Hierarchical production planning in forestry using price-directed decomposition

Samuel D. Pittman, B. Bruce Bare, and David G. Briggs

Abstract: Forest planning models have increased in size and complexity as planners address a growing array of economic, ecological, and societal issues. Hierarchical production models offer a means of better managing these large and complex models. Hierarchical production planning models decompose large models into a set of smaller linked models. For example, in this paper, a Lagrangian relaxation formulation and a modified Dantzig–Wolfe decomposition – column generation routine are used to solve a hierarchical forest planning model that maximizes the net present value of harvest incomes while recognizing specific geographical units that are subject to harvest flow and green-up constraints. This allows the planning model to consider forest-wide constraints such as harvest flow, as well as address separate subproblems for each contiguous management zone for which detailed spatial plans are computed. The approach taken in this paper is different from past approaches in forest hierarchical planning because we start with a single model and derive a hierarchical model that addresses integer subproblems using Dantzig–Wolfe decomposition. The decomposition approach is demonstrated by analyzing a set of randomly generated planning problems constructed from a large forest and land inventory data set.

Résumé : Comme les planificateurs considèrent un nombre croissant d'enjeux économiques, écologiques et sociaux, les modèles de planification forestière deviennent de plus en plus lourds et complexes. Les modèles hiérarchiques de production offrent une façon d'améliorer la gestion de ces gros modèles complexes. Les modèles hiérarchiques de planification de la production décomposent les gros modèles en un ensemble de modèles plus petits reliés entre eux. Par exemple, dans cet article, la formule de relaxation de Lagrange et une procédure modifiée de décomposition de Dantzig–Wolfe ou de génération de colonnes ont été utilisées pour résoudre un modèle hiérarchique de planification forestière qui maximise la valeur actuelle nette des revenus de récolte tout en considérant la spécificité des unités géographiques sujettes aux flux de récolte et aux contraintes de régénération. Ceci permet au modèle de planification de considérer des contraintes pour l'ensemble de la forêt telles que les flux de récolte et de considérer également des sous-problèmes spécifiques à chacune des zones contiguës d'aménagement pour lesquelles des plans spatialement détaillés sont élaborés. L'approche considérée dans cet article diffère des approches traditionnelles de planification forestière hiérarchique parce qu'elle débute par un modèle unique duquel dérive un modèle hiérarchique qui traite les sous-problèmes en entier par la méthode de décomposition de Dantzig–Wolfe. L'approche de décomposition est illustrée par l'analyse d'un ensemble de problèmes générés aléatoirement à partir d'une vaste forêt et d'un ensemble de données d'inventaire du territoire.

[Traduit par la Rédaction]

Introduction

Forest planning models are formulated to help produce management plans over extremely large and variable geographical areas involving multiple time periods. Forests often consist of many noncontiguous management zones, where each management zone is composed of several stands (units delineated to receive the same silvicultural treatment) that are mapped according to geographic information system (GIS) boundaries. Forest plans address forest-wide concerns such as harvest flow over time, as well as prescribing management treatments for each stand contained within a management zone. We refer to forest-wide constraints as joint constraints, as they link all management zones and stands together to satisfy requirements over the entire forest. The term management zone constraint is used for constraints that place restrictions on how a particular management zone is managed without consideration of other management zones.

Because of the need to consider multiple time and geographical scales, earlier hierarchical production planning models were developed such that management-zone-level problems and forest-level problems were formulated separately. These approaches also include mechanisms designed to achieve consistency among the resulting plans. Weintraub and Bare (1996) provide an extensive review of work defining the hierarchical model used in forest planning. Instead of formulating separate planning models to address different geographical scales, our paper demonstrates that one model can be formulated and then decomposed to realize a hierarchical model. An advantage of this approach is that one can naturally apply a decomposition algorithm, such as

Received 23 February 2006. Accepted 5 February 2007. Published on the NRC Research Press Web site at cjfr.nrc.ca on 2 November 2007.

S.D. Pittman,[1,2] B.B. Bare, and D.G. Briggs. College of Forest Resources, University of Washington, Box 352100, Seattle, WA 98195-2100, USA.

[1]Corresponding author (e-mail: sam.pittman@weyerhaeuser.com).
[2]Present address: Weyerhaeuser Co., WTC 1B20, P.O. Box 9777, Federal Way, WA 98063-9777, USA.

 doi:10.1139/X07-026

Dantzig–Wolfe (Dantzig and Wolfe 1961), to connect the different planning levels.

This paper demonstrates the utility of using price-directed decomposition in a hierarchical production planning model to coordinate the planning of several noncontiguous, spatially defined management zone planning problems linked by a joint harvest volume constraint. We first review the literature of applications of hierarchical production planning in forestry. We then present the planning problem formulation and the decomposition approach. We later discuss the data used to test the formulation and solution procedure and the hardware and software used to solve the problem. We present the results of the algorithm applied to the test data and finally outline our conclusions.

Literature review

The inclusion of spatial considerations in forest planning has resulted in large integer program planning models that are difficult to solve. Forest planners have contributed various algorithms to solve these integer programming problems using methodologies such as TABU search (Boston and Bettinger 1999), simulated annealing (Lockwood and Moore 1993), genetic algorithms (Lu and Eriksson 2000; Falcao and Borges 2001), simulation (Yoshimoto et al. 1994), integer programming (Nelson and Brodie 1990; Snyder and ReVelle 1997; Gunn and Richards 2005), and dynamic programming (Hoganson and Borges 1998). These algorithms address tactical forest planning problems that usually incorporate forest-wide constraints as desirable targets to be satisfied. In contrast, our hierarchical production planning method connects management-zone-level models in a framework that considers all geographical scales of the planning problem simultaneously.

Several methods have been proposed to link various planning models in a hierarchical fashion. A common theme in forest planning models is that strategic decisions derive from aggregate strata-based data (usually at the forest level), while tactical decisions address spatial issues (usually at the subforest level). Further, a framework has been adopted wherein strategic decisions guide tactical planning models. Recognizing the need for an iterative scheme to reconcile the output occurring at different levels, some researchers have proposed feedback and feed-forward mechanisms to connect levels that are considered to be modeling different aspects of the plan (Gunn 1991). The model of Weintraub and Cholaky (1991) utilizes this type of iterative scheme. Subforest or lower-level spatial problems are solved to meet harvest volume targets specified by a forest-level strategic problem. If feasible solutions at the lower level cannot be found, then the upper-level problem is solved again, specifying new harvest volume flow targets until feasible tactical solutions are found to be within some tolerance from the target. Bare and Liermann (1994) present a similar model structure, which is spatially decomposed and utilizes similar aggregation procedures. Hof and Pickens (1987) present a two-tiered model in which several subforest tactical plans are proposed for each management zone. Then the upper-tier problem selects plans for each management zone. Davis and Martell (1993) propose a hierarchical model that addresses a forest-wide constraint based on aggregate stand data. The solution from this strategic-level model is used to identify spatially feasible tactical solutions. The authors indicate that there is no mathematical linkage between the strategic and tactical models; rather, the strategic model is used to guide tactical-level planning. Nelson and Errico (1993) present a descriptive hierarchical process using simulation. They divide the forest into management zones that form spatial subproblems. Feasible spatial alternatives are constructed heuristically using the four-colour theorem, and forest-wide objectives and constraints are indirectly composed of spatial data aggregated from the spatial subproblems. Forest-wide objectives in this approach are satisfied using simulation rather than explicitly solved to optimality. Gunn (1995) proposes the use of stochastic programming with recourse at the tactical level within the hierarchical model. He simplifies the problem by decomposing the tactical model into a hierarchy of models addressing different elements within the plan.

