for long-term forest management planning by means of linear programming and a geographical information system. *Scandinavian Journal of Forest Research*, 12(1):77–88.

Navon, D. I. (1971). *Timber RAM: a Long-range Planning Method for Commercial Timber Lands Under Multiple-use Management*. Pacific Southwest Forest and Range Experiment Station, Forest

Service, US Department of Agriculture.

Neilly, P., Quigley, E., Stewart, B., and Keys, K. (2007). Forest disturbance ecology in Nova Scotia. Retrieved from http://www.novascotiaforests.ca/files/ND_Report.pdf.

Nelson, J., Brodie, J. D., and Sessions, J. (1991). Integrating short-term, area-based logging plans with long-term harvest schedules. *Forest Science*, 37(1):101–122.

NRCAN (2012). Natural Resources Canada. Forest statistics. Retrieved from http://www.nrcan.gc.ca/statistics-facts/forests/897.

NRCAN (2013). Natural Resources Canada. Forest certification. Retrieved from http://cfs.nrcan.gc.ca/pages/144.

NSDNR (2007). Nova Scotia Department of Natural Resources. Ecodistricts of Nova Scotia. Retrieved from http://www.gov.ns.ca/natr/forestry/ecological/pdf/ELC_Map.pdf.

NSDNR (2008). Nova Scotia Department of Natural Resources. Mapping Nova Scotia's natural disturbance regimes. Retrieved from http://novascotia.ca/natr/library/forestry/reports/NDRreport3.pdf.

NSDNR (2011). Nova Scotia Department of Natural Resources. The path we share: A natural resource strategy for Nova Scotia 2011-2020. Retrieved from http://novascotia.ca/natr/strategy/pdf/Strategy_Strategy.pdf.

NSDNR (2013). Nova Scotia Department of Natural Resources. Registry of buyers of primary forest products report for 2013-4. Retrieved from http://www.gov.ns.ca/natr/forestry/registry/annual/2013/2012AnnualReport.pdf.

NSENV (2011). Nova Scotia Environment. Primary Watersheds of Nova Scotia. Retrieved from http://www.gov.ns.ca/nse/water.strategy/docs/WaterStrategy_NSWatershedMap.pdf.

NSENV (2012). Nova Scotia Environment. Environmental goals and sustainable prosperity act: Progress report. Retrieved from http://gov.ns.ca/nse/egspa/docs/EGSPA.2012.Annual.Report.pdf.

Ohman, K. and Eriksson, L. O. (1998). The core area concept in forming contiguous areas for long-term forest planning. *Canadian Journal of Forest Research*, 28(7):1032–1039.

Ohman, K. and Eriksson, L. O. (2010). Aggregating harvest activities in long term forest planning by minimizing harvest area perimeters. *Silva Fennica*, 44(1):77–89.

O'Keefe, R. and McGrath, T. (2006). Nova scotia hardwood growth and yield model. Retrieved from http://novascotia.ca/natr/library/forestry/reports/REPORT78.PDF.

Paradis, G., LeBel, L., D'Amours, S., and Bouchard, M. (2013). On the risk of systematic drift under incoherent hierarchical forest management planning. *Canadian Journal of Forest Research*, 43(5):480–492.

Pickens, J. B., Kent, B. M., and Ashton, P. G. (1990). The declining even flow effect and the process of national forest planning. *Forest Science*, 36(3):665–679.

Plochmann, R. (1989). The forests of central europe: A changing view. Starker Lectures.

Rasinmäki, J., Mäkinen, A., and Kalliovirta, J. (2009). Simo: an adaptable simulation framework for multiscale forest resource data. *Computers and Electronics in Agriculture*, 66(1):76–84.

Richards, E. W. (1997). *A Tabu Search Method for a Tactical Forest Planning Problem*. PhD thesis, Dalhousie University.

Richards, E. W. and Gunn, E. A. (2000). A model and tabu search method to optimize stand harvest and road construction schedules. *Forest Science*, 46(2):188–203.

Robak, T. and Richards, E. (2001). Multi-year and annual forest operations planning using optimization and simulation. In *Proceedings of the 5th Brazilian Harvesting Symposium*, Porto Seguro, Brazil. The Organizing Committee.

Ronnqvist, M. (2003). Optimization in forestry. *Mathematical Programming*, 97(1):267–284.

Rouillard, D. and Moore, T. (2008). Patching together the future of forest modelling: Implementing a spatial model in the 2009 romeo malette forest management plan. *The Forestry Chronicle*, 84(5):718–730.

Snyder, S. and ReVelle, C. (1997). Dynamic selection of harvests with adjacency restrictions: the share model. *Forest Science*, 43(2):213–222.

Stewart, B. and Neilly, P. (2008). A procedural guide for ecological landscape analysis. Retrieved from http://novascotia.ca/natr/forestry/reports/Procedural-Guide-For-Ecological-Landscape-Analysis.pdf.

U.S. Congress (1960). The Multiple Use Sustained Yield Act of June 12, 1960. 74 Stat. 215; 16 U.S.C. 528-531, Public Law 86-517.

Vertinsky, I., Brown, S., Schreier, H., Thompson, W. A., and Kooten, G. C. V. (1994). A hierarchical-gis-based decision model for forest management: The systems approach. *Interfaces*, 24(4):38–53.

Ware, G. O. and Clutter, J. L. (1971). A mathematical programming system for the management of industrial forests. *Forest Science*, 17:428–445.

Webb, K. and Marshall, I. (1999). Environment Canada. ecoregions and ecodistricts of canada. retrieved from http://sis.agr.gc.ca/cansis/publications/surveys/ns/nsee/nsee_report.pdf.

Weintraub, A., Barahona, F., and Epstein, R. (1994). A column generation algorithm for solving general forest planning problems with adjacency constraints. *Forest science*, 40(1):142–161.

Weintraub, A. and Bare, B. B. (1996). New issues in forest land management from an operations research perspective. *Interfaces*, 26(5):9–25.

Weintraub, A. and Cholaky, A. (1991). A hierarchical approach to forest planning. *Forest Science*, 37(2):439–460.

Weintraub, A. and Davis, L. (1996). Hierarchical planning in forest resource management: Defining the dimensions of a subject area. In Martell, D.L., D. L. S. and Weintraub, A., editors, *Proceedings of a Workshop on Hierarchical Approaches to Forest Management in Public and Private Organizations*. Petwawa National Forestry Institute. Information Report PI-X-124. –8.

Weintraub, A. and Navon, D. (1976). A forest management planning model integrating silvicultural and transportation activities. *Management Science*, 22(12):1299–1309.

Weintraub, A. and Romero, C. (2006). Operations research models and the management of agricultural and forestry resources: a review and comparison. *Interfaces*, 36(5):446–457.

Wikström, P., Edenius, L., Elfving, B., Eriksson, L. O., Lämås, T., Sonesson, J., Öhman, K., Wallerman, J., Waller, C., and Klintebäck, F. (2011). The heureka forestry decision support system: an overview. *Mathematical and Computational Forestry & Natural-Resource Sciences (MCFNS)*, 3(2):Pages–87.

Yang, W. (2008). Advanced linear programming methods for spatial forest management. M.A.Sc, Dalhousie University.

Appendix A

Supplement to Chapter 4

A.1 Crown Central Forest Age Distribution

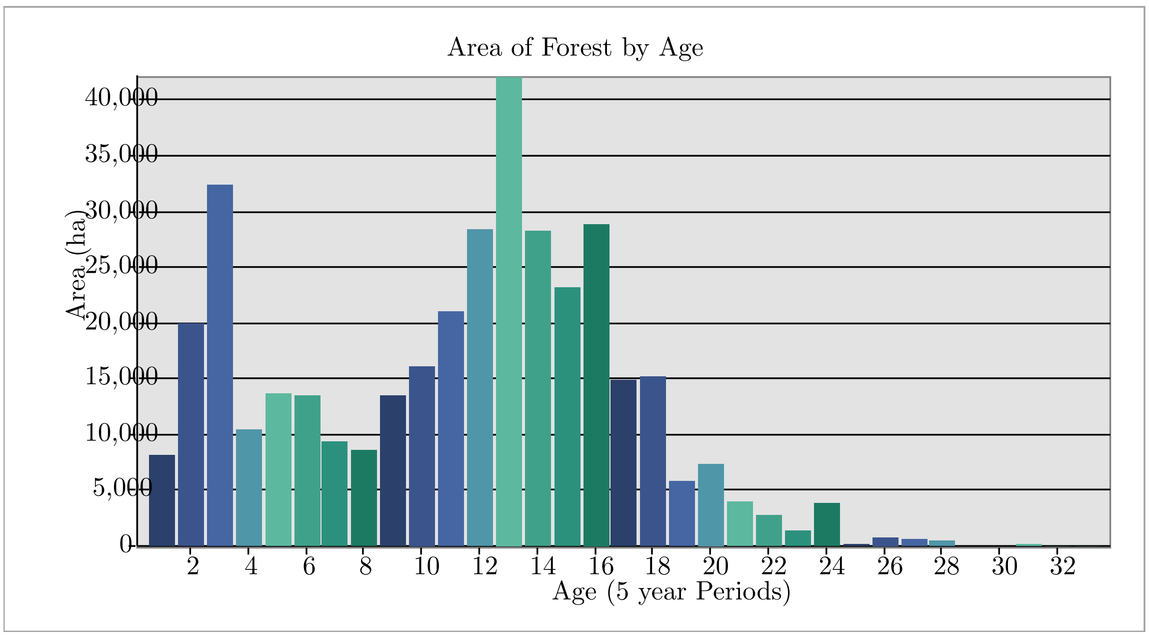

Figure A.1: Crown Central Forest Age Distribution

A.2 Model One Prescriptions

A.2.1 A Note on Model One Prescription Modelling

In models of forest management strategy, a prescription is a series of interventions spanning the planning horizon. The modelling of prescriptions employed here is similar to that of Woodstock in that interventions are abstracted into intervention ages, transitions, and have eligibility criteria. That being said, the method employed to generate prescriptions for Model One models is different from Woodstock prescription generation, largely because entire prescriptions are specified in Model One and single interventions in Model Two. The development of a means to easily generate Model One prescriptions could be an unadvertised contribution to forest management strategy modelling of this thesis.

The actual process of developing the prescription generating code involved writing it first in Visual Basic (VB), then rewriting it procedurally in Python, and then rewriting it using object oriented programming again in Python. This process of rewriting the source code several times was essential to distilling what started out as a monolithic block of VB code into a user friendly set of python modules. It also bears mentioning that if the prescription generator had languished as VB or procedural python code, it would not have been possible to achieve the objective function values and solution times presented in Chapter 4.4. This is due to the difficulty of adding and removing prescriptions to the early routines. The purpose of this note is to explain how Model One prescriptions were generated for models in this thesis. The source code for the prescription generator can be found in section A.2.5 of this appendix.

Programmatically, prescriptions were represented as a set of five lists that detail the age of the stand when each intervention takes place, the age a stand transitions to as a result of each intervention, the state change of a stand as a result of each intervention, the intervention methods applied, and the eligibility criteria for the prescription. An example will help illustrate how these are modelled.

Prescription: For site-class 5 stands in a natural state - Clearcut at age 75 (15 periods); regenerate naturally, then commercially thin at age 60 (12 periods), and clearcut the thinned stand at age 80; then, repeat the thinning and clearcut. Referring to the lists mentioned above, this prescription is represented as shown in Table A.1 .

List	Data
Intervention ages:	[15, 12, 16, 12, 16]
Transition ages:	[1, 13, 1, 13, 1]
Transition states:	[11, 01, 11, 01, 11]
Harvest method:	[11, 01, 11, 01, 11]
Eligibility:	[site-class=5, state=natural]

Table A.1: Example Prescription

Referring to Table A.1, intervention ages state that interventions occur at ages 15, 12, 16 periods. Transition ages state that as a result of these interventions the stand will transition to age 1 (regeneration) after the first clearcut, 13 (age advances as it would if stand was undisturbed) after the first thinning, then 1 again following the second clearcut entry, and the thinning and clearcut repeat. Transition states communicates that as a result of the clearcut the stand regenerates naturally. This has a state code of 11. Similarly the next entry in this list specifies that as a result of the thinning a stand's state code will change to 01 to reflect that it is a naturally regenerated stand in a commercial thinning state. All state codes are based on the Nova Scotia Department of Natural Resources (NSDNR) Growth and Yield Models; their application here mimics how they are used in the NSDNR Woodstock model. Next, harvest method communicates that the first, third and fifth interventions are clearcuts—harvest code 11—and the second, and fourth interventions are thinnings—harvest code 01. Finally, eligibility states that only site-class 5, natural stands, are eligible for this prescription. This allows for prescriptions to be designed to reflect growing capability of the land, and allows for specific prescriptions applying to only a small number of stands to be defined without bogging the model down by defining these prescriptions for all stands.

Interventions were modelled as described in Table A.2.

Intervention	Modelling
Clearcut	All volume was removed from a stand, age was reset to 1, and state regeneration state was either natural or as a plantation
Thinnings	Volume removed equal to the difference pre-thinning and post-thinning, age advanced as if undisturbed, state changed to reflect the thinning
Selection Harvest	Model the same way as thinnings with the exception that stand age was reset to 13 after the intervention
Shelterwoods	Two entry systems where the first entry took 40% of standing volume and the second entry took remaining 60%; stand regenerated as natural at age 2
Buffers	Volume removed equalled 30% of total volume and no state change

Table A.2: Modelling of Intervention Methods

A module, called the prescription generator, generates a list of prescription objects based on the above mentioned lists, specified as in Table A.1. Many prescriptions had what is called an alternate track, that is if a stand was too old for the prescription, for instance, in the example in Table A.1 if a stand was older than 15 periods in period 1 of the model, it could receive a clearcut in the first five periods, regenerate, and follow the prescription. Alternate tracks are discussed in "A Note on Modelling Prescriptions on Old Stands", found in the next subsection of this appendix chapter.

After the list of prescription objects is generated, a stand table is read in, and the progression of each stand along each is the prescriptions over the planning horizon is recorded in a series of tables. One maps stand, prescription, and period to state transitions; and two others map initial stand age, prescription, and period, to current stand age in one, and harvest method in the other. These tables are formatted for AMPL. Simultaneously, a list of yields required for the model is compiled, and using this list a yield file is constructed from the comprehensive yield table mentioned in Chapter 4.3.

A.2.2 Yield Table Headings

Table A.4 is to be read down columns. The first Yield table heading is "NDR", the second is "Ownership", and the twenty-fifth is "YEHV", etc. Any heading starting with a Y refers to an ecosystem or timber yield code. The yields used in thesis models and their corresponding codes are shown in table A.3.

Code	Yield
YDEVCLS	Development Class
YHRVCLS	Harvest Class
YIHBEROVLO	Intolerant Hardwood, Beech, Red Oak log volume
YIHBEROVLO	Intolerant Hardwood, Beech, Red Oak pulp volume
YPIEHTLVLO	Pine, Eastern Hemlock, Tamarack Larch log volume
YPIEHTLVPL	Pine, Eastern Hemlock, Tamarack Larch pulp volume
YSERCLS	Seral Class
YSMYBVLOG	Sugar Maple, Yellow Birch log volume
YSMYBVPLP	Sugar Maple, Yellow Birch pulp volume
YSPBFVLOG	Spruce-Fir log volume
YSPBFVPLP	Spruce-Fir pulp volume

Table A.3: Model Yields

Yield Table Headings				
NDR	YEHV	YPIEHTLI3	YSMAXMAI	YSVLOG
Ownership	YEHVLOG	YPIEHTLMAX	YSMRSEHV	YSVOLRATHW
Buffer	YEHVPLP	YPIEHTLV	YSMRSEHVLO	YSVOLRATSW
Species	YEXV	YPIEHTLVLO	YSMRSEHVP	YSVPLP
Stocking	YEXVLOG	YPIEHTLVPL	YSMRSEHVPL	YTCAI
Site-Class	YEXVPLP	YPIV	YSMV	YTHV
State	YFORCOMM	YPIVLOG	YSMVLOG	YTHVLOG
Excl	YHCAI	YPIVPLP	YSMVPLP	YTHVPLP
NewProt	YHMAI	YRMV	YSMYBMAXMA	YTLV
AGE	YHMAXMAI	YRMVLOG	YSMYBRORSW	YTLVLOG
AREA	YHP	YRMVPLP	YSMYBROR1	YTLVPLP
Y12SMRSEHV	YHRVCLS	YROV	YSMYBROR2	YTMAI
Y12SMYBROR	YHV	YROVLOG	YSMYBROR3	YTMAXMAI
Y12YBROWPV	YHVLOG	YROVPLP	YSMYBV	YTV
YBEV	YHVPLP	YRPV	YSMYBVLOG	YTVLOG
YBEVLOG	YIHBEROMAX	YRPVLOG	YSMYBVPLP	YTVPLP
YBEVPLP	YIHBEROV	YRPVPLP	YSP	YWBV
YBFV	YIHBEROVLO	YRSV	YSPBFMAXMA	YWBVLOG
YBFVLOG	YIHBEROVPL	YRSVLOG	YSPBFV	YWBVPLP
YBFVPLP	YIHV	YRSVPLP	YSPBFVLOG	YWPV
YBSV	YIHVLOG	YSCAI	YSPBFVPLP	YWPVLOG
YBSVLOG	YIHVPLP	YSERCLS	YSPV	YWPVPLP
YBSVPLP	YPIEHTLIHB	YSERDEVCLS	YSPVLOG	YWSV
YCOVTYPE	YPIEHTLI1	YSERSCORE	YSPVPLP	YWSVLOG
YDEVCLS	YPIEHTLI2	YSMAI	YSV	YWSVPLP

Table A.4: Yield Table Headings

A.2.3 A Note on Modelling Prescriptions on Old Stands

When defining Model One prescriptions, it is intuitive to define them by age. For instance, a clearcut prescription might clearcut at 18 periods (90 years) of age, then regenerate and clearcut at 14 periods, then regenerate and clearcut at 14 periods again. Stands that start the planning horizon younger than, or at, 18 periods of age are eligible for this prescription; stands that are older than 18 periods are not. This means old stands, stands older than 20 periods in the first period, are not eligible for any prescriptions since no prescription has an initial intervention after 20 periods of age. A method needed to be developed to accommodate old stands that was flexible and did not turn assigning prescriptions to stands into an odious task.

The approach taken in this thesis to modelling prescriptions to accommodate all ages of stand has been to break prescriptions up into two tracks: a main track and an alternate track. The main track details the ages and interventions that apply to a stand that is eligible for the first intervention. In the example above, this is any stand less than 19 periods of age. The alternate track is for stands that are too old to be eligible for the first entry. Stands that fall on the old track receive a clearcut in the first 5 periods of modelling and then after regeneration continue the modelling horizon as an eligible stand would on the main track. So in the clearcut example, the alternate track might be wait three periods and then clearcut, regenerate and clearcut at 14 periods, then clearcut at 14 periods again. This approach allows old stands to be harvested without allowing bizarre prescriptions, such as let a 23 period old stand grow for 10 periods before clearcutting.

Reading prescription listings

Each section consists of two listings. The first details prescription number, intervention age, and intervention method. The second details prescription number, transition age, and transition type as a result of the interventions in the first listing. Ages are given in periods (5 years). For the first listing, prescriptions with a first entry in the second column being a number less than 5 (in or not in single quotes), indicate a prescription where that many periods are waited, then the stand is clearcut and continues following the prescription. These correspond to the alternate tracks described above. Harvest Type of '01' is a (pre)-commercial thin, '11' is a clearcut, '12' is a selection harvest, '15' is a shelterwood entry, and '16' is a buffer harvest. Some prescriptions have 31 as an entry age, this is a way of signalling that the stand is not to be harvested after the initial interventions.

For the second listing stand types are as described in Table A.5. The transition age column displays the age a stand transitions to as a result of receiving the corresponding intervention in the first listing, and the transition type column displays what the stand's type changes to as a result of the intervention.

Stand Type Code	Forest State
01	Commercially thinned natural stand
02	Second entry commercially thinned natural stand
03	Second entry commercially thinned pre-commerically thinned stand
04	Second entry commercially thinned plantation stand
05	Commerically thinned pre-commercially thinned stand
06	Commercially thinned plantation stand
07	Pre-commerically thinned natural stand
08	Plantation
10	Natural uneven aged stand
11	Natural even aged stand
12	Selection management in natural
13	Selection management in pre-commerically thinned
14	Selection management in uneven aged natural

Table A.5: Stand Type Codes

A.2.4 Phase 1 Model

Number	Intervention Ages	Intervention Type
0	[100, 12, 24]	['16', '16', '16']
1	[14, 31, 31]	['11', '11', '11']
2	['0', 31, 31]	['11', '11', '11']
3	[15, 31, 31]	['11', '11', '11']
4	['1', 31, 31]	['11', '11', '11']
5	[16, 31, 31]	['11', '11', '11']
6	['2', 31, 31]	['11', '11', '11']
7	[17, 31, 31]	['11', '11', '11']
8	['3', 31, 31]	['11', '11', '11']
9	[18, 31, 31]	['11', '11', '11']
10	['4', 31, 31]	['11', '11', '11']
11	[13, 31, 31]	['11', '11', '11']
12	['0', 31, 31]	['11', '11', '11']
13	[14, 31, 31]	['11', '11', '11']
14	['1', 31, 31]	['11', '11', '11']
15	[15, 31, 31]	['11', '11', '11']
16	['2', 31, 31]	['11', '11', '11']
17	[16, 31, 31]	['11', '11', '11']
18	['3', 31, 31]	['11', '11', '11']
19	[17, 31, 31]	['11', '11', '11']
20	['4', 31, 31]	['11', '11', '11']
21	[12, 31, 31]	['11', '11', '11']
22	['0', 31, 31]	['11', '11', '11']
23	[13, 31, 31]	['11', '11', '11']
24	['1', 31, 31]	['11', '11', '11']
25	[14, 31, 31]	['11', '11', '11']

Number	Intervention Ages	Intervention Type
26	['2', 31, 31]	['11', '11', '11']
27	[15, 31, 31]	['11', '11', '11']
28	['3', 31, 31]	['11', '11', '11']
29	[16, 31, 31]	['11', '11', '11']
30	['4', 31, 31]	['11', '11', '11']
31	[11, 31, 31]	['11', '11', '11']
32	['0', 31, 31]	['11', '11', '11']
33	[12, 31, 31]	['11', '11', '11']
34	['1', 31, 31]	['11', '11', '11']
35	[13, 31, 31]	['11', '11', '11']
36	['2', 31, 31]	['11', '11', '11']
37	[14, 31, 31]	['11', '11', '11']
38	['3', 31, 31]	['11', '11', '11']
39	[15, 31, 31]	['11', '11', '11']
40	['4', 31, 31]	['11', '11', '11']
41	[14, 14, 14]	['11', '11', '11']
42	['0', 14, 14]	['11', '11', '11']
43	[15, 14, 14]	['11', '11', '11']
44	['1', 14, 14]	['11', '11', '11']
45	[16, 14, 14]	['11', '11', '11']
46	['2', 14, 14]	['11', '11', '11']
47	[17, 14, 14]	['11', '11', '11']
48	['3', 14, 14]	['11', '11', '11']
49	[18, 14, 14]	['11', '11', '11']
50	['4', 14, 14]	['11', '11', '11']
51	[13, 13, 13]	['11', '11', '11']

Number	Intervention Ages	Intervention Type
52	['0', 13, 13]	['11', '11', '11']
53	[14, 13, 13]	['11', '11', '11']
54	['1', 13, 13]	['11', '11', '11']
55	[15, 13, 13]	['11', '11', '11']
56	['2', 13, 13]	['11', '11', '11']
57	[16, 13, 13]	['11', '11', '11']
58	['3', 13, 13]	['11', '11', '11']
59	[17, 13, 13]	['11', '11', '11']
60	['4', 13, 13]	['11', '11', '11']
61	[12, 12, 12]	['11', '11', '11']
62	['0', 12, 12]	['11', '11', '11']
63	[13, 12, 12]	['11', '11', '11']
64	['1', 12, 12]	['11', '11', '11']
65	[14, 12, 12]	['11', '11', '11']
66	['2', 12, 12]	['11', '11', '11']
67	[15, 12, 12]	['11', '11', '11']
68	['3', 12, 12]	['11', '11', '11']
69	[16, 12, 12]	['11', '11', '11']
70	['4', 12, 12]	['11', '11', '11']
71	[11, 11, 11]	['11', '11', '11']
72	['0', 11, 11]	['11', '11', '11']
73	[12, 11, 11]	['11', '11', '11']
74	['1', 11, 11]	['11', '11', '11']
75	[13, 11, 11]	['11', '11', '11']
76	['2', 11, 11]	['11', '11', '11']
77	[14, 11, 11]	['11', '11', '11']

Number	Intervention Ages	Intervention Type
78	['3', 11, 11]	['11', '11', '11']
79	[15, 11, 11]	['11', '11', '11']
80	['4', 11, 11]	['11', '11', '11']
81	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
82	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
83	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
84	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
85	[14, 16, 14, 14]	['15', '15', '11', '11', '11']
86	[14, 16, 13, 13]	['15', '15', '11', '11', '11']
87	[14, 16, 12, 12]	['15', '15', '11', '11', '11']
88	[14, 16, 11, 11]	['15', '15', '11', '11', '11']
89	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
90	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
91	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
92	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
93	[17, 21, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
94	[17, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
95	[21, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
96	[19, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
97	[17, 8, 13, 8, 13]	['11', '01', '11', '01', '11']
98	[15, 19, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
99	[15, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
100	[13, 18, 8, 13, 8, 13]	['01', '11', '01', '11', '01', '11']
101	[10, 14, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
102	[10, 15, 19, 10, 15, 10, 15]	['01', '01', '11', '01', '11', '01', '11']
103	[8, 13, 17, 8, 13, 8, 13]	['01', '01', '11', '01', '11', '01', '11']

Number	Intervention Ages	Intervention Type
104	[8, 13, 8, 13, 8, 13]	['01', '11', '01', '11', '01', '11']
105	[8, 8, 13, 8, 13]	['11', '01', '11', '01', '11']
106	[12, 17, 21, 10, 15, 10, 15]	['01', '01', '11', '01', '11', '01', '11']
107	[12, 17, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
108	[17, 21, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01','11']
109	[17, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01','11']
110	[21, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01','11']
111	[19, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01','11']
112	[17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11']
113	[15, 19, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01','11']
114	[15, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01','11']
115	[13, 18, 8, 13, 17, 8, 13]	['01', '11', '01', '01', '11', '01', '01', '11']
116	[10, 14, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01','11']
117	[10, 15, 19, 10, 15, 19, 10, 15]	['01', '01', '11', '01', '01', '11', '01', '01','11']
118	[8, 13, 17, 8, 13, 17, 8, 13]	['01', '01', '11', '01', '01', '11', '01', '01', '11']
119	[8, 13, 8, 13, 17, 8, 13]	['01', '11', '01', '01', '11', '01', '01', '11']
120	[8, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11']
121	[12, 17, 21, 10, 15, 19, 10, 15]	['01', '01', '11', '01', '01', '11', '01', '01','11']
122	[12, 17, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01','11']
123	[13, 18, 14, 14]	['01', '11', '11', '11']
124	[8, 13, 14, 14]	['01', '11', '11', '11']
125	[8, 13, 17, 14, 14]	['01', '01', '11', '11', '11']
126	[13, 18, 13, 13]	['01', '11', '11', '11']
127	[8, 13, 13, 13]	['01', '11', '11', '11']

Number	Intervention Ages	Intervention Type
128	[8, 13, 17, 13, 13]	['01', '01', '11', '11', '11']
129	[13, 18, 12, 12]	['01', '11', '11', '11']
130	[8, 13, 12, 12]	['01', '11', '11', '11']
131	[8, 13, 17, 12, 12]	['01', '01', '11', '11', '11']
132	[13, 18, 11, 11]	['01', '11', '11', '11']
133	[8, 13, 11, 11]	['01', '11', '11', '11']
134	[8, 13, 17, 11, 11]	['01', '01', '11', '11', '11']
135	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
136	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
137	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
138	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
139	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
140	[16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12']
141	[12, 24, 12, 24]	['16', '16', '16', '16']

Number	Transition Ages	Transition Type
0	[1, 13, 1]	['11', '11', '11']
1	[1, 1, 1]	['11', '11', '11']
2	[1, 1, 1]	['11', '11', '11']
3	[1, 1, 1]	['11', '11', '11']
4	[1, 1, 1]	['11', '11', '11']
5	[1, 1, 1]	['11', '11', '11']
6	[1, 1, 1]	['11', '11', '11']
7	[1, 1, 1]	['11', '11', '11']
8	[1, 1, 1]	['11', '11', '11']
9	[1, 1, 1]	['11', '11', '11']
10	[1, 1, 1]	['11', '11', '11']
11	[1, 1, 1]	['11', '11', '11']
12	[1, 1, 1]	['11', '11', '11']
13	[1, 1, 1]	['11', '11', '11']
14	[1, 1, 1]	['11', '11', '11']
15	[1, 1, 1]	['11', '11', '11']
16	[1, 1, 1]	['11', '11', '11']
17	[1, 1, 1]	['11', '11', '11']
18	[1, 1, 1]	['11', '11', '11']
19	[1, 1, 1]	['11', '11', '11']
20	[1, 1, 1]	['11', '11', '11']
21	[1, 1, 1]	['11', '11', '11']
22	[1, 1, 1]	['11', '11', '11']
23	[1, 1, 1]	['11', '11', '11']
24	[1, 1, 1]	['11', '11', '11']
25	[1, 1, 1]	['11', '11', '11']

