available at www.sciencedirect.com

SCIENCE DIRECT®

journal homepage: www.elsevier.com/locate/ecolmodel

Optimisation algorithms for spatially constrained forest planning

Guoliang Liu[a,*], *Shijie Han*[a], *Xiuhai Zhao*[b], *John D. Nelson*[c], *Hongshu Wang*[d], *Weiying Wang*[e]

[a] *Institute of Applied Forest Ecology, Chinese Academy of Science, Shen Yang, China*
[b] *The Key Laboratory for Silviculture and Conservation of Ministry of Education, Beijing Forestry University, Beijing, China*
[c] *Faculty of Forestry, University of British Columbia, 2424 Main Mall, Vancouver, BC, Canada V6T 1Z4*
[d] *Faculty of Forest Economics, Northeast Forestry University of China*
[e] *Forestry Research Academy, Heilongjiang Province, China*

ARTICLE INFO

Article history:
Received 14 December 2004
Received in revised form 7 October 2005
Accepted 17 October 2005
Published on line 13 December 2005

Keywords:
Landscape design
Integrated forest resource planning
Harvest scheduling
Genetic algorithms
Simulated annealing

ABSTRACT

We compared genetic algorithms, simulated annealing and hill climbing algorithms on spatially constrained, integrated forest planning problems. There has been growing interest in algorithms that mimic natural processes, such as genetic algorithms and simulated annealing. These algorithms use random moves to generate new solutions, and employ a probabilistic acceptance/rejection criterion that allows inferior moves within the search space. Algorithms for a genetic algorithm, simulated annealing, and random hill climbing are formulated and tested on a same-sample forest-planning problem where the adjacency rule is strictly enforced. Each method was randomly started 20 times and allowed to run for 10,000 iterations. All three algorithms identified good solutions (within 3% of the highest found), however, simulated annealing consistently produced superior solutions. Simulated annealing and random hill climbing were approximately 10 times faster than the genetic algorithm because only one solution needs to be modified at each iteration. Performance of simulated annealing was essentially independent of the starting point, giving it an important advantage over random hill climbing. The genetic algorithm was not well suited to the strict adjacency problem because considerable computation time was necessary to repair the damage caused during crossover.

1. Introduction

Mathematical programming models used for planning broad-scale, multiple-use forest harvest scheduling problems are very difficult to solve due to the problem size and the constraint structure. A common objective of forest planning is to generate a long-term harvest schedule that maximises the volume harvested (or the net profit), subject to numerous constraints. Typical constraints specify the minimum harvest age, allowable opening size, and forest cover constraints. In addition, harvest flow and budget constraints are usually added to control resources in each time period. Further complexity is added by the dynamic growth of the forest through time, but it is primarily the integer solution required by the opening size rule (adjacency) that makes these problems difficult to solve.

* *Corresponding author at*: Forest Ecosystem Solutions Ltd., 210-275 Fell Avenue, North Vancouver, BC, Canada V7P 3R5. Tel.: +1 604 998 2222; fax: +1 604 986 0361.
E-mail address: gliu@forestecosystem.ca (G. Liu).
0304-3800/$ – see front matter
doi:10.1016/j.ecolmodel.2005.10.028

Many mathematical programming techniques have been used to solve these problems. Similar to Hill Climbing, an application of the Metropolis algorithm for forest harvest scheduling is used by Van, Deusen in 1999. Mixed Integer Programming (MIP) is a classical optimisation technique that has been applied to a number of problems (Bare and Eldon, 1969; Jones et al., 1991; McDill and Rebain, 2003; Bevers and Hof, 1999). Hoganson and Borges (1998) used dynamic programming and overlapping subproblems to address adjacency in harvest scheduling problems. However, most large size problems of operational dimensions remain outside the limits of reasonable computing resources. In response to these limitations, techniques designed for generating good, approximate solutions, have been investigated. One popular method, random searching of the feasible region, is attractive because of its simplicity and suitability to problems for which little specific information is available (Nelson and Brodie, 1990; Clements et al., 1990). Forest harvest scheduling problems with adjacency constraints are solved using the branch and bound algorithm (Crowe et al., 2003). Random searches are excellent exploratory tools, but typically they are inefficient in generating high-valued solutions.