Several authors have applied Dantzig–Wolfe decomposition to forest planning problems. Berck and Bible (1984) demonstrate that the formulations proposed in Johnson and Scheurman (1977) can be solved using Dantzig–Wolfe decomposition. In an example problem, the authors report that the decomposition method took less than half the time taken by the revised simplex method. Hoganson and Rose (1984) also demonstrate decomposition of the forest planning problem. Although their decomposition is similar to Dantzig–Wolfe, their method differs in how the dual variables are updated. The authors use economic interpretations of shadow prices to update the dual variables instead of the Dantzig–Wolfe approach that solves a master problem. Weintraub et al. (1994) propose a decomposition method to solve forest planning problems with joint resource constraints and management-zone-level constraints that include spatial stand adjacency constraints. They derive a solution to the entire planning problem by solving the master problem, whereas we employ a resource price vector derived from the master problem to guide the solution of each subproblem. Paredes (1995) provides an excellent conceptual interpretation of Dantzig–Wolfe decomposition applied to a forest planning problem from an economic viewpoint. His paper draws on many of the early classic publications in economics, such as Kornai and Liptak (1965), Heal (1973), Hurwicz (1973), and Arrow and Hurwicz (1977), which studied economic planning using resource and price allocation mechanisms. Paredes clearly describes the similarities of economic planning and hierarchical forest planning. Although Paredes' work is conceptual, it provides a stepping stone for the work to follow.

Our work, in applying decomposition, differs from previous studies in that the problems considered are of a mixed integer programming (MIP) type, because of spatial integer constraints at the management-zone-level, and our use of the hierarchical model under the interpretation of price-directed decomposition. Vielma et al. (2006) consider a similar problem; however, they address area restriction rather than unit restriction adjacency constraints (Murray 1999). In solving the multiperiod harvest planning problem with both area restriction adjacency and harvest flow constraints, Vielma et al. (2006) find that using harvest flow constraints considerably slows down the problem solution time because of excessive fractioning during the integer programming branch and bound procedure. To overcome this, they apply a penalty approach to the harvest volume constraints, updating the penalties applied to the volume constraints by adding hyperplane cuts to the constraint set during the solution of the integer programming problem. Using this methodology, the authors are able to find close approximate solutions to problems they were unable to solve with strict harvest volume constraints. Their method is similar to our work using decomposition because it updates the objective function during the algorithmic process and penalizes volume constraint violations. However, our method differs in that it uses column generation with a master problem to determine the penalty or price for constraint violation, and it decentralizes decisions regarding management zones.

Dantzig–Wolfe decomposition has been used to solve integer programming problems for determining optimality bounds and in branch and price algorithms (Barnhart et al. 1998; Vanderbeck 2000; Martin 1999). The branch and price method, which computes an optimal integer solution to an MIP problem, requires substantial additional work (algorithmically solving the master problem to an integer solution) beyond what is required for linear and convex programming problems. Our approach, unlike the branch and price method, does not attempt to produce integer variables in the master problem. Instead, we consider the utility of the shadow prices produced from solving the master problem as a linear program, but with the subproblems solved as integer programs.

Formulation of the price-directed decomposition approach

Figure 1 portrays an example of a hypothetical forest to demonstrate the hierarchy of elements within a forest planning model. The forest is modeled as a group of management zones, each consisting of one or more stands. In this paper, a management zone is defined as a collection of geographically grouped stands. Often a management zone will contain several groups of contiguous stands. In Fig. 1, management zone 1 has four stands and management zone 2 has four stands. Note that management zone 2 has two blocks of contiguous stands.

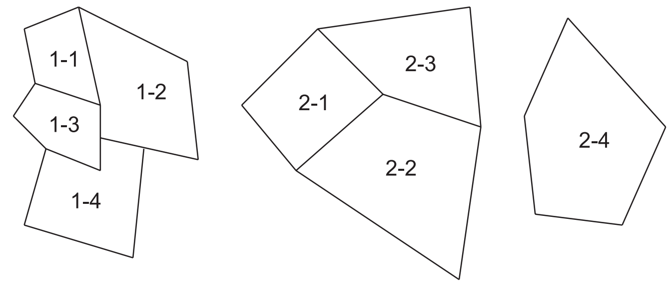


Fig. 1. Sample forest indicating forest structure. Polygons represent stands. Collections of stands are referred to as management zones. *v-i* indicates the *i*th stand of management zone *v*.

A planning model for the forest in Fig. 1 is given in [P1], where the objective is to maximize the net present value of harvested timber income subject to restrictions prohibiting the harvest of adjacent stands in the previous, current, and following planning periods and requiring the harvest flow to be within ±10% between time periods. Note that the adjacency constraints are essentially type I constraints (see Appendix A) (Murray and Church 1995). In utilizing this definition, we define management alternatives as being in conflict on adjacent stands if any harvest activity produces an adjacency violation between the two stands. As a consequence, our model formulation is able to consider adjacency constraints over multiple rotations (see Appendix A). We start with the following formulation:

$$
[\mathrm{P1}]\quad
\begin{aligned}
&\text{maximize} && \sum_{i=1}^{N}\sum_{j=1}^{J_i} c_{ij}x_{ij}\\
&\text{subject to :} && \\
&(1) && 0.9\sum_{i=1}^{N}\sum_{j=1}^{J_i} V_{ijt}x_{ij} - \sum_{i=1}^{N}\sum_{j=1}^{J_i} V_{ij,t+1}x_{ij} \le 0 \quad \forall t < T\\
&(2) && -\sum_{i=1}^{N}\sum_{j=1}^{J_i} V_{ijt}x_{ij} + 0.9\sum_{i=1}^{N}\sum_{j=1}^{J_i} V_{ij,t+1}x_{ij} \le 0 \quad \forall t < T\\
&(3) && \sum_{j=1}^{J_i} x_{ij} \le 1 \quad \forall i\\
&(4) && x_{i'j'}\sum_{i\neq i'}\sum_{j=1}^{J_i} x_{ij}I_{i'j',ij} \le 1 \quad \forall i'j'\\
&(5) && x_{ij} \in \{0,1\} \quad \forall i,j
\end{aligned}
$$

where i indexes the stand; j indexes the jth management alternative for stand i; t indexes the time period from 1 to T in our formulation; T is the number of planning periods in the problem; N is the number of stands in the planning problem (eight stands in the example shown in Fig. 1); c_{ij} is the net present value of stand i under the jth management alternative; J_i is the number of management alternatives for the ith stand; $I_{i'j',ij} = 1$ if stand i' under its j'th alternative produces an adjacency violation with stand i under its jth alternative; $I_{i'j',ij}$ = 0 otherwise; V_{ijt} is the volume harvested from stand i under management alternative j in period t; x_{ij} = 1 if management alternative j is selected for stand i; x_{ij} = 0 otherwise.

Note, that a management alternative is a specification of management treatments by period for a stand. This may include some level of harvesting, thinning, fertilization, or a combination of these activities. Constraints (1) and (2) in [P1] state that harvested volume between successive periods must be within ±10%. Constraint (3) states that each stand must be treated by at most one management alternative. Constraint (4) states that management alternative j on an adjacent stand i may not be selected if it produces a green-up restriction with stand i' under alternative j' if it is selected. Constraint (5) enforces whole selections of management alternatives by constraining the selection to be either 0 or 1.

