Risk management in forestry
- Possible solution approaches

Lars Relund Nielsen & Anders Ringgaard Kristensen

Dina Notat No. 111 · January 2005

Risk Management in Forestry - Possible Solution Approaches*

LARS RELUND NIELSEN† ANDERS RINGGAARD KRISTENSEN

Department of Large Animal Sciences
Royal Veterinary and Agricultural University
Grønnegårdsvej 2
DK-1870 Frederiksberg C
Denmark

January 13, 2005

Abstract

Risk management has received increased attention in the forest economics literature. Forest owners making harvesting decisions face many uncertain parameters such as price uncertainty, uncertainty about future growth and quality etc. The cost of ignoring these risk factors may be high.

This note presents a simple multi-level hierarchic Markov decision process modelling a forest stand. Three risk criteria are introduced, namely, the variance risk, the expected total consequence risk and the catastrophe avoidance risk criterion. Bicriterion solution procedures using directed hypergraphs are discussed.

Keywords: Risk modelling, Markov decision processes, stochastic dynamic programming, directed hypergraphs, hyperpaths.

*This research was supported by a grant from SNS - the Nordic Forest Research Co-operation Committee

†Corresponding author (lars@relund.dk)

Contents

1 Introduction

Risk management has received increased attention in the forest economics literature. However, with a few exceptions, the risks involved have not been subject to attention in the analyses.

Forest owners making harvesting decisions face many uncertain parameters such as price uncertainty, uncertainty about future growth and quality of retained stands and often forestry decisions about the management of the forest have a long time horizon. As a result, the size of the variation in consequences as a proportion of the decision-maker's wealth can be large and the cost of ignoring risk aversion may be high. Moreover, a forest owner's risk aversion may be important in the choice of an optimal rotation strategy.

Most analyses of optimal rotation in a stochastic setting are solved when uncertainties concerning long stand growth and quality, harvesting costs, planting costs etc. are taken into account. The problem is often formulated as a multistage decision process, and solved using a *Markov decision process* (*MDP*).

MDPs model sequential decision-making problems. At a specified point in time, a decision maker observes the state of a system and chooses an action. The action choice and the state produce two results: the decision maker incurs an immediate reward or cost, and the system evolves probabilistically to a new state at a subsequent discrete point in time. At this subsequent point in time, the decision maker faces a similar problem. The goal is to find an optimal *policy* of choosing actions (dependent on the observations of the state) which is minimal with respect to a certain criterion.

The majority of the work in the area of MDPs has focused on optimization criteria that are based on expected values of the rewards or costs, see e.g. Howard [6] and Puterman [16]. However, such risk-neutral approaches are not always applicable and expressive enough and risk sensitive criteria have to be considered.

A common problem with dynamic programming is "the curse of dimensionality". *Multi-level hierarchic Markov decision process* (*MLHMP*), see Kristensen [7], is a stochastic process that reduces the dimensionality difficulties. Moreover, the process is specially designed to solve dynamic decisions problems involving decisions with varying time horizon. The MLHMP approach has so far mainly been applied within animal production, but this approach also applies to other areas of agriculture and non-agriculture management.

In this paper we model a forest stand using an MLHMP. The forest stand owner's objective is in general to maximize the total expected reward. However, as pointed out above, risk management may also play an important role in which strategy/policy to choose. For instance, policies where the actual total reward obtained may deviate much from the total expected reward due to price changes, disasters or insect attacks will often be considered "risky". As a result the decision maker is not only interested in maximizing total expected reward but also in minimizing the risk measure. That is, we have a bicriterion optimization problem. In general it is not possible to find a single policy optimizing both objectives. Instead we are interested in finding a policy where the trade-off between reward and risk is acceptable among the set of so called *efficient policies* where the weight of the one criterion cannot be reduced without increasing the weight of the other criterion.

Note that the word risk may have many different meanings depending on how risk is modelled, e.g. we talk about risk of disasters, risk of price changes etc. Different risk criteria are pointed out in Section 4.

