Forest Management and Certification

Stand-level management planning - decision analysis in single species even-aged stands

Who? Manuel Arias-Rodil
Ulises Diéguez-Aranda

From? USC UNIVERSIDADE DE SANTIAGO DE COMPOSTELA
DEPARTAMENTO DE ENXEÑARÍA AGROFORESTAL
ESCOLA POLITÉCNICA SUPERIOR

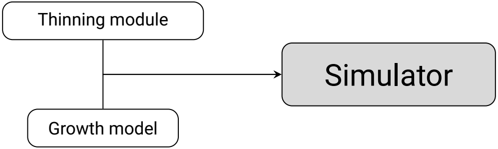
When? 17/10 - 04/11/2016

Introduction

Stand-level management planning

- Stand-level growth and yield models and thinning modules are needed $\rightarrow$ **simulation** of the stand evolution over time

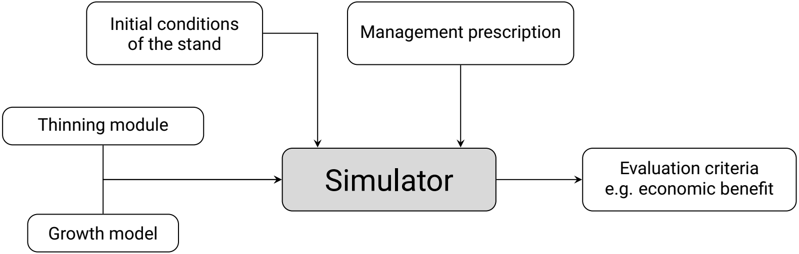
Simulation



Stand-level management planning

- Stand-level growth and yield and thinning modules are needed $\rightarrow$ prediction of the stand evolution over time
- Application of management prescriptions and economic variables $\rightarrow$ **Economic evaluation** (Net Present Value or Land Expectation Value)

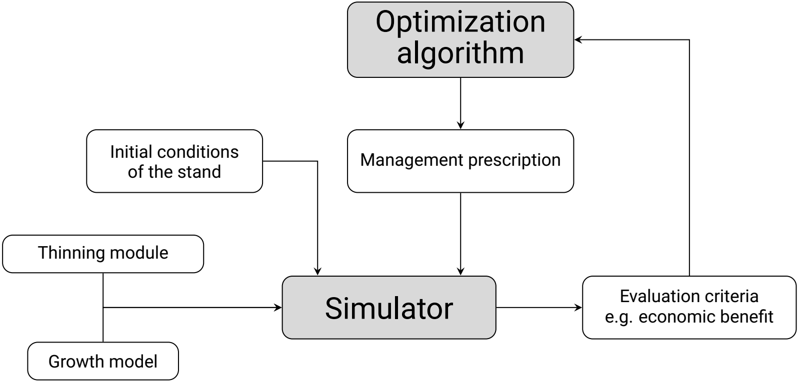
Economical evaluation



Stand-level management planning

- Stand-level growth and yield and thinning modules are needed $\rightarrow$ prediction of the stand evolution over time
- Application of management prescriptions and economic variables $\rightarrow$ **Economic criteria** (Net Present Value or Land Expectation Value)
- Automatic search of the optimal management procedure $\rightarrow$ **optimization**

Optimization



Dynamic programming

Dynamic programming (DP): introduction

- Optimization technique, presented by Bellman (1957)
- Objective → identify an optimal path within a **network**
- Dynamic programming network:
  - **Stages**→ positions in the problem where a decision should be made
  - **States**→ possible alternatives of the problem within each stage → each alternative is represented by a **node** in each stage
  - **Arcs** link nodes between consecutive stages