The hierarchical model and Lagrangian relaxation

As shown in formulation [P1], forest planning models incorporate constraints that consider management actions for all stands within a management zone and (or) across the entire forest (e.g., joint constraints that restrict timber harvest flow across the entire forest) and management zone constraints (e.g., spatial adjacency constraints) that restrict the management of a contiguous group of stands within a management zone. Constraints relating management zones to the entire forest are referred to as joint constraints, while constraints relating stands to a particular management zone (subsystem) are referred to as management zone constraints. For example, in Fig. 1, it can be seen that harvesting a stand in management zone 1 has no effect in terms of adjacency constraint violations to stands in management zone 2, since zone 1 does not border zone 2. However, a timber harvest flow constraint that considers timber harvests from all management zones and stands in the forest is usually required. Using the hierarchical components (forest, management zone, and stand), [P1] is reformulated so that it may be decomposed using the principles of Dantzig–Wolfe decomposition.

Generalizing [P1] to consider an arbitrary number of stands, management zones, and time periods leads to [P2]:

[P2]

$$\text{maximize} \quad \sum_{k=1}^{K}\sum_{i=1}^{N_k}\sum_{j=1}^{J_i} c_{kij}x_{kij}$$

subject to:

$$(1) \qquad (1-\rho)\sum_{k=1}^{K}\sum_{i=1}^{N_k}\sum_{j=1}^{J_i} V_{kijt}x_{kij} - \sum_{k=1}^{K}\sum_{i=1}^{N_k}\sum_{j=1}^{J_i} V_{kij,t+1}x_{kij} \le 0 \quad \forall t \le T$$

$$(2) \qquad -\sum_{k=1}^{K}\sum_{i=1}^{N_k}\sum_{j=1}^{J_i} V_{kijt}x_{kij} + (1-\rho)\sum_{k=1}^{K}\sum_{i=1}^{N_k}\sum_{j=1}^{J_i} V_{kijt+1}x_{kij} \le 0 \quad \forall t \le T$$

$$(3) \qquad x_{ki'j'} + \sum_{i=1, i\neq \bar{i}}^{N_k}\sum_{j=1}^{J_i} x_{kij}I_{ki'j',kij} \le 1 \quad \forall k, i', j'$$

$$(4) \qquad \sum_{j=1}^{J_i} x_{kij} \le 1 \quad \forall k, i$$

$$(5) \qquad x_{kij} \in \{0, 1\}$$

where K is the number of management zones; N_k is the number of stands in management zone k; and ρ is the maximum proportion of harvest flow deviation among periods.

The notation of [P2] is the same as for [P1], except that now kij indicates the ith stand of management zone k under the jth management alternative. Let $\boldsymbol{x}_v$ be the vector that contains all x_{kij} variables in management zone k, now indexed by v. Moreover let $\boldsymbol{c}_v$ be the vector that contains all of the coefficients c_{kij}. This allows the first two constraints in the problem to be represented compactly as [1], where the matrix $\mathbf{A}_v$ contains the coefficients in the first two constraints in [P2], and n is the number of management zones in the problem.

$$[1] \qquad \sum_{v=1}^{n} \mathbf{A}_v \le \boldsymbol{\beta} = 0$$

The last three constraints in [P2] can be written more generally as [2], where the matrix $\mathbf{B}_v$ and resource constraint $\boldsymbol{b}_v$ contain the constraints for management zone v.

$$[2] \qquad \begin{aligned} &\mathbf{B}_v \le \boldsymbol{b}_v \\ &\boldsymbol{x}_v \in \{0, 1\} \end{aligned}$$

This gives rise to [P3]:

[P3]

$$\text{maximize} \quad \sum_{v=1}^{n} \boldsymbol{c}_v\boldsymbol{x}_v$$

subject to :

$$(1) \qquad \sum_{v=1}^{n} \mathbf{A}_v\boldsymbol{x}_v \le \boldsymbol{\beta}$$

$$(2) \qquad \mathbf{B}_v\boldsymbol{x}_v \le \boldsymbol{b}_v \qquad v = 1, 2, ..., n$$

$$(3) \qquad \boldsymbol{x}_v \in \{0, 1\}$$

Lagrangian relaxation is applied to this formulation to obtain [P4], where constraints (1) are moved to the objective function with price $\boldsymbol{y}$ paid for their violation. The motivation for this is to obtain a decentralized formulation of [P3].

$$[\text{P4}]\quad \begin{array}{ll} \text{maximize} & \sum_{vi=1}^{n}\boldsymbol{c}_v\boldsymbol{x}_v + \boldsymbol{y}(\boldsymbol{\beta} - \sum_{v=1}^{n}\mathbf{A}_v\boldsymbol{x}_v) = \sum_{v=1}^{n}[\boldsymbol{c}_v\boldsymbol{x}_v - \boldsymbol{y}\,\mathbf{A}_v\boldsymbol{x}_v] - \boldsymbol{y}\,\boldsymbol{\beta} \\ \text{subject to :} & \\ (2) \quad \mathbf{B}_v\boldsymbol{x}_v \le \boldsymbol{b}_v & v = 1, 2, ..., n \\ (3) \quad \boldsymbol{x}_v \in \{0, 1\} & \end{array}$$

A slight amount of algebraic manipulation transforms [P4] into [P5], where the constant $\boldsymbol{y\beta}$ term in [P4] is dropped, since it does not effect the optimization.

$$[\text{P5}]\quad \begin{array}{lll} \text{maximize} & \sum_{v=1}^{n}(\boldsymbol{c}_v\boldsymbol{x}_v + \boldsymbol{y}\mathbf{A}_v\boldsymbol{x}_v) & \\ \text{subject to} & & \\ (2) & \mathbf{B}_v\boldsymbol{x}_v \le \boldsymbol{b}_v & v = 1, 2, ..., n \\ (3) & \boldsymbol{x}_v \in \{0, 1\} & \end{array}$$

In general, $\boldsymbol{y}$ represents the prices paid for joint, scarce resources consumed by the individual management zones. In formulation [P5], $\boldsymbol{y}$ represents the cost (penalty) of timber harvest flow restrictions. For a fixed $\boldsymbol{y}$, [P5] can be solved by simply solving (n) smaller optimization problems of the form [P6] and a master problem that guides the process.

$$[\text{P6}]\quad \begin{array}{ll} \text{maximize} & \boldsymbol{c}_v\boldsymbol{x}_v + \boldsymbol{y}\mathbf{A}_v\boldsymbol{x}_v \\ \text{subject to :} & \\ (2) & \mathbf{B}_v\boldsymbol{x}_v \le \boldsymbol{b}_v \\ (3) & \boldsymbol{x}_v \in \{0, 1\} \end{array}$$

Next, we discuss the decomposition algorithm that allows each management zone, given the appropriate price of joint resources (e.g., penalty costs for timber harvest flow violation), to be solved separately.

