

IMPROVED SOLUTION TECHNIQUES FOR MULTIPERIOD AREA-BASED HARVEST SCHEDULING PROBLEMS

Juan Pablo Vielma¹, Alan T. Murray², David Ryan³, Andres Weintraub⁴

ABSTRACT

Area-based harvest scheduling models, where management decisions are made for relatively small units subject to a maximum harvest area restriction, are known to be very difficult to solve by exact techniques. Previous research has developed good approaches for solving small and medium sized forestry applications based on projecting the problem onto a cluster graph for which cliques can be applied. However, as multiple time periods become of interest, current approaches encounter difficulties preventing successful identification of optimal solutions. In this paper we present an approach for elasticizing timber demand constraints, which lends itself to an efficient solution technique. It is also possible using this approach to examine trade-offs between objective value performance and maintaining demand constraints.

INTRODUCTION

Mathematical modeling has been frequently used for harvest schedule planning. This has allowed several regulations and requirements to be incorporated in the planning process. These regulations are generally incorporated as restrictions to a Linear Integer Programming model and often make the problem more difficult to solve.

Regulations limiting spatial disturbances have led to constraints, typically known as maximum area restrictions, limiting the size of clear cut areas (Thompson et al. 1973, Murray 1999). Several models using these constraints have been proposed over the years, but the model known as the Area Restriction Model (ARM) has been shown to deliver the most profitable harvest schedules (Murray and Weintraub 2002). Unfortunately the ARM has proven to be very difficult to solve computationally. Although several heuristics to solve this model have been proposed (Hokans 1983,

Lockwood and Moore 1993, Barrett et al. 1998, Clark et al. 1999, Richards and Gunn 2000, Boston and Bettinger 2001), exact methods have only recently been able to solve small and medium problem instances. One such method is that developed in Goycoolea et al. (2003), focusing on a strengthened formulation known as the Cluster Packing Problem. They were able to solve modest sized problems using a commercial integer programming solver for single period application instances. While solvability for multiple planning periods is possible, adding volume production restrictions creates significant complications for problem solution.

In this work we present an alternative way of structuring volume restrictions in order to restore most of the favorable properties of the single period Cluster Packing Problem. Application results are presented which demonstrate that near optimal solutions can be obtained quickly using the developed modeling approach.

In: Bevers, Michael; Barrett, Tara M., comps. 2005. *Systems Analysis in Forest Resources: Proceedings of the 2003 Symposium*. General Technical Report PNW-GTR-656. Portland, OR: U.S. Department of Agriculture, Forest Service, Pacific Northwest Research Station.

¹ Juan Pablo Vielma, Departamento de Ingenieria Matemática, Universidad de Chile, Casilla 2777, Santiago, Chile (Email: jvielma@dim.uchile.cl)

² Alan T. Murray, Department of Geography, The Ohio State University, 1036 Derby Hall, 154 North Oval Mall, Columbus, Ohio 43210, USA (Email: murray.308@osu.edu).

³ David Ryan, Department of Engineering Science, University of Auckland, Private Bag 92019, Auckland, New Zealand (Email: d.ryan@auckland.ac.nz).

⁴ Andres Weintraub, Departamento de Ingenieria Industrial, Universidad de Chile, Casilla 2777, Santiago, Chile (Email: aweintra@dii.uchile.cl).

HARVEST SCHEDULING WITH SPATIAL CONSTRAINTS

The harvest scheduling problem consists of selecting which areas of a forest will be harvested in different periods. Different types of requirements can be added to the generated harvested schedules. One environmental constraint that is generally enforced limits the contiguous area that can be harvested in any period. These constraints are generally known as maximum area restrictions (Thompson et al. 1973, Murray 1999).

We will assume that the forest is divided into sectors whose area is smaller than the maximum area that can be harvested contiguously and we will solve the harvest scheduling model known as Area Restriction Model (ARM). We will also assume a green-up time of one period. Finally we assume that each sector of the forest can only be harvested once during the planning horizon and that some kind of smoothing constraints over the volume of timber produced are desirable. Our base ARM formulation will be the Cluster Packing Problem developed in Goycoolea et al. (2003).