Number	Transition Ages	Transition Type
26	[1, 1, 1]	['11', '11', '11']
27	[1, 1, 1]	['11', '11', '11']
28	[1, 1, 1]	['11', '11', '11']
29	[1, 1, 1]	['11', '11', '11']
30	[1, 1, 1]	['11', '11', '11']
31	[1, 1, 1]	['11', '11', '11']
32	[1, 1, 1]	['11', '11', '11']
33	[1, 1, 1]	['11', '11', '11']
34	[1, 1, 1]	['11', '11', '11']
35	[1, 1, 1]	['11', '11', '11']
36	[1, 1, 1]	['11', '11', '11']
37	[1, 1, 1]	['11', '11', '11']
38	[1, 1, 1]	['11', '11', '11']
39	[1, 1, 1]	['11', '11', '11']
40	[1, 1, 1]	['11', '11', '11']
41	[1, 1, 1]	['11', '11', '11']
42	[1, 1, 1]	['11', '11', '11']
43	[1, 1, 1]	['11', '11', '11']
44	[1, 1, 1]	['11', '11', '11']
45	[1, 1, 1]	['11', '11', '11']
46	[1, 1, 1]	['11', '11', '11']
47	[1, 1, 1]	['11', '11', '11']
48	[1, 1, 1]	['11', '11', '11']
49	[1, 1, 1]	['11', '11', '11']
50	[1, 1, 1]	['11', '11', '11']
51	[1, 1, 1]	['11', '11', '11']

Number	Transition Ages	Transition Type
52	[1, 1, 1]	['11', '11', '11']
53	[1, 1, 1]	['11', '11', '11']
54	[1, 1, 1]	['11', '11', '11']
55	[1, 1, 1]	['11', '11', '11']
56	[1, 1, 1]	['11', '11', '11']
57	[1, 1, 1]	['11', '11', '11']
58	[1, 1, 1]	['11', '11', '11']
59	[1, 1, 1]	['11', '11', '11']
60	[1, 1, 1]	['11', '11', '11']
61	[1, 1, 1]	['11', '11', '11']
62	[1, 1, 1]	['11', '11', '11']
63	[1, 1, 1]	['11', '11', '11']
64	[1, 1, 1]	['11', '11', '11']
65	[1, 1, 1]	['11', '11', '11']
66	[1, 1, 1]	['11', '11', '11']
67	[1, 1, 1]	['11', '11', '11']
68	[1, 1, 1]	['11', '11', '11']
69	[1, 1, 1]	['11', '11', '11']
70	[1, 1, 1]	['11', '11', '11']
71	[1, 1, 1]	['11', '11', '11']
72	[1, 1, 1]	['11', '11', '11']
73	[1, 1, 1]	['11', '11', '11']
74	[1, 1, 1]	['11', '11', '11']
75	[1, 1, 1]	['11', '11', '11']
76	[1, 1, 1]	['11', '11', '11']
77	[1, 1, 1]	['11', '11', '11']

Number	Transition Ages	Transition Type
78	[1, 1, 1]	['11', '11', '11']
79	[1, 1, 1]	['11', '11', '11']
80	[1, 1, 1]	['11', '11', '11']
81	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
82	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
83	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
84	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
85	[15, 2, 1, 1]	['11', '11', '11', '11']
86	[15, 2, 1, 1]	['11', '11', '11', '11']
87	[15, 2, 1, 1]	['11', '11', '11', '11']
88	[15, 2, 1, 1]	['11', '11', '11', '11']
89	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
90	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
91	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
92	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
93	[18, 1, 11, 1, 11, 1]	['02', '07', '05', '07', '05', '07']
94	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
95	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
96	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
97	[1, 9, 1, 9, 1]	['08', '06', '08', '06', '08']
98	[16, 1, 11, 1, 11, 1]	['03', '07', '05', '07', '05', '07']
99	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
100	[14, 1, 9, 1, 9, 1]	['04', '08', '06', '08', '06', '08']
101	[11, 1, 11, 1, 11, 1]	['05', '07', '05', '07', '05', '07']
102	[11, 16, 1, 11, 1, 11, 1]	['05', '03', '07', '05', '07', '05', '07']
103	[9, 14, 1, 9, 1, 9, 1]	['06', '04', '08', '06', '08', '06', '08']

Number	Transition Ages	Transition Type
104	[9, 1, 9, 1, 9, 1]	[’06’, ’08’, ’06’, ’08’, ’06’, ’08’]
105	[1, 9, 1, 9, 1]	[’08’, ’06’, ’08’, ’06’, ’08’]
106	[13, 18, 1, 11, 1, 11, 1]	[’01’, ’02’, ’07’, ’05’, ’07’, ’05’, ’07’]
107	[13, 1, 11, 1, 11, 1]	[’01’, ’07’, ’05’, ’07’, ’05’, ’07’]
108	[18, 1, 11, 16, 1, 11, 16, 1]	[’02’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
109	[1, 11, 16, 1, 11, 16, 1]	[’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
110	[1, 11, 16, 1, 11, 16, 1]	[’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
111	[1, 11, 16, 1, 11, 16, 1]	[’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
112	[1, 9, 14, 1, 9, 14, 1]	[’08’, ’06’, ’04’, ’08’, ’06’, ’04’, ’08’]
113	[16, 1, 11, 16, 1, 11, 16, 1]	[’03’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
114	[1, 11, 16, 1, 11, 16, 1]	[’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
115	[14, 1, 9, 14, 1, 9, 14, 1]	[’04’, ’08’, ’06’, ’04’, ’08’, ’06’, ’04’, ’08’]
116	[11, 1, 11, 16, 1, 11, 16, 1]	[’05’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
117	[11, 16, 1, 11, 16, 1, 11, 16, 1]	[’05’, ’03’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
118	[9, 14, 1, 9, 14, 1, 9, 14, 1]	[’06’, ’04’, ’08’, ’06’, ’04’, ’08’, ’06’, ’04’, ’08’]
119	[9, 1, 9, 14, 1, 9, 14, 1]	[’06’, ’08’, ’06’, ’04’, ’08’, ’06’, ’04’, ’08’]
120	[1, 9, 14, 1, 9, 14, 1]	[’08’, ’06’, ’04’, ’08’, ’06’, ’04’, ’08’]
121	[13, 18, 1, 11, 16, 1, 11, 16, 1]	[’01’, ’02’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
122	[13, 1, 11, 16, 1, 11, 16, 1]	[’01’, ’07’, ’05’, ’03’, ’07’, ’05’, ’03’, ’07’]
123	[14, 1, 1, 1]	[’04’, ’08’, ’08’, ’08’]
124	[9, 1, 1, 1]	[’06’, ’08’, ’08’, ’08’]
125	[9, 14, 1, 1, 1]	[’06’, ’04’, ’08’, ’08’, ’08’]
126	[14, 1, 1, 1]	[’04’, ’08’, ’08’, ’08’]
127	[9, 1, 1, 1]	[’06’, ’08’, ’08’, ’08’]

Number	Transition Ages	Transition Type
128	[9, 14, 1, 1, 1]	[’06’, ’04’, ’08’, ’08’, ’08’]
129	[14, 1, 1, 1]	[’04’, ’08’, ’08’, ’08’]
130	[9, 1, 1, 1]	[’06’, ’08’, ’08’, ’08’]
131	[9, 14, 1, 1, 1]	[’06’, ’04’, ’08’, ’08’, ’08’]
132	[14, 1, 1, 1]	[’04’, ’08’, ’08’, ’08’]
133	[9, 1, 1, 1]	[’06’, ’08’, ’08’, ’08’]
134	[9, 14, 1, 1, 1]	[’06’, ’04’, ’08’, ’08’, ’08’]
135	[13, 13, 13, 13, 13, 13, 13, 13]	[’13’, ’13’, ’13’, ’13’, ’13’, ’13’, ’13’, ’13’]
136	[13, 13, 13, 13, 13, 13, 13, 13]	[’14’, ’14’, ’14’, ’14’, ’14’, ’14’, ’14’, ’14”]
137	[13, 13, 13, 13, 13, 13, 13, 13]	[’12’, ’12’, ’12’, ’12’, ’12’, ’12’, ’12’, ’12”]
138	[13, 13, 13, 13, 13, 13, 13, 13]	[’12’, ’12’, ’12’, ’12’, ’12’, ’12’, ’12’, ’12’]
139	[13, 13, 13, 13, 13, 13, 13, 13]	[’13’, ’13’, ’13’, ’13’, ’13’, ’13’, ’13’, ’13’]
140	[13, 13, 13, 13, 13, 13, 13, 13]	[’14’, ’14’, ’14’, ’14’, ’14’, ’14’, ’14’, ’14’]
141	[13, 1, 13, 1]	[’11’, ’11’, ’11’, ’11’]

A.2.5 Phase 2 Model

Number	Intervention Ages	Intervention Type
0	[100, 12, 24]	['16', '16', '16']
1	[14, 31, 31]	['11', '11', '11']
2	['0', 31, 31]	['11', '11', '11']
3	[15, 31, 31]	['11', '11', '11']
4	['1', 31, 31]	['11', '11', '11']
5	[16, 31, 31]	['11', '11', '11']
6	['2', 31, 31]	['11', '11', '11']
7	[17, 31, 31]	['11', '11', '11']
8	['3', 31, 31]	['11', '11', '11']
9	[18, 31, 31]	['11', '11', '11']
10	['4', 31, 31]	['11', '11', '11']
11	[13, 31, 31]	['11', '11', '11']
12	['0', 31, 31]	['11', '11', '11']
13	[14, 31, 31]	['11', '11', '11']
14	['1', 31, 31]	['11', '11', '11']
15	[15, 31, 31]	['11', '11', '11']
16	['2', 31, 31]	['11', '11', '11']
17	[16, 31, 31]	['11', '11', '11']
18	['3', 31, 31]	['11', '11', '11']
19	[17, 31, 31]	['11', '11', '11']
20	['4', 31, 31]	['11', '11', '11']
21	[12, 31, 31]	['11', '11', '11']
22	['0', 31, 31]	['11', '11', '11']
23	[13, 31, 31]	['11', '11', '11']
24	['1', 31, 31]	['11', '11', '11']
25	[14, 31, 31]	['11', '11', '11']

Number	Intervention Ages	Intervention Type
26	['2', 31, 31]	['11', '11', '11']
27	[15, 31, 31]	['11', '11', '11']
28	['3', 31, 31]	['11', '11', '11']
29	[16, 31, 31]	['11', '11', '11']
30	['4', 31, 31]	['11', '11', '11']
31	[11, 31, 31]	['11', '11', '11']
32	['0', 31, 31]	['11', '11', '11']
33	[12, 31, 31]	['11', '11', '11']
34	['1', 31, 31]	['11', '11', '11']
35	[13, 31, 31]	['11', '11', '11']
36	['2', 31, 31]	['11', '11', '11']
37	[14, 31, 31]	['11', '11', '11']
38	['3', 31, 31]	['11', '11', '11']
39	[15, 31, 31]	['11', '11', '11']
40	['4', 31, 31]	['11', '11', '11']
41	[14, 14, 14]	['11', '11', '11']
42	['0', 14, 14]	['11', '11', '11']
43	[15, 14, 14]	['11', '11', '11']
44	['1', 14, 14]	['11', '11', '11']
45	[16, 14, 14]	['11', '11', '11']
46	['2', 14, 14]	['11', '11', '11']
47	[17, 14, 14]	['11', '11', '11']
48	['3', 14, 14]	['11', '11', '11']
49	[18, 14, 14]	['11', '11', '11']
50	['4', 14, 14]	['11', '11', '11']
51	[13, 13, 13]	['11', '11', '11']
52	['0', 13, 13]	['11', '11', '11']
53	[14, 13, 13]	['11', '11', '11']
54	['1', 13, 13]	['11', '11', '11']
55	[15, 13, 13]	['11', '11', '11']
56	['2', 13, 13]	['11', '11', '11']
57	[16, 13, 13]	['11', '11', '11']
58	['3', 13, 13]	['11', '11', '11']
59	[17, 13, 13]	['11', '11', '11']
60	['4', 13, 13]	['11', '11', '11']
61	[12, 12, 12]	['11', '11', '11']
62	['0', 12, 12]	['11', '11', '11']
63	[13, 12, 12]	['11', '11', '11']
64	['1', 12, 12]	['11', '11', '11']
65	[14, 12, 12]	['11', '11', '11']

Number	Intervention Ages	Intervention Type
66	['2', 12, 12]	['11', '11', '11']
67	[15, 12, 12]	['11', '11', '11']
68	['3', 12, 12]	['11', '11', '11']
69	[16, 12, 12]	['11', '11', '11']
70	['4', 12, 12]	['11', '11', '11']
71	[11, 11, 11]	['11', '11', '11']
72	['0', 11, 11]	['11', '11', '11']
73	[12, 11, 11]	['11', '11', '11']
74	['1', 11, 11]	['11', '11', '11']
75	[13, 11, 11]	['11', '11', '11']
76	['2', 11, 11]	['11', '11', '11']
77	[14, 11, 11]	['11', '11', '11']
78	['3', 11, 11]	['11', '11', '11']
79	[15, 11, 11]	['11', '11', '11']
80	['4', 11, 11]	['11', '11', '11']
81	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
82	['0', 14, 16, 14]	['11', '15', '15', '15', '15']
83	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
84	['1', 14, 16, 14]	['11', '15', '15', '15', '15']
85	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
86	['2', 14, 16, 14]	['11', '15', '15', '15', '15']
87	[14, 16, 14, 16, 14, 16]	['15', '15', '15', '15', '15', '15']
88	['3', 14, 16, 14]	['11', '15', '15', '15', '15']
89	[14, 16, 14, 14]	['15', '15', '11', '11', '11']
90	[14, 16, 13, 13]	['15', '15', '11', '11', '11']
91	[14, 16, 12, 12]	['15', '15', '11', '11', '11']
92	[14, 16, 11, 11]	['15', '15', '11', '11', '11']
93	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
94	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
95	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
96	[14, 16, 3, 10, 15, 3, 10, 15]	['15', '15', '01', '01', '11', '01', '01', '11']
97	[17, 21, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
98	[3, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
99	[17, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
100	[4, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
101	[21, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
102	[0, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
103	[19, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
104	[1, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
105	[17, 8, 13, 8, 13]	['11', '01', '11', '01', '11']
106	[2, 8, 13, 8, 13, 8, 13]	['11', '01', '11', '01', '11', '01', '11', '01']
107	[15, 19, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
108	[3, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
109	[15, 10, 15, 10, 15]	['11', '01', '11', '01', '11']
110	[4, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']

Number	Intervention Ages	Intervention Type
111	[13, 18, 8, 13, 8, 13]	['01', '11', '01', '11', '01', '11']
112	[0, 8, 13, 8, 13, 8, 13]	['11', '01', '11', '01', '11', '01', '11', '01']
113	[10, 14, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
114	[1, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
115	[10, 15, 19, 10, 15, 10, 15]	['01', '01', '11', '01', '11', '01', '11']
116	[2, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
117	[8, 13, 17, 8, 13, 8, 13]	['01', '01', '11', '01', '11', '01', '11']
118	[3, 8, 13, 8, 13, 8, 13]	['11', '01', '11', '01', '11', '01', '11', '01']
119	[8, 13, 8, 13, 8, 13]	['01', '11', '01', '11', '01', '11']
120	[4, 8, 13, 8, 13, 8, 13]	['11', '01', '11', '01', '11', '01', '11', '01']
121	[8, 8, 13, 8, 13]	['11', '01', '11', '01', '11']
122	[0, 8, 13, 8, 13, 8, 13]	['11', '01', '11', '01', '11', '01', '11', '01']
123	[12, 17, 21, 10, 15, 10, 15]	['01', '01', '11', '01', '11', '01', '11']
124	[1, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
125	[12, 17, 10, 15, 10, 15]	['01', '11', '01', '11', '01', '11']
126	[2, 3, 10, 15, 3, 10, 15, 3]	['11', '01', '01', '11', '01', '01', '11', '01', '01', '11']
127	[17, 21, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01', '11']
128	[3, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
129	[17, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01', '11']
130	[4, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
131	[21, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01', '11']
132	[0, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
133	[19, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01', '11']
134	[1, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
135	[17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11']
136	[2, 8, 13, 17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11', '01']
137	[15, 19, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01', '11']
138	[3, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
139	[15, 10, 15, 19, 10, 15]	['11', '01', '01', '11', '01', '01', '11']
140	[4, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
141	[13, 18, 8, 13, 17, 8, 13]	['01', '11', '01', '01', '11', '01', '01', '11']
142	[0, 8, 13, 17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11', '01']
143	[10, 14, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01', '11']
144	[1, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
145	[10, 15, 19, 10, 15, 19, 10, 15]	['01', '01', '11', '01', '01', '11', '01', '01', '11']
146	[2, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
147	[8, 13, 17, 8, 13, 17, 8, 13]	['01', '01', '11', '01', '01', '11', '01', '01', '11']
148	[3, 8, 13, 17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11', '01']
149	[8, 13, 8, 13, 17, 8, 13]	['01', '11', '01', '01', '11', '01', '01', '11']
150	[4, 8, 13, 17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11', '01']
151	[8, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11']
152	[0, 8, 13, 17, 8, 13, 17, 8, 13]	['11', '01', '01', '11', '01', '01', '11', '01']
153	[12, 17, 21, 10, 15, 19, 10, 15]	['01', '01', '11', '01', '01', '11', '01', '01', '11']
154	[1, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']
155	[12, 17, 10, 15, 19, 10, 15]	['01', '11', '01', '01', '11', '01', '01', '11']
156	[2, 3, 10, 15, 19, 3, 10, 15]	['11', '01', '01', '01', '11', '01', '01', '01', '11']

Number	Intervention Ages	Intervention Type
157	[17, 21, 14, 14]	['01', '11', '11', '11']
158	[15, 19, 14, 14]	['01', '11', '11', '11']
159	[13, 18, 14, 14]	['01', '11', '11', '11']
160	[10, 14, 14, 14]	['01', '11', '11', '11']
161	[10, 15, 19, 14, 14]	['01', '01', '11', '11', '11']
162	[8, 13, 14, 14]	['01', '11', '11', '11']
163	[8, 13, 17, 14, 14]	['01', '01', '11', '11', '11']
164	[12, 17, 21, 14, 14]	['01', '01', '11', '11', '11']
165	[12, 17, 14, 14]	['01', '11', '11', '11']
166	[17, 21, 13, 13]	['01', '11', '11', '11']
167	[15, 19, 13, 13]	['01', '11', '11', '11']
168	[13, 18, 13, 13]	['01', '11', '11', '11']
169	[10, 14, 13, 13]	['01', '11', '11', '11']
170	[10, 15, 19, 13, 13]	['01', '01', '11', '11', '11']
171	[8, 13, 13, 13]	['01', '11', '11', '11']
172	[8, 13, 17, 13, 13]	['01', '01', '11', '11', '11']
173	[12, 17, 21, 13, 13]	['01', '01', '11', '11', '11']
174	[12, 17, 13, 13]	['01', '11', '11', '11']
175	[17, 21, 12, 12]	['01', '11', '11', '11']
176	[15, 19, 12, 12]	['01', '11', '11', '11']
177	[13, 18, 12, 12]	['01', '11', '11', '11']
178	[10, 14, 12, 12]	['01', '11', '11', '11']
179	[10, 15, 19, 12, 12]	['01', '01', '11', '11', '11']
180	[8, 13, 12, 12]	['01', '11', '11', '11']
181	[8, 13, 17, 12, 12]	['01', '01', '11', '11', '11']
182	[12, 17, 21, 12, 12]	['01', '01', '11', '11', '11']
183	[12, 17, 12, 12]	['01', '11', '11', '11']
184	[17, 21, 11, 11]	['01', '11', '11', '11']
185	[15, 19, 11, 11]	['01', '11', '11', '11']
186	[13, 18, 11, 11]	['01', '11', '11', '11']
187	[10, 14, 11, 11]	['01', '11', '11', '11']
188	[10, 15, 19, 11, 11]	['01', '01', '11', '11', '11']
189	[8, 13, 11, 11]	['01', '11', '11', '11']
190	[8, 13, 17, 11, 11]	['01', '01', '11', '11', '11']
191	[12, 17, 21, 11, 11]	['01', '01', '11', '11', '11']
192	[12, 17, 11, 11]	['01', '11', '11', '11']

Number	Intervention Ages	Intervention Type
193	[16, 16, 16, 16, 16, 16, 16, 16,16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
194	[4, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12"]
195	[16, 16, 16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
196	[0, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12']
197	[16, 16, 16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
198	[1, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12']
199	[16, 16, 16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
200	[2, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12']
201	[16, 16, 16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
202	[3, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12']
203	[16, 16, 16, 16, 16, 16, 16, 16, 16, 16]	['12', '12', '12', '12', '12', '12', '12', '12', '12', '12']
204	[4, 3, 16, 16, 16, 16, 16, 16, 16, 16]	['11', '01', '12', '12', '12', '12', '12', '12', '12']
205	[12, 24, 12, 24]	['16', '16', '16', '16']

Number	Transition Ages	Transition Type
0	[1, 13, 1]	['11', '11', '11']
1	[1, 1, 1]	['11', '11', '11']
2	[1, 1, 1]	['11', '11', '11']
3	[1, 1, 1]	['11', '11', '11']
4	[1, 1, 1]	['11', '11', '11']
5	[1, 1, 1]	['11', '11', '11']
6	[1, 1, 1]	['11', '11', '11']
7	[1, 1, 1]	['11', '11', '11']
8	[1, 1, 1]	['11', '11', '11']
9	[1, 1, 1]	['11', '11', '11']
10	[1, 1, 1]	['11', '11', '11']
11	[1, 1, 1]	['11', '11', '11']
12	[1, 1, 1]	['11', '11', '11']
13	[1, 1, 1]	['11', '11', '11']
14	[1, 1, 1]	['11', '11', '11']
15	[1, 1, 1]	['11', '11', '11']
16	[1, 1, 1]	['11', '11', '11']
17	[1, 1, 1]	['11', '11', '11']
18	[1, 1, 1]	['11', '11', '11']
19	[1, 1, 1]	['11', '11', '11']
20	[1, 1, 1]	['11', '11', '11']
21	[1, 1, 1]	['11', '11', '11']
22	[1, 1, 1]	['11', '11', '11']
23	[1, 1, 1]	['11', '11', '11']
24	[1, 1, 1]	['11', '11', '11']
25	[1, 1, 1]	['11', '11', '11']

Number	Transition Ages	Transition Type
26	[1, 1, 1]	['11', '11', '11']
27	[1, 1, 1]	['11', '11', '11']
28	[1, 1, 1]	['11', '11', '11']
29	[1, 1, 1]	['11', '11', '11']
30	[1, 1, 1]	['11', '11', '11']
31	[1, 1, 1]	['11', '11', '11']
32	[1, 1, 1]	['11', '11', '11']
33	[1, 1, 1]	['11', '11', '11']
34	[1, 1, 1]	['11', '11', '11']
35	[1, 1, 1]	['11', '11', '11']
36	[1, 1, 1]	['11', '11', '11']
37	[1, 1, 1]	['11', '11', '11']
38	[1, 1, 1]	['11', '11', '11']
39	[1, 1, 1]	['11', '11', '11']
40	[1, 1, 1]	['11', '11', '11']
41	[1, 1, 1]	['11', '11', '11']
42	[1, 1, 1]	['11', '11', '11']
43	[1, 1, 1]	['11', '11', '11']
44	[1, 1, 1]	['11', '11', '11']
45	[1, 1, 1]	['11', '11', '11']
46	[1, 1, 1]	['11', '11', '11']
47	[1, 1, 1]	['11', '11', '11']
48	[1, 1, 1]	['11', '11', '11']
49	[1, 1, 1]	['11', '11', '11']
50	[1, 1, 1]	['11', '11', '11']
51	[1, 1, 1]	['11', '11', '11']
52	[1, 1, 1]	['11', '11', '11']
53	[1, 1, 1]	['11', '11', '11']
54	[1, 1, 1]	['11', '11', '11']
55	[1, 1, 1]	['11', '11', '11']
56	[1, 1, 1]	['11', '11', '11']
57	[1, 1, 1]	['11', '11', '11']
58	[1, 1, 1]	['11', '11', '11']
59	[1, 1, 1]	['11', '11', '11']
60	[1, 1, 1]	['11', '11', '11']
61	[1, 1, 1]	['11', '11', '11']
62	[1, 1, 1]	['11', '11', '11']
63	[1, 1, 1]	['11', '11', '11']
64	[1, 1, 1]	['11', '11', '11']
65	[1, 1, 1]	['11', '11', '11']

Number	Transition Ages	Transition Type
66	[1, 1, 1]	['11', '11', '11']
67	[1, 1, 1]	['11', '11', '11']
68	[1, 1, 1]	['11', '11', '11']
69	[1, 1, 1]	['11', '11', '11']
70	[1, 1, 1]	['11', '11', '11']
71	[1, 1, 1]	['11', '11', '11']
72	[1, 1, 1]	['11', '11', '11']
73	[1, 1, 1]	['11', '11', '11']
74	[1, 1, 1]	['11', '11', '11']
75	[1, 1, 1]	['11', '11', '11']
76	[1, 1, 1]	['11', '11', '11']
77	[1, 1, 1]	['11', '11', '11']
78	[1, 1, 1]	['11', '11', '11']
79	[1, 1, 1]	['11', '11', '11']
80	[1, 1, 1]	['11', '11', '11']
81	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
82	[1, 15, 2, 15, 2]	['11', '11', '11', '11']
83	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
84	[1, 15, 2, 15, 2]	['11', '11', '11', '11']
85	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
86	[1, 15, 2, 15, 2]	['11', '11', '11', '11']
87	[15, 2, 15, 2]	['11', '11', '11', '11', '11', '11']
88	[1, 15, 2, 15, 2]	['11', '11', '11', '11']
89	[15, 2, 1, 1]	['11', '11', '11', '11']
90	[15, 2, 1, 1]	['11', '11', '11', '11']
91	[15, 2, 1, 1]	['11', '11', '11', '11']
92	[15, 2, 1, 1]	['11', '11', '11', '11']
93	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
94	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
95	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
96	[15, 2, 4, 11, 1, 4, 11, 1]	['11', '11', '01', '01', '11', '01', '01', '11']
97	[18, 1, 11, 1, 11, 1]	['02', '07', '05', '07', '05', '07']
98	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
99	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
100	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
101	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
102	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
103	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
104	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
105	[1, 9, 1, 9, 1]	['08', '06', '08', '06', '08']
106	[1, 9, 1, 9, 1, 9, 1, 9, 1]	['08', '06', '08', '06', '08', '06', '08']
107	[16, 1, 11, 1, 11, 1]	['03', '07', '05', '07', '05', '07']
108	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
109	[1, 11, 1, 11, 1]	['07', '05', '07', '05', '07']
110	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']

Number	Transition Ages	Transition Type
111	[14, 1, 9, 1, 9, 1]	['04', '08', '06', '08', '06', '08']
112	[1, 9, 1, 9, 1, 9, 1, 9, 1]	['08', '06', '08', '06', '08', '06', '08']
113	[11, 1, 11, 1, 11, 1]	['05', '07', '05', '07', '05', '07']
114	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
115	[11, 16, 1, 11, 1, 11, 1]	['05', '03', '07', '05', '07', '05', '07']
116	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
117	[9, 14, 1, 9, 1, 9, 1]	['06', '04', '08', '06', '08', '06', '08']
118	[1, 9, 1, 9, 1, 9, 1, 9, 1]	['08', '06', '08', '06', '08', '06', '08']
119	[9, 1, 9, 1, 9, 1]	['06', '08', '06', '08', '06', '08']
120	[1, 9, 1, 9, 1, 9, 1, 9, 1]	['08', '06', '08', '06', '08', '06', '08']
121	[1, 9, 1, 9, 1]	['08', '06', '08', '06', '08']
122	[1, 9, 1, 9, 1, 9, 1, 9, 1]	['08', '06', '08', '06', '08', '06', '08']
123	[13, 18, 1, 11, 1, 11, 1]	['01', '02', '07', '05', '07', '05', '07']
124	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
125	[13, 1, 11, 1, 11, 1]	['01', '07', '05', '07', '05', '07']
126	[1, 4, 11, 1, 4, 11, 1, 4]	['11', '07', '05', '11', '07', '05', '11', '07']
127	[18, 1, 11, 16, 1, 11, 16, 1]	['02', '07', '05', '03', '07', '05', '03', '07']
128	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
129	[1, 11, 16, 1, 11, 16, 1]	['07', '05', '03', '07', '05', '03', '07']
130	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
131	[1, 11, 16, 1, 11, 16, 1]	['07', '05', '03', '07', '05', '03', '07']
132	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
133	[1, 11, 16, 1, 11, 16, 1]	['07', '05', '03', '07', '05', '03', '07']
134	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
135	[1, 9, 14, 1, 9, 14, 1]	['08', '06', '04', '08', '06', '04', '08']
136	[1, 9, 14, 1, 9, 14, 1, 9, 14, 1, 9, 1]	['08', '06', '04', '08', '06', '04', '08', '06', '08']
137	[16, 1, 11, 16, 1, 11, 16, 1]	['03', '07', '05', '03', '07', '05', '03', '07']
138	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
139	[1, 11, 16, 1, 11, 16, 1]	['07', '05', '03', '07', '05', '03', '07']
140	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
141	[14, 1, 9, 14, 1, 9, 14, 1]	['04', '08', '06', '04', '08', '06', '04', '08']
142	[1, 9, 14, 1, 9, 14, 1, 9, 14, 1, 9, 1]	['08', '06', '04', '08', '06', '04', '08', '06', '08']
143	[11, 1, 11, 16, 1, 11, 16, 1]	['05', '07', '05', '03', '07', '05', '03', '07']
144	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
145	[11, 16, 1, 11, 16, 1, 11, 16, 1]	['05', '03', '07', '05', '03', '07', '05', '03', '07']
146	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
147	[9, 14, 1, 9, 14, 1, 9, 14, 1]	['06', '04', '08', '06', '04', '08', '06', '04', '08']
148	[1, 9, 14, 1, 9, 14, 1, 9, 14, 1, 9, 1]	['08', '06', '04', '08', '06', '04', '08', '06', '08']
149	[9, 1, 9, 14, 1, 9, 14, 1]	['06', '08', '06', '04', '08', '06', '04', '08']
150	[1, 9, 14, 1, 9, 14, 1, 9, 14, 1, 9, 1]	['08', '06', '04', '08', '06', '04', '08', '06', '08']
151	[1, 9, 14, 1, 9, 14, 1]	['08', '06', '04', '08', '06', '04', '08']
152	[1, 9, 14, 1, 9, 14, 1, 9, 14, 1, 9, 1]	['08', '06', '04', '08', '06', '04', '08', '06', '08']
153	[13, 18, 1, 11, 16, 1, 11, 16, 1]	['01', '02', '07', '05', '03', '07', '05', '03', '07']
154	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']
155	[13, 1, 11, 16, 1, 11, 16, 1]	['01', '07', '05', '03', '07', '05', '03', '07']
156	[1, 4, 11, 16, 1, 4, 11, 16, 1]	['11', '07', '05', '03', '11', '07', '05', '03', '11', '07']