Dantzig–Wolfe decomposition and column generation

Column generation (see Dantzig 1963) provides a mechanism for iterative resource price determination in response to plans generated for each separate management zone. The prices are determined by solving a master problem (see Dantzig 1963, p. 448). Example problem [E1], with two management zones, is used to demonstrate the approach before moving to a general formulation. [E1] represents an instance of [P3] with two management zones.

$$[\text{E1}]\quad \begin{array}{lll} \text{minimize} & \boldsymbol{c}_1\boldsymbol{x}_1 + \boldsymbol{c}_2\boldsymbol{x}_2 & \\ \text{subject to :} & & \\ (1) & \mathbf{A}_1\boldsymbol{x}_1 + \mathbf{A}_2\boldsymbol{x}_2 \le \boldsymbol{\beta} & \\ (2) & \mathbf{B}_1\boldsymbol{x}_1 \le \boldsymbol{b}_1 & \\ (3) & \mathbf{B}_2\boldsymbol{x}_2 \le \boldsymbol{b}_2 & \\ (4) & \boldsymbol{x}_1 \in \{0, 1\}, & \boldsymbol{x}_2 \in \{0, 1\} \end{array}$$

In initiating the solution, a resource price of $\boldsymbol{y} = 0$ is used in formulation [P6], indicating that the harvest flow constraints are being ignored during the first iteration. Using the solution to [P6] for each management zone, we formulate the following linear program (i.e., the master problem):

$$[\text{E2}]\quad \begin{array}{lll} \text{maximize} & (\boldsymbol{c}_1\boldsymbol{x}_1^0)z_1^0 + (\boldsymbol{c}_2\boldsymbol{x}_2)z_2^0 & \\ \text{subject to :} & & \\ (1) & (\mathbf{A}_1\boldsymbol{x}_1^0)z_1^0 + (\mathbf{A}_2\boldsymbol{x}_2^0)z_2^0 \le \boldsymbol{\beta} & \\ (2.1) & z_1^0 \le 1 & \\ (2.2) & z_2^0 \le 1 & \\ (3) & z_v^h \le 0 & \forall v, h \end{array}$$

where $\boldsymbol{x}_1^0$ is the solution for management zone 1 at iteration 0, which is generated by solving [P6] with resource prices set to 0. In general, $\boldsymbol{x}_v^h$ is the solution to [P6] for management zone v at the hth iteration of the algorithm. z_1^0 is a solution of the master problem for management zone 1 at the 0th iteration (the first time the master problem is solved). z_1^0 indicates the proportion of plan 0 for management zone 1 that is used in the solution of the master problem. In general, z_v^h indicates the proportion of the hth plan submitted for management zone v that is used in the solution of the master problem. The master problem maximizes the value of the entire forest planning problem by selecting the level of use from the various plans generated from solving the subproblems. Note that the joint constraints (1) in this problem ensure that the joint constraints in the original problem are satisfied, while the constraints specify that a convex combination of the plans for each management zone does not exceed unity. Upon solving this problem, a Lagrange multiplier, $\boldsymbol{y}^1$, for constraint (1) is determined. The Lagrange multiplier gives the rate of change in the objective function with respect to the rate of change in the right-hand side of constraint (1). In the sample problem, it may be interpreted as the cost to the objective function per unit of even harvest flow. Next, $\boldsymbol{y}^1$ is passed back to each of the subproblems so that they may be solved again using this new penalty (Lagrange multiplier) for the joint constraints. [P6] is solved again for each management zone. This determines a new proposed plan for each management zone that gets entered into the master problem as follows:

$$
\text{[E3]}\quad
\begin{array}{ll}
\text{maximize} & (c_1x_1^0)z_1^0 + (c_1x_1^1)z_1^1 + (c_2x_2^0)z_2^0 + (c_2x_2^1)z_2^1 \\
\text{subject to :} & \\
(1) & (\mathbf{A}_1x_1^0) + (\mathbf{A}_1x_1^1)z_1^1 + (\mathbf{A}_2x_2^0)z_2^0 + (\mathbf{A}_2x_2^1)z_2^1 \le \boldsymbol{\beta} \\
(2.1) & z_1^0 + z_1^1 \le 1 \\
(2.2) & z_2^0 + z_2^1 \le 1 \\
(3) & z_v^h \ge 0 \qquad \forall v, h
\end{array}
$$

Each time the master problem is solved with the new proposed plans submitted for each subproblem, it determines a convex combination of the plans for each subproblem by choosing z_v^h, so that it maximizes the value of the plan for the entire forest and adheres to the harvest flow constraints. This procedure of generating columns to enter the master problem is continued until the algorithm converges.

Generalizing to n management zones, the master problem at the hth iteration is described by formulation [P7]. Note that instead of subscripting $\boldsymbol{x}$ with the iteration index h, we have denoted its dependence on the price $\boldsymbol{y}^h$, which is indexed by h.

$$
\text{[P7]}\quad
\begin{array}{lll}
\text{maximize} & \sum_{v=1}^{n}\sum_{h=0}^{m-1} \boldsymbol{c}_v\boldsymbol{x}_v(\boldsymbol{y}^h)\, z_v^h & \\
\text{subject to :} & & \\
(1) & \sum_{v=1}^{n}\sum_{h=0}^{m-1} \mathbf{A}_v\boldsymbol{x}_v(\boldsymbol{y}^h) z_v^h \le \boldsymbol{\beta} & \\
(2) & \sum_{h=0}^{m-1} z_v^h \le 1 & v = 1, 2, ..., n \\
(3) & z_v^h \ge 0 & \forall v, h
\end{array}
$$

Each column of the master problem represents an extreme point of the polyhedron of the corresponding subproblem for management zone v. The master problem finds a convex combination of the extreme points for each subproblem generated up to iteration $m - 1$ to produce a solution, which is likely not feasible, to the original problems [P2] and [P3]. Each time the master problem is solved, its solution produces vector $\boldsymbol{y}^h$, with a length equal to the length of $\boldsymbol{\beta}$, which approximates the dual variables of constraint (1) in [P3]. This Lagrange multiplier ($\boldsymbol{y}^h$) is used in each management zone subproblem to solve [P6]. Note also that the z variables are the variables in the master problem [P7] and that the fixed $\boldsymbol{x}_v$ variables determined at the hth iteration of the solution of [P6] are dependent upon the hth approximation of the resource prices, as denoted in [P7]. In actuality, the master problem does not directly need $\boldsymbol{x}_v$. It only requires the aggregate objective coefficients $\boldsymbol{c}_v\boldsymbol{x}_v$ (the dot product) and the columns $\mathbf{A}_v\boldsymbol{x}_v$ (matrix product) if there is a change in resource use by the vth subsystem to determine the next approximation of the resource prices. Most mathematical programming software allows the dual variables to be retrieved; hence, $\boldsymbol{y}$ can be obtained by solving [P7] directly, as done in this study, as opposed to solving its dual formulation.

Each time the master problem is solved, all subproblems are solved with the new penalty or price. After subproblem v is solved, at the hth iteration, its optimal value α_v^h is compared with σ_v^h, the Lagrange multiplier of the convexity constraint (constraint (2) of [P7]) for subproblem v in the master problem at the hth iteration.

$$\text{[3]}\quad \alpha_v^h - \sigma_v^h < 0$$

If [3] holds, then the generated column for subproblem v, which represents an extreme point of the subproblem's feasible region, is added to the master problem. Condition [3] indicates that a cost-reducing column has been found that needs to be added to the basis of the master problem. When no more cost-reducing columns can be added to the basis, [3] will not be satisfied for any subproblem. The algorithm should be allowed to continue making iterations between solving the master problem and then solving subproblems until it is determined that no more cost-reducing columns exist to add to the master problem.

It is possible for the Dantzig–Wolfe decomposition algorithm to quickly produce solutions that are close to optimal, but then make only infinitesimal improvements as it nears convergence in successive iterations. For this reason, we employ condition [C1] to determine if the difference between the optimal value of the master problem and the summation over the objective function values of all subproblems is less than a prespecified tolerance, ε.

$$\text{[C1]}\quad \left|\sum_{v=1}^{n}\alpha_v^h - \alpha_h\right| \le \varepsilon$$

where ε is a small number (in this paper ε = 0.000 000 01), and α^h is the optimal value of the master problem at iteration h.