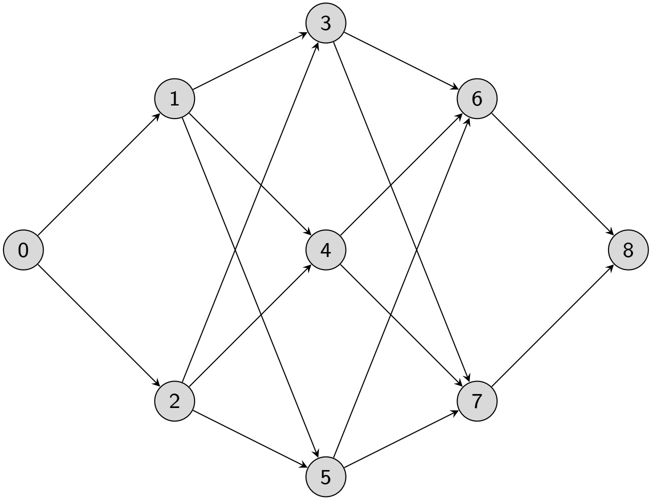
Dynamic programming (DP): introduction

- Optimization technique, presented by Bellman (1957)
- Objective $\rightarrow$ identify an optimal path within a **network**
- Dynamic programming network:
  - **Stages** $\rightarrow$ positions in the problem where a decision should be made
  - **States** $\rightarrow$ possible alternatives of the problem within each stage $\rightarrow$ each alternative is represented by a **node** in each stage
  - **Arcs** link nodes between consecutive stages

Dynamic programming (DP): introduction

- Optimization technique, presented by Bellman (1957)
- Objective $\rightarrow$ identify an optimal path within a **network**
- Dynamic programming network:
  - **Stages** $\rightarrow$ positions in the problem where a decision should be made
  - **States** $\rightarrow$ possible alternatives of the problem within each stage $\rightarrow$ each alternative is represented by a **node** in each stage
  - **Arcs** link nodes between consecutive stages

Dynamic programming network



Stage 1 Stage 2 Stage 3 Stage 4

DP solution

- To solve a problem, DP uses **recursion** in order to find the optimal path within the network
- The procedure consists on a selection of the optimal path to get to each node in each stage
- This implies that the optimal paths to the nodes of a stage are considered to find the optimal paths to the nodes of the next stage, without considering the details of the optimal path followed to get to the nodes of the previous stage → this is known as **forward recursion**
  - **Forward recursion**→ moves from the first to the last stage
  - **Backward recursion**→ moves from the last to the first stage

DP solution

- To solve a problem, DP uses **recursion** in order to find the optimal path within the network
- The procedure consists on a selection of the optimal path to get to each node in each stage
- This implies that the optimal paths to the nodes of a stage are considered to find the optimal paths to the nodes of the next stage, without considering the details of the optimal path followed to get to the nodes of the previous stage $\rightarrow$ this is known as **forward recursion**
  - **Forward recursion**$\rightarrow$ moves from the first to the last stage
  - **Backward recursion**$\rightarrow$ moves from the last to the first stage

DP solution

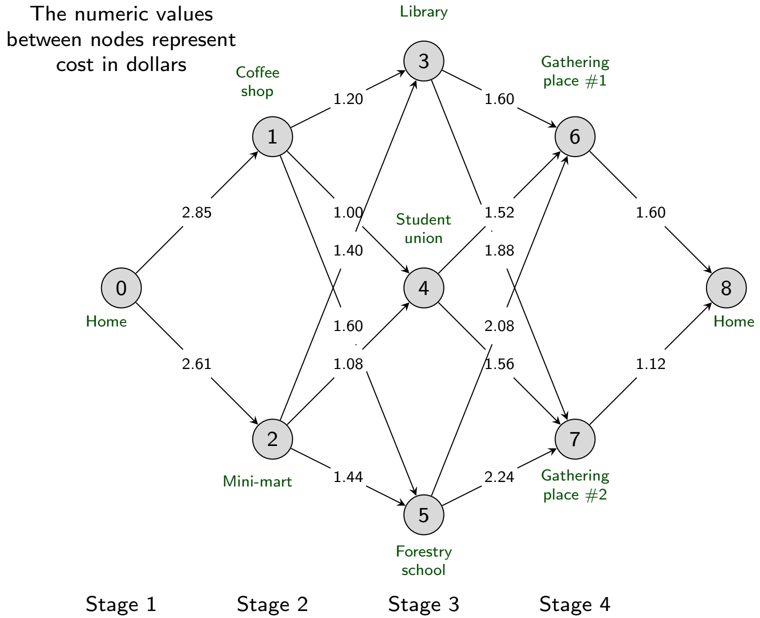
- To solve a problem, DP uses **recursion** in order to find the optimal path within the network
- The procedure consists on a selection of the optimal path to get to each node in each stage
- This implies that the optimal paths to the nodes of a stage are considered to find the optimal paths to the nodes of the next stage, without considering the details of the optimal path followed to get to the nodes of the previous stage $\rightarrow$ this is known as **forward recursion**
  - **Forward recursion**$\rightarrow$ moves from the first to the last stage
  - **Backward recursion**$\rightarrow$ moves from the last to the first stage