Number	Transition Ages	Transition Type
157	[18, 1, 1, 1]	['02', '11', '11', '11']
158	[16, 1, 1, 1]	['03', '11', '11', '11']
159	[14, 1, 1, 1]	['04', '08', '08', '08']
160	[11, 1, 1, 1]	['05', '11', '11', '11']
161	[11, 16, 1, 1, 1]	['05', '03', '11', '11', '11']
162	[9, 1, 1, 1]	['06', '08', '08', '08']
163	[9, 14, 1, 1, 1]	['06', '04', '08', '08', '08']
164	[13, 18, 1, 1, 1]	['01', '02', '11', '11', '11']
165	[13, 1, 1, 1]	['01', '11', '11', '11']
166	[18, 1, 1, 1]	['02', '11', '11', '11']
167	[16, 1, 1, 1]	['03', '11', '11', '11']
168	[14, 1, 1, 1]	['04', '08', '08', '08']
169	[11, 1, 1, 1]	['05', '11', '11', '11']
170	[11, 16, 1, 1, 1]	['05', '03', '11', '11', '11']
171	[9, 1, 1, 1]	['06', '08', '08', '08']
172	[9, 14, 1, 1, 1]	['06', '04', '08', '08', '08']
173	[13, 18, 1, 1, 1]	['01', '02', '11', '11', '11']
174	[13, 1, 1, 1]	['01', '11', '11', '11']
175	[18, 1, 1, 1]	['02', '11', '11', '11']
176	[16, 1, 1, 1]	['03', '11', '11', '11']
177	[14, 1, 1, 1]	['04', '08', '08', '08']
178	[11, 1, 1, 1]	['05', '11', '11', '11']
179	[11, 16, 1, 1, 1]	['05', '03', '11', '11', '11']
180	[9, 1, 1, 1]	['06', '08', '08', '08']
181	[9, 14, 1, 1, 1]	['06', '04', '08', '08', '08']
182	[13, 18, 1, 1, 1]	['01', '02', '11', '11', '11']
183	[13, 1, 1, 1]	['01', '11', '11', '11']
184	[18, 1, 1, 1]	['02', '11', '11', '11']
185	[16, 1, 1, 1]	['03', '11', '11', '11']
186	[14, 1, 1, 1]	['04', '08', '08', '08']
187	[11, 1, 1, 1]	['05', '11', '11', '11']
188	[11, 16, 1, 1, 1]	['05', '03', '11', '11', '11']
189	[9, 1, 1, 1]	['06', '08', '08', '08']
190	[9, 14, 1, 1, 1]	['06', '04', '08', '08', '08']
191	[13, 18, 1, 1, 1]	['01', '02', '11', '11', '11']
192	[13, 1, 1, 1]	['01', '11', '11', '11']

Number	Transition Ages	Transition Type
193	[13, 13, 13, 13, 13, 13, 13, 13]	['13', '13', '13', '13', '13', '13', '13', '13']
194	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
195	[13, 13, 13, 13, 13, 13, 13, 13]	['14', '14', '14', '14', '14', '14', '14', '14']
196	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
197	[13, 13, 13, 13, 13, 13, 13, 13]	['12', '12', '12', '12', '12', '12', '12', '12']
198	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
199	[13, 13, 13, 13, 13, 13, 13, 13]	['12', '12', '12', '12', '12', '12', '12', '12']
200	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
201	[13, 13, 13, 13, 13, 13, 13, 13]	['13', '13', '13', '13', '13', '13', '13', '13']
202	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
203	[13, 13, 13, 13, 13, 13, 13, 13]	['14', '14', '14', '14', '14', '14', '14', '14']
204	[1, 4, 13, 13, 13, 13, 13, 13, 13, 13, 13]	['11', '07', '13', '13', '13', '13', '13']
205	[13, 1, 13, 1]	['11', '11', '11', '11']

A.2.6 Prescription Generator Python Code

main_pres.py

```
import csv
import pres_builder as pb


global model_type
model_type = 'A'
def main(model_type=model_type, directory=None):
    """
    The central routine from which stand_types,
    stand_types_eco, harv_types, and stand_age_period
    files are generated.

    First generate a list of prescription objects
    through calling the pres_builder.py module.
    Then read in a stand table line by line and
    simulate each stand following each prescription.
    Store the results of these simulations in lists.
    Once all stands have been simulated, write the
    contents of the lists to files that will
    supply parameters to the LP.

    There is a global parameter and
    function argument 'model_type'. If set
```

```
to 'A' this means, build the full prescription
set, as used in the phase 2 and industry models
of Andrew B. Martin's thesis. If set to 'B', a
reduced prescription set is built that is the same
as the phase 1 prescription set from the same thesis.

kwargs directory states the directory where
the stand table can be found, and where
output files should be written .
"""

# Generate list of prescriptions
prescriptions = pres_generator()



# Do nothing will alwyas be the first prescription
nothing = prescriptions[0]
buffers = [p for p in prescriptions if p.name == 'buffer']

# Any prescription with number >= 200 is a buffer prescription.
# Only buffer stands are elligible for these, and nothing else
# so separate buffer prescriptions from prescriptions for
# non-buffer stands
prescriptions = [p for p in prescriptions if p.number < 200]

# Print to screen a list of prescriptions
# see function description below
list_pres(prescriptions)
```

```python
# current working directory
# change as appropriate
if directory:
    cwd = directory
else:
    cwd = 'C:\\Users\\Andrew_Martin\\Desktop\\29_July\\'

# FBF1 file in form: stand, age, buffer, excl, type; for each stand
fbf1 = csv.reader(open(cwd+'FBF1.tab'), delimiter='\t')

# These are the master lists that are appended after a stand runs through
# all the prescriptions
stand_types = []
stand_types_eco = []

# Age and Harv_type tables are indexed by initial stand age,
# so use sets to ensure duplicates are not written
stand_age = set()
harv_types = set()

# list of all master lists
all_list = [stand_types, stand_types_eco, stand_age, harv_types]

# file names that will be written from each master list - in order
files = ['stand_types', 'stand_types_eco', 'stand_age_period',
         'harv_types']
```

```
# headers to be written for each master list file
headers = [['Stand','Pres','Period','Stand_Type'],
                              ['Stand','Pres','Period','Stand_Type'],
                              ['Initial_Age','Pres','Period','Age'],
                              ['Initial_Age','Pres','Period','Cut_Type']]

# For each stand in fbf1, run through all the prescriptions
for stand in fbf1:

    # sets that keep lines to written for this stand
    # their contents will be appended to the master lists
    # after the current stand has run through all
    # prescriptions
    temp_lists = [ set(), set(), set(), set() ]

    # store entries to master list for this
    # prescription. The entries in this list
    # will populate the sets of temp_lists
    # above, after each prescription is
    # run through
    temp_temp_list = []

    # Read info from fbf1 row into variables
    number, age, buff, excl, stype = stand_unpack(stand)

    # Give some idea of progress
    if int(number)%1000 == 0:
        print 'Feeding Stand {0} through prescriptions'.format(number)
```

```python
# Stand site-class, used for shelterwood and CC prescriptions.
# This is to determine age of Max MAI for intervention scheduling
sc = stype[-3]

# Forest state of the stand, used for eligibility tests
state = stype[len(stype)-5:len(stype)-3]

# Excl stands are only eligible for the do nothing prescription
# Buffer stands are only eligible for do nothing and buffer harvest
if not buff and not excl:
    for pres in prescriptions:
        # Eligibility is dependent on state and/or site-class
        if eligible(sc, state, age, pres):
            # Stores file entries from assigning current pres
            # to current stand
            temp_temp_list = pres.assign_stand(stand)

        # append current pres lists to current stand lists
        set_unpack(temp_temp_list, temp_lists)
elif buff and not excl:
    for pres in buffers:
        if eligible(sc, state, age, pres):
            temp_temp_list = pres.assign_stand(stand)

    set_unpack(temp_temp_list, temp_lists)
else:
    temp_temp_list = nothing.assign_stand(stand)
```

```
            # Don't need to do this, but it keeps the three
            # streams consistent
            set_unpack(temp_temp_list, temp_lists)

        # Append stand list entries to master lists after running
        # through all prescriptions
        list_unpack(temp_lists, all_list)

    #write each list to files
    for a, f in enumerate(files):
        file_writer = csv.writer(open(f+'.tab', 'w'), delimiter='\t')
        # universal header for ampl
        # 3 indicates 3 index columns, 1 indicates 1 data column
        file_writer.writerow(['ampl.tab 3 1'])
        file_writer.writerow(headers[a])
        for row in all_list[a]:
            file_writer.writerow(row)

def list_unpack(lists_1, lists_2):
    """
    Copy the contents from each list/set in lists_1
    into lists_2.
    Stand Type, Stand Type Eco will always be lists
    and will always occupy the first two indices of
    both lists.
    Stand Age and Harv Type are sets will always be
    indices 2,3 respectively in the both lists.
    """
```

```
    for a,i in enumerate(lists_1):
        # stand Type and stand type eco are lists
        if a < 2:
            for j in i:
                lists_2[a].append(j)
        # stand age and harv type are sets
        else:
            for j in i:
                lists_2[a].add(j)

def set_unpack(sets_1, sets_2):
    """
    Like list unpack, but for two sets
    """
    for a,i in enumerate(sets_1):
        sets_2[a].update(i)



def eligible(sc, state, age, pres):
    """
    Test a stand's site-class and forest state
    against the restriction set of a prescription.
    Return true if the stand is eligible for the prescription;
    false otherwise.
    """
    # pres.restrictions is a dictionary
    # if it has a key called 'sc' then the
    # value corresponding to that key, in this
```

```
# case a list of ineligible sit-class numbers,
# will be returned; if it does not contain
# this key, then a list containing 0 will
# be returned.
if int(sc) not in pres.restrictions.get('sc', [0]):
    if state not in pres.restrictions.get('state', [0]):
        try:
            # If the prescription has
            # an alternate track, then
            # every stand is eligible for
            # it regardless of age.
            if pres.alt_i_sched:
                return True
        # This exception will be raised if
        # a prescription does not have an
        # alternate track. In which case
        # if the stand is older than the
        # initial age of the prescription,
        # it is ineligible.
        except AttributeError:
            if age > pres.in_age:
                return False
            else:
                return True

    return False
return False
```

```python
def stand_unpack(stand):
    """
    Break up a row from the FBF1 table
    into stand number, age, exclusion status,
    buffer status, and stand type.
    Return a list of these values.
    """
    number = stand[0]
    age = int(stand[1])
    stype = stand[4]
    excl = int(stand[3])
    buff = int(stand[2])
    return [number, age, buff, excl, stype]

def pres_generator():
    """
    Generate and return a list of prescription
    objects.

    This is called from main() if no UI
    is attached; otherwise it is not called.
    """

    # Track prescription number
    count = 0
    # Store all generated prescriptions here
    pres_list = []
```

```
# Generate do nothing prescription
# Set initial intervention age at 50
# this ensures no stands will ever
# reach it, since stands die at 40
nothing = [50]

# g function calls are used throughout
# they pass action to functions defined
# below that feed parameters into the
# module that performs all the heavy lifting
temp_list = g_buffer(nothing, count)
for t in temp_list:
    pres_list.append(t)
count+=1

# Generate clearcut prescriptions
# One list for each site-class

cc_tuples = [0]*4
# Initial CC intervention then
# no more interventions for the
# rest of the planning horizon
cc_nothing = [0]*4

# Initial CC intervention in
# +\- 2 periods from Max MAI
# based on site-class, then have
# a second entry based on
```

```
# earliest intervention age
cc_reg = [0]*4

# Generate a 5 period window for each site-class
# 18-14 for sc = 1 to 10-14 sc = 4, etc
for a,i in enumerate(cc_tuples):

        # range gives the initial intervention ages
        # [a+1] signifies the site-class
        cc_reg[a] = zip(range(14-a, 19-a), [a+1]*5)

        cc_nothing[a] = zip(range(14-a, 19-a) ,[a+1]*5)



all_cc = cc_tuples + cc_nothing

# Cut then no more interventions
for cc in cc_nothing:
    temp_list = g_clearcuts(cc, count, nothing=1)
    #add new prescriptions to prescription list
    for t in temp_list:
        pres_list.append(t)

    count+=len(temp_list)
# Regular two entry Clearcut
for cc in cc_reg:
    temp_list = g_clearcuts(cc, count)
    for t in temp_list:
```

```
        pres_list.append(t)
    count+=len(temp_list)


# Note that more clearcut prescriptions can be
# added by following the above two examples




# Generate shelterwood prescriptions
# For now, this is also done by site-class
# First tuple index gives age of first entry
# the second gives site class.
# For now, all initial entries are followed
# by a final felling 2 periods later
shl_tuples = [(14,1), (14,2), (14,3),(14,4)]
shl_tuples_2 = [(15,1), (15,2), (15,3),(15,4)]
shl_tuples_3 = [(16,1), (16,2), (16, 3), (16,4)]

# all tuples in one list, so generate all
# shelterwood pres in one fell swoop
temp_list = g_shelterwood(shl_tuples, count)
for t in temp_list:
    pres_list.append(t)
count+=len(temp_list)

# SHL with second entry CC
# Note that these prescriptions,
```

```
# i.e. ones with non-clearcut entries
# as the initial entry have no need for
# an alternate track, since it would be
# equivalent to a stand following the
# alternate track on a clearcut prescription
temp_list = g_shelterwood(shl_tuples_2, count)
for t in temp_list:
    pres_list.append(t)
count+=len(temp_list)

# SHL with second entry CT
temp_list = g_shelterwood(shl_tuples_3, count)
for t in temp_list:
    pres_list.append(t)
count+=len(temp_list)



# Generate commercial thin prescriptions

# Dictionary mapping (state, initial intervention age)
# to number of interventions in the prescription.
# ('11', 12, 3) :3 means stands with state 11,
# natural regenration state, can receive a three entry
# clearcut, starting at age 12 periods
ct_tuples = {     ('11', 12,3) : 3, ('11', 12, 2):2,
                ('07', 10,3) : 3, ('07', 10,2) : 2, ('05', 15,2) : 2,
                ('05', 15,1) : 1, ('03', 19,1) : 1, ('01', 17,2) : 2,
```

```
                ('01', 17,1) : 1, ('02', 21,1) : 1, ('08', 8,3) : 3,
                ('08', 8,2) : 2, ('08', 8, 1) : 1, ('04', 17, 1) : 1,
                ('06', 13,2) : 2, }

ct_tuples_2 = {  ('11', 12,3) : 3, ('11', 12, 2):2,
                ('07', 10,3) : 3, ('07', 10,2) : 2, ('05', 15,2) : 2,
                 ('01', 17,2) : 2,
                 ('08', 8,3) : 3,
                ('08', 8,2) : 2,
                ('06', 13,2) : 2,  }

if model_type == 'A':
    ct_tuples_2 = {
                 ('08', 8,3) : 3,
                ('08', 8,2) : 2,
                ('06', 13,2) : 2,  }

# Dicts are not ordered, so have a list of the tuple keys
# given by state and intervention age
in_age_state = sorted(ct_tuples.keys(),
                      key = lambda x: (x[0],x[1]))

# Again, generate all commercial thin prescriptions
# in one fell swoop.
# Two entry regeneration harvests here
temp_list = g_commercialthin(in_age_state, count, ct_tuples)
for t in temp_list:
    pres_list.append(t)
```

```
count+=len(temp_list)

# Three entry regeration harvests here
temp_list = g_commercialthin(in_age_state,count,
                             ct_tuples, regen=3)
for t in temp_list:
    pres_list.append(t)
count+=len(temp_list)

in_age_state = sorted(ct_tuples_2.keys(),
                      key = lambda x: (x[0],x[1]))


for a in range(4):
    # clearcut regeneration harvests
    temp_list = g_commercialthin(in_age_state,count,
                ct_tuples_2, sc=a+1, cc=True, regen=4)
    for t in temp_list:
        pres_list.append(t)
    count+=len(temp_list)


# Generate selection prescriptions

# Selection harvest prescriptions all
# occur at 16 periods of age. Make a
# list of all eligible forest states tupled
# with '16', the age of initial intervention
```

```
sel_tuples = zip(['07', '10', '11', '12', '13', '14'],
                 [16]*6)

# Selection harvests are modelled the
# same way as thinnings except the age
# is reset each entry. Use the commercial thin
# generator to generate selection prescriptions
temp_list = g_commercialthin(sel_tuples,
                             count, sel=True)
for t in temp_list:
    pres_list.append(t)
count+=len(temp_list)

# Generate buffer prescriptions
# one option: go in at 12 periods of age
in_age = [12]

# Set count to 200 to indicate that
# these are buffers.
# Note that if more prescriptions are added, then it
# may be necessary to increase this to 300 or 500
count = 200

temp_list=g_buffer(in_age, count)

for t in temp_list:
    pres_list.append(t)
```

```
    # Return a list containing all
    # the prescription objects
    # generated above
    return pres_list

# In the following functions g stands for "GENERATE"
def g_clearcuts(in_age_sc, start_number, nothing=""):
    """
    Take a list of tuples mapping
    initial_age of prescriptions to corresponding
    site_classes. Start number lets us know what
    prescription this is.

    Return a list of Pres objects for clearcuts.
    Each clearcut has an alt_age option that is
    0-5 periods after modelling begins. Could be
    interesting to have it as 0-10 and randomly selected.
    Would need to pass a set or something to make sure
    duplicates were not recorded

    key word argument 'nothing' if true means
    that after the first intervention, nothing is
    done to the stand; otherwise, the stand follows
    a normal prescription
    """

    # Store pres objects here
    pres_list = []
```

```
    # in_age_0 specifies the minimum age
    # for this set of prescriptions. If
    # earliest prescription first entry occurs
    # at 12 periods of age, then no stand older
    # than 12 periods is elligible for the main track
    # of these prescriptions. They fall on the alternate
    # track. This prevents duplicate prescriptions
    # from being written.
    in_age_0 = in_age_sc[0][0]
    for a,in_age in enumerate(in_age_sc):

        # each site-class has a 5 period window
        # in which entries can take place
        # this means a%5 will give 0-4 period
        # leave times as the alt entry age
        alt_age = a%5
        pres = pb.clearcut(in_age[0], alt_age, in_age[1],
                    start_number+a, in_age_0, nothing=nothing)

        pres_list.append(pres)

    return pres_list

def g_shelterwood(in_age_sc, start_number):
    """
    Take a list of tuples mapping
    initial_age of prescriptions to corresponding
```

```
    site_classes. Start number lets us know what
    prescription this is.

    Return a list of Pres objects for shelterwoods.
    14 period shelterwoods have an alt_age option that is
    0-5 periods after modelling begins. This is chosen
    as a random number for each stand.
    """
    pres_list = []

    # Only want alt_age prescriptions for 14 period
    # Shelterwoods.
    # If model_type is 'B' then an alternate
    # track prescription is written using
    # site-class to set the initial entry age;
    # otherwise, no alternate track is defined.
    for a, in_age in enumerate(in_age_sc):
        if model_type == 'B':
            alt_age = a
        else:
            alt_age = ""
        pres = pb.shelterwood(in_age[0], in_age[1],
                    start_number + a, alt_age=alt_age)
        pres_list.append(pres)

    return pres_list

def g_commercialthin(in_age_state, start_number, entries="",
```

```
                    regen=2, sel="", cc="", sc=""):
"""
Take a list of tuples mapping state to initial intervention age.
start_number supplies the prescription numbers, and entries
indicates how many entries in the prescription. With a slight mod
this generation routine can apply to selection harvests, and the sel
named arg tells us whether to go that route.

key word args:
entries: for commercial thinning prescriptions indicates
    the number of entries in the initial prescription.
regen: indicates the number of entries after regeneration
sel: whether this is a selection harvest prescription
cc: whether this is a clearcut harvest prescription
sc: for clearcut prescriptions it is necessary to know
    the site-class
"""
pres_list = []

for a, in_age in enumerate(in_age_state):

    # 'x' indicates that alt_age
    # will be figure out in the pb
    # module
    if model_type == 'B':
        alt_age = 'x'
    else:
        alt_age = ""
```

```
        # Xommercial thin track
        if not sel and not cc:
            entry = entries[in_age]
            pres = pb.commercialthin(in_age[1], start_number + a,
                            in_age[0], entry, alt_age, regen=regen)
        #selection track
        elif not cc:
            pres = pb.selection(in_age[1], in_age[0],
                                start_number+a, alt_age)
        else:
            entry = entries[in_age]
            pres = pb.commercialthin(in_age[1], start_number+a,
                        in_age[0], entries=entry, alt_age=alt_age,
                        cc=True, sc=sc, regen=regen)

        pres_list.append(pres)

    return pres_list

def g_buffer(in_age, start_number):
    """
    Take a list of initial ages, and a number
    indicating what to count up from for prescription
    numbers. Like the other partial harvest prescriptions,
    generate an alt_age pres based on a random number between
    0 and 5 only for the earliest entry prescription.
    """
```

```
    #If you have any questions read any of the
    # 'g_' functions above.
    pres_list = []

    for a,i in enumerate(in_age):
        if 0 == a and start_number >= 100:
            alt_age = 'x'
        else:
            alt_age = ""
        pres = pb.buffer_harv(i, start_number+a, alt_age)
        pres_list.append(pres)

    return pres_list

def list_pres(pres_list):
    """
    List the name and number
    and scheds of each prescription
    """

    for p in pres_list:
        try:
            print p.number, p.name, p.i_sched, p.alt_i_sched
        except AttributeError:
            print p.number, p.name, p.i_sched
```