CLUSTER PACKING HARVEST SCHEDULING MODEL

The Cluster Packing Problem uses geographic information system (GIS) based data to model the harvest scheduling model. This data partitions the forest into small units for which area, volume and harvest profit information is available. The area of each unit is generally smaller than the maximum clear cut size specified for the Maximum Area Restrictions, so some groups of adjacent units may be harvested together.

We will define the set of Feasible Clusters (Λ) as all groups of adjacent units whose combined area does not exceed the maximum clear cut size. All of these clusters will be generated a priori by enumeration. This can be done efficiently as the maximum area restrictions generally limit the number of units in a cluster to 4 or 5 (Goycoolea et al. 2003). We will say that two clusters are incompatible if they share a unit or if they are adjacent. Forbidding the simultaneous harvesting of incompatible clusters will assure compliance with the maximum clear cut restrictions. This requirement is modeled by Goycoolea et al. (2003) using maximal cliques to impose incompatibilities. These restrictions give the formulation integrality properties that make it relatively easy to solve. Almost all instances of the single period problem are solved to optimality in the root Branch & Bound (B&B) node by CPLEX 8.1.

Maximize	$\sum_{S,t} c_{S,t} x_{S,t}$	
subject to	$\sum_{S \in \Lambda(K)} x_{S,t} \leq 1$	for each maximal clique K and for each period t
	$\sum_{S,t: u \in S} x_{S,t} \leq 1$	for each unit u
	$x_S \in \{0,1\}$	for each cluster $S \in \Lambda$ and for each period t
where:		
	• $c_{S,t}$ is net present value of the profit of cluster S for period t	
	• $\Lambda(K)$ is the set of all clusters that intersect maximal clique K	

Formulation 1—Multi-period period cluster packing problem.

The multi-period version of the model allows harvesting over several periods, but only allows each cell to be harvested once in the planning horizon. This model is presented in formulation 1. In this formulation variable $x_{S,t}$ is 1 if cluster S is harvested in period t and 0 otherwise. The objective is to maximize the net present value of the profit associated with the harvest schedule. The first set of constraints is a strengthened version of the constraints that force compliance with the area restrictions by forbidding two incompatible clusters from being harvested in the same period. Finally the last two sets of constraints forbid units from being harvested more than once in the planning horizon and force the variables to be binary, respectively.

This formulation preserves most of the good properties of the single period formulation and is easily solvable, as the computational results will show.

The multi-period model can be complemented with different kinds of restrictions on the volume harvested in each period. The most common restrictions include the production smoothing volume constraints and upper/lower bounds over the volume production.

One typical restriction on the harvested volume is to require total volume in a period to be within $\pm\Delta\%$ of previous periods. This can be achieved by adding the following restrictions to the multi-period model for each time period $t > 1$:

$$\left(1 - \frac{\Delta}{100}\right) \sum_S v_{S,t-1} x_{S,t-1} \leq \sum_S v_{S,t} x_{S,t} \leq \left(1 + \frac{\Delta}{100}\right) \sum_S v_{S,t-1} x_{S,t-1}$$

where $v_{S,t}$ the volume harvested if cluster S is selected to be harvested in period t.

Other restrictions that are frequently applied are minimum and maximum harvested volumes. This can be achieved by adding the following restrictions to the multi-period model for each time period t :

$$L \leq \sum_S v_{S,t} x_{S,t} \leq U$$

where U and L are the maximum and minimum volume allowed to be harvested in each period.

For both types of restrictions it is common that one of the inequalities is active, and hence acts as a fractional generating cut on the LP polytope. This fractional generating effect causes solutions to the LP relaxation to have many fractions. Furthermore, these fractions are difficult to eliminate by variable fixing. As the computational results will show, this makes the problem very difficult to solve.

ELASTIC VOLUME CONSTRAINT MODEL

One technique that can be used to minimize the fractional generating effects of volume constraints is to use an elastic version of the constraints. An elastic constraint allows a violation of the restriction, but penalizes this violation in the objective function. In this manner the volume constraints will no longer act as strong cuts, and hence will generate almost no new fractional extreme points to the LP polytope. This will restore practically all the integrality properties of the multi-period model without volume constraints. Elastic constraints have been successfully used in similar problems (see Ehrgott and Ryan 2003).