Example 1: planning an evening

Example from Bettinger et al. (2009, p. 114-116)

1. An evening, a student plans to leave home in his car
2. Then, he/she wants to pick up a **coffee** or a **soft drink**
3. After that, he/she will go to study to the **library**, the **student union**, or the **forestry school**
4. Afterward, he/she will visit some friends to watch their favorite TV show, but the student can decided where they will meet between **two gathering places**
5. Finally, he/she will drive back home

Example 1: planning an evening



Example 1: planning an evening

- The student wants to minimize the cost of the evening activities
- Some useful definitions of dynamic programming network
  - **From-node**$\rightarrow$ where a branch starts
  - **To-node**$\rightarrow$ where a branch ends
  - **Cost**$\rightarrow$ accumulated cost of a route
  - **Route**$\rightarrow$ path through the network
  - $r_{i\rightarrow j}\rightarrow$ cost associated with going from node i to j
  - $R_j\rightarrow$ minimum cost to get to node j $\rightarrow$ calculated as the minimum value of $R_i + r_{i\rightarrow j}$ for all nodes i that lead to node j
  - $P_j\rightarrow$ previous node that represent the path with the minimum cost to get node j

Stand-level management planning: application of DP

What makes dynamic programming interesting for stand-level planning?

- Forest management and forest growth responses are sequences of actions that may follow similar paths at various points in time
- Selecting the optimal path may require multiple passes through the same point (**node**)
- Recursive procedures of DP become useful

What makes dynamic programming interesting for stand-level planning?

- Forest management and forest growth responses are sequences of actions that may follow similar paths at various points in time
- Selecting the optimal path may require multiple passes through the same point (**node**)
- Recursive procedures of DP become useful

What makes dynamic programming interesting for stand-level planning?

- Forest management and forest growth responses are sequences of actions that may follow similar paths at various points in time
- Selecting the optimal path may require multiple passes through the same point (**node**)
- Recursive procedures of DP become useful

Discretization of state variables

- Stand variables should be discretized to represent the alternative states of stand in a stage $\rightarrow$ state variables
- Variables to consider
  - Univariate, e.g. residual basal area
  - Multivariate, e.g. residual basal area and number of stems

Discretization of state variables

- Stand variables should be discretized to represent the alternative states of stand in a stage $\rightarrow$ state variables
- Variables to consider
  - Univariate, e.g. residual basal area
  - Multivariate, e.g. residual basal area and number of stems

DP network in stand-level planning

- **Stages** $\rightarrow$ represented by age
- **States** $\rightarrow$ defined by one or more stand variables representing the residual value after a management action has been applied
- **Optimal path** $\rightarrow$ best course of management actions to be applied over a stand
- **Rotation length**
  - **Fixed** $\rightarrow$ select the best set of management activities **except** the clearcut timing
  - **Variable** $\rightarrow$ select the best set of management activities **including** the clearcut timing

DP network in stand-level planning

- **Stages**$\rightarrow$ represented by age
- **States**$\rightarrow$ defined by one or more stand variables representing the residual value after a management action has been applied
- **Optimal path**$\rightarrow$ best course of management actions to be applied over a stand
- **Rotation length**
  - **Fixed** $\rightarrow$ select the best set of management activities **except** the clearcut timing
  - **Variable** $\rightarrow$ select the best set of management activities **including** the clearcut timing

DP network in stand-level planning

- **Stages** $\rightarrow$ represented by age
- **States** $\rightarrow$ defined by one or more stand variables representing the residual value after a management action has been applied
- **Optimal path** $\rightarrow$ best course of management actions to be applied over a stand
- **Rotation length**
  - **Fixed** $\rightarrow$ select the best set of management activities **except** the clearcut timing
  - **Variable** $\rightarrow$ select the best set of management activities **including** the clearcut timing