pres_builder.py

```
class Pres(object):
    """
    Pres objects have methods that when supplied a
    stand, generate the data to write stand_types,
    stand_type_eco, stand_age_period, and
    harv_type files.

    A stand is eligible for a prescription if it
    does not match any of the restrictions on
    site-class and state, and if it is younger
    than the initial entry age. Some prescriptions
    will have alt_age entries that are offset 0-5
    periods from beginning of the planning horizon.

    A Pres has three master lists i_sched, t_sched,
    and r_sched that detail when interventions happen,
    how a stand transitions as a result of the intervention,
    and the age the stand assumes after the intervention,
    respectively. For stands that have alt_age alt_i_sched,
    alt_t_sched, and al_r_sched supply the same information
    for a management track that targets stands that fall
    outside of the intitial intervention age.

    Each entry in a Pres also has a harv_type
    {01, 11, 12, 15, 16} indicating whether
```

```
    it is a thin, clearcut, selection, shelterwood,
    or buffer, respectively.

    Pres objects also have a name,
    and a number.

    The primary method of a Pres object is assign_stand,
    which simulates a stand following the prescription
    over the planning horizon. Lists are compiled during
    the simulation that track the stand's stand_type,
    and age, while also recording harvest methods, along
    the prescription.
    """

    def __init__(self, restrictions, name, number, sched,
                 in_age, harv_type):
        self.name = name
        self.number = number
        self.i_sched = sched[0]
        self.t_sched = sched[1]
        self.r_sched = sched[2]
        self.in_age = in_age
        self.harv_type = harv_type
        self.restrictions = restrictions

    # Not all prescriptions have alternate age tracks;
    # they have to be added after the object is instantiated
    def add_alt(self, sched, harv_type):
```

```
        self.alt_i_sched = sched[0]
        self.alt_t_sched = sched[1]
        self.alt_r_sched = sched[2]
        self.alt_harv_type = harv_type


    # Filter out exclusion and buffer beforehand
    # i.e. assign_stand performs no eligibility
    # check, except for age checking
    def assign_stand(self, stand):
        """
        Take in a stand.
        Depending on age of the stand either write
        primary or alternative prescription.

        Return a list of lists.

        Stand_Type
        Stand_Type_Eco
        Stand_Age - actually a set
        Harv_Type - actually a set too

        The main() method from main_pres (mp)
        will take these lists and write them to
        their respective files
        """
        # sa - stand_age, st - stand_type, se - stand_type_eco
        # ht - harv_type
```

```
sa_list = set() # misnomer, actually a set
st_list = []
se_list = []
ht_list = set() # misnomer, actually a set

i_sched, t_sched, r_sched = [], [], []


# Get essential information about the stand
# being assigned the prescription
number,age,stype = stand[0],int(stand[1]),stand[4]

name = self.name

# Keep all lists together for easy passing
list_of_lists = [st_list, se_list, sa_list, ht_list]

# Local versions of the master versions of the lists
# held by the prescription object
sched_list = [i_sched, r_sched, t_sched]

# This is what happens in the following section:
# the Pres object's list are pasted into local
# copies of the lists which are then
# passed to the assignment function, which
# runs the current stand through the prescription

# Can only take alt_age track if the prescription
```

```
# has such a thing
if age > self.in_age and hasattr(self, 'alt_i_sched'):
    # Set all the schedules to be alternate
    # Populate them from the master copies
    sched_list[0] = [i for i in self.alt_i_sched]
    sched_list[1] = [t for t in self.alt_t_sched]
    sched_list[2] = [r for r in self.alt_r_sched]


    # For entries that occur at unknown age, they will be
    # entered in the lists as strings
    # when going through the lists, if an entry
    # is a string that means add the age of the stand
    # to the integer value of the string to get the
    # age the entry occurs at.
    # same practice will apply to the regeneration list.
    resolve_alt(sched_list, age, name)
    # don't have to return anything cause the list
    # is being passed and changes will be made to it

    alt_harv = [h for h in self.alt_harv_type]
    assignment(sched_list, list_of_lists, stand,
               self.number, alt_harv)

else:

    # Populate lists from master copies
    sched_list[0] = [i for i in self.i_sched]
```

```
        sched_list[1] = [t for t in self.t_sched]
        sched_list[2] = [r for r in self.r_sched]
        harv = [h for h in self.harv_type]
        # Don't return anything, list
        # will be updated in function
        assignment(sched_list, list_of_lists, stand,
                   self.number, harv)

    return list_of_lists


def resolve_alt(sched, age, name):
    """
    Every case except shelterwood deals with
    alt prescriptions in a straight forward way:
    knock the stand down in the next 5 periods and
    start managing it.
    Shelterwoods are a little different, cause
    we want to finish them off shelterwoods the first
    time round. This function adjusts the regen list
    and intervention list to suit all intervention methods
    """
    # Add stand age to the period offset for the prescription
    # for everything except shelterwoods, only the first
    # entry in alt_i_sched will need to be updated

    # multipliers apply at 16 under shelterwood
    # prescriptions therefore no clearcuts can occur
```

```
    # at 16 under shelterwood pres. This works out
    # without a problem, luckily.
    if name == 'shelterwood':
        if int(sched[0][0]) + age > 16:
            sched[0][0] = int(sched[0][0]) + age
        else:
            sched[0][0] = 17

        # for shelterwoods, the second entry in alt_i_sched
        # and the first entry in alt_r_sched need to be updated
        sched[0][1] = sched[0][0] + 2
        sched[2][0] = sched[0][0] + 1

    else:
        sched[0][0] = int(sched[0][0]) + age

    # This function modifies a list
    # so no need to return anything

def intervention(stand_type, trans='11', stock = '3'):
    """
    change the type of a stand as the result of an
    intervention taking place.

    named args:
    regen - defaults to '11' (NRG) the state the stand
        takes as a result of the intervention
    stock - defaults to '3'. The stocking of the
```

```
        stand as a result of the intervention.
    """
    new_type = (stand_type[:-5] + trans + stand_type[-3]
                + stock + stand_type[-1])
    return new_type

def assignment(sched, list_of_lists, stand,
               pres_number, harv_type):
    """
    Update a list of lists with the results of
    applying a prescription to the provided stand.

    Age
    Stand Type
    Stand Type Eco
    Harv Type

    Are updated as the stand ages
    and passes through transitions
    """

    # Unpack info from stand list
    number,age,stype = stand[0],int(stand[1]),stand[4]

    # House period by period entries for the age, stand type,
    # stand_type eco and harv type lists
    age_tuple, st_tuple, se_tuple, ht_tuple = (0,),(0,),(0,),(0,)
    list_of_tuples = [st_tuple, se_tuple, age_tuple, ht_tuple]
```

```
# Initialize period, current age, stocking
period = 0
cur_age = age
stock = stype[-2:-1]

# PLT stand_types that cannot be naturally regenerated
# without changing the species
dead_plt = ['1111335','1111435','1311435','1311235']

# intervention, transition and regeneration schedules,
# respectively
i_sched, t_sched, r_sched = sched[0], sched[1], sched[2]

#31 is the number of periods.
# This could change were more or less modelled.
for i in range(31):
    for a,tup in enumerate(list_of_tuples):
        list_of_tuples[a] = (0,)

    list_of_tuples[2] = (age, pres_number, period, cur_age)

    # If sched has entries left, and the current age of
    # the stand corresponds to an entry age, and the period
    # is less than 30, then simulate the intervention.
    if sched[0] and cur_age == sched[0][0] and period < 30:

        # After applying an intervention, remove
```

```
# it from the list
pop_hold = sched[0].pop(0)
# Same thing with harv_types
# there was a problem with
# the harv_type list getting
# exhausted prematurely, hence the
# try-except clause.
try:
    harv = harv_type.pop(0)
except IndexError:
    print pres_number, period, age
# Get the type resulting from the intervention
try:
    # check if regeneration comes
    # as a result of a final felling. In this
    # case sched[2][0] will equal 1 or 2.
    # These are the regeneration ages.
    if sched[2][0] in (1, 2):
        stock = '3'

    new_type = intervention(stype, trans=sched[1].pop(0),
                            stock=stock)

except IndexError:
    print sched, pres_number

# Record the intervention cutting the pre-intervention type
list_of_tuples[0] = (number, pres_number, period, stype)
```

```
        # If type changes due to intervention write a line in eco
        if stype != new_type:

            # This type comes up once and there is no yield for it
            # brush it under the rug
            if new_type in dead_plt:
                new_type = intervention(new_type, trans='08')
            list_of_tuples[1] = (number, pres_number,
                                 period + 1, new_type)
        #record the intervention type

        list_of_tuples[3]= (age, pres_number,
                            period, harv)
        # how the stand's age changes as a
        # result of the prescription
        cur_age = sched[2].pop(0)
        # update the stand's type
        stype = new_type


    # no intervention takes place, stand ages as usual
    elif cur_age < 40 and period < 30:
        cur_age += 1
    # Stand is 40 and it is time to die and regenerate
    elif cur_age == 40 and period < 30:
        cur_age = 0
        # Default death intervention with regen = '11'
        new_type = intervention(stype)
```

```
        if new_type in dead_plt:
            new_type = intervention(new_type, trans='08')
        # If type changes as a result of regenerating than
        # write a line a in the stand eco file
        if stype != new_type:
            list_of_tuples[1]=(number, pres_number,
                                    period+1, new_type)
            stype = new_type



    # Each time we go through this cycle record any
    # harvests, or aging in their respective lists
    for a,tup in enumerate(list_of_tuples):
        if tup != (0,) and tup not in list_of_lists[a]:
            # stand_types and stand_types_eco will occupy
            # the first two indices and they are lists
            if a < 2:
                list_of_lists[a].append(tup)
            # stand_age and harv_types are will occupy the
            # third and fourth indices and they are sets
            else:
                list_of_lists[a].add(tup)


    period += 1

def clearcut(in_age, alt_age, sc, code, in_age_0, nothing=""):
    """
    Given an initial age and a site class
```

```
and an offset from initial period for stands older
than the initial age, and the number of the prescription,
generate and return a prescription
object for a clearcut.

Alt_age tracks are always generated.

Key word args
nothing: indicates that after the first intervention
    nothing is done to the stand
"""

# Only applicable to stands of a certain site-class
site_classes = [1,2,3,4]

# Check restrictions by having a dictionary that
# you check each stands attributes with using .get(key, 0)
restrictions = { 'sc' : [s for s in site_classes if s!=sc],
                        'state' : ['01', '02', '03', '04',
                                   '05', '06', '07', '08',
                                   '12','13','14']  }

harv_type = '11'
name = 'clearcut'
number = code

# primary and alternate tracks respectively
sched = [0]*3
```

```
alt_sched = [0]*3


# Regen harvests at earliest eligible age
if 2 == nothing:
    regen = { 1 : 16, 2 : 15, 3 : 14, 4 : 13 }
elif 1 == nothing:
    # planning horizon is 30 periods, so
    # second entry at 31 periods ensures
    # not second entry will take place
    regen = { 1 : 31, 2 : 31, 3 : 31, 4 : 31 }
else:
    #Regens are based on Age for max Mai on give site
    regen = { 1 : 18, 2 : 17, 3 : 16, 4 : 15 }

# Simple prescription: Cut, cut, cut
# intervention ages
sched[0] = [in_age, regen[sc], regen[sc]]
# state changes as a result of intervention
sched[1] = ['11', '11', '11']
# age as a result of intervention
sched[2] = [1, 1, 1]
# harvest type of each intervention
harv_type = ['11', '11', '11']

# instantiate new prescription object
new_pres = Pres(restrictions, name, number, sched,
                in_age_0, harv_type)
```

```
    # Clearcuts always have an alternate track
    alt_sched[0] = [str(alt_age),regen[sc], regen[sc]]
    alt_sched[1] = ['11', '11', '11']
    alt_sched[2] = [1,1,1]
    harv_type = ['11', '11', '11']

    # Add alternate prescription for old stands to object
    new_pres.add_alt(alt_sched, harv_type)

    return new_pres

def shelterwood(in_age, sc, code, alt_age=""):
    """
    Given an initial age, and a prescription number
    generate a shelterwood prescription object.

    Named args:
    alt_age - true if the prescription has an
        alternate track
    """
    site_classes = [1,2,3,4]
    restrictions = { 'sc' : [s for s in site_classes if s!=sc],
                                    'state' : ['08', '06', '04', '01',
                                               '02', '05', '07','12',
                                               '13', '03', '14']  }
    harv_type = '15'
```

```
name = 'shelterwood'
number = code

# Primary and alternate tracks, respectively
sched = [0]*3
alt_sched = [0]*3

# Regeneration ages if shelterwood is regeneration
# harves, or if clearcuts are regeneration harvests.
regen = { 1 : 14, 2 : 14, 3 : 14, 4 : 14 }
regen_cc = { 1 : 18, 2 : 17, 3 : 16, 4 : 15 }
if in_age == 14:

    sched[0] = [in_age, in_age+2, regen[sc], regen[sc] + 2,
                regen[sc], regen[sc] + 2]
    sched[1] = ['11', '11', '11', '11', '11', '11']
    sched[2] = [in_age+1, 2, regen[sc]+1, 2]
    harv_type = ['15', '15', '15', '15', '15', '15']

# indicates after a shelterwood system the stand
# moves onto a clearcut track
elif in_age == 15:
    alt_age = ""
    in_age = 14
    #simple too cut, grow two years, fell
    sched[0] = [in_age, in_age+2, regen_cc[sc], regen_cc[sc]]
    sched[1] = ['11', '11', '11', '11']
    sched[2] = [in_age+1, 2, 1, 1]
```

```
        harv_type = ['15', '15', '11', '11', '11']

    # indicates after a shleterwood system the stand
    # moves onto a commercial thinning track
    elif in_age == 16:
        alt_age = ""
        in_age = 14
        sched[0] = [in_age, in_age+2, 3, 10, 15, 3, 10, 15]
        sched[1] = ['11', '11', '01', '01', '11', '01',
                    '01', '11']
        sched[2] = [in_age+1, 2, 4, 11, 1, 4, 11, 1]
        harv_type = ['15', '15', '01', '01', '11', '01',
                     '01', '11']

    new_pres = Pres(restrictions, name, number, sched,
                    in_age, harv_type)

    # If an alternate track is to be defined
    if alt_age != "":
        in_age = alt_age
        # First entry is a clearcut, then move onto shelterwood
        # management track.
        alt_sched[0] = [str(in_age), regen[sc], regen[sc]+2,
                        regen[sc]]
        alt_sched[1] = [ '11', '11', '11', '11']
        alt_sched[2] = [ 1, regen[sc]+1, 2, regen[sc]+1, 2]
        harv_type = ['11', '15', '15', '15', '15']
```

```
        new_pres.add_alt(alt_sched, harv_type)

    return new_pres

def commercialthin(in_age, code, state, entries,
                   alt_age="", cc="", sc="", regen=2):
    """
    Given initial age, a prescription number
    and optional pct and alt_age arguments.
    Generate a precommercial thin prescription.

    Unlike others, this will have a state it is aimed at.

    Key word args
    alt_age: true if this prescription has an alternate track
    cc: true if first set of entries are clearcuts before
        stand moves onto a commerical thinning track.
    sc: indicates the site-class for use if the first entry
        is a clearcut
    regen: indicates if after first regeneration the stand
        receives a two or three entry commercial thinning system
    """
    # Primary and alternate tracks, respectively
    sched = [0]*3
    alt_sched = [0]*3

    name = 'commercial_thin'
    number = code
```

```
harv_type = '01'
site_classes = [1,2,3,4]

# Mapping of initial states to how they respond
# to a thinning intervention, i.e. what state
# they change to.
states = { '11' : '01', '01' : '02', '07' : '05',
           '05' : '03', '08' : '06', '06' : '04',
           '10' : '10', '02' : '11',
            '03' : '11', '04' : '08'}

# Mapping of stand initial state to the state
# it will regenerate as.
initial_states = { '11' : '11', '01' : '11', '02' : '11',
                   '07' : '11','05' : '11', '03' : '11',
                   '08' : '08', '06' : '08',
                    '04' : '08' }

regen_state = initial_states[state]
regen_cc = { 1 : 18, 2 : 17, 3 : 16, 4 : 15 }

# Only the specified state can take this prescription
restrictions = { 'state' : [s for s in states.keys()
                             if s !=state] }


if cc:
    restrictions = { 'sc' : [s for s in site_classes
                              if s!=sc],
```

```
                    'state' : [s for s in states.keys()
                              if s !=state]
                    }

# Plantation stands are treated differently from naturals
if state in ('08', '06', '04'):

    if regen == 2:
        sched[0] = [in_age, in_age+5, in_age+9, 8, 13,
                8, 13]
        sched[1] = [states[state], states[states[state]],
                    '08', '06' ,'08', '06', '08']
        sched[2] = [in_age+1, in_age+6, 1, 9, 1, 9, 1]
        harv_type = ['01', '01', '11', '01',
                     '11', '01', '11']
    elif regen == 3:
        sched[0] = [in_age, in_age+5, in_age+9, 8, 13, 17,
                8, 13] #might change
        sched[1] = [states[state], states[states[state]],
                    '08', '06' ,'04', '08', '06','04', '08']
        sched[2] = [in_age+1, in_age+6, 1, 9, 14, 1, 9,
                    14, 1]
        harv_type = ['01', '01', '11', '01', '01', '11',
                     '01', '01', '11']
    elif regen == 4:
        sched[0] = [in_age, in_age+5, in_age+9, regen_cc[sc],
                    regen_cc[sc]] #might change
        sched[1] = [states[state], states[states[state]],
```

```
                 '08', '08', '08']

        sched[2] = [in_age+1, in_age+6, 1, 1, 1]
        harv_type = ['01', '01', '11', '11', '11']

    # If less than three entries are requested,
    # remove index 2 then index 1 from each sched[0]
    # and sched[1] remove index 1 then index 0 from
    # sched[2]
    for i in range(3 - entries):
        sched[0].remove(sched[0][2-i])
        sched[1].remove(sched[1][1-i])
        sched[2].remove(sched[2][1-i])
        harv_type.remove(harv_type[1-i])

# This where non plantation stands get shunted
else:
    if regen == 2:
        sched[0] = [in_age, in_age+5, in_age+9,
                    10, 15, 10,15]
        sched[1] = [states[state], states[states[state]],
                    '07','05', '07',  '05', '07' ]
        sched[2] = [in_age+1, in_age+6, 1, 11, 1, 11, 1]
        harv_type = ['01', '01', '11', '01',
                     '11', '01', '11']
    elif regen == 3:
        sched[0] = [in_age, in_age+5, in_age+9,
                    10, 15, 19, 10,15]
```

```
        sched[1] = [states[state], states[states[state]],'07',
                    '05', '03', '07', '05', '03', '07' ]
        sched[2] = [in_age+1, in_age+6, 1, 11, 16, 1,
                    11, 16, 1]
        harv_type = ['01', '01', '11', '01', '01',
                     '11', '01', '01' '11']
    elif regen == 4:
        sched[0] = [in_age, in_age+5, in_age+9,
                    regen_cc[sc], regen_cc[sc]] #might change
        sched[1] = [states[state], states[states[state]],
                    '11', '11', '11']

        sched[2] = [in_age+1, in_age+6, 1, 1, 1]
        harv_type = ['01', '01', '11', '11', '11']

    if entries < 3:
        for i in range(3 - entries):
            sched[0].remove(sched[0][2-i])
            sched[1].remove(sched[1][1-i])
            sched[2].remove(sched[2][1-i])
            harv_type.remove(harv_type[1-i])
        if entries == 2 and int(state) <= 7:
            sched[0][1] = in_age+4

new_pres = Pres(restrictions, name, number, sched,
                in_age, harv_type)
if regen == 4:
    alt_age = ""
```

```
if alt_age != "":
    # waiting time for the initial alternate entry
    # is determined by the number of the prescription
    # modulo 5
    wait = code%5

    # Again, plantation and natural are separated
    if state in ('08', '06', '04'):
        if regen == 2:
            alt_sched[0] = [wait, 8, 13, 8, 13, 8, 13]
            alt_sched[1] = ['08', '06', '08', '06',
                            '08', '06', '08']
            alt_sched[2] = [1, 9, 1, 9, 1, 9, 1, 9, 1]
            harv_types = ['11', '01', '11', '01', '11',
                          '01', '11', '01']
        elif regen == 3:
            alt_sched[0] = [wait, 8, 13, 17, 8, 13, 17, 8,
                            13]
            alt_sched[1] = ['08', '06', '04','08', '06',
                            '04', '08','06', '08']
            alt_sched[2] = [1, 9, 14, 1, 9, 14, 1, 9, 14, 1,
                            9, 1]
            harv_types = ['11', '01', '01', '11', '01', '01',
                          '11', '01']

    else:
        if regen == 2:
```

```
            alt_sched[0] = [wait, 3, 10, 15, 3, 10, 15, 3]
            alt_sched[1] = ['11', '07', '05', '11','07',
                            '05', '11', '07']
            alt_sched[2] = [1, 4,11,1,4,11,1,4]
            harv_types = ['11', '01', '01', '11', '01',
                          '01','11', '01', '01', '11']
        elif regen == 3:
            alt_sched[0] = [wait, 3,10, 15, 19, 3, 10, 15]
            alt_sched[1] = ['11','07','05', '03', '11',
                            '07', '05', '03','11','07' ]
            alt_sched[2] = [1, 4, 11, 16, 1, 4, 11, 16, 1]
            harv_types = ['11', '01', '01', '01', '11',
                          '01', '01','01', '11']

        # Add alternate track
        new_pres.add_alt(alt_sched, harv_types)

    new_pres.state = state

    return new_pres


def selection(in_age, state, code, alt_age = ""):
    """
    Return a selection harvest prescription based
    on initial age, state of forest it applies to
    and prescription number.
    An alternative age option can be specified as
```

```
well.

Selection harvests, after the initial intervention
occur every 4 years, and are modelled by setting the
stands age back to 12 after harvest of desirable species.

Key word args
alt_age: indicates whether this prescription has
    an alternate track
"""

# Boilerplate prescription object details
name = 'selection'
number = code
harv_type = '12'
sched = [0]*3
alt_sched = [0]*3

# Mapping of how selection harvests change
# the state of the forest
states = {'11': '12', '12': '12', '07' : '13',
          '13' : '13', '14' : '14', '10' : '14' }
# Dictionary of stands that are not applicable
never = ['01', '02', '03', '04', '05', '06', '08']
states_list = [s for s in states.keys() if s!= state]
for s in states_list:
    never.append(s)
restrictions = {'state' : never }
```

```
# Intervention, transition, and regeneration
# schedules respectively
sched[0] = [in_age, 16, 16, 16, 16, 16, 16,
            16, 16, 16]
sched[1] = [states[state] for i in range(10)]
sched[2] = [13 for i in range(10)]
harv_type = ['12', '12', '12', '12', '12',
             '12', '12', '12', '12','12']

# Instantiate prescription object and
# give it a new attribute that holds the
# state of the forest it applies to
# Note on Oct. 14 2013 - I don't think
# this attirbute is ever used
new_pres = Pres(restrictions, name, number,
                sched, in_age, harv_type)
new_pres.state = state

# Define potential alt_age prescription
if alt_age != "":
    wait = code%5
    alt_sched[0] = [wait, 3, 16, 16, 16, 16, 16, 16,
                    16, 16, 16, 16]
    alt_sched[1] = ['11', '07', '13', '13', '13', '13',
                    '13', '13', '13']
    alt_sched[2] = [1, 4, 13,13,13,13,13,13,13,
                    13,13,13,13]
```

```
        harv_type = ['11', '01', '12', '12', '12', '12',
                     '12', '12', '12','12', '12']

        new_pres.add_alt(alt_sched, harv_type)

    return new_pres

def buffer_harv(in_age, code, alt_age=""):
    """
    Define buffer prescription objects.
    Just like all the other ones.
    """
    name = 'buffer'
    number = code
    harv_type = '16'
    sched = [0]*3
    alt_sched = [0]*3

    # Buffer restrictions are only that it
    # must be  a buffer stand, and not in an excl.
    # These are checked prior to the stand getting
    # access to the prescription. This means
    # all states are eligible.
    restrictions = {'state' : ['08', '06', '04', '01',
                         '02', '03','05', '07','12',
                              '13', '14']}
    # Two tracks depending on the initial
    # age of the prescription
```

```
if in_age > 12:
    sched[0] = [in_age, 12, 24]
    sched[1] = [ '11', '11', '11']
    sched[2] = [1, 13, 1]
    harv_type = ['16', '16', '16']
else:
    sched[0] = [in_age, 24, 12, 24]
    sched[1] = ['11', '11', '11', '11']
    sched[2] = [13, 1, 13, 1]
    harv_type = ['16', '16', '16', '16']

new_pres = Pres(restrictions, name, number,
                sched, in_age, harv_type)
alt_age = ""
if alt_age != "":
    wait = random.randint(0,5)
    alt_sched[0] = [wait, 12, 24]
    alt_sched[1] = ['11', '11', '11']
    alt_sched[2] = [0, 13, 0]

    new_pres.add_alt(alt_sched)

return new_pres
```