The summation over the objective function values of all subproblems provides an upper bound to the optimal value of the master problem, while the master problem provides an upper bound to the optimal value of the original problems [P2] and [P3]. Hence, when [C1] is met, the current (iteration h) master problem objective function value is within ε of the true master problem optimum value; therefore, when [C1] is met we consider the Dantzig–Wolfe routine converged. Alternatively, the algorithm can be terminated when an iteration is reached for which there are no cost-reducing columns identified for every subproblem. The convergence of the Dantzig–Wolfe algorithm applied in this study will be discussed further with the results.

Upon termination of the algorithm, if $\boldsymbol{x}$ were a fractional variable instead of integer, the convex combination:

$$\text{[4]}\quad \boldsymbol{x}_v = \sum_{h=0}^{m} z_v^h \boldsymbol{x}_v(\boldsymbol{y}^h)$$

would yield an optimal solution to the original problems

[P2] and [P3]. However, since the convex combination [4] is not guaranteed to be a binary (0–1, integer) vector, since z is not constrained to be integer, this solution, in general, does not solve [P5] and therefore also not [P2] and [P3], since [P2], [P3], and [P5] require $\boldsymbol{x}$ to be a binary (0–1, integer) vector. However, solving the problem using this algorithm supplies two important pieces of information without ever having to solve the original problem, which in many cases is not possible because of computational complexity: (1) a tight upper bound on the optimal value of the original problems [P2] and [P3] and (2) resource prices that can be used to obtain close approximate solutions of the original problems [P2] and [P3]. Since the subproblems [P6] are solved with the requirement of integrality, the optimal value of [P7] provides a tighter bound on the optimal value of [P3] than that obtained by simply relaxing the integrality constraints (3) in [P3] (see Martin 1999). Thus, a tighter upper bound (the optimal value of the master problem) is obtained for the optimal value of [P2] and [P3] than would be produced by solving [P2] and [P3] as a linear program (see Martin 1999). The bound for the actual objective function of [P2] and [P3] measures an approximate solution's verity. The optimal value of [P2] and [P3] will be less than the objective function value of [P2] and [P3] using the relaxed solution determined by [4], which will be less than the optimal value of [P2] and [P3] solved as a pure linear program with the integer constraints relaxed. This paper makes use of the tighter upper bound provided by the master problem's optimal value by comparing the objective function value of approximate solutions to the optimal value of the master problem.

This paper considers the solutions of each subproblem (management zone plan computation), solved separately using the approximate prices (those produced at convergence of [P7]), as a candidate solution to [P2] and [P3]. Hence, a convex combination of the solutions produced in the master problem is used for obtaining a bound and for obtaining a harvest flow penalty price, but is not used directly for determining the approximate price solution to the planning problem [P2] and [P3]. Instead, we consider the solution to each subproblem solved using the converged upon approximate resource prices. The solutions for each subproblem obtained from using the converged prices, considered together, provide a solution for the entire forest. In general, owing to the integer requirements, this solution does not match the solution or optimal value of [P2] and [P3]. It also does not match the optimal value obtained from solving [P7], because [P7] takes a convex combination of plans (see [4]) solved under various resource prices computed before and up to convergence of the algorithm. The forest-wide solution, constructed from solving [P6] for each management zone using the final converged upon approximate price vector, is denoted as the price-directed decomposition solution. This solution may not be optimal or feasible. The results of this paper inspect the deviation of the approximate price solution from the actual solution to [P2] and [P3] by using the bound provided in [P7] and by inspecting the violation of the relaxed, joint constraints in [P5].

Price-directed decomposition case study

Data description

To evaluate the utility of the price-directed decomposition approach, randomly assembled planning problems were generated from an actual forest data set. Forest inventory data containing spatial and volume forecast information were obtained from an anonymous source. These stands were organized into 237 management zones, ensuring that adjacent stands were grouped in the same management zone. A management zone, on average, contains 36 stands, and each stand contains an average of 5.45 management alternatives. A planning problem was formed by randomly selecting between 6 and 20 (the number selected with equal probability) management zones without replacement from the population of 237. This procedure was followed to produce 75 unique planning problems. Because of the large number of planning problems involved, any block of stands with more than 150 contiguous stands was omitted from the population. The average number of stands per planning problem was 485.87. The largest planning problem had 806 stands and the smallest had 187 stands.

Each planning problem consisted of eight 5-year planning periods, giving a 40-year planning horizon. The allowable harvest volume deviation among consecutive periods was constrained to be within ±10%. Adjacent stands were precluded from harvest within the same 5-year period and one period before and after harvest. The objective was to maximize the net present value of the timber harvest income using an 8% discount rate. Note that including additional rotations does not change the inherent block angular structure of the problem or the algorithm presented; however, additional rotations do increase the size of the planning problem, particularly the number of columns.

Solution procedure

Each of the 75 planning problems gives an instance of [P2] and [P3]. Each was decomposed and solved according to the methodology discussed, using CPLEX 8.1 to solve the subproblems and the master problem. CPLEX was run under its default parameters.

Choosing $\boldsymbol{x}_v = 0$ for all management zones v provides an initial feasible solution to the planning problem formulations [P2], [P3], and [P5]. This corresponds to a plan in which no management is done in the master problem. This means that the columns of the master problem and the objective coefficients are all zero except for the convexity constraints (see [P7]). This translates to an initial resource price of zero ($\boldsymbol{y}$ = 0), because the marginal change in the objective function with respect to $\boldsymbol{\beta}$, the right-hand side of constraint (1) in [P7], is zero. Hence, the subproblems are initially solved using a resource price of $\boldsymbol{y} = 0$. The algorithm proceeds as the example considered in [E1]–[E3].

Results

The results derived from solving the 75 planning problems are used to evaluate the utility of the decomposition procedure. This includes an analysis of the convergence of the decomposition procedure, the optimality of the plans computed relative to the upper bound discussed, the violation of timber harvest flow restrictions due to their relaxation, and the resulting solution times. Figure 2 presents the size distribution of the 75 randomly generated planning problems in terms of the number of stands considered.

Fig. 2. The number of stands in the solved planning problems.

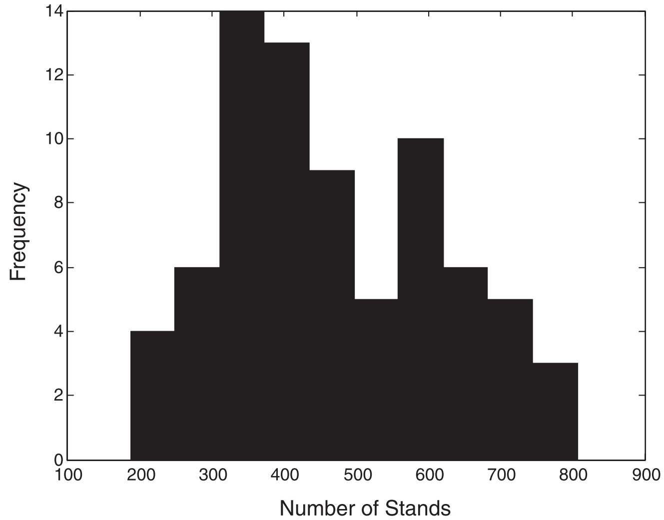


Fig. 3. Solution time for each randomly generated problem plotted against the number of stands in the problem.