DP network in stand-level planning

- **Stages** $\rightarrow$ represented by age
- **States** $\rightarrow$ defined by one or more stand variables representing the residual value after a management action has been applied
- **Optimal path** $\rightarrow$ best course of management actions to be applied over a stand
- **Rotation length**
  - **Fixed** $\rightarrow$ select the best set of management activities **except** the clearcut timing
  - **Variable** $\rightarrow$ select the best set of management activities **including** the clearcut timing

Alternative to the strict discretization of state variables

- Discretization of state variables can be sometimes **difficult**
  - If the stand state is defined by more than one variable $\rightarrow$ the probability of two paths passing through the same node decreases
  - If decisions on each stage lead to non-discrete state variables, e.g. if we decide on the percentage to cut within a thinning instead of using the residual value of a variable (number of trees, basal area...)
- Brodie and Kao (1979) suggested to consider each node as a neighborhood which all stand states with similar continuous values (within a predefined range) belong to

Alternative to the strict discretization of state variables

- Discretization of state variables can be sometimes **difficult**
  - If the stand state is defined by more than one variable $\rightarrow$ the probability of two paths passing through the same node decreases
  - If decisions on each stage lead to non-discrete state variables, e.g. if we decide on the percentage to cut within a thinning instead of using the residual value of a variable (number of trees, basal area...)
- Brodie and Kao (1979) suggested to consider each node as a neighborhood which all stand states with similar continuous values (within a predefined range) belong to

Alternative to the strict discretization of state variables

- Discretization of state variables can be sometimes **difficult**
  - If the stand state is defined by more than one variable $\rightarrow$ the probability of two paths passing through the same node decreases
  - If decisions on each stage lead to non-discrete state variables, e.g. if we decide on the percentage to cut within a thinning instead of using the residual value of a variable (number of trees, basal area...)
- Brodie and Kao (1979) suggested to consider each node as a neighborhood which all stand states with similar continuous values (within a predefined range) belong to

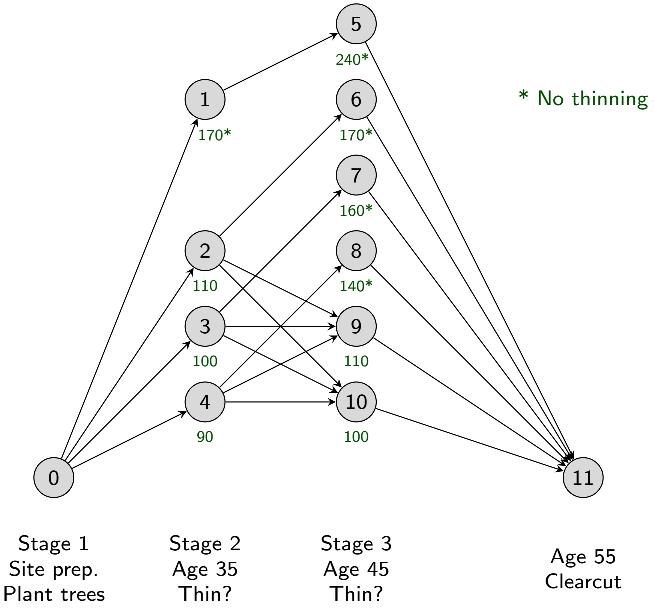
Example$\rightarrow$ a node is defined by $G = 20$ and a bandwidth of 2 $\rightarrow$ all stand states with G between 18 and 22 would be considered to behave equally and represented by the mentioned node

Example 2: stand-level planning with **fixed** rotation length

Example from Bettinger et al. (2009, p. 116-118)

- A landowner is interested on maximizing the value of a investment on a stand that will be cut at the age of 55 years
- Several alternative treatments are considered to be applied when the stand is 35 and 45 years old $\rightarrow$ intermediate cuttings differing on the intensity
- The intermediate states of the dynamic programming network are characterized by the residual basal area left after a thinning
- Only a rotation length (R) is considered$\rightarrow$ 55 years
- Economic variables
  - Plantation cost$\rightarrow$ $250
  - Volume price$\rightarrow$ $400 per MBF
  - Discount rate$\rightarrow$ 5%