A hypergraph model is used to model the MLHMP. Directed hypergraphs are an extension of directed graphs and undirected hypergraphs introduced by Berge [2]. The concept of a hyperpath and a shortest hyperpath was introduced by Nguyen and Pallottino [9] and later the definition of a hyperpath in a directed hypergraph and a general formulation of the shortest hyperpath problem were given by Gallo, Longo, Pallottino, and Nguyen [5]. For a general overview on directed hypergraphs see Ausiello, Franciosa, and Frigioni [1].

Recently, the study of directed hypergraphs has become an important aspect in finding optimal strategies/paths in stochastic time-dependent networks, see Nielsen [10], Pretolani [15] and Nielsen, Andersen, and Pretolani [12]. Moreover, algorithms to solve bicriterion problems in stochastic time-dependent networks have been developed, see Nielsen [10], Nielsen, Andersen, and Pretolani [11].

As pointed out in a recent paper by Nielsen and Kristensen [13], by having a look on the hypergraph model for stochastic time-dependent networks it is apparent that, hypergraphs also can be used to model finite-horizon MDPs. Here an MDP can be modelled using a *state-expanded directed hypergraph* and the problem of finding the optimal policy under different optimality criteria can be formulated as a shortest hyperpath problem.

The paper is organized as follows. Notation for MLHMP is introduced in Section 2. The MLHMP model for the forest stand is given in Section 3. Section 4 considers different risk criteria. In Section 5 bicriterion solution techniques are discussed. Conclusions and directions for further research are pointed out in Section 6. Finally, Appendix A describes how to build the state-expanded hypergraph.

2 Multi-level hierarchic Markov decision processes - some definitions

We consider a multi-level hierarchic Markov process which is an infinite-horizon Markov decision process with parameters defined in a special way, but nevertheless in accordance with all usual rules and conditions relating to such processes. The basic idea is to expand stages of the processes to so-called child processes, which again may be expanded to further child processes. Only one process in the structure is not the child of a parent process denoted the *founder process* ρ^0. The index 0 indicate that the process has no ancestral processes. Since ρ^0 is running over a infinite number of stages, we assume that all stages have identical state and action spaces. Each stage of ρ^0 is represented by a child process with a planning horizon equal to the length of the state of ρ^0. Often several alternative child processes are available (depending on the action taken and the state of the parent level). A child process may further expand some stages to further child processes.

Consider a finite-horizon child process ρ^l with l ancestors. At stage n the system occupies a *state*. At stage n the set of finite system states is S_n^l. Given the decision maker observes state $s \in S_n^l$, he may choose an *action* a from the set of finite allowable actions $A_{s,n}^l$, generating *cost* $c_n^l(s,a)$ (a reward if negative). Moreover, let $0 < \lambda_n^l(s,a) \leq 1$ denote the corresponding discount factor and let $p_n^l(\cdot \mid s,a)$ denote the *probability distribution* or *transition probabilities* of obtaining states $s' \in S_{n+1}^l$ at stage $n+1$.

Assume that ρ^l have N decision epochs. Since no decision is made at the end of stage $N-1$, the cost at this time point is a function of the state $s \in S_N^l$ denoted $c_N^l(s, a_N)$ which is often referred to as the *salvage cost* or *scrap cost*. Here a_N denotes a deterministic (dummy) action.

Note that we may use the same notation for the founder process as defined above. Furthermore, since the founder process runs over an infinite time-horizon we may drop index n.

A process ρ^l with l ancestors is uniquely identified by the stage, actions and states of its ancestors necessary to start the process. Often we assume that these stages, actions and states are know implicitly and just write ρ^l; however, we may write ρ^l explicitly using the following notation

$$\rho^l = \left(\left(s^0, a^0\right), \left(n^1, s^1, a^1\right), ..., \left(n^{l-1}, s^{l-1}, a^{l-1}\right)\right)$$

A *policy* or *strategy* δ is a function which specifies the action to choose for all (child) processes given its stage and state. That is, a policy provide the decision maker with a management plan.