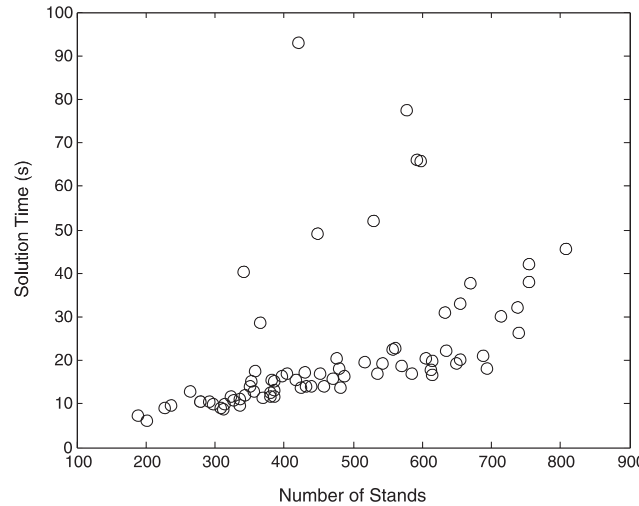
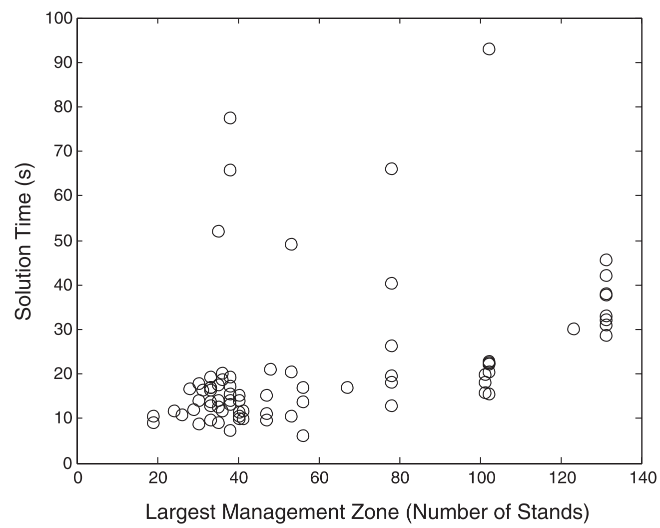


Fig. 4. Largest contiguous block of stands in a management zone plotted against solution time.



Solution effort: number of iterations and computing time

Each planning problem follows formulation [P2] and is solved using the price-directed decomposition algorithm; thus, models [P2]–[P7] are all relevant descriptions of the formulation and solution process. In using this algorithm, the maximum number of planning iterations (the number of times the master problem is solved) was set to equal 85. The planning iterations are indexed by h in [P7], and m indicates the number of iterations used. Generally, far fewer planning iterations than 85 are required because of convergence of the algorithm (condition [C1] is met with ε = 0.000 000 01). After a maximum of 85 iterations, the algorithm terminates with the approximate prices and the corresponding solutions for each management zone. Seven times out of the 75 problems solved, 85 iterations were reached prior to convergence. On average, however, 29.7 iterations were required, including the seven problems that did not converge by 85 iterations. Excluding the problems that did not converge, only 24 iterations on average were required. It is possible that including an additional stopping criterion would reduce the number of iterations in those problems requiring 85 iterations. As previously mentioned, imposing condition [C1] with an additional condition (a planning iteration without any identified reduced cost columns) may further reduce the number of iterations required in the nonconverging problems.

If this iterative pricing approach is used to help guide the actual planning process as described in Dantzig (1963, p. 455), an average of 29.7 iterations may pose a challenge from an implementation viewpoint. However, one might mitigate this challenge by first determining an approximate resource price before integrating this type of iterative solution process in the planning cycles. At any point during the solution process, resource prices can be communicated to zone managers so that they can build a plan that is completely independent of the other zones in the planning problem. Highly complex spatial decisions must be made at the management zone level, making the use of one large planning model ineffective. Thus, a method that allows spatial plans at the management zone level to be computed in a way that preserves near optimality of the plan across all management zones (i.e., the entire forest plan) seems beneficial.

A computing machine with a 756 MH processor and 384 MB of RAM was used for all computations. Figure 3 shows the solution times plotted against problem size, measured by the number of stands. The mean solution time is 22 s. The longer solution times can be partially explained by examining the number of stands in the planning problem and by the largest block of contiguous stands in a management zone. Figure 4 shows the trend between the largest block of contiguous stands in a problem plotted against the solution times. Problems with large contiguous blocks of stands take longer to solve, an expected outcome, since solution times are expected to grow exponentially for nonpolynomial hard integer programming problems. Using decomposition reduces the size of the integer programming problems and, therefore, the computational time required.

Deviation from optimality and constraint violations

Resource prices carry information that allows planning to

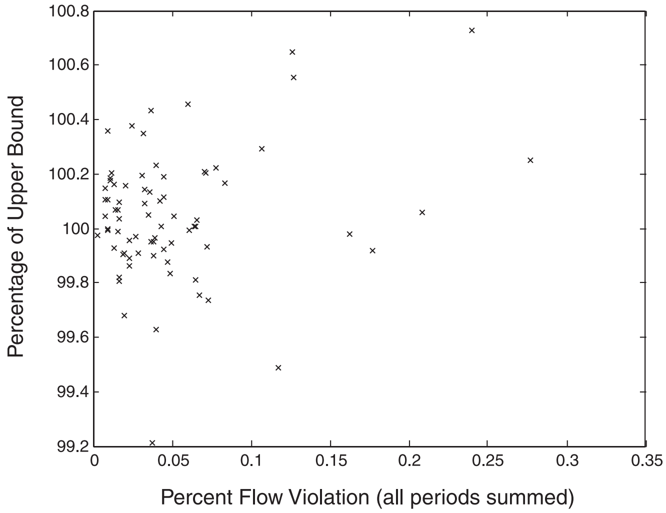


Fig. 5. Percentage of the upper bound (defined as the price-directed approach optimal value divided by the optimal value of [P7]) attained, plotted against harvest flow violation.

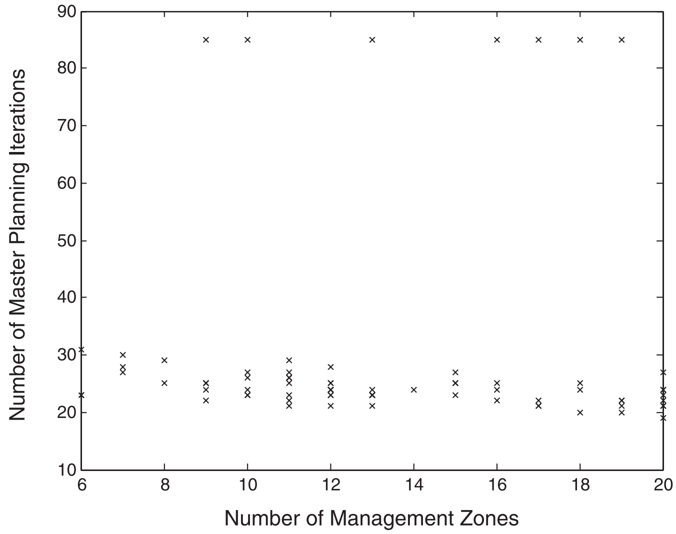


Fig. 6. Number of planning iterations plotted against the number of management zones for each of the 75 sample planning problems solved.