It is very difficult to find penalties that will lead to integer solutions that do not violate volume restrictions. For this reason it is a good idea to start penalizing before the restrictions are really violated. So, for example, if we wanted to solve the problem with $\pm 15\%$ production smoothing volume constraints⁵, we could add a $\pm 14\%$ production smoothing volume constraint⁶, allow violations to these constraints, and penalize their violation in the objective function. In this way, if we just keep the violations controlled (below 1%), we will be complying with our target 15% volume constraint.

In the following section we will describe the elastic constraints for the production smoothing volume constraints. The corresponding relaxations for the upper/lower bound volume constraints are analogous.

⁵ i.e. with $\Delta=15\%$ in the original model

⁶ i.e. with $\Delta_E=14\%$ in the elastic model

<p>Maximize $\sum_{S,t} c_{S,t} x_{S,t} - \sum_{t>1} \underline{p}_t l_t - \sum_{t>1} \bar{p}_t u_t$</p> <p>$\sum_{S \in \Lambda(K)} x_{S,t} \leq 1$ for each maximal clique K for each period t</p> <p>$\sum_{S,j \in \mathcal{U}(S)} x_{S,j} \leq 1$ for each unit u in V</p> <p>$(1 - \frac{\Delta_E}{100}) \sum_S v_{S,t-1} x_{S,t-1} - \sum_S v_{S,t} x_{S,t} \leq l_t$ for each period $t > 1$</p> <p>$\sum_S v_{S,t} x_{S,t} - (1 + \frac{\Delta_E}{100}) \sum_S v_{S,t-1} x_{S,t-1} \leq u_t$ for each period $t > 1$</p> <p>$x_{S,t} \in \{0,1\}$ $l_t, u_t \geq 0$ for each cluster $S \in \Lambda$ and for each period t</p> <p>with $\Delta_E \leq \Delta$ and $\underline{p}_t, \bar{p}_t$ to be determined independently for each t</p>

Formulation 2—Multi-period period cluster packing problem with elastic volume constraints

If we add elastic volume constraints to the multi-period model, we obtain the following formulation:

SHAPE * MERGEFORMAT

Formulation 2. Multi-period period cluster packing problem with elastic volume constraints

INTEGER ALLOCATION

Although penalties can be easily adjusted to control volume constraint violations for the root B&B node, it might be very difficult to do this and get integer solutions. General purpose LP based heuristics tend to have problems generating solutions with small volume constraint violations. For this reason a custom integer allocation heuristic was developed. The heuristic fixes variables and re-solves the linear relaxation of the model while trying to account for any violations that are too big.

The elastic volume constraints are crucial for the performance of the heuristic. The fractional generating effect of the volume constraints makes it very difficult to develop an LP based heuristic for the strict volume constraint model. Fixing some fractional variables to integrality in this model generally ends in the appearance of an alternate set of fractional variables, making the integer allocation process very slow. This does not happen with the elastic constraint model as the fractional generating effect of the strict volume constraints is not present. On the other hand, if the penalties are big enough, the violations will probably be reasonably controlled. Some corrections of the violations are still necessary, but they are very few due to the penalties.

Table 1—Characteristic features.

Instance	# of cells	# of Feasible clusters	Total # of restrictions for 15 period model without volume constraints	Total # of variables for 15 period model without volume constraints
El Dorado	1363	21412	32938	321180
rand 12 by 12 t15	144	2056	1959	30840

Table 2—Multi-period model without volume constraints results

Map	Time periods	IP time [s]	B&B nodes	1 st sol under 1% time [s]	1 st feasible time [s]	1 st feasible gap [%]
El Dorado 15	12	720	30	448	173	6.58
El Dorado 15	15	147	0	101	77	3.02
ran 12 by 12	12	501	789	24	14	33.61
ran 12 by 12	15	524	732	32	19	38.32

COMPUTATIONAL RESULTS

Computational tests were run over two instances: a real forest in Northern California called El Dorado and a randomly generated square grid with 144 units. Table 1 shows a summary of the problem characteristics.