A.3 AMPL Code for Model One models

```
# AMPL Code for Model One models
# from chapters 4 and 5.
# All models were constructed from
# .lp files generated in AMPL
# using this code. After .lp
# generation, models were modified
# either by removing constraints
# or changing variable objective
# coeffecients. .mps files for
# all models presented in this thesis
# are available from the author

# To make the industrial expansion
#  model used in Chapter 5.4.3
# replace all tables related to the
# shipping network with new tables
# that include the new mill. The
# model is otherwise unchanged.



/*Sets and Parameters*/

# Planning horizon is 30 periods from 0 to 29 inclusive
# Some constraints (e.g. NDY) only have indices going to
# 28, and some have indices going to 30: hence the
```

```
# x and y Periods parameters.
# Originally, ownerships were not modelled over the
# entire horizon, so they have their own parameter.
# Mills are only modelled for periods 0,1,2,3,4, and they
# have their own time horizon parameter as well
param yPeriods:=30;
param Periods:= 29;
param xPeriods:= 28;
param Mill_Periods:=4;
param Own_Periods:=29;

# a list,of stand ID numbers
set stands;

# set of all possible stand_types
set all_stand_types;

# list of all prescriptions
set all_pres;

# Prescription costs by intervention
# not included in modelling
set pres_cost;
param Pres_Cost{j in all_pres} default 0;

# a list of yield types modelled
set yield_types;
```

```
# set of yield_types that correspond to timber values
# 45,46: IHREBO logs and pulp
# 56,57: PIEHTL logs and pulp
# 92,93: SMYB logs and pulp
# 97,98: SPBF logs and pulp
set timber_yield_types;

# set of yield_types that correspond to ecosystem values
# 27: dev class
# 39: Hrv class
# 71 - Ser class
set eco_yield_types;

# a list of ecodistricts
set ecodistricts;

# list of mills
set mills;

# reduced list of mills
# used throughout the model
set red_mills;

# list of sawmills
set saw_mills;

# list of pulpmills
set pulp_mills;
```

```
# number of segments mill demand curves are broken up into
# not included in modelling
set segs;

# a list of the timbersheds under consideration
set tsheds;

# list of natural disturbance regimes
set ndrs;

# list of buffer status'
set buffers;

# list of exclusion status'
set excl;

# list of ownershps
set ownerships;

# Possible Newprot status
set new_prot;

# mapping of stand to buffer status
set stand_buffer;
param Stand_Buffer{stands};

# mapping of stand ID to stand area
```

```
set stand_areas;
param Stand_Area{stands};

#mapping of stand ID to initial stand_age
set in_stand_age;
param ISA{stands};

# Mapping of stand ID to stand_type in periods of each prescription
# when harvests take place. Periods where a stand is harvested
# have the current stand type recorded, periods where no harvest
# takes place have an entry of zero
set stand_types dimen 3;
param Stand_Type{i in stands, all_pres, 0..yPeriods} default 0;

# Mapping of initial stand age to current stand age in each period
#  of the model under each prescription
set stand_ages dimen 3;
param Stand_Age{a in 0..40, b in all_pres, t in 0..yPeriods} default 0;

# 15 July 2013
# Speed up model generation by telling the model
# which stands are eligible for each prescription
# rather than have the model determine based on Stand_Type
# parameter
set stand_pres dimen 2;
param Stand_Pres{i in stands, j in all_pres} default 0;

# Nothing goes past 30 periods, so for the purposes
```

```
# of this model, Stand_Age_2 is equivalent to Stand_Age.
# Note that Stand_Age_2 is used throughout the model
param Stand_Age_2{a in 0..40, b in all_pres, t in 0..yPeriods+1}:=
 if t = 31 then Stand_Age[a,b,t-1]+1
else Stand_Age[a,b,t];

# Prescriptions stand i is eligible for. All stands are eligible for
# prescription 0, i.e. "do nothing"
set pres{i in stands}:= setof{ j in all_pres : Stand_Pres[i,j] > 0 or j = 0 } j;

# Mapping of stand ID to stand_type in periods when stand_type
# changes as a result of stand death,
# or an intervention taking place.
set eco_stand_types dimen 3;
param Eco_Stand_Type{i in stands, j in pres[i], 0..yPeriods} default 0;

# Mapping of stand ID to initial stand_type
set in_stand_type;
param In_Stand_Type{stands};

# Fill in the stand_types of a particular stand between harvests.
# IST (intermediate stand_type) is identical to the Eco_Stand_Type table
#  (the one that maps stand_type transistions),
# except that it includes the stand_type in period 0 for every stand.
# Stand_Type_Eco has a entry for each stand, prescription and period,
#  recording the current stand_type
# The IST allows the parameter table of Stand_Type_eco
# to be populated entriely in GLPK.
```

```
# Unless the stand_type changes,
# Stand_Type_Eco[i,j,t] = Stand_Type_eco[i,j,t-1]
param IST{i in stands, j in pres[i], t in 0..yPeriods}:=
if t=0 then In_Stand_Type[i]
else Eco_Stand_Type[i,j,t];
param Stand_Type_Eco{i in stands, j in pres[i], t in 0..yPeriods}:=
 if IST[i,j,t]>0 then IST[i,j,t] else Stand_Type_Eco[i,j,t-1];

# Mapping of ISA[i] (initial stand age of i), j (prescription) and t (period)
# to the harvest type that is applied (if there is one applied for that (i,j,t) )
set harv_types dimen 3;
param Harv_Types{0..40, all_pres, 0..yPeriods} default 0;

# Mapping of Stand ID to stand ecodistrict
set stand_ecod;
param Stand_Ecod{stands};

# Mapping of Stand ID to stand timbershed
set stand_tshed;
param Stand_Tshed{stands};

# Mapping of stand to natural disturbance regime
set stand_ndr;
param Stand_Ndr{stands};

# Mapping of stand to exclusion status
set stand_excl;
param Stand_Excl{stands};
```

```
# Mapping of stand to ownership
set stand_ownership;
param Stand_Ownership{stands};

# Mapping of stand to NewProt status
set stand_newprot;
param Stand_Newprot{stands};

# Mapping of the yield of yield_type obtained from a stand of stand_type
# at age 0..40
set yields dimen 3;
param Yield{all_stand_types, yield_types, 0..40} default 0;

# Each mill has different demands, the volume to demand converter takes
# total harvested volume and converts it to dollars.
# demand_converter_own is used throughout the model. demand_converter
# was used before ownerships were introduced.
set demand_converter_own dimen 3;
param Demand_Converter_Own{ownerships, mills, yield_types} default 0;
set demand_converter dimen 2;
param Demand_Converter{mills, yield_types} default 0;

# Multipliers so that shelterwood harvests and buffer harvests
# can remove less than 100% of a stand’s volume
set multipliers dimen 2;
param Multipliers{all_pres,0..40} default 1;
```

```
# Mapping of the price wood will fetch in each segment of
# the demand curve at each mill
# Not included in thesis models
set price_segs dimen 2;
param Price_Segs{mills,segs} default 0;

# Mapping of the cost of shipping wood from a particular
# timbershed to a specific mill
set ship_cost dimen 2;
param Ship_Cost{mills, tsheds} default 0;

# Mapping of cost of shipping chips from a particular
# sawmill to the pulpmill
set intermill_ship_cost dimen 2;
param Intermill_Ship_Cost{mills, mills} default 0;

# Each mill has a level of demand that it must achieve
set min_demand;
param Min_Demand{mills};

# Each demand segment has a maximum demand;
# after this demand is reached,
# we move down to the next price segment
# not included in thesis models
set max_seg_demand dimen 2;
param Max_Seg_Demand{mills, segs};

# Mill capacities
```

```
set mill_cap;
param Mill_Cap{mills};

# list of watersheds in the model,
# along with mapping of stands to the
# watersheds they belong to
set watersheds;
set stand_watershed;
param Stand_Watershed{stands};

# list of protection statuses,
# mapping of stands to protection status
set protections;
set stand_protection;
param Stand_Protection{stands};



#Derived Sets#



# Mapping of stand_type, timber_yield_type
# and period to timber yield generated.
# Derived from the set yields and param Yield
set timber_yields:= setof{ (i,y,t) in yields: y in timber_yield_types } (i,y,t);
param Timber_Yield{ (i,y,t) in timber_yields }, default Yield[i,y,t];
```

```
# Mapping of stand_type, eco_yield_type and period to eco yield generated.
# Derived from the set yields and param Yield
set eco_yields:= setof{ (i,y,t) in yields: y in eco_yield_types } (i,y,t);
param Eco_Yield{ (i,y,t) in eco_yields }, default Yield[i,y,t];

# Mapping of stand_type, piece_yield and period to yield generated.
# Derived from the set yields and param Yield.
# Piecesize constraints were not included in thesis models
set piece_yields:=setof{ (i,y,t) in yields: y in {38,94} } (i,y,t);
param Piece_Yield{ (i,y,t) in piece_yields }, default Yield[i,y,t];

# For each period the set of stands that could be
# assigned a buffer harvest in that period
set buffered{t in 0..yPeriods}:=
setof{ i in stands, j in pres[i] : Stand_Buffer[i] = 1 and
Eco_Yield[Stand_Type_Eco[i,j,t], 39, Stand_Age_2[ISA[i],j,t]]>=1 and
Stand_Type[i,j,t]>0} (i,j);

# set of all stands with exclusion status
set excled:= setof{ i in stands : Stand_Excl[i] = 1 } i;

# Set of Harvestable Stands in a given period
set harvestable{t in 0..yPeriods}:= setof{ i in stands, j in pres[i] :
Eco_Yield[Stand_Type_Eco[i,j,t], 39, Stand_Age_2[ISA[i],j,t]]>=1 and
 Stand_Type[i,j,t]>0 } (i,j);

#Set of Harvest types that are considered partial harvest
# in each ecodistrict in each period
```

```
set partial{k in ecodistricts, t in 0..Periods}:=
setof{ (i,j) in harvestable[t] union buffered[t] : (Harv_Types[ISA[i],j,t] = 1 or
Harv_Types[ISA[i],j,t] = 12  or Harv_Types[ISA[i],j,t] = 16 or
Harv_Types[ISA[i],j,t] = 15) and Stand_Ecod[i] = k } (i,j);

# Set of stands in each ecodistrict and each period that could
# recieve a shelterwood harvest in that period
set shelt_harv{k in ecodistricts, t in 0..Periods}:=
setof{ (i,j) in partial[k,t] : Harv_Types[ISA[i],j,t] = 15 } (i,j);

# Set of stands and prescriptions belonging to ecodistrict k in period t
# that are eligible for harvest
set piece_harv{k in ecodistricts, t in 0..Periods}:=
setof{ (i,j) in harvestable[t] union buffered[t] : Stand_Ecod[i] = k  } (i,j);

# Only create transportation variables (z and p)
# for transportations that are feasible
# Only create sw pulp trans variables for mills
# that can process wood into softwood pulp
# If Demand_Converter_Own[u,m,y] it means
# that mill m accepts wood from onwership u of type y
set trans_own :=
setof{u in ownerships, m in red_mills, y in timber_yield_types :
Demand_Converter_Own[u,m,y] >0 } (u,m,y);
set pulp_trans_own :=
setof{ u in ownerships, m in saw_mills, n in pulp_mills, y in {46,57,93,98} :
Demand_Converter_Own[u,n,y]>0 and
 Demand_Converter_Own[u,m,y-1]>0} (u,m,n,y);
```

```
# set of stands and prescriptions (i,j) that are in development classes
# 5, 4 or 5, and 3,4 or 5, respectively
# indices are ecodistricts, natural disturbance regimes and periods
# Development class 5
set mat_area1{k in ecodistricts, n in ndrs, t in 0..yPeriods}:=
setof{ i in stands, j in pres[i] : Stand_Ecod[i] = k and Stand_Ndr[i]=n and
Eco_Yield[Stand_Type_Eco[i,j,t], 27, Stand_Age_2[ISA[i],j,t]] =5 } (i,j);
# Development class 4 or 5
set mat_area2{k in ecodistricts, n in ndrs, t in 0..yPeriods}:=
setof{ i in stands, j in pres[i] : Stand_Ecod[i] = k and Stand_Ndr[i]=n and
Eco_Yield[Stand_Type_Eco[i,j,t], 27, Stand_Age_2[ISA[i],j,t]] >=4 } (i,j);
# Development class 3, 4, or 5
set mat_area3{k in ecodistricts, n in ndrs, t in 0..yPeriods}:=
setof{ i in stands, j in pres[i] : Stand_Ecod[i] = k and Stand_Ndr[i]=n and
Eco_Yield[Stand_Type_Eco[i,j,t], 27, Stand_Age_2[ISA[i],j,t]] >=3 } (i,j);

# set of stands and prescriptions in a late seral stage in each
# ecodistrict, natural disturbance regime and period
set late_area{k in ecodistricts, n in ndrs, t in 0..yPeriods}:=
setof{ i in stands, j in pres[i] : Stand_Ecod[i] = k and Stand_Ndr[i]=n and
Eco_Yield[Stand_Type_Eco[i,j,t], 74, Stand_Age_2[ISA[i],j,t]] = 3 } (i,j);

# Set of all stands belonging to each ecodistrict
# and natural disturbance regime
set ndr_ecod{k in ecodistricts, n in ndrs}:=
setof{ i in stands : Stand_Ecod[i]=k and Stand_Ndr[i] = n } i;
```

```
# All stands and prescriptions belonging to ownership u
# with a clearcut intervention scheduled for period t
set clearcuts_own{u in ownerships, t in 0..Own_Periods}:=
setof{ (i,j) in harvestable[t] : (Harv_Types[ISA[i],j,t] = 11 or
Harv_Types[ISA[i],j,t] = 8) and Stand_Ownership[i] = u } (i,j);

# All stands and prescirptions belonging to ownership u with a
# shelterwood or buffer prescription scheduled for period t
set shelt_buffer_own{u in ownerships, t in 0..Own_Periods}:= \
setof{ (i,j) in harvestable[t] union buffered[t] : (Harv_Types[ISA[i],j,t] = 15 or
Harv_Types[ISA[i],j,t] = 16) and Stand_Ownership[i] = u } (i,j);

# All stands (in ownership u) and prescriptions with a selection
# entry occuring in period t
set sel_cuts_own{u in ownerships, t in 0..Own_Periods}:=
setof{ (i,j) in harvestable[t] : Harv_Types[ISA[i], j, t] = 12
and Stand_Ownership[i] = u } (i,j);

# All stands and prescriptions in ownership u and prescriptions
# with a thinning occuring  in period t
set thin_cuts_own{u in ownerships, t in 0..Own_Periods}:=
setof{ (i,j) in harvestable[t] : Harv_Types[ISA[i], j, t] = 1 and
Stand_Ownership[i] = u} (i,j);

# All stands and prescriptions with a partial entry (thinning or selection)
# in ownership u in period t
set partial_cuts_own{u in ownerships, t in 0..Own_Periods}:=
setof{ (i,j) in sel_cuts_own[u,t] union thin_cuts_own[u,t] } (i,j);
```

```
# All stands and prescriptions belonging to watershed w
# existing in an establishment development class in period t
set young_area_watershed{w in watersheds, t in 0..Periods}:=
setof{ i in stands, j in pres[i] : Stand_Watershed[i] = w and
Eco_Yield[Stand_Type_Eco[i,j,t], 27, Stand_Age_2[ISA[i],j,t]] = 1} (i,j);


/*Data*/

# Stand ID numbers
table tab_Stands IN "C:\Users\Andrew Martin\Desktop
\Documents\Models\16 August Model\stands.tab":
stands <- [Stand];
read table tab_Stands;


# List of prescriptions
table tab_All_Pres IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\all_pres.tab":
all_pres <- [Pres];
read table tab_All_Pres;


# All possible stand_types
table tab_All_Stand_Types IN "C:\Users\Andrew Martin\Desktop\
29_July\stand_types_list.tab":
all_stand_types <- [Stand_Type];
```

```
read table tab_All_Stand_Types;

# Listing of all yield types
table tab_Yield_Types IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\yield_types.tab":
yield_types <- [Yield_Types];
read table tab_Yield_Types;

# Listing of all Timber Yield Types
table tab_Timber_Yield_Types IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\timber_yield_types.tab":
timber_yield_types <- [Yield_Types];
read table tab_Timber_Yield_Types;

# Al Ecosystem Yield Types
table tab_Eco_Yield_Types IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\eco_yield_types.tab":
 eco_yield_types <- [Yield_Types];
read table tab_Eco_Yield_Types;

# Ecodistricts
table tab_Ecodistricts IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\ecodistricts.tab":
 ecodistricts <- [Ecod];
read table tab_Ecodistricts;

# Mills (deprecated)
table tab_Mills IN "C:\Users\Andrew Martin\Desktop\
```

```
Documents\Models\16 August Model\mills.tab":
 mills <- [Mill];
read table tab_Mills;

# Mills list as of 2 July 2013
table tab_Mills_Red IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\red_mills.tab":
 red_mills <- [Mill];
read table tab_Mills_Red;

# Pulp Mills
table tab_Pulp_Mills IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\pulp_mills.tab":
 pulp_mills <- [Mill];
read table tab_Pulp_Mills;

# Saw Mills
table tab_Saw_Mills IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\saw_mills.tab":
 saw_mills <- [Mill];
read table tab_Saw_Mills;

# Demand Segments (deprecated)
table tab_Segs IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\segs.tab":
 segs <- [Seg];
read table tab_Segs;
```

```
# Timbersheds
table tab_Tsheds IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\tsheds.tab":
tsheds <- [Tshed];
read table tab_Tsheds;

# Natural Disturbance Regimes
table tab_Ndrs IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\ndrs.tab":
ndrs <- [Ndr];
read table tab_Ndrs;

# Exclusion Statuses
table tab_Excl IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\excl.tab":
excl <- [Excl];
read table tab_Excl;

# Buffer statuses
table tab_Buffer IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\buffers.tab":
buffers <- [Buffer];
read table tab_Buffer;

# Mapping Stand ID to Stand Area
table tab_Stand_Area IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_areas.tab":
stand_areas <- [Stand], Stand_Area ~Area;
```

```
read table tab_Stand_Area;

# Mapping stand ID to initial age
table tab_In_Stand_Age IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_ages_pro.tab":
in_stand_age <- [Stand], ISA ~Age;
read table tab_In_Stand_Age;

#Mapping stand ID to stand age in each period under each prescription
table tab_Stand_Age IN "C:\Users\Andrew Martin\Desktop\
29_July\stand_age_period.tab":
stand_ages <- [Initial_Age, Pres, Period], Stand_Age ~Age;
read table tab_Stand_Age;

# Mapping stand ID to initial stand_type
table tab_In_Stand_Type IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_types_pro.tab":
 in_stand_type <- [Stand], In_Stand_Type ~Stand_Type;
read table tab_In_Stand_Type;

#15 July try to speed up generation time by reducing lookups
# for stand prescriptions
# Mapping stands to eligible prescriptions
table tab_Stand_Pres IN "C:\Users\Andrew Martin\Desktop\
29_July\stand_pres.tab":
 stand_pres <- [Stand, Pres], Stand_Pres ~Val;
read table tab_Stand_Pres;
```

```
# Mapping Stand ID to stand_type for timber yields
table tab_Stand_Types IN "C:\Users\Andrew Martin\Desktop\\
29_July\stand_types.tab":
stand_types <- [Stand, Pres, Period], Stand_Type ~Stand_Type;
read table tab_Stand_Types;

# Mapping Stand ID to stand_type for ecosystem yields
table tab_Stand_Types_Eco IN "C:\Users\Andrew Martin\Desktop\
29_July\stand_types_eco.tab":
eco_stand_types <- [Stand, Pres, Period],
Eco_Stand_Type ~Stand_Type;
read table tab_Stand_Types_Eco;

# Mapping Stand ID to ecodistrict
table tab_Stand_Ecod IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_ecod.tab":
 stand_ecod <- [Stand], Stand_Ecod ~Ecod;
read table tab_Stand_Ecod;

# Stand ID to timbershed
table tab_Stand_Tshed IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_tshed.tab":
stand_tshed <- [Stand], Stand_Tshed ~Tshed;
read table tab_Stand_Tshed;

# Stand ID to ndr
table tab_Stand_Ndr IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_ndr.tab":
```

```
stand_ndr <- [Stand], Stand_Ndr ~Ndr;
read table tab_Stand_Ndr;

# Mapping Stand ID to buffer status
table tab_Stand_Buffer IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_buffer.tab":
stand_buffer <- [Stand], Stand_Buffer ~Buffer;
read table tab_Stand_Buffer;

# Mapping Stand ID to buffer status
table tab_Stand_Excl IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_excl.tab":
stand_excl <- [Stand], Stand_Excl ~Excl;
read table tab_Stand_Excl;

# Mapping of Stand Type, Yield Type and Stand Age to
# Yield value
table tab_Yields IN "C:\Users\Andrew Martin\Desktop\
29_July\yields.tab":
 yields <- [Stand_Type,Yield_Type,Age], Yield ~Yield;
read table tab_Yields;

# Timber yields from partial cuts such as Buffers and Shelterwoods
# are given a multiplier < 1 to reflect that not all the volume is
#  being harvested
table tab_Multipliers IN "C:\Users\Andrew Martin\Desktop\
29_July\multipliers.tab":
multipliers <- [Pres,Age], Multipliers ~Mult;
```

```
read table tab_Multipliers;

# Shipping costs from timbershed to mill
table tab_Ship_Cost IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\ship_cost.tab":
 ship_cost <- [Mill,Tshed], Ship_Cost ~Cost;
read table tab_Ship_Cost;

# Shipping costs from mills to other mills
table tab_Inter_Mill IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\inter_mill.tab":
 intermill_ship_cost <- [Mill_1, Mill_2], Intermill_Ship_Cost ~Cost;
read table tab_Inter_Mill;

# Minimum level of demand at each mill
table tab_Min_Demand IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\min_demand.tab":
min_demand <- [Mill], Min_Demand ~Min_Demand;
read table tab_Min_Demand;

# Conversion rate of total volume to dollars at each mill
# deprecated, see new demand converter below
table tab_Demand_Converter IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\demand_converter.tab":
 demand_converter <- [Mill, Yield], Demand_Converter ~Conv;
read table tab_Demand_Converter;

# Harvest type for each intervention of each prescription
```

```
table tab_Harv_Types IN "C:\Users\Andrew Martin\Desktop\
29_July\harv_types.tab":
harv_types <- [Initial_Age, Pres, Period], Harv_Types ~Cut_Type;
read table tab_Harv_Types;

# Mill capacities
table tab_Mill_Cap IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\mill_cap.tab":
 mill_cap <- [Mill], Mill_Cap ~Cap;
read table tab_Mill_Cap;

# listing of ownerships
table tab_Ownerships IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\ownerships.tab":
ownerships <- [Ownership];
read table tab_Ownerships;

# Mapping of Stand Id to ownership
table tab_Stand_Ownership IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_ownership.tab":
stand_ownership <- [Stand], Stand_Ownership ~Ownership;
read table tab_Stand_Ownership;

# Conversion of cubic meters to dollars of wood
# of each type, from each ownership, to
# each mill
table tab_Demand_Converter_Own IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\demand_converter_own.tab":
```

```
 demand_converter_own <- [Ownership, Mill, Yield], Demand_Converter_Own ~Conv;
read table tab_Demand_Converter_Own;

# Listing of protection statuses
table tab_protections IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\protections.tab":
 protections <- [Protection];
read table tab_protections;

# Mapping stand ID to protection status
table tab_stand_protection IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_protection.tab":
stand_protection <- [Stand], Stand_Protection ~Protection;
read table tab_stand_protection;

# Listing of watersheds
table tab_watersheds IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\watersheds.tab":
 watersheds <- [Wshed];
read table tab_watersheds;

# Mapping stand ID to watershed
table tab_Stand_Watershed IN "C:\Users\Andrew Martin\Desktop\
Documents\Models\16 August Model\stand_watershed.tab":
 stand_watershed <- [Stand], Stand_Watershed ~Watershed;
read table tab_Stand_Watershed;
```

```
/*Model*/

#Model Variables#

# Number of hectares of stand i alloted to prescription j
var x{i in stands, j in pres[i]} >=0;

# Demand ($) supplied to mill m in period t
var d{ m in mills, t in 0..Mill_Periods} >=0;

# Volume of wood of type y shipped from ownership u
# and timbershed r to mill m in period t
var z{r in tsheds, (u,m,y) in trans_own, t in 0..Mill_Periods} >=0;

# Volume of chips transported of type y from ownership u
# to mill n from mill m in period t
var p{ (u,m,n,y) in pulp_trans_own, t in 0..Mill_Periods}>=0;


#Inventory Variables#

# Softwood Volumes harvested each period each ownership
# Total softwood volume harvested from ownership u in period t
var SWVOL_OWN{u in ownerships, t in 0..Periods};
# Spruce-Fir volume harvested from ownership u in period t
var SPBFVOL_OWN{u in ownerships, t in 0..Periods};
# Pine, Eastern Hemlock, Tamarack Larch harvested from ownership u
```

```
# in period t
var PIEHTLVOL_OWN{u in ownerships, t in 0..Periods};
# Spruce-fir operable inventory in each ownership and period
var SWINV_OWN{ u in ownerships, t in 0..yPeriods};

#Total Volume harvested each period from each ownership
var TOTVOL_OWN{u in ownerships, t in 0..Periods};

# Volumes of hardwood harvested each period and each ownership
# Total hardwood harvest by ownership and period
var HWVOL_OWN{u in ownerships, t in 0..Periods};
# Sugar-Maple, Yellow-Birch harvest by ownership and period
var SMYBVOL_OWN{u in ownerships, t in 0..Periods};
# Intolerant hardwood, beech, red oak harvest by ownership and period
var IHBEROVOL_OWN{u in ownerships, t in 0..Periods};

# Volume of timber yield type y from ownership u harvested from
# timbershed r in period t
var YIELDVOL_OWN{u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods};

# Volume of type y from ownership u and timbershed r
# harvested in period t using selection
var SEL_CUT_OWN{u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods};
# Volume of type y from ownership u and timbershed r harvested
#  in period t using selection
var THIN_CUT_OWN{u in ownerships, y in timber_yield_types,
```

```
 r in tsheds, t in 0..Own_Periods};
# Sum or SEL_CUT_OWN and THIN_CUT_OWN
var PART_CUTS_OWN{u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods};
# Volume of type y from ownership u and timbershed r in period t
# harvested by buffer harvest or shelterwood harvest
var SHELT_BUFF_OWN{u in ownerships, y in timber_yield_types,
 r in tsheds,t in 0..Own_Periods};
# Volume of type y from ownership u and timbershed r in period t
# harvested by clearcut
var CC_OWN{u in ownerships, y in timber_yield_types,
r in tsheds, t in 0..Own_Periods};

# Variables to inventory the area in development class
# 5, 4 and 5, and 3,4, and 5 respectively
var OLDAREA1{k in ecodistricts, n in ndrs, t in 0..yPeriods};
var OLDAREA2{k in ecodistricts, n in ndrs,t in 0..yPeriods};
var OLDAREA3{k in ecodistricts, n in ndrs,t in 0..yPeriods};

# The area ineach watershed that is in establishment
# class in each period
var YOUNGAREAWAT{w in watersheds, t in 0..yPeriods};

# The area of late seral stage forest in each ecodistrict,
# and ndr in each period
var LATEAREA{k in ecodistricts, n in ndrs,t in 0..yPeriods};

#Area harvested in each ecodistrict and period
```

```
var AREAHARV{k in ecodistricts, t in 0..Periods};



#Feasibility Variables#

# Violation of MatDevCls and SerCl constraints
# This is what is penalized in the objective function
var J{0..3,k in ecodistricts, n in ndrs, t in 0..yPeriods}>=0;



# Economic Variables#

# Revenue generated at mill from timber
var SALES{m in red_mills,t in 0..Mill_Periods};
# Transportation costs of shipping wood from ownership u
# to mill m in period t
var TRANS{u in ownerships, m in red_mills, t in 0..Mill_Periods};
# Net of sales and trans at mill m in period t
var MILLREV{red_mills, t in 0..Mill_Periods};
# Sum of all revenue at all mills
var Rev{t in 0..Mill_Periods};


# For Chapter 5 models this would be maximize Obj: sum(t in 0..Mill_Periods} Rev[t];
# Below is Chapter 4 objective function
maximize Obj: sum{u in ownerships, t in 0..Periods} SPBFVOL_OWN[u,t];
```

```
#Define Inventory Variables#

# Sales are demand supplied at the mill in dollars
# plus $20 per cubic meter incentive to ship chips
s.t. Sales{ m in red_mills, t in 0..Mill_Periods}:
SALES[m,t] =  d[m,t]+ 20*sum{(u,m,n,y) in pulp_trans_own} p[u,m,n,y,t];

# $6.50 fixed cost plus $0.07/km/m^3 for both chips and solid wood
# Cost is charged to the mill receiving the wood
s.t. Trans{ u in ownerships, m in red_mills, t in 0..Mill_Periods}:
TRANS[u,m,t] =
sum{ (u,m,y) in trans_own, r in tsheds} z[r,u,m,y,t]*(6.5+Ship_Cost[m,r]) +
sum{ (u,n,m,y) in pulp_trans_own} p[u,n,m,y,t]*(6.5+Intermill_Ship_Cost[n,m]);

# Net of SALES and TRANS
# 5% per period discount rate applies
s.t. MillRevenue{m in red_mills, t in 0..Mill_Periods}:
MILLREV[m,t] = (0.95**t)*( SALES[m,t] - sum{u in ownerships}TRANS[u,m,t] );

# Total Revenue of the system
s.t. Revenue{t in 0..Mill_Periods}:
Rev[t] = sum{m in red_mills} MILLREV[m,t];

# All harvests are modelled are they are in the NSDNR Woodstock model.
# Yields from selection harvests are computed as the difference in yield
# values before the intervention takes place and after the intervention takes place
```

```
s.t. sel_cut_own{ u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods}:
SEL_CUT_OWN[u,y,r,t] = (sum{ (i,j) in sel_cuts_own[u,t] : Stand_Tshed[i] = r}
if ((Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]] -
Timber_Yield[Stand_Type_Eco[i,j,t+1],y, Stand_Age_2[ISA[i],j,t+1]-1])) >0 then
((Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]] -
Timber_Yield[Stand_Type_Eco[i,j,t+1],y, Stand_Age_2[ISA[i],j,t+1]-1]))*x[i,j]
else 0);

# Yields from thinnings are computed the same way as for selection harvests
s.t. thin_cut_own{ u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods}:
THIN_CUT_OWN[u,y,r,t] = (sum{ (i,j) in thin_cuts_own[u,t] : Stand_Tshed[i] = r}
if ((Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]] -
Timber_Yield[Stand_Type_Eco[i,j,t+1],y, Stand_Age_2[ISA[i],j,t+1]-1])) >0 then
((Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]] -
Timber_Yield[Stand_Type_Eco[i,j,t+1],y, Stand_Age_2[ISA[i],j,t+1]-1]))*x[i,j]
else 0);

# Sum volume from thinnings and selection harvests
s.t. part_cut_own{ u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods}:
PART_CUTS_OWN[u,y,r,t] = THIN_CUT_OWN[u,y,r,t] +
SEL_CUT_OWN[u,y,r,t];

# Clearcuts take all volume off a stand.
s.t. cc_own{u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods}:
```

```
CC_OWN[u,y,r,t] = (sum{ (i,j) in clearcuts_own[u,t] : Stand_Tshed[i] = r }
(Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]]*x[i,j]));

# Shelterwoods and buffers take a fraction of the volume off a stand.
# This is communicated to the model via the Multipliers parameters
s.t. shelterwoodbuffer_own{ u in ownerships, y in timber_yield_types,
 r in tsheds, t in 0..Own_Periods}:
SHELT_BUFF_OWN[u,y,r,t] = sum { (i,j) in shelt_buffer_own[u,t] : Stand_Tshed[i] = r}
Multipliers[j,Stand_Age_2[ISA[i],j,t]]*
Timber_Yield[Stand_Type[i,j,t],y, Stand_Age_2[ISA[i],j,t]]*x[i,j];

# Variables storing the hardwood volume harvested in each period
# IHBERO - Intolerant Hardwood, Beech, Red Oak
# SMYB - Sugar Maple and Yellow Birch

s.t. IHBEROLog_Own{u in ownerships, y in {45}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. IHBEROPulp_Own{u in ownerships, y in {46}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. IHBEROVolOwn{u in ownerships, t in 0..Periods}:
IHBEROVOL_OWN[u,t] =
sum{ r in tsheds} (YIELDVOL_OWN[u,45,r,t] + YIELDVOL_OWN[u,46,r,t]);

s.t. SMYBLog_Own{u in ownerships, y in {92}, r in tsheds, t in 0..Own_Periods}:
```

```
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. SMYBPulp_Own{u in ownerships, y in {93}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. SMYBVolOwn{u in ownerships, t in 0..Own_Periods}:
SMYBVOL_OWN[u,t] =
sum{ r in tsheds} (YIELDVOL_OWN[u,92,r,t] + YIELDVOL_OWN[u,93,r,t]);

s.t. HardwoodVolOwn{u in ownerships, t in 0..Periods}:
HWVOL_OWN[u,t] =
SMYBVOL_OWN[u,t] + IHBEROVOL_OWN[u,t];

# Variables storing the softwood volume harvested from each period
# PIEHTL - Pine, Eastern Hemlock, Tamarck Larch
# SPBF - Sprcue Fir Volume

s.t. PIEHTLLog_Own{u in ownerships, y in {56}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. PIEHTLPulp_Own{u in ownerships, y in {57}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. PIEHTLVolOwn{u in ownerships, t in 0..Periods}:
```

```
PIEHTLVOL_OWN[u,t] =
sum{r in tsheds}(YIELDVOL_OWN[u,56,r,t] + YIELDVOL_OWN[u,57,r,t]);

s.t. SPBFLog_Own{u in ownerships, y in {97}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. SPBFPulp_Own{u in ownerships, y in {98}, r in tsheds, t in 0..Own_Periods}:
YIELDVOL_OWN[u,y,r,t] =
CC_OWN[u,y,r,t]+ PART_CUTS_OWN[u,y,r,t] + SHELT_BUFF_OWN[u,y,r,t];

s.t. SPBFVolOwn{u in ownerships, t in 0..Periods}:
SPBFVOL_OWN[u,t] =
sum{r in tsheds} (YIELDVOL_OWN[u,97,r,t] + YIELDVOL_OWN[u,98,r,t]);

s.t. SoftwoodVolOwn{u in ownerships, t in 0..Periods}:
SWVOL_OWN[u,t] =
 SPBFVOL_OWN[u,t] + PIEHTLVOL_OWN[u,t];

s.t. TotVolOwn{u in ownerships, t in 0..Periods}:
TOTVOL_OWN[u,t] =
SWVOL_OWN[u,t] + HWVOL_OWN[u,t];

# Variables storing the area of the forest in a mature
# development class, for each ecodistrict and ndr
# Area in Development class 5
s.t. MatArea1{t in 0..yPeriods, k in ecodistricts, n in ndrs}:
OLDAREA1[k,n,t] =  sum{ (i,j) in mat_area1[k,n,t] } x[i,j];
```

```
# Area in late-Mat and old Age development classes (4 and 5)
s.t. MatArea2{t in 0..yPeriods, k in ecodistricts, n in ndrs}:
OLDAREA2[k,n,t] =  sum{ (i,j) in mat_area2[k,n,t] } x[i,j];

#Area in early-Mat, late-Mat, and old Age (3,4 5)
s.t. MatArea3{t in 0..yPeriods, k in ecodistricts, n in ndrs}:
OLDAREA3[k,n,t] =  sum{ (i,j) in mat_area3[k,n,t] } x[i,j];

# Area in a late seral stage in each period
s.t. LateArea{t in 0..yPeriods, k in ecodistricts, n in ndrs}:
LATEAREA[k,n,t] = sum{ (i,j) in late_area[k,n,t] } x[i,j];

# Area harvested in each ecodistrict in each time period
s.t. AreaHarvested{k in ecodistricts, t in 0..Periods}:
AREAHARV[k,t] =
sum{ (i,j) in harvestable[t] union buffered[t] : Stand_Ecod[i]=k } x[i,j];

# Operable Spruce-Fir growing stock
s.t. SwinvOwn{u in ownerships,t in 0..yPeriods}:
SWINV_OWN[u,t] =
sum{ i in stands, j in pres[i] : Stand_Ownership[i] = u and
Stand_Excl[i] = 0 and
Eco_Yield[Stand_Type_Eco[i,j,t], 39, Stand_Age_2[ISA[i],j,t] ]>=1}
( Yield[Stand_Type_Eco[i,j,t], 97, Stand_Age_2[ISA[i],j,t]]+
Yield[Stand_Type_Eco[i,j,t], 98, Stand_Age_2[ISA[i],j,t]] )*x[i,j];

# Area in each watershed in an establishment development class
```

```
s.t. YoungAreaWat{w in watersheds, t in 0..Periods}:
YOUNGAREAWAT[w,t] = sum{ (i,j) in young_area_watershed[w,t] } x[i,j];



/*Constraints*/


#Timber Constraints#

# Area Accounting
s.t. Area_Accounting{i in stands}:
sum{j in pres[i]} x[i,j] = Stand_Area[i];

# Non-decreasing flow of spruce-fir from each ownership
s.t. SW_Even_Flow1{u in ownerships, t in 0..xPeriods}:
SPBFVOL_OWN[u,t]  <= SPBFVOL_OWN[u,t+1];

# Non-declining operable growing stock of Sruce-fir from each ownership
s.t. ndInv{u in ownerships, t in 12..Periods}:
SWINV_OWN[u,t] <= SWINV_OWN[u,t+1];

# Limit low-value species to less than 25% of harvest
s.t. Other_Vol{u in ownerships, t in 0..Periods}:
PIEHTLVOL_OWN[u,t] + IHBEROVOL_OWN[u,t] <= 0.25*TOTVOL_OWN[u,t];

# Development Class Seral Score Constraints. At least X% of the forest in each ndr
# and in each ecodistrict must be in each seral/dev class
```

```
# FREQ NDR
s.t. MatDevCls1a{k in ecodistricts, t in 11..yPeriods}:
OLDAREA1[k,0,t] + J[0,k,0,t] >=0.08*sum{ i in ndr_ecod[k,0] } Stand_Area[i];
s.t. MatDevCls1b{k in ecodistricts, t in 11..yPeriods}:
OLDAREA2[k,0,t] + J[1,k,0,t] >=0.16*sum{ i in ndr_ecod[k,0] } Stand_Area[i];
s.t. MatDevCls1c{k in ecodistricts, t in 11..yPeriods}:
OLDAREA3[k,0,t] + J[2,k,0,t] >=0.4*sum{ i in ndr_ecod[k,0] } Stand_Area[i];

# INFREQ NDR
s.t. MatDevCls2a{k in ecodistricts, t in 11..yPeriods}:
OLDAREA1[k,1,t] + J[0,k,1,t] >=0.16*sum{ i in ndr_ecod[k,1] } Stand_Area[i];
s.t. MatDevCls2b{k in ecodistricts, t in 11..yPeriods}:
OLDAREA2[k,1,t] + J[1,k,1,t] >=0.27*sum{ i in ndr_ecod[k,1] } Stand_Area[i];
s.t. MatDevCls2c{k in ecodistricts, t in 11..yPeriods}:
OLDAREA3[k,1,t] + J[2,k,1,t] >=0.6*sum{ i in ndr_ecod[k,1] } Stand_Area[i];

# GAP NDR
s.t. MatDevCls3a{k in ecodistricts, t in 11..yPeriods}:
OLDAREA1[k,2,t] + J[0,k,2,t] >=0.24*sum{ i in ndr_ecod[k,2] } Stand_Area[i];
s.t. MatDevCls3b{k in ecodistricts, t in 11..yPeriods}:
OLDAREA2[k,2,t] + J[1,k,2,t] >=0.38*sum{ i in ndr_ecod[k,2] } Stand_Area[i];
s.t. MatDevCls3c{k in ecodistricts, t in 11..yPeriods}:
OLDAREA3[k,2,t] + J[2,k,2,t] >=0.8*sum{ i in ndr_ecod[k,2] } Stand_Area[i];

# FREQ, INFREQ and GAP in order
s.t. SeralClsa{k in ecodistricts, t in 11..yPeriods}:
LATEAREA[k,0,t] + J[3,k,0,t] >=0.2*sum{ i in ndr_ecod[k,0] } Stand_Area[i];
s.t. SeralClsb{k in ecodistricts, t in 11..yPeriods}:
```

```
LATEAREA[k,1,t] + J[3,k,1,t] >=0.4*sum{ i in ndr_ecod[k,1] } Stand_Area[i];
s.t. SeralClsc{k in ecodistricts, t in 11..yPeriods}:
LATEAREA[k,2,t] + J[3,k,2,t] >=0.7*sum{ i in ndr_ecod[k,2] } Stand_Area[i];


#Demand Constraints#


# The amount of wood of each type shipped from each timbershed to all mills
# has to equal the amount of wood of each type harvested from each timbershed
s.t. ShipAcct{u in ownerships, r in tsheds, y in timber_yield_types,
 t in 0..Mill_Periods}:
sum{ (u,m,y) in trans_own} z[r,u,m,y,t] = YIELDVOL_OWN[u,y,r,t];

# Convert cubic meters to dollars at mills
s.t. WoodConv{m in red_mills, t in 0..Mill_Periods}:
d[m,t] =
sum{r in tsheds, (u,m,y) in trans_own} Demand_Converter_Own[u,m,y]*z[r,u,m,y,t] +
sum{ (u,n,m,y) in pulp_trans_own } Demand_Converter_Own[u,m,y]*p[u,n,m,y,t];

# Mill capcity
s.t. MillCapacity{m in red_mills, t in 0..Mill_Periods}:
sum{r in tsheds, (u,m,y) in trans_own} z[r,u,m,y,t] +
sum{ (u,n,m,y) in pulp_trans_own} p[u,n,m,y,t] <= Mill_Cap[m];

# Mill minimum demand levels
s.t. MinDemand{m in red_mills, t in 0..Mill_Periods}:
d[m,t] >= Min_Demand[m];
```

```
# Amount of wood shipped as pulp chips can equal up to 1/2 the volume
# of solid wood of that type
s.t. PulpChips{ u in ownerships, m in red_mills, y in {46,57,93,98},
 t in 0..Mill_Periods :
(u,m,y-1) in trans_own }:
sum{(u,m,n,y) in pulp_trans_own} p[u,m,n,y,t] <=
0.5*sum{r in tsheds, (u,m,y) in trans_own} z[r,u,m,y-1,t];

# Less than 10% of sw pulp mills feedstock can come from low-value species
s.t. FeedStockPulp{ m in {1,7}, t in 0..Mill_Periods}:
sum{ r in tsheds, u in ownerships : (u,m,57) in trans_own }
(z[r,u,m,57,t]-0.1*z[r,u,m,98,t]) +
sum{u in ownerships, n in saw_mills : (u,n,m,57) in pulp_trans_own}
(p[u,n,m,57,t] - 0.1*p[u,n,m,98,t]) <= 0;
# Less than 10% of sw saw mill feedstock can come from low-value species
s.t. FeedStockSaw{ m in {0,4,6}, t in 0..Mill_Periods}:
sum{ r in tsheds, (u,m,56) in trans_own}
(z[r,u,m,56, t] - 0.1*z[r,u,m,97,t]) <= 0;

#in the first two periods at least 25% of the forest must be not clearcut
# (shelter, selection, buffer, ct)
s.t.Part_Harv2{k in ecodistricts, t in 0..2}:
sum{ (i,j) in partial[k,t] } x[i,j] >=
0.25*AREAHARV[k,t];

#in the first 18 periods at least 50% of the forest must be not clearcut
s.t. Part_Harv3{k in ecodistricts,t in 2..Periods}:
sum{ (i,j) in partial[k,t] } x[i,j] >=
```

```
0.5*AREAHARV[k,t];

# Alternate Regulation - each mill is to receive a non-declining value of
# Wood over the planning horizon
s.t. MillProf{m in red_mills, t in 0..Mill_Periods-1}:
MILLREV[m,t] <= MILLREV[m,t+1];

# Each watershed must have no more than 40% of its area
# in an establishment development class
s.t. MinForestCover{ w in watersheds, t in 5..Periods}:
 YOUNGAREAWAT[w,t] <=
 0.4*( sum{ i in stands : Stand_Watershed[i] = w} Stand_Area[i]);

end;
```