be decentralized. However, owing to the spatial adjacency constraints, the prices yield solutions that do not exactly meet the original formulation specifications [P2] and [P3]. The optimal value found by using the price-directed decomposition algorithm may be quite different from that of [P2], and the fulfillment of the timber harvest flow constraints may not be met, although the spatial restrictions will be preserved. [P5] reflects the multiobjective nature of using Lagrangian relaxation where harvest flow violation is weighed in the objective function against the net present value of timber income. Indeed, the optimal value of the master problem is an upper bound to [P2] (Martin 1999). Moreover, for each iteration h in the algorithm, the solution computed for each management zone can be implemented as a plan that preserves all of the restrictions of [P2] and [P3], except the joint timber harvest flow constraints. A measure of the plan's utility relative to the desired solution of [P2] and [P3] can be depicted in terms of its optimal value relative to the upper bound of the optimal value of [P2] and [P3] (the optimal value of the master problem) while considering the violation of timber harvest flow constraints. Using the optimal value of the master problem as an upper bound on the optimal value of [P2] and [P3] for each of the planning problems solved, Fig. 5 displays the percentage of the upper bound attained using the price-directed decomposition solution plotted against the percentage of timber harvest flow violation over all time periods. The percent timber harvest flow violation is defined as the summation of violation for all harvest flow constraints divided by the total volume scheduled for harvest. If these constraints are violated when using the price-directed decomposition solution, the total harvest volume required to make the solution feasible is the sum of the absolute values of the amount of harvest volume that must be added to, or subtracted from, the right-hand side of the timber harvest flow constraints. The mean percent difference between the master problem's optimal value and the objective function value of the decomposition algorithm is 0.047%, with standard deviation 0.2%. The mean percent timber harvest flow violation is 0.05%, with a standard deviation of 0.05%. Although the timber harvest violations are consistently small, they may still be unacceptable. If timber harvest flow constraints are extremely rigid — for example, if timber harvest flow must not deviate more than ±10% among periods — then a method is needed to eliminate such violations. One option would be to manually adjust the harvest plan without rerunning the model. Another option would be to rerun the model using a value slightly less than ±10%, like 9.99%, so that the calculated flow is within ±10% when the solution is found. In terms of optimality and joint constraint violation, the price-directed approach that allows the management zone planning problems to be solved separately supplies solutions that are extremely close to the optimal solution without substantially violating any constraints. In measuring this aspect against the objective, it appears that the algorithm can produce solutions that satisfy the original problem formulation [P2] to within 0.047% of optimality and to within 0.05% (on average) of timber harvest flow violation. The small standard deviation associated with the results shown in Fig. 5 indicates that the procedure is consistent.

Relationship between planning iterations and number of management zones considered

Figure 6 shows the relationship between the number of iterations required by the master problem in each of the 75 sample problems plotted against the number of management zones in each problem. The graph indicates that there is no increasing trend in the number of planning iterations required as the number of management zones increases. The number of planning iterations required for most of the problems is between 20 and 30. Figure 6 shows that the number of planning iterations appears to decrease slightly with the number of management zones considered. Also shown in Fig. 6 are the problems that did not converge within the specified tolerance required to stop the algorithm. These problems were stopped when the algorithm reached 85 iterations. Figure 5 shows that although condition [C1] was not met, good solutions (within 0.8% of optimality) were always found. Although not presented here, there was also no relationship found between the size of the management zones and the number of planning iterations required.

Discussion and conclusion

To formulate and solve the hierarchical forest production planning problem, a resource-pricing approach derived from Lagrangian relaxation, referred to as price-directed decomposition, was used. This allowed the resulting planning problem to be solved using column generation, also known as Dantzig–Wolfe decomposition. To investigate the impact of this method in terms of objectives being met and constraints being violated, problems were randomly generated from real forest inventory data. The 75 test problems allowed the price-directed algorithm to be analyzed over several problems as opposed to a single problem. This larger set of test problems shows trends that would be otherwise unobservable. The results suggest that there is little sacrifice in joint constraint violation and that objectives are achieved using the proposed method.

We view price-directed decomposition as a refinement to the hierarchical planning model and having major benefits. It starts with a single model formulation and applies Dantzig–Wolfe decomposition to achieve a decentralized interpretation and solution procedure. Providing a formulation of the planning problem in absence of the solution procedure is a noteworthy difference from past methods that have addressed the MIP hierarchical forest planning problem.

Computational complexity and centralized planning have been discussed as two of the major drawbacks of FORPLAN, a planning model developed by the USDA Forest Service capable of solving large-scale forest planning problems (see Bare and Field 1986; Iverson and Alston 1986; Johnson 1986). Following FORPLAN, there was an emphasis on computing feasible plans at the management zone level and then aggregating these plans to provide a forest-level plan (i.e., a bottom-up approach). While this approach improves tractability and implementation at the management zone level, it can result in suboptimal achievement of forest-wide (strategic) objectives. Because of the rule-based heuristics that must be applied at the management zone level, the objective function often suffers when managers attempt to meet joint constraints in a bottom-up approach. On the other hand, a top-down method to solve a strategic linear program that provides goals or production targets for management zone level problems may suboptimize the overall forest plan because of insufficient information about individual management zones. The recognition that both bottom-up and top-down approaches specify solution procedures that operate on underspecified information, leading to suboptimal plans, has prompted the consideration of feedback and feed-forward mechanisms (Gunn 1991) and the hierarchical approach used herein.

This paper has developed and implemented an iterative approach that uses shadow prices on joint constraints to convey forest-level information to each management zone level and aggregated management-zone-level solutions under the existing shadow prices to convey information to the strategic-level problem. The cost of this approach, if carried out organizationally as opposed to computationally, is that plans have to be computed multiple times. The trade-off between improving the solution and the cost of planning can be assessed to indicate when to terminate the feedback and feed-forward iterations, as suggested in Burton and Obel (1977). Although previous research primarily focused on spatial planning, which can be viewed as tactical, there is a need to translate management zone level activity to forest-level objectives. The approach presented here creates the information link between forest-level planning and management zone level planning.

Acknowledgements

This research was conducted while Sam Pittman was a Ph.D. candidate at the University of Washington. Partial funding was provided by the Precision Forestry Cooperative and the Rachel A. Woods Professorship. Moreover, the authors acknowledge Professors Bruce Lippke, Terry Rockafellar, and Paul Tseng for many insightful discussions.

References

Arrow, K.J., and Hurwicz, L. 1977. Studies in resource allocation processes. Cambridge University Press, Cambridge, N.Y.

Bare, B.B., and Field, R. 1986. An evaluation of FORPLAN from an operations research perspective. Presented at the Symposium FORPLAN: An Evaluation of a Forest-Planning Tool. USDA For. Serv. GTR RM-140.

Bare, B.B., and Liermann, M. 1994. Incorporating spatial relationships in forest planning models: the development of a modeling strategy. Final Report. USDA Forest Service, Pacific Northwest Forest Research Station. Portland, Ore.

Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., and Vance, P.H. 1998. Branch and price: column generation for solving huge integer programs. Oper. Res. **46**: 316–329.

Berck, P., and Bible, T. 1984. Solving large-scale harvest scheduling problems by duality and decomposition. For. Sci. **30**: 173–182.

Boston, K., and Bettinger, P. 1999. An analysis of Monte Carlo integer programming, simulated annealing, and tabu search heuristics for solving spatial harvest scheduling problems. For. Sci. **45**: 292–301.

Burton, R.M., and Obel, B. 1977. The multilevel approach to organizational issues of the firm — a critical review. Omega, **5**: 395–413. doi:10.1016/0305-0483(77)90004-4.

Dantzig, G.B. 1963. Linear programming and extensions. Princeton University Press, Princeton, N.J.

Dantzig, G.B., and Wolfe, P. 1961. The decomposition algorithm for linear programming. Econometrica, **29**: 767–778. doi:10.2307/1911818.

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. Can. J. For. Res. **23**: 1078–1095.