Example 2: stand-level planning (fixed R)



Example: calculating Net Present Value

- Plantation cost$\rightarrow$ \$250 (year 0)
- Discount rate$\rightarrow$ 5%
- Calculating revenues$\rightarrow$ \$400 per MBF

$$\mathrm{NPV} = \sum_{t=0}^{R} \frac{I_t - C_t}{(1+i)^t}$$

Example 2: stand-level planning (fixed R)

From-node	To-node	Volume harvested	Revenues	$r_{i\rightarrow j}$
0	1		0	-250
0	2	3.378	1351.2	-5.04
0	3	4.360	1744	66.17
0	4	5.352	2140.8	138.11
1	5		0	0
2	6		0	0
2	9	7.931	3172.4	353.08
2	10	9.343	3737.2	415.94
3	7		0	0
3	9	6.188	2475.2	275.48
3	10	7.636	3054.4	339.94
4	8		0	0
4	9	4.347	1738.8	193.52
4	10	5.762	2304.8	256.52
5	11	44.858	17943.2	1225.99
6	11	36.620	14648	1000.85
7	11	33.804	13521.6	923.88
8	11	31.031	12412.4	848.09
9	11	24.500	9800	669.60
10	11	23.000	9200	628.60

DP: advantages

- It guarantees finding the optimal path within the network
- Avoid evaluating all possible management options $\rightarrow$ the problem is decomposed into a set of smaller, inter-related problems $\rightarrow$ **stages**
- Suboptimal decisions are ignored $\rightarrow$ reduce the solution space

DP: advantages

- It guarantees finding the optimal path within the network
- Avoid evaluating all possible management options $\rightarrow$ the problem is decomposed into a set of smaller, inter-related problems $\rightarrow$ **stages**
- Suboptimal decisions are ignored $\rightarrow$ reduce the solution space

DP: advantages

- It guarantees finding the optimal path within the network
- Avoid evaluating all possible management options $\rightarrow$ the problem is decomposed into a set of smaller, inter-related problems $\rightarrow$ **stages**
- Suboptimal decisions are ignored $\rightarrow$ reduce the solution space

DP: disadvantages

- Lack of shadow prices $\rightarrow$ we are not able to evaluate how the objective function would change in case we modify the constraints in one unit
- States and stages should be carefully defined to avoid
  - Excessive discretization that would imply too high computing times without gaining a significative improvement in the optimal solution
  - Rough discretization that would cause the optimal solution achieved could be far from the true optimal solution
- It is deterministic, therefore it does not consider uncertainty in prices, pests, cost...

DP: disadvantages

- Lack of shadow prices$\rightarrow$ we are not able to evaluate how the objective function would change in case we modify the constraints in one unit
- States and stages should be carefully defined to avoid
  - Excessive discretization that would imply too high computing times without gaining a significative improvement in the optimal solution
  - Rough discretization that would cause the optimal solution achieved could be far from the true optimal solution
- It is deterministic, therefore it does not consider uncertainty in prices, pests, cost...

DP: disadvantages

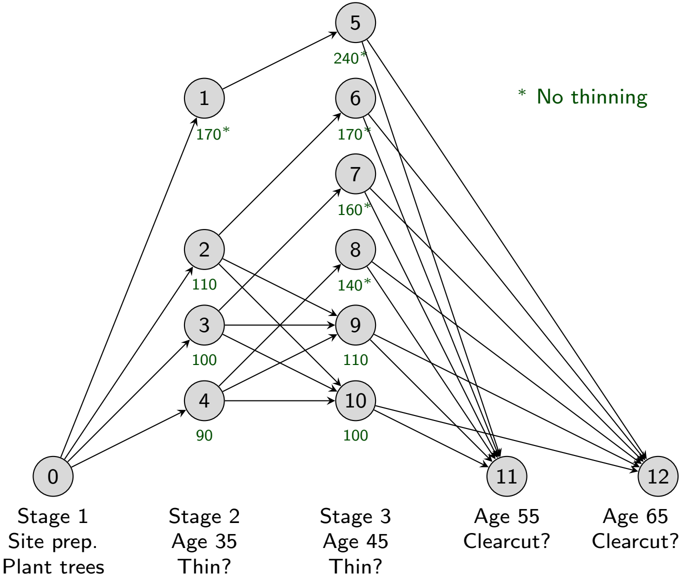
- Lack of shadow prices $\rightarrow$ we are not able to evaluate how the objective function would change in case we modify the constraints in one unit
- States and stages should be carefully defined to avoid
  - Excessive discretization that would imply too high computing times without gaining a significative improvement in the optimal solution
  - Rough discretization that would cause the optimal solution achieved could be far from the true optimal solution
- It is deterministic, therefore it does not consider uncertainty in prices, pests, cost...