2.1 Finding the optimal policy

Note that an MLHMP may be considered as a stochastic process $\{X_i\}_{i=1,...,\infty}$ where random variable X_i denote the state of the process at decision epoch i. Moreover, we define random variable l_i and n_i used to identify the corresponding child process and its stage at decision epoch i.

In general the goal of the decision maker is to find an optimal policy according to a certain criterion which is a function of X_i, $i = 1, ..., \infty$.

Definition 1 Given policy δ let random variable TDC_s^δ denote the *total discounted economic cost* when the initial state is s. That is,

$$TDC_s^\delta = \sum_{i=1}^{\infty} \left(\prod_{j=1}^{i-1} \lambda_{n_j}^{l_j}\left(X_j, \delta\left(X_j\right)\right) \right) c_{n_i}^{l_i}\left(X_i, \delta\left(X_i\right)\right), \quad X_1 = s \tag{1}$$

For a risk neutral forest owner operating under no risk, a well-known criterion is to find the policy that minimize the expected total discounted economic cost

$$ETDC_s^\delta = \mathbb{E}\left(TDC_s^\delta\right) \tag{2}$$

```
procedure policy_ite()
    choose a policy δ;
    optimal := false
    while (not optimal) do
        given δ find the unique solution to equations (3);
        determine policy δ′ using equations (4);
        if (δ = δ′) then optimal := true;
        else δ := δ′;
    end while
    return δ′;
end procedure
```

Figure 1: Policy iteration procedure (ETDC).

In the following we summarize how to find $f_s^\delta = ETDC_s^\delta$ using policy iteration and value iteration.

Given an MDP it is well-known that f_s^δ, $s \in S^0$ can be found by solving the following equations

$$f_s^\delta = c^0 (s, \delta (s)) + \lambda^0 (s, \delta (s)) \sum_{s' \in S^0} p^0 \left(s' \mid s, \delta (s)\right) f_{s'}^\delta \tag{3}$$

Furthermore, policy δ can be improved by for each $s \in S^0$ selecting action $a' \in A_s^0$ minimizing

$$a'_s = \arg \min_{a \in A_s^0} \left\{ c^0 (s, a) + \lambda^0 (s, \delta (s)) \sum_{s' \in S^0} p^0 \left(s' \mid s, \delta (s)\right) f_{s'}^\delta \right\} \tag{4}$$

Let δ' denote the policy with $\delta' (s) = a'_s$, $s \in S^0$, then it is easy to see that δ' is an improved policy. Moreover, if $\delta = \delta'$ then δ is optimal. The policy iteration procedure, shown in Figure 1, repeats the above operations until an optimal policy are found.

If we consider a MLHMP instead the policy iteration procedure only have to be changed slightly. In this case $c^0 (s, \delta (s))$ and $\lambda^0 (s, \delta (s))$ are the expected cost and discount rate of the child processes when using policy δ. Moreover, a'_s in (4) is the optimal actions for the whole child process. These values can be calculated using value iteration, see Kristensen and Jørgensen [8] or a shortest hyperpath algorithm as pointed out in Appendix A.

3 The forest stand model

We consider a single forest stand over an infinite time-horizon. We start by considering the parameters and variables under consideration.

3.1 Model parameters and variables

Forest management involves adoption to several variables and parameters from planting to the final felling. We consider a simple model with the following assumptions (unrealistic model assumptions are pointed out in Section 6).

Species Given a forest stand we assume that the species is fixed.

Site qualities The production functions of forest stands differ according to their site quality, which gives the production possibility to a certain location. Different parameters are used to determine the site quality, e.g. tree height growth is almost perfectly correlated with the site quality. For instance, the height of dominating trees at 40 years of age (i.e. 40 years from the trees reached their breast height - 1.3 m) is used in Norway. In this paper we assume that at a given time the site quality is known with (approximate) certainty. As a result site quality is not included in the state space of the model.