Falcao, A.O., and Borges, J.G. 2001. Designing an evolution program for solving integer forest management scheduling models: an application in Portugal. For. Sci. **47**: 158–168.

Gunn, E.A. 1991. Some aspects of hierarchical production planning in forest management. *In* Proceedings of the Symposium on Systems Analysis in Forest Resources, Charleston, South Carolina, 3–7 March 1991. USDA For. Serv. GTR-74. pp. 54–62.

Gunn, E.A. 1995. Hierarchical planning processes in forestry: a stochastic programming – decision analytic perspective. Petawawa National Forestry Institute, Petawawa, Ont. Rep. PI-X-124.

Gunn, E.A., and Richards, E.W. 2005. Solving the adjacency problem with stand-centered constraints. Can. J. For. Res. **35**: 832–842. doi:10.1139/x05-013.

Hoganson, H.M., and Borges, J.G. 1998. Using dynamic programming and overlapping subproblems to address adjacency in large scale harvest scheduling problems. For. Sci. **44**: 526–538.

Hoganson, H.M., and Rose, D.W. 1984. A simulation approach for optimal timber management scheduling. For. Sci. **30**(1): 220–238.

Heal, G.M. 1973. Theory of economic planning. North-Holland Pub. Co., Amsterdam; American Elsevier Pub. Co., New York.

Hof, J.G., and Pickens, J.B. 1987. A pragmatic multilevel approach to large-scale renewable resource optimization: a test case. Nat. Res. Modeling, **1**: 245–262.

Hurwicz, L. 1973. The design of mechanisms for resource allocation. Am. Econ. Rev. **63**: 1–30.

Iverson, D.C., and Alston, R.M. 1986. The genesis of FORPLAN: a historical and analytical review of forest planning models. USDA For. Serv. GTR-INT-214.

Johnson, N.K. 1986. Reflections on the development of FORPLAN. Presented at the Symposium FORPLAN: An Evaluation of a Forest-Planning Tool. USDA For. Serv. GTR-RM-140.

Johnson, N.K., and Scheurman, L.H. 1977. Techniques prescribing optimal timber harvest and investment under different objectives — discussion and synthesis. For. Sci. Monograph No. 18.

Kornai, J., and Liptak, T. 1965. Two-level planning. Econometrica, **33**(1): 141–169.

Lockwood, C., and Moore, T. 1993. Harvest scheduling with spatial constraints: a simulated annealing approach. Can. J. For. Res. **23**: 468–478.

Lu, F., and Eriksson, L.O. 2000. Formation of harvest units with genetic algorithms. For. Ecol. Manage. **130**: 57–67. doi:10.1016/S0378-1127(99)00185-1.

Martin, R.K. 1999. Large-scale linear and integer optimization: a unified approach. Kluwer Academic, Boston.

Murray, A.T. 1999. Spatial restrictions in harvest scheduling. For. Sci. **45**: 45–52.

Murray, A.T., and Church, R.L. 1995. Measuring the efficacy of adjacency constraint structure in forest planning models. Can. J. For. Res. **25**: 1416–1424.

Nelson, J., and Brodie, J.D. 1990. Comparison of a random search algorithm and mixed integer programming for solving area-based forest plans. Can. J. For. Res. **20**: 934–942.

Nelson, J.D., and Errico, D. 1993. Multiple-pass harvesting and spatial constraints: an old technique applied to a new problem. For. Sci. **39**: 137–151.

Paredes, V.G.L. 1995. Design of a resource allocation method for multiple use forest planning. *In* Hierarchical approaches to forest management in public and private organizations. Petawawa National Forestry Institute, Petawawa, Ont. Rep. PI-X-124.

Snyder, S., and ReVelle, C. 1997. Dynamic selection of harvests with adjacency restrictions: the SHARE model. For. Sci. **43**: 213–222.

Vanderbeck, F. 2000. On Dantzig–Wolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm. Oper. Res. **48**: 111–128.

Vielma, P.J., Murray, A.T., Ryan, D.M., and Weintraub, A. 2006. Improving computational capabilities for addressing volume constraints in forest harvest scheduling problems. Eur. J. Oper. Res. **176**: 1246–1264.

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

Weintraub, A., and Cholaky, A. 1991. A hierarchical approach to forest planning. For. Sci. **37**: 439–460.

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

Yoshimoto, A., Brodie, J.D., and Sessions, J. 1994. A new heuristic to solve spatially constrained long-term harvest scheduling problems. For. Sci. **40**: 365–396.

Appendix A. Constructing adjacency constraints from management alternatives

Here, we present the formulation used in this study to enforce green-up adjacency restrictions. Implementing adjacency constraints in this way makes it possible to easily consider multiple rotations. The example illustrates two rotations with green-up restrictions that are only enforced during the same harvest period instead of one period before and after as presented in the body of the paper.

Consider the contiguous block of three stands in management zone 2 in Fig. 1. Further, consider several management alternatives that have been prepared for each of these stands, for example, with two rotations embedded within each alternative. Suppose that each stand has two management alternatives. A management alternative specifying when a harvest occurs can be represented as a binary column C_{ij}, indicating when stand i is harvested under management alternative j. An example matrix indicating harvest periods that may require green-up restrictions is presented in Table A1.

In generating adjacency constraints, one can define an indicator function I.

$$\text{[A.1]} \qquad I_{i'j',i,j} = 1$$

if stand i′, under its j'th management alternative, produces a green-up violation with stand i under its jth management alternative; otherwise $I_{i'j',i,j} = 0$. Consider writing the adjacency constraint for stand 1 under management alternative 1. Using option C11 precludes the option of C32 only. Therefore, if $\boldsymbol{x}_{ij}$ is the decision variable for a particular management option, the following constraint [A.2] is generated:

$$\text{[A.2]} \qquad \boldsymbol{x}_{11} + \boldsymbol{x}_{32} \leq 1$$

Consider writing the constraint for management alternative C12. Alternative C12 produces a green-up violation with stand 2 under alternative 1 during its first harvest and also a green-up violation with stand 3 under its second alternative. This induces constraint [A.3].

$$\text{[A.3]} \qquad \boldsymbol{x}_{12} + \boldsymbol{x}_{21} + \boldsymbol{x}_{32} \leq 1$$

This processing of each stand and alternative can be continued until all stands and alternatives have been considered. Some of the constraints generated in this fashion will be replicated. In the matrix generation process used in this paper, green-up violations between stands that were considered previously were not repeated. Hence, for example, when stand 3, management alternative 2 is processed, a constraint restricting selection of C12 and C32 will not be generated because it would be redundant; it was generated when stand 1, alternative 2 was processed.

Suppose we only want to consider adjacency in the first three periods of the problem. The indicator function *I* (see [A.1]) is simply changed to only yield a value of 1 when a green up violation is produced in the first three periods; otherwise, $I = 0$. It should be evident that the number of harvest rotations embedded in a management alternative does not affect the logic of this algorithm. Generally, more harvest rotations increase the number of columns in a planning problem; however, additional rotations do not affect the ability to perform this adjacency constraint method.

Table A1. An example matrix indicating harvest activity.

Time period	C11	C12	C21	C22	C31	C32
1	1	0	0	0	0	1
2	0	1	1	0	0	0
3	0	0	0	1	0	0
4	0	0	0	0	1	0
5	1	0	0	0	0	0
6	0	1	0	0	0	1
7	0	0	1	0	0	0
8	0	0	0	1	1	0

Note: Harvests for a time period (1–8) and management alternative (C11–C32) pair are coded as 1; nonharvests are shown with 0.