A.4 Model Two Phase 1 Woodstock Files

A.4.1 Prescription Files

Action File

```
;Precommercial Thinning in Natural Stands
*ACTION aPT   N  "Precommercial Thinning"
  *OPERABLE aPT _CP 1.._LENGTH
   ? ? NBUF ? ? ? ? ? 4567 NRG NOEXCLSET NOEXCLSET _AGE = 3

;Clearcut based harvest/silviculture Systems
;-------------------------------------------
;Clearcut and Leave for Natural
*ACTION aCC_LN Y "Clear-Cut Harvest & Leave For Natural"
 *OPERABLE aCC_LN _CP  8.._LENGTH
   ? ? NBUF ? ? ? ? ? 3 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 14<=_AGE<=18
   ? ? NBUF ? ? ? ? ? 4 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 13<=_AGE<=17
   ? ? NBUF ? ? ? ? ? 5 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 12<=_AGE<=16
   ? ? NBUF ? ? ? ? ? 6 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 11<=_AGE<=15

   ? ? NBUF ? ? ? ? ? ? CTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
   ? ? NBUF ? ? ? ? ? ? CTCTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 21
   ? ? NBUF ? ? ? ? ? ? CTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 15
   ? ? NBUF ? ? ? ? ? ? CTCTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 19
   ? ? NBUF ? ? ? ? ? ? CTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 13
   ? ? NBUF ? ? ? ? ? ? CTCTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
*OPERABLE acc_LN _CP 6..7
```

```
  ? ? NBUF ? ? ? ? ? 3 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 14 <=_AGE <= 18
  ? ? NBUF ? ? ? ? ? 4 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 13 <=_AGE <= 17
  ? ? NBUF ? ? ? ? ? 5 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 12<=_AGE <=  16
  ? ? NBUF ? ? ? ? ? 6 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and 11<=_AGE <=15
  ? ? NBUF ? ? ? ? ? ? PLT NOEXCLSET NOEXCLSET yHrvcls >= 1 and _AGE = 8
  ? ? NBUF ? ? ? ? ? ? CTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
  ? ? NBUF ? ? ? ? ? ? CTCTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 21
  ? ? NBUF ? ? ? ? ? ? CTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 15
  ? ? NBUF ? ? ? ? ? ? CTCTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 19
  ? ? NBUF ? ? ? ? ? ? CTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 13
  ? ? NBUF ? ? ? ? ? ? CTCTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
 *OPERABLE aCC_LN _CP 1..5
  ? ? NBUF ? ? ? ? ? 3 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE >= 14
  ? ? NBUF ? ? ? ? ? 4 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE >= 13
  ? ? NBUF ? ? ? ? ? 5 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE >= 12
  ? ? NBUF ? ? ? ? ? 6 NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE >= 11

  ? ? NBUF ? ? ? ? ? ? PLT NOEXCLSET NOEXCLSET yHrvcls >= 1 and _AGE = 8
  ? ? NBUF ? ? ? ? ? ? CTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
  ? ? NBUF ? ? ? ? ? ? CTCTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 21
  ? ? NBUF ? ? ? ? ? ? CTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 15
  ? ? NBUF ? ? ? ? ? ? CTCTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 19
  ? ? NBUF ? ? ? ? ? ? CTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 13
  ? ? NBUF ? ? ? ? ? ? CTCTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17

;*ACTION aCC_PLT Y "Clearcut for Plantations"
; *OPERABLE aCC_LN _CP 1..7
```

```
;*ACTION CC_CT Y "Final Felling for Commercial Thins"
 ;*OPERABLE CC_CT _CP 1.._LENGTH
   ;? ? NBUF ? ? ? ? ? ? CTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
   ;? ? NBUF ? ? ? ? ? ? CTCTH NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 21
   ;? ? NBUF ? ? ? ? ? ? CTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 15
   ;? ? NBUF ? ? ? ? ? ? CTCTPCT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 19
   ;? ? NBUF ? ? ? ? ? ? CTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 13
   ;? ? NBUF ? ? ? ? ? ? CTCTPLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 17
  ;; ? ? NBUF ? ? ? ? ? ? PLT NOEXCLSET NOEXCLSET yHrvCls >= 1 and _AGE = 9


;Partial Harvest / Silviculture Systems
;-------------------------------------------
;Commercial Thinning
*ACTION aCT Y "Commercial Thinning in Natural Regeneration"
  *OPERABLE aCT _CP 1.._LENGTH
  ? ? NBUF ? ? ? ? ? ? NRG   NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 12
  ? ? NBUF ? ? ? ? ? ? CTH   NOEXCLSET NOEXCLSET   yHrvCls >= 1 and _AGE = 17
  ? ? NBUF ? ? ? ? ? ? PCT   NOEXCLSET NOEXCLSET   yHrvCls >= 1 and _AGE = 10
  ? ? NBUF ? ? ? ? ? ? CTPCT NOEXCLSET NOEXCLSET   yHrvCls >= 1 and _AGE = 15
  ? ? NBUF ? ? ? ? ? ? PLT   NOEXCLSET NOEXCLSET    yHrvCls >= 1 and _AGE = 8
    ? ? NBUF ? ? ? ? ? ? CTPLT NOEXCLSET NOEXCLSET   yHrvCls >= 1 and _AGE = 13
  *PARTIAL aCT
   yRSv yBSv yWSv yBFv yWPv yRPv yEHv yTLv yRMv yWBv yXAv ySMv yBEv yYBv yROv yEXv
   ySPv yPIv yIHv yTHv ySv yHv yTv
   ySPBFv ySMYBv yPIEHTLv yIHBEROv yPIEHTLIHBEROv
   yRSvLog yBSvLog yWSvLog yBFvLog yWPvLog yRPvLog yEHvLog yTLvLog yRMvLog
```

```
   yWBvLog yXAvLog ySMvLog yBEvLog yYBvLog yROvLog yEXvLog
   ySPvLog yPIvLog yIHvLog yTHvLog ySvLog yHvLog yTvLog
   ySPBFvLog ySMYBvLog yPIEHTLvLog yIHBEROvLog yPIEHTLIHBEROvLog


;Shelterwood Harvest (see regime section for details)
*REGIME rSL
*ACTION aSL N "Shelterwood Harvest (Regime)"
  *PARTIAL aSL
   yRSv yBSv yWSv yBFv yWPv yRPv yEHv yTLv yRMv yWBv yXAv ySMv yBEv yYBv yROv yEXv
   ySPv yPIv yIHv yTHv ySv yHv yTv
   ySPBFv ySMYBv yPIEHTLv yIHBEROv yPIEHTLIHBEROv
   yRSvLog yBSvLog yWSvLog yBFvLog yWPvLog yRPvLog yEHvLog yTLvLog yRMvLog
   yWBvLog yXAvLog ySMvLog yBEvLog yYBvLog yROvLog yEXvLog
   ySPvLog yPIvLog yIHvLog yTHvLog ySvLog yHvLog yTvLog
   ySPBFvLog ySMYBvLog yPIEHTLvLog yIHBEROvLog yPIEHTLIHBEROvLog
   yHrvCls
;Shelterwood Harvest - Overstory REmoval
*ACTION aOR Y "Overstory Removal Harvest (Regime)"

;20m Water Buffer Harvest (see regime section for details)
*REGIME rBH

*ACTION aBH N "Buffer Harvest (Regime)"
  *PARTIAL aBH
   yRSv yBSv yWSv yBFv yWPv yRPv yEHv yTLv yRMv yWBv yXAv ySMv yBEv yYBv yROv yEXv
   ySPv yPIv yIHv yTHv ySv yHv yTv
   ySPBFv ySMYBv yPIEHTLv yIHBEROv yPIEHTLIHBEROv
```

```
  yRSvLog yBSvLog yWSvLog yBFvLog yWPvLog yRPvLog yEHvLog yTLvLog yRMvLog
 yWBvLog  yXAvLog ySMvLog yBEvLog yYBvLog yROvLog yEXvLog
  ySPvLog yPIvLog yIHvLog yTHvLog ySvLog yHvLog yTvLog
  ySPBFvLog ySMYBvLog yPIEHTLvLog yIHBEROvLog yPIEHTLIHBEROvLog
  yHrvCls
;20m Water Buffer Harvest (regenerate action - no harvest volume generated)
*ACTION aBR Y "Buffer Harvest - Regenerate Action - No Harvest Volume Generated (Regime)"


;Selection Harvest
*ACTION aSH Y "Selection Harvest"
 *OPERABLE aSH _CP 1.._LENGTH
 ? ? NBUF ? ? ? ? ? ? NAENRG NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 16
 ? ? NBUF ? ? ? ? ? ? NAU    NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 16
 ? ? NBUF ? ? ? ? ? ? PCT    NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 16
 ? ? NBUF ? ? ? ? ? ? SEL    NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 16
 *PARTIAL aSH
  yRSv yBSv yWSv yBFv yWPv yRPv yEHv yTLv yRMv yWBv yXAv ySMv
  ySPv yPIv yIHv yTHv ySv yHv yTv yBEv yYBv yROv
  ySPBFv ySMYBv yPIEHTLv yIHBEROv yPIEHTLIHBEROv
  yRSvLog yBSvLog yWSvLog yBFvLog yWPvLog yRPvLog yEHvLog yTLvLog yRMvLog
  yWBvLog yXAvLog ySMvLog yBEvLog yYBvLog yROvLog
  ySPvLog yPIvLog yIHvLog yTHvLog ySvLog yHvLog yTvLog
  ySPBFvLog ySMYBvLog yPIEHTLvLog yIHBEROvLog yPIEHTLIHBEROvLog
  yHrvCls

;All Partial Harvest Actions
*AGGREGATE aPH
```

```
  aCT aSH aSL aOR aBH
  rSL rBH

;All Full/Total Harvest Actions
*AGGREGATE aCC
  aCC_LN

;All Harvest Actions
*AGGREGATE aHarvest
  aCC_LN
  aCT aSH aSL aOR aBH
  rSL rBH
*AGGREGATE aHarvestNoBH
  aCC_LN
  aCT aSH aSL aOR aBH
  rSL

;All Actions
*AGGREGATE aTreated
  aPT
  aCC_LN
  aCT aSH aSL aOR aBH
  rSL rBH
```

Regimes File

This file has been truncated for readability.

```
*REGIME rSL Shelterwood Regime

  *OPERABLE rSL

    ? ? NBUF ? ? ? ? ? ? NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE = 14


  *PRESCRIPTION Remove40%

    *OPERABLE ? ? ? ? ? ? ? ? ? ? ? ? 

      _RXPERIOD _ACTION  _ENTRY   yHrvCls yTv  yHv  ySv  ySPBFv ySMYBv yPIEHTLv

          0     aSL      _INITIAL 100%    100% 100% 100% 100%   100%   100%

          0      -           -  100%     60%  60%  60%  60%    60%    60%

          2     aOR      _FINAL   100%     60%  60%  60%  60%    60%    60%

         ; 2     aOR      _FINAL   100%     60%  60%  60%  60%    60%    60%



*REGIME rBH Buffer Harvest Regime

  *OPERABLE rBH

    ? ? WBUF ? ? ? ? ? ? NAENRGNAU NOEXCLSET NOEXCLSET  yHrvCls >= 1 and _AGE >= 12


 *PRESCRIPTION Remove30%

    *OPERABLE ? ? ? ? ? ? ? ? ? ? ? ? 

      _RXPERIOD _ACTION  _ENTRY   yHrvCls yTv  yHv  ySv  ySPBFv ySMYBv yPIEHTLv

          0     aBH      _INITIAL 100%    100% 100% 100% 100%   100%   100%

          0      -           -    100%    70%  70%  70%  70%    70%    70%      70%

          12    aBH         -      100%    100%  100%  100%  100%    100%

          12     -         -    100%    70%  70%  70%  70%    70%    70%

          13     aBR        _FINAL 100%    0%  0%  0%  0%    0%    0%      0%
```

Transition File

```
{=================================================================}
{                          /\     Nova Scotia                     }
{                         /  \    Department of Natural Resources }
{                        /    \   Forestry Division               }
{ Tel:   (902)-893-5696 /______\  Forest Management Planning      }
{ Fax:   (902)-893-6102    |      Resource Modeling Section       }
{=================================================================}
{ Project     : Crown Lands Strategic Forest Model                }
{=================================================================}

;CASE aCC_Simp
;  *SOURCE ? ? ? ? ? ? ? ? ? FTGminusSEL ? ?
 ;   *TARGET ? ? ? ? ? ? ? D ? NRG ? ? 100 _AGE 0 _LOCK 1



;Claercut and Leave for Natural
*CASE  aCC_LN ;Clear cut Transitions ;NewRegenDataBasedTransitionsForCC
  ;Hardwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? 3 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 3 NRG ? ? 100 _AGE 0 _LOCK 18
  *SOURCE ? ? ? ? ? ? ? ? 4 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 4 NRG ? ? 100 _AGE 0 _LOCK 17
  *SOURCE ? ? ? ? ? ? ? ? 5 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 5 NRG ? ? 100 _AGE 0 _LOCK 16
  *SOURCE ? ? ? ? ? ? ? ? 6 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 6 NRG ? ? 100 _AGE 0 _LOCK 15
```

```
*SOURCE ? ? ? ? ? ? ? ? ? CTX ? ?
  *TARGET ? ? ? ? ? ? ? D ? PCT ? ? 100 _AGE 0
*SOURCE ? ? ? ? ? ? ? ? ? CTPLTCTCTPLT ? ?
  *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0
*SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
  *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0
;*SOURCE ? ? ? ? ? ? ? ? 3 PLall ? ?
  ;*TARGET ? ? ? ? ? ? ? D 3 PLT ? ? 100 _AGE 0 _LOCK 18
;*SOURCE ? ? ? ? ? ? ? ? 4 PLall ? ?
  ;*TARGET ? ? ? ? ? ? ? D 4 PLT ? ? 100 _AGE 0 _LOCK 17
;*SOURCE ? ? ? ? ? ? ? ? 5 PLall ? ?
  ;*TARGET ? ? ? ? ? ? ? D 5 PLT ? ? 100 _AGE 0 _LOCK 16
;*SOURCE ? ? ? ? ? ? ? ? 6 PLall ? ?
  ;*TARGET ? ? ? ? ? ? ? D 6 PLT ? ? 100 _AGE 0 _LOCK 15

;*SOURCE ? ? ? ? ? ? ? ? 3 FTGminusSEL ? ?
  ;*TARGET ? ? ? ? ? ? ? D 3 NRG ? ? 100 _AGE 0 _LOCK 18
;*SOURCE ? ? ? ? ? ? ? ? 4 FTGminusSEL ? ?
  ;*TARGET ? ? ? ? ? ? ? D 4 NRG ? ? 100 _AGE 0 _LOCK 17
;*SOURCE ? ? ? ? ? ? ? ? 5 FTGminusSEL ? ?
  ;*TARGET ? ? ? ? ? ? ? D 5 NRG ? ? 100 _AGE 0 _LOCK 16
;*SOURCE ? ? ? ? ? ? ? ? 6 FTGminusSEL ? ?
  ;*TARGET ? ? ? ? ? ? ? D 6 NRG ? ? 100 _AGE 0 _LOCK 15
 ;; *SOURCE ? ? ? ? ? ? ? ? ? PLall ? ?
 ;;  *TARGET ? ? ? ? ? ? ? ? ? PLT ? ? 100 _AGE 0

; *SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
 ;  *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0 _LOCK 9
```

```
;Precommercial Thinning
*CASE aPT ;Precommercial Thinning Transitions
 *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
   *TARGET ? ? ? ? ? ? ? ? ? PCT ? ?  100 _LOCK 2

;Commercial Thinning
*CASE aCT ;Commercial Thinning Transitions
  *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTH ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTH ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTH ? ? 100 _AGE 17 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PCT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTPCT ? ? 100 _AGE 10 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTPCT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTPCT ? ? 100 _AGE 15 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTPLT ? ? 100 _AGE 8 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTPLT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTPLT ? ? 100 _AGE 13 _LOCK 4


;Shelterwood
*CASE aSL
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ? 100 _LOCK 2
;Shelterwood Harvest - Ovberstory Removal
*CASE aOR
```

```
;Hardwood Cover Types
*SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
  *TARGET ? ? ? ? ? ? ? D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,101)
  ;*TARGET  ? ? ? ? ? ? HIHw  D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,102)
  ;*TARGET  ? ? ? ? ? ? HITHw D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,103)
  ;*TARGET  ? ? ? ? ? ? HTHw  D ? NRG ? ? 100 _AGE 1
;;Mixedwood Cover Types
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,201)
  ;*TARGET  ? ? ? ? ? ? MIHwHS D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,202)
  ;*TARGET  ? ? ? ? ? ? MIHwSH D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,203)
  ;*TARGET  ? ? ? ? ? ? MTHw   D ? NRG ? ? 100 _AGE 1
;;Softwood Cover Types
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,301)
  ;*TARGET  ? ? ? ? ? ? SrSbSDom D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,302)
  ;*TARGET  ? ? ? ? ? ? SwSDom   D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,303)
  ;*TARGET  ? ? ? ? ? ? SbFDom   D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,304)
  ;*TARGET  ? ? ? ? ? ? SSpbFDom D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,305)
  ;*TARGET  ? ? ? ? ? ? SPiDom   D ? NRG ? ? 100 _AGE 1
;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,306)
```

```
    ;*TARGET  ? ? ? ? ? ? SMHePiSp D ? NRG ? ? 100 _AGE 1

;Buffer Harvest (30% removal)
*CASE aBH
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ? 100 _LOCK 2
;Buffer Harvest (regenerate action - no harvest volume generated)
*CASE aBR
  ;Hardwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,101)
    *TARGET  ? ? ? ? ? ? HIHw  D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,102)
    *TARGET  ? ? ? ? ? ? HITHw D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,103)
    *TARGET  ? ? ? ? ? ? HTHw  D ? NRG ? ? 100 _AGE 0
  ;Mixedwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,201)
    *TARGET  ? ? ? ? ? ? MIHwHS D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,202)
    *TARGET  ? ? ? ? ? ? MIHwSH D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,203)
    *TARGET  ? ? ? ? ? ? MTHw   D ? NRG ? ? 100 _AGE 0
  ;Softwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,301)
    *TARGET  ? ? ? ? ? ? SrSbSDom D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,302)
    *TARGET  ? ? ? ? ? ? SwSDom   D ? NRG ? ? 100 _AGE 0
```

```
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,303)
    *TARGET  ? ? ? ? ? ? SbFDom   D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,304)
    *TARGET  ? ? ? ? ? ? SSpbFDom D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,305)
    *TARGET  ? ? ? ? ? ? SPiDom   D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,306)
    *TARGET  ? ? ? ? ? ? SMHePiSp D ? NRG ? ? 100 _AGE 0


;SELECTION HARVEST
*CASE aSH {SELECTION HARVEST};Selection Harvest Transitions
  *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
    *TARGET ? ? ? ? ? ? ? ? ? SELNE ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? NAU ? ?
    *TARGET ? ? ? ? ? ? ? ? ? SELNU ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PCT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? SELNP ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? SEL ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ?     100 _AGE 12 _LOCK 4

{STAND BREAK-UP}
*CASE _DEATH  {What Happens at Lifespan}
  *SOURCE  ? ? ? ? ? ? ? ? ? PLTCTPLTCTCTPLT ? ?
    *TARGET  ? ? ? ? ? ? ? ? ? ESC ? ? 100 _AGE 0  _LOCK 20
  *SOURCE  ? ? ? ? ? ? ? ? ? FTG ? ?
    *TARGET  ? ? ? ? ? ? ? ? ? ? ? ? 100 _AGE 0 _LOCK 20
```