Example 3: stand-level planning with **variable** rotation length

Example adapted from Bettinger et al. (2009, p. 116-118)

- The landowner of the previous example is now considering the possibility of harvesting when the stand is 65 years old, i.e. varying R between 55 and 65 years
- The aim is to evaluate the optimal path through the network, but in this case we will have two ending nodes: 55 and 65 years $\rightarrow$ the optimal path will also indicate in this case when it is optimal to harvest

Example 3: stand-level planning (variable R)



Example 3: stand-level planning (variable R)

Extension for clearcut at age 65

From-node	To-node	Volume harvested	Revenues	$r_{i \to j}$
5	12	50.625	20250	849.42
6	12	43.236	17294.4	725.44
7	12	38.471	15388.4	645.49
8	12	34.612	13844.8	580.74
9	12	27.851	11140.4	467.30
10	12	25.692	10276.8	431.08

Example 3: stand-level planning (variable R)

How we compare paths that imply different rotation lengths?

- **Land Expectation Value (LEV)**
- Convert the **Net Present Value (NPV)** to a future value at the end of each rotation

$$\mathrm{LEV} = \frac{\mathrm{NPV}\,(1+i)^R}{(1+i)^R - 1} = \frac{\mathrm{NPV}}{1 - \dfrac{1}{(1+i)^R}}$$

where

$$\mathrm{NPV} = \sum_{t=0}^{R} \frac{I_t - C_t}{(1+i)^t}$$

Example 3: stand-level planning (variable R)

How we compare paths that imply different rotation lengths?

- **Land Expectation Value (LEV)**
- Convert the **Net Present Value (NPV)** to a future value at the end of each rotation

$$\mathrm{LEV} = \frac{\mathrm{NPV}\,(1+i)^R}{(1+i)^R - 1} = \frac{\mathrm{NPV}}{1 - \dfrac{1}{(1+i)^R}}$$

where

$$\mathrm{NPV} = \sum_{t=0}^{R} \frac{I_t - C_t}{(1+i)^t}$$

Example 3: stand-level planning (variable R)

How we compare paths that imply different rotation lengths?

- **Land Expectation Value (LEV)**
- Convert the **Net Present Value (NPV)** to a future value at the end of each rotation

$$\mathrm{LEV} = \frac{\mathrm{NPV}\,(1+i)^R}{(1+i)^R - 1} = \frac{\mathrm{NPV}}{1 - \dfrac{1}{(1+i)^R}}$$

where

$$\mathrm{NPV} = \sum_{t=0}^{R} \frac{I_t - C_t}{(1+i)^t}$$

References I

Bellman, R. (1957). *Dynamic programming*. Princeton University Press, p. 340.

Bettinger, P, K Boston, J. Siry, and D. Grebner (2009). *Forest management and planning*. San Diego, CA: Academic Press.

Borges, J. G., L. Diaz-Balteiro, M. E. McDill, and L. C. Rodriguez (2014). *The Management of Industrial Forest Plantations*. Ed. by J. G. Borges, L. Diaz-Balteiro, M. E. McDill, and L. C. Rodriguez. Vol. 33. Managing Forest Ecosystems. Dordrecht: Springer, p. 543.

Brodie, J. D. and C. Kao (1979). "Optimizing thinning in Douglas-fir with three-descriptor dynamic programming to account for accelerated diameter growth". In: *For. Sci.* 25.4, pp. 665–672.