As more problem-specific information becomes available, more efficient algorithms can be designed to take advantage of specific structures. Prioritising harvest units within simulation models has produced good results (Nelson and Finn, 1991; O'Hara et al., 1989). Simulations combined with random search methods have also been applied to tactical forest planning problems (Sessions and Sessions, 1991). Rosie (1990) used a non-linear penalty function method to analyse the spatial arrangement of stands. Penalty functions within simulated annealing algorithms have also proved promising for forest planning problems (Lockwood and Moore, 1993). Other recent applications include interchange, simulated annealing, and Tabu searches (Murray and Church, 1996). A series of tabu search (TS) methods for solving the stand harvesting and road access optimisation problem was developed and evaluated by Richards and Gunn (2003).

All have tended to use an element of randomness to globally search the solution space. The common thread in these algorithms is to accept inferior solutions in the short-term to avoid convergence on local optima. Pukkala and Kurttila (2005) have compared six heuristic optimisation techniques in different forest planning problems. Bettinger et al. (1999) have intensified a heuristic forest harvest scheduling search procedure with 2-opt decision choices. More complicated problems such as wildlife habitat planning problems were solved with different heuristic search algorithms (Bettinger et al., 2002). Heinonen and Pukkala (2004) compared one- and two-compartment neighbourhoods in heuristic search with spatial forest management goals.

Other random, mathematical programming methods that have received much attention in non-forestry fields include the genetic algorithm and evolution algorithms. In classical genetic algorithm studies, the chromosome that represents the potential solution is composed of a fixed length binary string. Evolution algorithms are more general in their solution representation, allowing base 10 numbers, tables or other data structures (Davis, 1991). Aside from these differences in solution representation, the two methods are essentially identical in the way they mimic natural processes like mutation, spread of genetic material, and natural selection. Genetic algorithms have been applied to travelling salesman problems (Whitley and Starkweather, 1989), transportation problems (Vignaux and Michalewicz, 1989), resource scheduling (Syswerda and Palmucci, 1991), communication networks (Michalewicz, 1991), placement and routing of integrated circuits (Sechen, 1988), and production scheduling (Husbands et al., 1990). These probabilistic algorithms have an adaptive acceptance criterion that progressively gets more stringent as the number of iterations increases. The genetic algorithms and evolution algorithms are particularly well suited to problems where there is little information about processes and interactions, like the field of genetics, where the algorithms were developed. The algorithms are able to perform fairly efficient searches even when prior knowledge is lacking, and only an evaluation of the objective function is possible. This makes genetic algorithms and evolution algorithms useful search techniques for many problems for which the only alternative is some form of random search. However, research has shown that when additional knowledge about the problem in question can be incorporated, the genetic algorithm can be considerably more powerful (Grefenstette, 1987; Michalewicz, 1992).

Like evolution algorithms, simulated annealing and random hill climbing rely on random moves within the search space, however, the population is limited to one, and the acceptance criteria for new solutions are different. These methods are suited to ill-structured problems where little prior information is available (van Laarhoven and Aarts, 1987), but we anticipated that their performance could be improved by incorporating problem-specific information.Evolution program formulation for forest planning

The objectives of this paper are: (1) to present formulations for an evolution program, simulated annealing, and random hill climbing in a forest planning problem that incorporate problem-specific knowledge about forest harvest scheduling, and (2) to compare the evolution program, simulated annealing and random hill climbing methods on a sample forest planning problem. We chose a relatively simple planning problem to demonstrate the algorithms. The objective is to maximize timber volumes over a single rotation, subject to minimum harvest ages, strict adjacency rules, and even flow. Modifications for more complicated problems, such as meeting patch size distributions, are discussed near the end of the paper. The first section of the paper describes the formulation and procedures for the evolution program. Second, the simulated annealing and random hill climbing formulations are described. In the third section, the algorithms are tested on the sample forest-planning problem, and these results are presented and discussed. Finally, some general conclusions and opportunities for further research are presented.

2. Evolution program formulation for forest planning

An evolution program begins with a population of randomly generated solutions. Solutions are then randomly selected from the population to undergo crossover that exchanges