A.4.2 Other Files

Optimize File

```
*VARIABLE
 SPBFVOL

*OBJECTIVE
 _MAX SPBFVOL - _PENALTY(_ALL)  _LENGTH

*CONSTRAINTS
     SPBFVOL - _SUM(ohvSFF) = 0 20


{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
{1:ECOSYSTEM VALUES: LANDSCAPE COMPOSITIONAL INDICATORS}
{---------------------------------------------------------------------}
oOld(330FREQ) >= 0.080 * oELCHa(330FREQ) 11.._LENGTH  _GOAL(FREQ01, 120)
oM2o(330FREQ) >= 0.160 * oELCHa(330FREQ) 11.._LENGTH  _GOAL(FREQ02, 120)
oMto(330FREQ) >= 0.400 * oELCHa(330FREQ) 11.._LENGTH  _GOAL(FREQ03, 120)
oLat(330FREQ) >= 0.200 * oELCHa(330FREQ) 11.._LENGTH  _GOAL(FREQ04, 120)
oOld(330GAP) >= 0.240 * oELCHa(330GAP) 11.._LENGTH  _GOAL(GAP11, 120)
oM2o(330GAP) >= 0.380 * oELCHa(330GAP) 11.._LENGTH  _GOAL(GAP12, 120)
oMto(330GAP) >= 0.800 * oELCHa(330GAP) 11.._LENGTH  _GOAL(GAP13, 120)
oLat(330GAP) >= 0.700 * oELCHa(330GAP) 11.._LENGTH  _GOAL(GAP14, 120)
oOld(340FREQ) >= 0.080 * oELCHa(340FREQ) 11.._LENGTH  _GOAL(FREQ21, 120)
oM2o(340FREQ) >= 0.160 * oELCHa(340FREQ) 11.._LENGTH  _GOAL(FREQ22, 120)
oMto(340FREQ) >= 0.400 * oELCHa(340FREQ) 11.._LENGTH  _GOAL(FREQ23, 120)
```

```
oLat(340FREQ) >= 0.200 * oELCHa(340FREQ) 11.._LENGTH  _GOAL(FREQ24, 120)
oOld(340GAP) >= 0.240 * oELCHa(340GAP) 11.._LENGTH  _GOAL(GAP31, 120)
oM2o(340GAP) >= 0.380 * oELCHa(340GAP) 11.._LENGTH  _GOAL(GAP32, 120)
oMto(340GAP) >= 0.800 * oELCHa(340GAP) 11.._LENGTH  _GOAL(GAP33, 120)
oLat(340GAP) >= 0.700 * oELCHa(340GAP) 11.._LENGTH  _GOAL(GAP34, 120)
oOld(340INFREQ) >= 0.160 * oELCHa(340INFREQ) 11.._LENGTH  _GOAL(INFREQ41, 120)
oM2o(340INFREQ) >= 0.270 * oELCHa(340INFREQ) 11.._LENGTH  _GOAL(INFREQ42, 120)
oMto(340INFREQ) >= 0.600 * oELCHa(340INFREQ) 11.._LENGTH  _GOAL(INFREQ43, 120)
oLat(340INFREQ) >= 0.400 * oELCHa(340INFREQ) 11.._LENGTH  _GOAL(INFREQ44, 120)
oOld(350FREQ) >= 0.080 * oELCHa(350FREQ) 11.._LENGTH  _GOAL(FREQ51, 120)
oM2o(350FREQ) >= 0.160 * oELCHa(350FREQ) 11.._LENGTH  _GOAL(FREQ52, 120)
oMto(350FREQ) >= 0.400 * oELCHa(350FREQ) 11.._LENGTH  _GOAL(FREQ53, 120)
oLat(350FREQ) >= 0.200 * oELCHa(350FREQ) 11.._LENGTH  _GOAL(FREQ54, 120)
oOld(350GAP) >= 0.240 * oELCHa(350GAP) 11.._LENGTH  _GOAL(GAP61, 120)
oM2o(350GAP) >= 0.380 * oELCHa(350GAP) 11.._LENGTH  _GOAL(GAP62, 120)
oMto(350GAP) >= 0.800 * oELCHa(350GAP) 11.._LENGTH  _GOAL(GAP63, 120)
oLat(350GAP) >= 0.700 * oELCHa(350GAP) 11.._LENGTH  _GOAL(GAP64, 120)
oOld(350INFREQ) >= 0.160 * oELCHa(350INFREQ) 11.._LENGTH  _GOAL(INFREQ71, 120)
oM2o(350INFREQ) >= 0.270 * oELCHa(350INFREQ) 11.._LENGTH  _GOAL(INFREQ72, 120)
oMto(350INFREQ) >= 0.600 * oELCHa(350INFREQ) 11.._LENGTH  _GOAL(INFREQ73, 120)
oLat(350INFREQ) >= 0.400 * oELCHa(350INFREQ) 11.._LENGTH  _GOAL(INFREQ74, 120)
oOld(370FREQ) >= 0.080 * oELCHa(370FREQ) 11.._LENGTH  _GOAL(FREQ81, 120)
oM2o(370FREQ) >= 0.160 * oELCHa(370FREQ) 11.._LENGTH  _GOAL(FREQ82, 120)
oMto(370FREQ) >= 0.400 * oELCHa(370FREQ) 11.._LENGTH  _GOAL(FREQ83, 120)
oLat(370FREQ) >= 0.200 * oELCHa(370FREQ) 11.._LENGTH  _GOAL(FREQ84, 120)
oOld(370GAP) >= 0.240 * oELCHa(370GAP) 11.._LENGTH  _GOAL(GAP91, 120)
oM2o(370GAP) >= 0.380 * oELCHa(370GAP) 11.._LENGTH  _GOAL(GAP92, 120)
oMto(370GAP) >= 0.800 * oELCHa(370GAP) 11.._LENGTH  _GOAL(GAP93, 120)
```

```
oLat(370GAP) >= 0.700 * oELCHa(370GAP) 11.._LENGTH  _GOAL(GAP94, 120)
oOld(370INFREQ) >= 0.160 * oELCHa(370INFREQ) 11.._LENGTH  _GOAL(INFREQ101, 120)
oM2o(370INFREQ) >= 0.270 * oELCHa(370INFREQ) 11.._LENGTH  _GOAL(INFREQ102, 120)
oMto(370INFREQ) >= 0.600 * oELCHa(370INFREQ) 11.._LENGTH  _GOAL(INFREQ103, 120)
oLat(370INFREQ) >= 0.400 * oELCHa(370INFREQ) 11.._LENGTH  _GOAL(INFREQ104, 120)
oOld(380FREQ) >= 0.080 * oELCHa(380FREQ) 11.._LENGTH  _GOAL(FREQ111, 120)
oM2o(380FREQ) >= 0.160 * oELCHa(380FREQ) 11.._LENGTH  _GOAL(FREQ112, 120)
oMto(380FREQ) >= 0.400 * oELCHa(380FREQ) 11.._LENGTH  _GOAL(FREQ113, 120)
oLat(380FREQ) >= 0.200 * oELCHa(380FREQ) 11.._LENGTH  _GOAL(FREQ114, 120)
oOld(380GAP) >= 0.240 * oELCHa(380GAP) 11.._LENGTH  _GOAL(GAP121, 120)
oM2o(380GAP) >= 0.380 * oELCHa(380GAP) 11.._LENGTH  _GOAL(GAP122, 120)
oMto(380GAP) >= 0.800 * oELCHa(380GAP) 11.._LENGTH  _GOAL(GAP123, 120)
oLat(380GAP) >= 0.700 * oELCHa(380GAP) 11.._LENGTH  _GOAL(GAP124, 120)
oOld(380INFREQ) >= 0.160 * oELCHa(380INFREQ) 11.._LENGTH  _GOAL(INFREQ131, 120)
oM2o(380INFREQ) >= 0.270 * oELCHa(380INFREQ) 11.._LENGTH  _GOAL(INFREQ132, 120)
oMto(380INFREQ) >= 0.600 * oELCHa(380INFREQ) 11.._LENGTH  _GOAL(INFREQ133, 120)
oLat(380INFREQ) >= 0.400 * oELCHa(380INFREQ) 11.._LENGTH  _GOAL(INFREQ134, 120)
oOld(410GAP) >= 0.240 * oELCHa(410GAP) 11.._LENGTH  _GOAL(GAP141, 120)
oM2o(410GAP) >= 0.380 * oELCHa(410GAP) 11.._LENGTH  _GOAL(GAP142, 120)
oMto(410GAP) >= 0.800 * oELCHa(410GAP) 11.._LENGTH  _GOAL(GAP143, 120)
oLat(410GAP) >= 0.700 * oELCHa(410GAP) 11.._LENGTH  _GOAL(GAP144, 120)
oOld(410INFREQ) >= 0.160 * oELCHa(410INFREQ) 11.._LENGTH  _GOAL(INFREQ151, 120)
oM2o(410INFREQ) >= 0.270 * oELCHa(410INFREQ) 11.._LENGTH  _GOAL(INFREQ152, 120)
oMto(410INFREQ) >= 0.600 * oELCHa(410INFREQ) 11.._LENGTH  _GOAL(INFREQ153, 120)
oLat(410INFREQ) >= 0.400 * oELCHa(410INFREQ) 11.._LENGTH  _GOAL(INFREQ154, 120)
oOld(420FREQ) >= 0.080 * oELCHa(420FREQ) 11.._LENGTH  _GOAL(FREQ161, 120)
oM2o(420FREQ) >= 0.160 * oELCHa(420FREQ) 11.._LENGTH  _GOAL(FREQ162, 120)
oMto(420FREQ) >= 0.400 * oELCHa(420FREQ) 11.._LENGTH  _GOAL(FREQ163, 120)
```

```
oLat(420FREQ) >= 0.200 * oELCHa(420FREQ) 11.._LENGTH  _GOAL(FREQ164, 120)
oOld(420GAP) >= 0.240 * oELCHa(420GAP) 11.._LENGTH  _GOAL(GAP171, 120)
oM2o(420GAP) >= 0.380 * oELCHa(420GAP) 11.._LENGTH  _GOAL(GAP172, 120)
oMto(420GAP) >= 0.800 * oELCHa(420GAP) 11.._LENGTH  _GOAL(GAP173, 120)
oLat(420GAP) >= 0.700 * oELCHa(420GAP) 11.._LENGTH  _GOAL(GAP174, 120)
oOld(420INFREQ) >= 0.160 * oELCHa(420INFREQ) 11.._LENGTH  _GOAL(INFREQ181, 120)
oM2o(420INFREQ) >= 0.270 * oELCHa(420INFREQ) 11.._LENGTH  _GOAL(INFREQ182, 120)
oMto(420INFREQ) >= 0.600 * oELCHa(420INFREQ) 11.._LENGTH  _GOAL(INFREQ183, 120)
oLat(420INFREQ) >= 0.400 * oELCHa(420INFREQ) 11.._LENGTH  _GOAL(INFREQ184, 120)
oOld(430FREQ) >= 0.080 * oELCHa(430FREQ) 11.._LENGTH  _GOAL(FREQ191, 120)
oM2o(430FREQ) >= 0.160 * oELCHa(430FREQ) 11.._LENGTH  _GOAL(FREQ192, 120)
oMto(430FREQ) >= 0.400 * oELCHa(430FREQ) 11.._LENGTH  _GOAL(FREQ193, 120)
oLat(430FREQ) >= 0.200 * oELCHa(430FREQ) 11.._LENGTH  _GOAL(FREQ194, 120)
oOld(430GAP) >= 0.240 * oELCHa(430GAP) 11.._LENGTH  _GOAL(GAP201, 120)
oM2o(430GAP) >= 0.380 * oELCHa(430GAP) 11.._LENGTH  _GOAL(GAP202, 120)
oMto(430GAP) >= 0.800 * oELCHa(430GAP) 11.._LENGTH  _GOAL(GAP203, 120)
oLat(430GAP) >= 0.700 * oELCHa(430GAP) 11.._LENGTH  _GOAL(GAP204, 120)
oOld(440FREQ) >= 0.080 * oELCHa(440FREQ) 11.._LENGTH  _GOAL(FREQ211, 120)
oM2o(440FREQ) >= 0.160 * oELCHa(440FREQ) 11.._LENGTH  _GOAL(FREQ212, 120)
oMto(440FREQ) >= 0.400 * oELCHa(440FREQ) 11.._LENGTH  _GOAL(FREQ213, 120)
oLat(440FREQ) >= 0.200 * oELCHa(440FREQ) 11.._LENGTH  _GOAL(FREQ214, 120)
oOld(440GAP) >= 0.240 * oELCHa(440GAP) 11.._LENGTH  _GOAL(GAP221, 120)
oM2o(440GAP) >= 0.380 * oELCHa(440GAP) 11.._LENGTH  _GOAL(GAP222, 120)
oMto(440GAP) >= 0.800 * oELCHa(440GAP) 11.._LENGTH  _GOAL(GAP223, 120)
oLat(440GAP) >= 0.700 * oELCHa(440GAP) 11.._LENGTH  _GOAL(GAP224, 120)
oOld(440INFREQ) >= 0.160 * oELCHa(440INFREQ) 11.._LENGTH  _GOAL(INFREQ231, 120)
oM2o(440INFREQ) >= 0.270 * oELCHa(440INFREQ) 11.._LENGTH  _GOAL(INFREQ232, 120)
oMto(440INFREQ) >= 0.600 * oELCHa(440INFREQ) 11.._LENGTH  _GOAL(INFREQ233, 120)
```

```
oLat(440INFREQ) >= 0.400 * oELCHa(440INFREQ) 11.._LENGTH  _GOAL(INFREQ234, 120)
oOld(450FREQ) >= 0.080 * oELCHa(450FREQ) 11.._LENGTH  _GOAL(FREQ241, 120)
oM2o(450FREQ) >= 0.160 * oELCHa(450FREQ) 11.._LENGTH  _GOAL(FREQ242, 120)
oMto(450FREQ) >= 0.400 * oELCHa(450FREQ) 11.._LENGTH  _GOAL(FREQ243, 120)
oLat(450FREQ) >= 0.200 * oELCHa(450FREQ) 11.._LENGTH  _GOAL(FREQ244, 120)
oOld(450GAP) >= 0.240 * oELCHa(450GAP) 11.._LENGTH  _GOAL(GAP251, 120)
oM2o(450GAP) >= 0.380 * oELCHa(450GAP) 11.._LENGTH  _GOAL(GAP252, 120)
oMto(450GAP) >= 0.800 * oELCHa(450GAP) 11.._LENGTH  _GOAL(GAP253, 120)
oLat(450GAP) >= 0.700 * oELCHa(450GAP) 11.._LENGTH  _GOAL(GAP254, 120)
oOld(450INFREQ) >= 0.160 * oELCHa(450INFREQ) 11.._LENGTH  _GOAL(INFREQ261, 120)
oM2o(450INFREQ) >= 0.270 * oELCHa(450INFREQ) 11.._LENGTH  _GOAL(INFREQ262, 120)
oMto(450INFREQ) >= 0.600 * oELCHa(450INFREQ) 11.._LENGTH  _GOAL(INFREQ263, 120)
oLat(450INFREQ) >= 0.400 * oELCHa(450INFREQ) 11.._LENGTH  _GOAL(INFREQ264, 120)
oOld(530FREQ) >= 0.080 * oELCHa(530FREQ) 11.._LENGTH  _GOAL(FREQ271, 120)
oM2o(530FREQ) >= 0.160 * oELCHa(530FREQ) 11.._LENGTH  _GOAL(FREQ272, 120)
oMto(530FREQ) >= 0.400 * oELCHa(530FREQ) 11.._LENGTH  _GOAL(FREQ273, 120)
oLat(530FREQ) >= 0.200 * oELCHa(530FREQ) 11.._LENGTH  _GOAL(FREQ274, 120)
oOld(530GAP) >= 0.240 * oELCHa(530GAP) 11.._LENGTH  _GOAL(GAP281, 120)
oM2o(530GAP) >= 0.380 * oELCHa(530GAP) 11.._LENGTH  _GOAL(GAP282, 120)
oMto(530GAP) >= 0.800 * oELCHa(530GAP) 11.._LENGTH  _GOAL(GAP283, 120)
oLat(530GAP) >= 0.700 * oELCHa(530GAP) 11.._LENGTH  _GOAL(GAP284, 120)
oOld(530INFREQ) >= 0.160 * oELCHa(530INFREQ) 11.._LENGTH  _GOAL(INFREQ291, 120)
oM2o(530INFREQ) >= 0.270 * oELCHa(530INFREQ) 11.._LENGTH  _GOAL(INFREQ292, 120)
oMto(530INFREQ) >= 0.600 * oELCHa(530INFREQ) 11.._LENGTH  _GOAL(INFREQ293, 120)
oLat(530INFREQ) >= 0.400 * oELCHa(530INFREQ) 11.._LENGTH  _GOAL(INFREQ294, 120)
oOld(540FREQ) >= 0.080 * oELCHa(540FREQ) 11.._LENGTH  _GOAL(FREQ301, 120)
oM2o(540FREQ) >= 0.160 * oELCHa(540FREQ) 11.._LENGTH  _GOAL(FREQ302, 120)
oMto(540FREQ) >= 0.400 * oELCHa(540FREQ) 11.._LENGTH  _GOAL(FREQ303, 120)
```

```
oLat(540FREQ) >= 0.200 * oELCHa(540FREQ) 11.._LENGTH  _GOAL(FREQ304, 120)
oOld(540GAP) >= 0.240 * oELCHa(540GAP) 11.._LENGTH  _GOAL(GAP311, 120)
oM2o(540GAP) >= 0.380 * oELCHa(540GAP) 11.._LENGTH  _GOAL(GAP312, 120)
oMto(540GAP) >= 0.800 * oELCHa(540GAP) 11.._LENGTH  _GOAL(GAP313, 120)
oLat(540GAP) >= 0.700 * oELCHa(540GAP) 11.._LENGTH  _GOAL(GAP314, 120)
oOld(540INFREQ) >= 0.160 * oELCHa(540INFREQ) 11.._LENGTH  _GOAL(INFREQ321, 120)
oM2o(540INFREQ) >= 0.270 * oELCHa(540INFREQ) 11.._LENGTH  _GOAL(INFREQ322, 120)
oMto(540INFREQ) >= 0.600 * oELCHa(540INFREQ) 11.._LENGTH  _GOAL(INFREQ323, 120)
oLat(540INFREQ) >= 0.400 * oELCHa(540INFREQ) 11.._LENGTH  _GOAL(INFREQ324, 120)
oOld(550FREQ) >= 0.080 * oELCHa(550FREQ) 11.._LENGTH  _GOAL(FREQ331, 120)
oM2o(550FREQ) >= 0.160 * oELCHa(550FREQ) 11.._LENGTH  _GOAL(FREQ332, 120)
oMto(550FREQ) >= 0.400 * oELCHa(550FREQ) 11.._LENGTH  _GOAL(FREQ333, 120)
oLat(550FREQ) >= 0.200 * oELCHa(550FREQ) 11.._LENGTH  _GOAL(FREQ334, 120)
oOld(550INFREQ) >= 0.160 * oELCHa(550INFREQ) 11.._LENGTH  _GOAL(INFREQ341, 120)
oM2o(550INFREQ) >= 0.270 * oELCHa(550INFREQ) 11.._LENGTH  _GOAL(INFREQ342, 120)
oMto(550INFREQ) >= 0.600 * oELCHa(550INFREQ) 11.._LENGTH  _GOAL(INFREQ343, 120)
oLat(550INFREQ) >= 0.400 * oELCHa(550INFREQ) 11.._LENGTH  _GOAL(INFREQ344, 120)
oOld(560FREQ) >= 0.080 * oELCHa(560FREQ) 11.._LENGTH  _GOAL(FREQ351, 120)
oM2o(560FREQ) >= 0.160 * oELCHa(560FREQ) 11.._LENGTH  _GOAL(FREQ352, 120)
oMto(560FREQ) >= 0.400 * oELCHa(560FREQ) 11.._LENGTH  _GOAL(FREQ353, 120)
oLat(560FREQ) >= 0.200 * oELCHa(560FREQ) 11.._LENGTH  _GOAL(FREQ354, 120)
oOld(560GAP) >= 0.240 * oELCHa(560GAP) 11.._LENGTH  _GOAL(GAP361, 120)
oM2o(560GAP) >= 0.380 * oELCHa(560GAP) 11.._LENGTH  _GOAL(GAP362, 120)
oMto(560GAP) >= 0.800 * oELCHa(560GAP) 11.._LENGTH  _GOAL(GAP363, 120)
oLat(560GAP) >= 0.700 * oELCHa(560GAP) 11.._LENGTH  _GOAL(GAP364, 120)
oOld(560INFREQ) >= 0.160 * oELCHa(560INFREQ) 11.._LENGTH  _GOAL(INFREQ371, 120)
oM2o(560INFREQ) >= 0.270 * oELCHa(560INFREQ) 11.._LENGTH  _GOAL(INFREQ372, 120)
oMto(560INFREQ) >= 0.600 * oELCHa(560INFREQ) 11.._LENGTH  _GOAL(INFREQ373, 120)
```

```
oLat(560INFREQ) >= 0.400 * oELCHa(560INFREQ) 11.._LENGTH  _GOAL(INFREQ374, 120)
oOld(620FREQ) >= 0.080 * oELCHa(620FREQ) 11.._LENGTH  _GOAL(FREQ381, 120)
oM2o(620FREQ) >= 0.160 * oELCHa(620FREQ) 11.._LENGTH  _GOAL(FREQ382, 120)
oMto(620FREQ) >= 0.400 * oELCHa(620FREQ) 11.._LENGTH  _GOAL(FREQ383, 120)
oLat(620FREQ) >= 0.200 * oELCHa(620FREQ) 11.._LENGTH  _GOAL(FREQ384, 120)
oOld(620GAP) >= 0.240 * oELCHa(620GAP) 11.._LENGTH  _GOAL(GAP391, 120)
oM2o(620GAP) >= 0.380 * oELCHa(620GAP) 11.._LENGTH  _GOAL(GAP392, 120)
oMto(620GAP) >= 0.800 * oELCHa(620GAP) 11.._LENGTH  _GOAL(GAP393, 120)
oLat(620GAP) >= 0.700 * oELCHa(620GAP) 11.._LENGTH  _GOAL(GAP394, 120)
oOld(620INFREQ) >= 0.160 * oELCHa(620INFREQ) 11.._LENGTH  _GOAL(INFREQ401, 120)
oM2o(620INFREQ) >= 0.270 * oELCHa(620INFREQ) 11.._LENGTH  _GOAL(INFREQ402, 120)
oMto(620INFREQ) >= 0.600 * oELCHa(620INFREQ) 11.._LENGTH  _GOAL(INFREQ403, 120)
oLat(620INFREQ) >= 0.400 * oELCHa(620INFREQ) 11.._LENGTH  _GOAL(INFREQ404, 120)
oOld(630FREQ) >= 0.080 * oELCHa(630FREQ) 11.._LENGTH  _GOAL(FREQ411, 120)
oM2o(630FREQ) >= 0.160 * oELCHa(630FREQ) 11.._LENGTH  _GOAL(FREQ412, 120)
oMto(630FREQ) >= 0.400 * oELCHa(630FREQ) 11.._LENGTH  _GOAL(FREQ413, 120)
oLat(630FREQ) >= 0.200 * oELCHa(630FREQ) 11.._LENGTH  _GOAL(FREQ414, 120)
oOld(630GAP) >= 0.240 * oELCHa(630GAP) 11.._LENGTH  _GOAL(GAP421, 120)
oM2o(630GAP) >= 0.380 * oELCHa(630GAP) 11.._LENGTH  _GOAL(GAP422, 120)
oMto(630GAP) >= 0.800 * oELCHa(630GAP) 11.._LENGTH  _GOAL(GAP423, 120)
oLat(630GAP) >= 0.700 * oELCHa(630GAP) 11.._LENGTH  _GOAL(GAP424, 120)
oOld(630INFREQ) >= 0.160 * oELCHa(630INFREQ) 11.._LENGTH  _GOAL(INFREQ431, 120)
oM2o(630INFREQ) >= 0.270 * oELCHa(630INFREQ) 11.._LENGTH  _GOAL(INFREQ432, 120)
oMto(630INFREQ) >= 0.600 * oELCHa(630INFREQ) 11.._LENGTH  _GOAL(INFREQ433, 120)
oLat(630INFREQ) >= 0.400 * oELCHa(630INFREQ) 11.._LENGTH  _GOAL(INFREQ434, 120)
oOld(710GAP) >= 0.240 * oELCHa(710GAP) 11.._LENGTH  _GOAL(GAP441, 120)
oM2o(710GAP) >= 0.380 * oELCHa(710GAP) 11.._LENGTH  _GOAL(GAP442, 120)
oMto(710GAP) >= 0.800 * oELCHa(710GAP) 11.._LENGTH  _GOAL(GAP443, 120)
```

```
oLat(710GAP) >= 0.700 * oELCHa(710GAP) 11.._LENGTH  _GOAL(GAP444, 120)
oOld(720FREQ) >= 0.080 * oELCHa(720FREQ) 11.._LENGTH  _GOAL(FREQ451, 120)
oM2o(720FREQ) >= 0.160 * oELCHa(720FREQ) 11.._LENGTH  _GOAL(FREQ452, 120)
oMto(720FREQ) >= 0.400 * oELCHa(720FREQ) 11.._LENGTH  _GOAL(FREQ453, 120)
oLat(720FREQ) >= 0.200 * oELCHa(720FREQ) 11.._LENGTH  _GOAL(FREQ454, 120)
oOld(720GAP) >= 0.240 * oELCHa(720GAP) 11.._LENGTH  _GOAL(GAP461, 120)
oM2o(720GAP) >= 0.380 * oELCHa(720GAP) 11.._LENGTH  _GOAL(GAP462, 120)
oMto(720GAP) >= 0.800 * oELCHa(720GAP) 11.._LENGTH  _GOAL(GAP463, 120)
oLat(720GAP) >= 0.700 * oELCHa(720GAP) 11.._LENGTH  _GOAL(GAP464, 120)
oOld(720INFREQ) >= 0.160 * oELCHa(720INFREQ) 11.._LENGTH  _GOAL(INFREQ471, 120)
oM2o(720INFREQ) >= 0.270 * oELCHa(720INFREQ) 11.._LENGTH  _GOAL(INFREQ472, 120)
oMto(720INFREQ) >= 0.600 * oELCHa(720INFREQ) 11.._LENGTH  _GOAL(INFREQ473, 120)
oLat(720INFREQ) >= 0.400 * oELCHa(720INFREQ) 11.._LENGTH  _GOAL(INFREQ474, 120)
oOld(740GAP) >= 0.240 * oELCHa(740GAP) 11.._LENGTH  _GOAL(GAP481, 120)
oM2o(740GAP) >= 0.380 * oELCHa(740GAP) 11.._LENGTH  _GOAL(GAP482, 120)
oMto(740GAP) >= 0.800 * oELCHa(740GAP) 11.._LENGTH  _GOAL(GAP483, 120)
oLat(740GAP) >= 0.700 * oELCHa(740GAP) 11.._LENGTH  _GOAL(GAP484, 120)
oOld(780FREQ) >= 0.080 * oELCHa(780FREQ) 11.._LENGTH  _GOAL(FREQ491, 120)
oM2o(780FREQ) >= 0.160 * oELCHa(780FREQ) 11.._LENGTH  _GOAL(FREQ492, 120)
oMto(780FREQ) >= 0.400 * oELCHa(780FREQ) 11.._LENGTH  _GOAL(FREQ493, 120)
oLat(780FREQ) >= 0.200 * oELCHa(780FREQ) 11.._LENGTH  _GOAL(FREQ494, 120)
oOld(780GAP) >= 0.240 * oELCHa(780GAP) 11.._LENGTH  _GOAL(GAP501, 120)
oM2o(780GAP) >= 0.380 * oELCHa(780GAP) 11.._LENGTH  _GOAL(GAP502, 120)
oMto(780GAP) >= 0.800 * oELCHa(780GAP) 11.._LENGTH  _GOAL(GAP503, 120)
oLat(780GAP) >= 0.700 * oELCHa(780GAP) 11.._LENGTH  _GOAL(GAP504, 120)
oOld(780INFREQ) >= 0.160 * oELCHa(780INFREQ) 11.._LENGTH  _GOAL(INFREQ511, 120)
oM2o(780INFREQ) >= 0.270 * oELCHa(780INFREQ) 11.._LENGTH  _GOAL(INFREQ512, 120)
oMto(780INFREQ) >= 0.600 * oELCHa(780INFREQ) 11.._LENGTH  _GOAL(INFREQ513, 120)
```

```
oLat(780INFREQ) >= 0.400 * oELCHa(780INFREQ) 11.._LENGTH  _GOAL(INFREQ514, 120)
oOld(820FREQ) >= 0.080 * oELCHa(820FREQ) 11.._LENGTH  _GOAL(FREQ521, 120)
oM2o(820FREQ) >= 0.160 * oELCHa(820FREQ) 11.._LENGTH  _GOAL(FREQ522, 120)
oMto(820FREQ) >= 0.400 * oELCHa(820FREQ) 11.._LENGTH  _GOAL(FREQ523, 120)
oLat(820FREQ) >= 0.200 * oELCHa(820FREQ) 11.._LENGTH  _GOAL(FREQ524, 120)
oOld(910FREQ) >= 0.080 * oELCHa(910FREQ) 11.._LENGTH  _GOAL(FREQ531, 120)
oM2o(910FREQ) >= 0.160 * oELCHa(910FREQ) 11.._LENGTH  _GOAL(FREQ532, 120)
oMto(910FREQ) >= 0.400 * oELCHa(910FREQ) 11.._LENGTH  _GOAL(FREQ533, 120)
oLat(910FREQ) >= 0.200 * oELCHa(910FREQ) 11.._LENGTH  _GOAL(FREQ534, 120)
oOld(910GAP) >= 0.240 * oELCHa(910GAP) 11.._LENGTH  _GOAL(GAP541, 120)
oM2o(910GAP) >= 0.380 * oELCHa(910GAP) 11.._LENGTH  _GOAL(GAP542, 120)
oMto(910GAP) >= 0.800 * oELCHa(910GAP) 11.._LENGTH  _GOAL(GAP543, 120)
oLat(910GAP) >= 0.700 * oELCHa(910GAP) 11.._LENGTH  _GOAL(GAP544, 120)
oOld(910INFREQ) >= 0.160 * oELCHa(910INFREQ) 11.._LENGTH  _GOAL(INFREQ551, 120)
oM2o(910INFREQ) >= 0.270 * oELCHa(910INFREQ) 11.._LENGTH  _GOAL(INFREQ552, 120)
oMto(910INFREQ) >= 0.600 * oELCHa(910INFREQ) 11.._LENGTH  _GOAL(INFREQ553, 120)
oLat(910INFREQ) >= 0.400 * oELCHa(910INFREQ) 11.._LENGTH  _GOAL(INFREQ554, 120)


{XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX}
{2:FOREST PRODUCT VALUES: INDICATORS}
{--------------------------------------------------------------------}

  ;Fiber Flow:

    _NDY(ohvSFF(_EACH))                1.._LENGTH
    ohvOTF(_EACH) <= 0.25 * ohvt(_EACH) 1.._LENGTH
```

```
   _NDY(oogSFF(_EACH))          12.._LENGTH   ;>Total Spruce-Fir Harvest


  ;  oWatShed(_EACH) <= 0.4 * oWatShedArea(_EACH) 5.._LENGTH



*EXCLUDE
 {Actions}
 ;aCC_LN   1.._LENGTH  ;Clearcut Harvest and Leave for natural
 ; aCC_Old 6.._LENGTH

 aPT   1.._LENGTH  ;Precommercial Thinning in Natural Stands

 ;aCT   1.._LENGTH  ;Commercial Thinning
;aSH   1.._LENGTH  ;Selection Harvest
 ;aOR   1.._LENGTH

 ;rSL   1.._LENGTH  ;Shelterwood Regime
 ;aSL   1.._LENGTH
 ;rBH   1.._LENGTH  ;Buffer Harvest Regime
 ;aBR   1.._LENGTH
 ;aBH   1.._LENGTH

*FORMAT GUROBI
```