Multi-period applications containing 12 and 15 periods were tested for both instances. The runs were made on a Pentium 4 2.0GHz PC with 2.0 Gb of RAM running Linux. CPLEX 8.1 was used as the MIP solver and problem generation and heuristics were programmed in C++.

Multi-period model without volume constraints

Table 2 shows computational results for the multi-period model without volume constraints. A time limit of 4 hours was imposed, but for all instances it was possible to declare optimality long before that time limit. The first two columns show the instances characteristics. Columns 3 and 4 show the time and B&B nodes needed to declare optimality. Finally the last three columns show information regarding integer feasible solutions found before declaring optimality. Column 5 shows when the first solution with an LP gap⁷ of under 1% was found and columns 6 and 7 show the time the first feasible solution was found and its LP gap.

The integrality properties of this model help CPLEX 8.1 find feasible solutions very quickly and also declare optimality in little time.

Production smoothing volume constraint model

Table 3 shows the results for the production smoothing volume constraint model as solved directly by CPLEX 8.1. All tests for this table were run for 8 hours. The format of table 3 is similar to that of table 2. Additionally column 3 shows the level used for the volume constraints. As optimality could not be declared, columns 5 and 6 show the time the best feasible solution was found and its LP gap. Finally column 4 shows the total number of B&B nodes processed in the allotted time. A dash (-) indicates that a feasible solution with the required characteristics was not found.

It can be seen that CPLEX has a lot of trouble finding integer solutions. Although eventually it does find good solutions for El Dorado, computational effort is significant. No integer solutions are found for the grid instances.

Production smoothing elastic volume constraint method

Table 4 shows the results for the elastic constraint method. This method is essentially B&B over the multiple penalties elastic constraint model with constraint branching plus the integer allocation heuristic.

⁷ gap=(obj_best_lp - obj_ip)/obj_ip*100, where obj_best_lp is the greatest linear relaxation optimal value among the B&B nodes to be processed. obj_ip is the objective value of the particular integer feasible solution.

Table 3—Volume Constraint Model Results

Map	Time periods	Δ	B&B nodes	Best solution time [s]	Gap [%]	1 st sol under 1% gap [s]	1 st feasible time [s]	1 st feasible gap
El Dorado 15	12	10%	2133	18606	1.47	—	18606	1.47
El Dorado 15	15	10%	1575	18315	0.83	10839	10839	1.00
ran 12 by 12	12	10%	388	—	—	—	—	—
ran 12 by 12	15	10%	388	—	—	—	—	—
El Dorado 15	12	15%	2087	11211	0.50	10719	2323	1.51
El Dorado 15	15	15%	2067	20733	0.59	20274	20274	0.77
ran 12 by 12	12	15%	634	—	—	—	—	—
ran 12 by 12	15	15%	342	—	—	—	—	—

Table 4—Elastic Volume Constraint Method Results

Map	Time periods	Δ	B&B nodes	Best solution time [s]	Gap [%]	1 st sol under 1% gap [s]	1 st feasible time [s]	1 st feasible gap
El Dorado 15	12	10%	23	5555	0.41	1706	1706	0.43
El Dorado 15	15	10%	13	12541	0.44	4307	4307	0.45
ran 12 by 12	12	10%	75	5059	3.43	—	663	8.70
ran 12 by 12	15	10%	25	13856	4.52	—	614	14.42
El Dorado 15	12	15%	18	12216	0.30	1160	1160	0.33
El Dorado 15	15	15%	13	9916	0.29	2387	2387	0.34
ran 12 by 12	12	15%	199	9684	2.29	—	312	5.07
ran 12 by 12	15	15%	20	9124	4.97	—	504	7.99

The format of table 4 is the same as table 3 with the exception of the meaning of Δ and how the gaps are calculated. Δ corresponds to the strict volume constraint we are trying to comply with. Again we use $\Delta_E=(\Delta-1)\%$ and allow only 1% violation to solve the exact volume constraint with level $\Delta\%$. LP gaps are calculated with respect to the LP solution of the corresponding exact $\Delta\%$ volume constraint model, so they can be compared to the gaps reported in table 3.