Landscape File

```
;THEME
;0    _INDEX(ecod=520)
;1    _INDEX(ecod=520)
;2    _INDEX(ecod=520)

*THEME {1}
COLLAPSE   _INDEX(ecod=530)
 128FREQ    _INDEX(ecod=820)
 345FREQ    _INDEX(ecod=530)
 345GAP     _INDEX(ecod=530)
 345INFREQ _INDEX(ecod=530)
 679FREQ   _INDEX(ecod=720)
 679GAP    _INDEX(ecod=720)
 679INFREQ _INDEX(ecod=720)
 330FREQ   _INDEX(ecod=530)
 330GAP    _INDEX(ecod=530)
 340FREQ   _INDEX(ecod=530)
 340GAP    _INDEX(ecod=530)
 340INFREQ _INDEX(ecod=530)
 350FREQ   _INDEX(ecod=530)
 350GAP    _INDEX(ecod=530)
 350INFREQ _INDEX(ecod=530)
 370FREQ   _INDEX(ecod=530)
 370GAP    _INDEX(ecod=530)
 370INFREQ _INDEX(ecod=530)
```

```
380FREQ   _INDEX(ecod=530)
380GAP    _INDEX(ecod=530)
380INFREQ _INDEX(ecod=530)
410GAP    _INDEX(ecod=530)
410INFREQ _INDEX(ecod=530)
420FREQ   _INDEX(ecod=530)
420GAP    _INDEX(ecod=530)
420INFREQ _INDEX(ecod=530)
430FREQ   _INDEX(ecod=530)
430GAP    _INDEX(ecod=530)
440FREQ   _INDEX(ecod=530)
440GAP    _INDEX(ecod=530)
440INFREQ _INDEX(ecod=530)
450FREQ   _INDEX(ecod=530)
450GAP    _INDEX(ecod=530)
450INFREQ _INDEX(ecod=530)
530FREQ   _INDEX(ecod=530)
530GAP    _INDEX(ecod=530)
530INFREQ _INDEX(ecod=530)
540FREQ   _INDEX(ecod=530)
540GAP    _INDEX(ecod=530)
540INFREQ _INDEX(ecod=530)
550FREQ   _INDEX(ecod=530)
550INFREQ _INDEX(ecod=530)
560FREQ   _INDEX(ecod=530)
560GAP    _INDEX(ecod=530)
560INFREQ _INDEX(ecod=530)
620FREQ   _INDEX(ecod=720)
```

```
 620GAP    _INDEX(ecod=720)
 620INFREQ _INDEX(ecod=720)
 630FREQ   _INDEX(ecod=720)
 630GAP    _INDEX(ecod=720)
 630INFREQ _INDEX(ecod=720)
 710GAP    _INDEX(ecod=720)
 720FREQ   _INDEX(ecod=720)
 720GAP    _INDEX(ecod=720)
 720INFREQ _INDEX(ecod=720)
 740GAP    _INDEX(ecod=720)
 780FREQ   _INDEX(ecod=720)
 780GAP    _INDEX(ecod=720)
 780INFREQ _INDEX(ecod=720)
 820FREQ   _INDEX(ecod=820)
 910FREQ   _INDEX(ecod=720)
 910GAP    _INDEX(ecod=720)
 910INFREQ _INDEX(ecod=720)


*THEME {2 - MANGT UNIT}
COLLAPSE
 CRNLICNPG
 CRNLICNPL
 CRNUNLICE
 CRNUNLICC
 CRNUNLICW
 INDALL
 PRVABW
```

```
 PRVIRV
 PRVNPL
 PRVE
 PRVC
 PRVW

 *AGGREGATE CRNUNLIC
   CRNUNLICE CRNUNLICC CRNUNLICW
 *AGGREGATE CRNLIC
   CRNLICNPG CRNLICNPL
 *AGGREGATE CRNALL
   CRNUNLIC CRNLIC
 *AGGREGATE CRNnoNPG
   CRNUNLIC CRNLICNPL
 *AGGREGATE PRVIND
   PRVABW PRVIRV PRVNPL INDALL
 *AGGREGATE PRVOTH
    PRVE  PRVC  PRVW
 *AGGREGATE PRVALL
    PRVIND PRVOTH
 *AGGREGATE MUALL
    PRVALL CRNALL
 *AGGREGATE noNPGIND
    CRNUNLIC CRNLICNPL PRVOTH

*THEME {3 - WATER BUFFER ZONE}
 COLLAPSE ;THEME COLLAPSED
 WBUF ;20m WATER BUFFER
```

```
 NBUF ;NON-BUFFER

*THEME
COLLAPSE
  1BT
 1DE
 1DF
 1DG
 1DH
 1DJ
 1DK
 1DL
 1DM
 1DN
 1DO
 1DP
 1DQ
 1DR
 1EH
 1EJ
 1EK
 1EL
 1EM
 1EN
 1EO
 X

*THEME {5}
```

COLLAPSE

*THEME {6}

COLLAPSE

330

340

350

370

380

410

420

430

440

450

530

540

550

560

620

630

710

720

740

780

820

910

```
*THEME {7 - FOREST COMMUNITIES}
 ;Forest Communities
 ;-------------------
  ;Hardwood
   HIHw      _INDEX(fc=101) ;Intolerant Hardwood
   HITHw     _INDEX(fc=102) ;Mixed Intolerant/Tolarant Hardwood
   HTHw      _INDEX(fc=103) ;Tolerant Hardwood
  ;Mixedwood
   MIHwHS    _INDEX(fc=201) ;Intolerant Hardwood - Hardwood Leading
   MIHwSH    _INDEX(fc=202) ;Intolerant Hardwood - Softwood Leading
   MTHw      _INDEX(fc=203) ;Tolerant Hardwood
  ;Softwood
   SrSbSDom  _INDEX(fc=301) ;Red/Black Spruce Dominant
   SwSDom    _INDEX(fc=302) ;White/Other Spruce Dominant
   SbFDom    _INDEX(fc=303) ;Balsam Fir Dominant
   SSpbFDom  _INDEX(fc=304) ;Spruce/Fir Dominant
   SPiDom    _INDEX(fc=305) ;Pine Dominant
   SMHePiSp  _INDEX(fc=306) ;Mixed Spruce/Pine/Hemlock

 ;Managed Stand Types
 ;-------------------------
   SrSPL     _INDEX(fc=401)  ;Softwood Plantation: Native Red Spruce
   SbSPL     _INDEX(fc=402) ;Softwood Plantation: Native Black Spruce
   SPiPL     _INDEX(fc=403)  ;Softwood Plantation: Native Pine
   SwSPL     _INDEX(fc=404) ;Softwood Plantation: White Spruce
   SExPL     _INDEX(fc=405) ;Softwood Plantation: Exotic Species: Norway Spruce/xLarch

 *AGGREGATE SWD
```

```
  SrSbSDom SwSDom SbFDom SSpbFDom SPiDom SMHePiSp
  SrSPL SbSPL SPiPL SwSPL SExPL
 *AGGREGATE MWD
  MIHwHS MIHwSH  MTHw
 *AGGREGATE HWD
  HIHw HITHw HTHw

*THEME {8 - CROWN CLOSURE CLASSES}
 B _INDEX(stk=58,cc=2);CrownClosure Between 31-50%    [stk used for plantations]
 C _INDEX(stk=70,cc=3);CrownClosure Between 51-70%    [stk used for plantations]
 D _INDEX(stk=80,cc=4);CrownClosure Between 71-100%   [stk used for plantations]

 *AGGREGATE CD
  C D

*THEME {9 - SITE CLASSES}
;SW LC = 1-3 & HW LC = 1
 3 _INDEX(lc=3,si=11.85,siNPG=11.85,siIND=11.85,siOTH=11.85)
 ;SW LC = 4-4 & HW LC = 1
 4 _INDEX(lc=4,si=13.60,siNPG=11.85,siIND=13.60,siOTH=11.85)
 ;SW LC = 5-5 & HW LC = 2
 5 _INDEX(lc=5,si=15.38,siNPG=11.85,siIND=15.38,siOTH=13.60)
 ;SW LC = 6-6 & HW LC = 2
 6 _INDEX(lc=6,si=17.06,siNPG=13.60,siIND=17.06,siOTH=15.38)
 ;SW LC = > 6 & HW LC = > 2
 7 _INDEX(lc=7,si=18.65,siNPG=15.38,siIND=18.65,siOTH=17.06)

 *AGGREGATE 34
```

```
  3 4
 *AGGREGATE 4567
  4 5 6 7
 *AGGREGATE 567
  5 6 7

*THEME {10 - FOREST STATE INDICATOR}
;NATURAL UNMANAGED STAND - EVENAGED
 NAE     _INDEX(fs=10)
 ;NATURAL UNMANAGED STAND - UNEVENAGED
 NAU     _INDEX(fs=20)
 ;2ND ROTATION UNMANAGED
 NRG     _INDEX(fs=30)
 ;MANAGED STAND - PLANTATION
 PLT     _INDEX(fs=40)
 ;MANAGED STAND - PRECOMMERCIAL THINNING
 PCT     _INDEX(fs=50)
 ;MANAGED STAND - COMMERCIAL THINNING IN NATURAL STANDS
 CTH     _INDEX(fs=61)(NAE OR NRG)
 ;MANAGED STAND - COMMERCIAL THINNING IN COMMERCIALLY THINNED NATURAL STANDS
 CTCTH   _INDEX(fs=62)
 ;MANAGED STAND - COMMERCIAL THINNING IN PRECOMMERCIALLY THINNED STANDS
 CTPCT   _INDEX(fs=63)
 ;MANAGED STAND - COMMERCIAL THINNING IN PREVIOUSLY CT'D AND PCT'd  STANDS
  CTCTPCT _INDEX(fs=64)
  ;MANAGED STAND - COMMERCIAL THINNING IN PLANTATIONS
 CTPLT   _INDEX(fs=65)
 ;MANAGED STAND - COMMERCIAL THINNING IN PREVIOUSLY COMMERCIALLY THINNED PLANTATIONS
```

```
CTCTPLT _INDEX(fs=66)
;MANAGED STAND - SELECTION HARVESTING IN NATURAL EVENAGED STANDS (NAE OR NRG)
SELNE   _INDEX(fs=71)
 ;MANAGED STAND - SELECTION HARVESTING IN NATURAL EVENADED PCT'd STANDS (PCT)
SELNP   _INDEX(fs=72)
;MANAGED STAND - SELECTION HARVESTING IN NATURAL UNEVENAGED STANDS (NAU)
SELNU   _INDEX(fs=73)
;TRACK MANAGED STANDS THAT ESCAPE NORMAL WINDOW
ESC     _INDEX(fs=0)

*AGGREGATE PLTPCT
  PLT PCT
*AGGREGATE CTX ;COMMERCIAL THINNING TYPES
  CTH CTCTH CTPCT CTCTPCT


*AGGREGATE PTall ;COMMERCIAL THINNING TYPES
  PCT CTPCT CTCTPCT
*AGGREGATE PTX ;COMMERCIAL THINNING TYPES
  PTall
*AGGREGATE PLall ;COMMERCIAL THINNING TYPES
  PLT CTPLT CTCTPLT
*AGGREGATE PLX ;COMMERCIAL THINNING TYPES
  PLall

*AGGREGATE CTPLTCTCTPLT
  CTPLT CTCTPLT
*AGGREGATE CTHCTCTH ;CTH & CT'ed CTH
```

```
  CTH CTCTH

*AGGREGATE PCTCTPCT ;PCT & CT'ed PCT

  PCT CTPCT

*AGGREGATE CTPCTCTCTPCT ;CTPCT & CT'ed CTPCT

  CTPCT CTCTPCT

*AGGREGATE CTPCTCTCTPCTSELNP ;CTPCT & CT'ed CTPCT & SH'ed PCTs

  CTPCT CTCTPCT SELNP

*AGGREGATE PCTCTPCTCTCTPCTSELNP ;PCT TYPES

  PCT CTPCT CTCTPCT SELNP

*AGGREGATE PLTCTPLT ;PLANTATIONS & CT'ed PLANTATIONS

  PLT CTPLT

*AGGREGATE PLTCTPLTCTCTPLT ;PLANTATIONS & CT'ed PLT  & CT'ed CT'ed PLT

  PLT CTPLT CTCTPLT

*AGGREGATE NAENRG ;NATURAL EVENAGED STANDS

  NAE NRG

*AGGREGATE NRGPCT ;NATURAL EVENAGED STANDS WITH PRECOMMERCIAL THINNING

  PCT NRG

*AGGREGATE NAENRGPCT ;NATURAL EVENAGED STANDS WITH PRECOMMERCIAL THINNING

  NAE NRG PCT


*AGGREGATE NAENRGNAU ;NATURAL STANDS

  NAE NRG NAU

*AGGREGATE NAT ;NATURAL STANDS

  NAE NRG NAU


*AGGREGATE SEL ;EVENAGED TYPES BASED ON NATURAL GROWTH CURVES

  SELNE SELNP SELNU
```

```
 *AGGREGATE NGU ;TYPES BASED ON NATURAL UNENENAGED GROWTH CURVES
   NAU SELNU
 *AGGREGATE NFG ;TYPES BASED ON NATURAL EVENAGED GROWTH CURVES
   NAE NRG PCT CTH CTCTH CTPCT CTCTPCT SELNE SELNP

 *AGGREGATE NFGminusSEL ;
  NAE NRG PCT CTH CTCTH CTPCT CTCTPCT

 *AGGREGATE FGE ;EVENAGED STANDS GROWING TREES
   NAE NRG PLT PCT CTX
 *AGGREGATE FGU ;UNEVENAGED STANDS GROWING TREES
   NAU SEL
 *AGGREGATE FTG ;ALL STANDS GROWING TREES
   FGE FGU
 *AGGREGATE FTGminusSEL ;ALL STANDS GROWING TREES MINUS THOSE UNDER SEL MANAGEMENT
   FGE NAU
  *AGGREGATE CCElig
  NRG NAU NAE

*THEME {11 - HARVESTING EXCLUSIONS BASED ON RESTRICTION ZONES}
  NOEXCL   ;NO EXCLUSION
  COLLAPSE ;EXCLUSIONS
  NEWPROT  ;12 Percent Committee
 *AGGREGATE NOEXCLSET
   NOEXCL NEWPROT

*THEME {12 - HARVESTING EXCLUSIONS BASED ON INVENTORY ATTRIBUTES}
  NOEXCL   ;NO EXCLUSION
```

```
 COLLAPSE ;EXCLUSIONS
*AGGREGATE NOEXCLSET
  NOEXCL
```

A.5 Model Two Phase 2 Woodstock Files

A.5.1 Prescription Files

All files are identical to the Phase 1 model. With the exception of the optimize and transition files.The transition file is included below, and the only change in the optimize file for the phase two model is aPT is commented out in the exclude section.

Transition File

```
{==================================================================}
{                         /\     Nova Scotia                       }
{                        /  \    Department of Natural Resources  }
{                       /    \   Forestry Division                 }
{ Tel:   (902)-893-5696 /______\  Forest Management Planning       }
{ Fax:   (902)-893-6102    |      Resource Modeling Section        }
{==================================================================}
{ Project     : Crown Lands Strategic Forest Model                 }
{==================================================================}

;CASE aCC_Simp
;  *SOURCE ? ? ? ? ? ? ? ? ? FTGminusSEL ? ?
 ;   *TARGET ? ? ? ? ? ? ? D ? NRG ? ? 100 _AGE 0 _LOCK 1
```

```
;Claercut and Leave for Natural
*CASE  aCC_LN ;Clear cut Transitions ;NewRegenDataBasedTransitionsForCC
  ;Hardwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? 3 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 3 NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? 4 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 4 NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? 5 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 5 NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? 6 NAENRGNAU ? ?
    *TARGET ? ? ? ? ? ? ? D 6 NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? CTX ? ?
    *TARGET ? ? ? ? ? ? ? D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? CTPLTCTCTPLT ? ?
    *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
    *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0
  ;*SOURCE ? ? ? ? ? ? ? ? 3 PLall ? ?
    ;*TARGET ? ? ? ? ? ? ? D 3 PLT ? ? 100 _AGE 0 _LOCK 18
  ;*SOURCE ? ? ? ? ? ? ? ? 4 PLall ? ?
    ;*TARGET ? ? ? ? ? ? ? D 4 PLT ? ? 100 _AGE 0 _LOCK 17
  ;*SOURCE ? ? ? ? ? ? ? ? 5 PLall ? ?
    ;*TARGET ? ? ? ? ? ? ? D 5 PLT ? ? 100 _AGE 0 _LOCK 16
  ;*SOURCE ? ? ? ? ? ? ? ? 6 PLall ? ?
    ;*TARGET ? ? ? ? ? ? ? D 6 PLT ? ? 100 _AGE 0 _LOCK 15
```

```
  ;*SOURCE ? ? ? ? ? ? ? ? 3 FTGminusSEL ? ?
    ;*TARGET ? ? ? ? ? ? ? D 3 NRG ? ? 100 _AGE 0 _LOCK 18
  ;*SOURCE ? ? ? ? ? ? ? ? 4 FTGminusSEL ? ?
    ;*TARGET ? ? ? ? ? ? ? D 4 NRG ? ? 100 _AGE 0 _LOCK 17
  ;*SOURCE ? ? ? ? ? ? ? ? 5 FTGminusSEL ? ?
    ;*TARGET ? ? ? ? ? ? ? D 5 NRG ? ? 100 _AGE 0 _LOCK 16
  ;*SOURCE ? ? ? ? ? ? ? ? 6 FTGminusSEL ? ?
    ;*TARGET ? ? ? ? ? ? ? D 6 NRG ? ? 100 _AGE 0 _LOCK 15
   ;; *SOURCE ? ? ? ? ? ? ? ? ? PLall ? ?
   ;;  *TARGET ? ? ? ? ? ? ? ? ? PLT ? ? 100 _AGE 0



 ; *SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
  ;  *TARGET ? ? ? ? ? ? ? D ? PLT ? ? 100 _AGE 0 _LOCK 9

;Precommercial Thinning
*CASE aPT ;Precommercial Thinning Transitions
 *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
    *TARGET ? ? ? ? ? ? ? ? ? PCT ? ?  100 _LOCK 2

;Commercial Thinning
*CASE aCT ;Commercial Thinning Transitions
  *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTH ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTH ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTH ? ? 100 _AGE 17 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PCT ? ?
```

```
    *TARGET ? ? ? ? ? ? ? ? ? CTPCT ? ? 100 _AGE 10 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTPCT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTPCT ? ? 100 _AGE 15 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PLT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTPLT ? ? 100 _AGE 8 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? CTPLT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? CTCTPLT ? ? 100 _AGE 13 _LOCK 4


;Shelterwood
*CASE aSL
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ? 100 _LOCK 2
;Shelterwood Harvest - Ovberstory Removal
*CASE aOR
  ;Hardwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
    *TARGET ? ? ? ? ? ? ? D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,101)
    ;*TARGET  ? ? ? ? ? ? HIHw  D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,102)
    ;*TARGET  ? ? ? ? ? ? HITHw D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,103)
    ;*TARGET  ? ? ? ? ? ? HTHw  D ? NRG ? ? 100 _AGE 1
  ;;Mixedwood Cover Types
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,201)
    ;*TARGET  ? ? ? ? ? ? MIHwHS D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,202)
```

```
    ;*TARGET  ? ? ? ? ? ? MIHwSH D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,203)
    ;*TARGET  ? ? ? ? ? ? MTHw   D ? NRG ? ? 100 _AGE 1
  ;;Softwood Cover Types
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,301)
    ;*TARGET  ? ? ? ? ? ? SrSbSDom D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,302)
    ;*TARGET  ? ? ? ? ? ? SwSDom   D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,303)
    ;*TARGET  ? ? ? ? ? ? SbFDom   D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,304)
    ;*TARGET  ? ? ? ? ? ? SSpbFDom D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,305)
    ;*TARGET  ? ? ? ? ? ? SPiDom   D ? NRG ? ? 100 _AGE 1
  ;*SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,306)
    ;*TARGET  ? ? ? ? ? ? SMHePiSp D ? NRG ? ? 100 _AGE 1


;Buffer Harvest (30% removal)
*CASE aBH
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ? 100 _LOCK 2
;Buffer Harvest (regenerate action - no harvest volume generated)
*CASE aBR
  ;Hardwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,101)
    *TARGET  ? ? ? ? ? ? HIHw  D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,102)
```

```
    *TARGET  ? ? ? ? ? ? HITHw D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,103)
    *TARGET  ? ? ? ? ? ? HTHw  D ? NRG ? ? 100 _AGE 0
  ;Mixedwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,201)
    *TARGET  ? ? ? ? ? ? MIHwHS D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,202)
    *TARGET  ? ? ? ? ? ? MIHwSH D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,203)
    *TARGET  ? ? ? ? ? ? MTHw   D ? NRG ? ? 100 _AGE 0
  ;Softwood Cover Types
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,301)
    *TARGET  ? ? ? ? ? ? SrSbSDom D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,302)
    *TARGET  ? ? ? ? ? ? SwSDom   D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,303)
    *TARGET  ? ? ? ? ? ? SbFDom   D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,304)
    *TARGET  ? ? ? ? ? ? SSpbFDom D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,305)
    *TARGET  ? ? ? ? ? ? SPiDom   D ? NRG ? ? 100 _AGE 0
  *SOURCE ? ? ? ? ? ? ? ? ? ? ? ? @YLD(yForComm,306)
    *TARGET  ? ? ? ? ? ? SMHePiSp D ? NRG ? ? 100 _AGE 0


;SELECTION HARVEST
*CASE aSH {SELECTION HARVEST};Selection Harvest Transitions
  *SOURCE ? ? ? ? ? ? ? ? ? NAENRG ? ?
```

```
    *TARGET ? ? ? ? ? ? ? ? ? SELNE ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? NAU ? ?
    *TARGET ? ? ? ? ? ? ? ? ? SELNU ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? PCT ? ?
    *TARGET ? ? ? ? ? ? ? ? ? SELNP ? ? 100 _AGE 12 _LOCK 4
  *SOURCE ? ? ? ? ? ? ? ? ? SEL ? ?
    *TARGET ? ? ? ? ? ? ? ? ? ? ? ?     100 _AGE 12 _LOCK 4


{STAND BREAK-UP}
*CASE _DEATH  {What Happens at Lifespan}
  *SOURCE  ? ? ? ? ? ? ? ? ? PLTCTPLTCTCTPLT ? ?
    *TARGET  ? ? ? ? ? ? ? ? ? ESC ? ? 100 _AGE 0  _LOCK 20

  *SOURCE  ? ? ? ? ? ? ? ? ? FTG ? ?
    *TARGET  ? ? ? ? ? ? ? ? ? ? ? ? 100 _AGE 0 _LOCK 20
```

A.6 Penalties

The violations to the ecosystem constraints for the M1 phase 2 scenario 3 model can be found in Table A.6, they sum to 12,267 ha of violation. The violations on the equivalent M2 model were almost exactly 5,000 ha. Weighted at 120 m^3/ha in the objective function these correspond to 1,472,074 and 600,000 m^3 respectively.

Period	Phase 2 Scenario 3 (ha)	Base Industry Model (ha)
11	1586.5	1877.9
12	954.4	382.5
13	671.6	75.741
14	372.3	43.9
15	314.9	44.9
16	82.4	31.2
17	60.1	86.2
18	22.7	46.7
19	4.4	35.5
20	4.6	33.5
21	38.0	33.7
22	5.7	40
23	20.6	14.5
24	59.9	24.4
25	84.8	56.85
26	169.9	91.9
27	356.9	137.5
28	999.6	462.8
29	2417.9	1323.4
30	4039.9	2415.6

Table A.6: Per Period Violations on Model One Phase 2 Scenario 3 and Industry Models

Appendix B

Supplement to Chapter 5

B.1 Model Formulation

B.1.1 Mill Descriptions

Table B.1 shows the per period capacity in cubic meters and minimum demand levels in dollars at each of the mills. The second column displays the products the mill accepts. Refer to Figure **??** to see location of mills.

Mill	Accepts	Capacity (m^3)	Minimum Demand ($)
1	all softwood sawlogs	4,000,000	600,000
2	all softwood pulp	300,000	300,000
3	valuable hardwood sawlogs	5,000,000	250,000
4	all softwood sawlogs	1,000,000	200,000
5	all softwood sawlogs	6,000,000	400,000
6	all softwood pulp	400,000	300,000
7	low-value softwood and all hardwood	1,000,000	0
8	all softwood sawlogs	1,000,000	600,000

Table B.1: Mills, the Wood Types they accept, Capacity and Minimum Demand

B.1.2 Shipping Costs

The following tables, B.2 and B.3 show the cost per cubic meter to ship wood from each timbershed to each mill; and from each mill to each other mill. The costs reflect valuing shipping at $0.07 per kilometer.

Mills	Timbersheds					
	1	2	3	4	5	6
1	8.01	5.14	1.80	6.81	2.50	6.98
2	11.06	6.89	3.36	11.56	6.43	10.91
3	8.53	5.02	0.68	8.20	3.28	7.89
4	1.62	3.48	6.66	6.17	3.94	1.82
5	4.66	3.78	4.43	4.70	1.36	3.35
6	8.75	5.34	1.04	8.11	3.41	8.01
7	4.42	4.13	4.80	4.72	1.35	3.74
8	1.62	3.48	6.66	6.17	3.94	1.82

Table B.2: Per cubic meter transportation costs: Timbersheds to Mills

Mills	Mills							
	1	2	3	4	5	6	7	8
1	0	5.14	1.43	6.43	3.64	1.3	6.43	3.4
2	5.14	0	3.36	9.77	7.77	3.46	9.77	5.7
3	1.43	3.36	0	7.05	4.61	0.36	7.05	4.3
4	6.43	9.77	7.05	0	3.05	7.24	0.5	3.3
5	3.64	7.77	4.61	3.05	0	4.70	3.05	0.4
6	1.3	3.46	0.36	7.24	4.70	0	7.24	4.4
7	6.43	9.77	7.05	0.5	3.05	7.24	0	3.3
8	3.4	5.7	4.3	3.3	0.4	3.3	4.4	0

Table B.3: Per cubic meter transportation costs: Mills to Mills

B.2 Example Analysis

All Δ values are the difference between the relevant model and the base model.

B.2.1 Base Model

Model Name:	Base
Size:	rows - 107,370, cols - 768,747, NZ - 71,967,679
Solution Time (secs)	1723.49
Objective ($)	140,000,000

Table B.4: Base Model Summary

Period	Revenue ($)
1	32,065,370
2	32,461,017
3	32,615,434
4	33,074,475
5	36,994,258

Table B.5: Base Model Revenue

Period	Spruce-Fir (m^3)
1	1,423,314
2	1,423,314
3	1,423,314
4	1,423,314
5	1,423,314
Long-Run Harvest	1,485,596

Table B.6: Base Model Spruce-Fir Harvests

Penalties on the Base Industrial Model are listed in dollar ($) values in Table B.7; Table A.6 contains the number of hectares of violations these penalties are computed from.

Period	Penalty ($)
11	5,633,875
12	1,147,603
13	227,235
14	131,877
15	134,587
16	93,601
17	258,543
18	140,163
19	106,405
20	100,477
21	101,030
22	120,237
23	43,525
24	73,094
25	170,569
26	275,672
27	412,539
28	1,388,365
29	3,970,218
30	7,246,706

Table B.7: Per Period Penalties ($) on Base Model

B.2.2 Clearcut Restriction

Model Name	Clearcut Restriction
Size:	rows - 107,970, cols - 768,747, NZ - 71,198,299
Solution Time (secs)	2912
Objective ($)	140,000,000

Table B.8: Clearcut Restriction: Summary

Period	ΔRevenue ($)
1	-59,409
2	90,767
3	305,880
4	364,638
5	-974,424

Table B.9: Clearcut Restriction: Revenue

Period	ΔSpruce-Fir (m^3)
1	-665.59
2	-665.59
3	-665.59
4	-665.59
5	-665.59
Long-Run Harvest	-5566.40

Table B.10: Clearcut Restriction: Spruce-Fir Harvests

B.2.3 Profit-Based Regulation

Model Name:	Mill Regulation
Size:	rows - 107,386; cols - 768,747; NZ - 71,197,711
Solution Time (secs)	3011.42 seconds
Objective ($)	130,900,000

Table B.11: Mill Regulation: Summary

Period	Revenue ($)	Δ Revenue with Base ($)
1	27,323,202	-4,742,168
2	29,482,522	-2,978,495
3	31,630,368	-985,065
4	32,844,028	-230,446
5	36,867,101	-127,156

Table B.12: Mill Regulation: Revenue

Period	Spruce-Fir (m^3)	ΔSpruce-Fir (m^3)
1	1,366,404	-56,910
2	1,415,342	-7972
3	1,474,931	51,617
4	1,398,595	-24,719
5	1,371,514	-51,799
Long-Run Harvest	1,371,514	-114,081

Table B.13: Mill Regulation: Spruce-Fir Harvests

Model Name:	Mean Regulation
Size:	rows - 107,370; cols - 768,747; NZ - 71,200,115
Solution Time (secs)	1426.70 seconds
Objective ($)	140,500,000

Table B.14: Mean Regulation: Summary

Period	Revenue ($)	Δ Revenue with Base ($)
1	32,278,605	213,234
2	32,466,883	5865
3	32,641,494	26,060
4	32,972,273	102,201
5	37,324,488	330,230

Table B.15: Mean Regulation: Revenue

Period	Spruce-Fir (m^3)	S Δspruce-Fir (m^3)
1	1,476,061	52,746
2	1,440,267	16,952
3	1,416,559	-6755
4	1,415,119	-8195
5	1,387,234	-36,080
Long-Run Harvest	1,312,076	-173,520

Table B.16: Mean Regulation: Spruce-Fir Harvests

B.2.4 Industrial Expansion

Model Name:	Industrial Expansion
Size:	rows - 107,415; cols - 768,787; NZ - 71,197,804
Solution Time	886,58 seconds
Objective	192,500,000

Table B.17: Industrial Expansion: Summary

Period	Revenue ($)	Δ Revenue ($)
1	43,242,479	11,177,109
2	43,516,651	11,055,633
3	43,715,726	11,100,292
4	43,729,955	10,655,479
5	45,845,995	8,851,737

Table B.18: Industrial Expansion: Revenue

Period	Spruce-Fir (m^3)	Δ Spruce-Fir (m^3)
1	1,946,396	523,081
2	1,946,396	523,081
3	1,946,396	523,081
4	1,946,396	523,081
5	1,946,396	523,081
Long-Run Harvest	1,980,359	494,762

Table B.19: Industrial Expansion: Spruce-Fir Harvests

B.2.5 Leaving Wood in the Forest

Model Name:	Leaving Wood
Size:	rows - 109,296; cols - 768,747; NZ - 72,296,671
Solution Time (secs)	879.96 seconds
Objective ($)	150,100,000

Table B.20: Leaving Wood: Summary

Period	Δ Revenue with Base ($)	Revenue ($)
1	32,896,781	831,410
2	34,108,972	1,647,955
3	33,884,744	1,269,310
4	36,331,511	3,257,035
5	40,063,787	3,069,529

Table B.21: Leaving Wood: Revenue