Penalties for each constraint are set independently so that the root LP has less than 1% violation, but they are then kept fixed in the B&B tree. A time limit of only 4 hours, instead of 8, was used for these tests.

Although with this method fewer B&B nodes are processed, we can get good solutions quickly for El Dorado

and we can also get integer solutions for the grid instances quickly. However, their quality is not good. It should be noted though that the grids were purposefully generated so that it is very difficult to get integer solutions that comply with the volume constraints tightly. Thus, large gaps between the IP and LP solutions for the grid cases are expected. If we compare these results with table 3, we see that the elastic constraint method is much faster than the strict volume constraint model. Integer feasible solutions with similar objective values are found up to 150 times faster⁸ with this method.

CONCLUSIONS

By eliminating the fractional generating effect of the strict volume constraints, it is much easier to obtain integer feasible solutions. For this reason the elastic constraint method allows good solutions to be obtained much earlier

⁸ CPLEX was run for 24 hours for the strict volume constraint model for the ran12by12 instance with 15 periods. Only one solution with a 9% gap was found after 22 hours.

than when solving the strict volume constraint model directly.

It should be noted also that restrictions on harvested volume are generally guides instead of strict requirements, so small violations would likely be acceptable. It is clear that allowing these small violations (for example by allowing violations slightly over 1% of the 14% volume constraint) will give superior results. This provides yet another reason for not using strict volume constraints.

During the computational analysis, it was found that the integer allocation heuristic worked better when the initial LP had little or no violations of the target volume constraints. Because of this, it might be useful to adjust penalties each time a volume restriction is violated in the B&B tree. This would also guarantee that integer solutions found in leafs of the B&B tree would comply with the target volume constraints. We are currently implementing this dynamic adjustment of penalties to be added to the B&B based integer allocation method.

ACKNOWLEDGEMENTS

Partial funding for this research was received from FONDECYT under grant number 1000959. Research support for the second author was provided by the National Science Foundation (Geography and Regional Science Program and the Decision, Risk, and Management Science Program) under grant BCS-0114362.

LITERATURE CITED

- Barrett, T., Gillies, J. and Davis, L. 1998.** "Economic and fragmentation effects of clearcut restrictions." *Forest Science* 44: 569-577.
- Boston, K. and Bettinger, P. 2001.** "The economic impact of green-up constraints in the southeastern United States." *Forest Ecology and Management* 145: 191-202.
- Clark, M.M., R.D. Meller and McDonald, T.P. 2000.** "A three-stage heuristic for harvest scheduling with access road network development." *Forest Science* 46: 204-218.
- Ehrgott, M. and Ryan, D. 2003.** "The Method of Elastic Constraints for Multiobjective Combinatorial Optimization and its Application in Airline Crew Scheduling", in "Multi-Objective Programming and Goal Programming", Springer Verlag, Berlin.

- Goycoolea, M., Murray A., Barahona F., Epstein R., and Weintraub A. 2005.** "Harvest scheduling subject to maximum area restrictions: exploring exact approaches." *Operations Research*, 53:3: 490-500.
- Hokans, R.H. 1983.** "Evaluating spatial feasibility of harvest schedules with simulated stand-selection decisions." *Journal of Forestry* 81: 601-603,613.
- Lockwood, C. and T. Moore. 1993.** "Harvest scheduling with spatial constraints: a simulated annealing approach." *Canadian Journal of Forest Research* 23: 468-478.
- Murray, A. 1999.** "Spatial Restrictions in Harvest Scheduling", *Forest Science*. Vol. 45, No. 1. 1-8.
- Murray, A.T. and Weintraub, A. 2002.** "Scale and unit specification influences in harvest scheduling with maximum area restrictions." *Forest Science* 48: 779-789.
- 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: 188-203.
- Thompson, E.F., B.G. Halterman, T.S. Lyon and R.L. Miller. 1973.** "Integrating timber and wildlife management planning." *Forestry Chronicle* 47: 247-250.