Reforestation/planting The scope of reforestation is to start the growth of a new stand after final felling as soon as possible. Planting is especially important where the site quality is good and future competition from other vegetation is high. On dryer soil (for pine) natural regeneration is more (cost) efficient. Some trees are left during the final felling in order to supply the site with seeds enough for a new stand to growth. In order to improve such regeneration, mechanical soil improvement (scarification) is often conducted.

Silviculture The scope of silviculture is to remove competitive vegetation and trees in order to improve growth conditions. This activity involves only costs, but is supposed to improve tree quality and diameter growth. Silviculture activities are conducted from stand age 5 to 30, depending on the site quality. A silviculture strategy could be to do silviculture every second year.

Thinnings Thinnings involve reducing the stem number in order to improve diameter growth for the rest of the trees. Thinnings are conducted at age 30 to 60, and there can be several thinnings (or none at all), and are supposed to give some revenues from timber sales (even though the costs associated can exceed the income). Thinnings will in the short run decrease the volume yield in a specific stand (a shift downwards in the production function) but since it improves the growth condition the stand will recover this yield over time (thus the production function also gets steeper).

Timber price The timber price is defined as a random autoregressive time series, i.e. if the current timber price is $\hat{p}$, then the price t time periods after is

$$p_t = \exp\left(-\alpha t\right)\left(\hat{p} - \mu\right) + \varepsilon$$

where $\varepsilon \sim N\left(\mu, \sigma^2\right)$ with μ equal to the expected timber price and σ^2 the variance. The timber price is illustrated in Figure 2. It is assumed that the

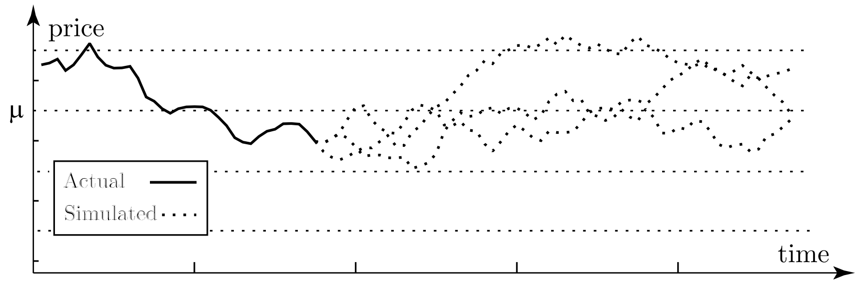

Figure 2: Timber prices for the random autoregressive process.

forest owner does not have any effect on the timber price. Moreover, note that the forest stand cannot be harvested before year 60, i.e. price monitoring is only needed from year 60 until felling where it must be predicted 5 years ahead. Since there are at least 60 years between each felling, we assume that the price between rotations are independent.

Catastrophe risk By using a random process to model the timber price we introduce price risks into the model. Risk of catastrophes (e.g. fire) are introduced into the model differently inspired by studies on catastrophe avoidance for hazardous materials route planning, see Erkut and Ingolfsson [4].

Given the current state of the system we assume that the probability of a catastrophe and e.g. the number of trees affected can be calculated. Further details will be pointed out in Section 4.

3.2 MLHMP formulation

An MLHMP with 2 levels is used to model the forest stand

Founder process (level 0) A stage of the founder process is one rotation, i.e. from felling to felling.

State space Since we assume that the site quality is a known function of e.g. the tree height, we define a single dummy state.

Action space The choice between planting versus natural regeneration.

Child level 1 The child process at level 1 begins at planting/reforestation. The length is equal to the duration of a rotation. From year 0 to year 30 silviculture is done according to a selected silviculture strategy (one stage). It is assumed that thinning is considered every 10'th year from year 30 to year 60, i.e. there will, as a maximum, be 3 thinnings. For the remaining years felling is considered each 5. year. Moreover, we assume that felling is conducted before year 100.

Founder process (level 0):

- *Horizon*: Infinite.
- *Stage length*: From felling to felling (one rotation).
- *States*: $s \in$\{*dummy*\} (stage independent).
- *Actions*: $a \in$\{*plant,natural regeneration*\}.
- **Child process (level 1)**:
  - *Given*: reforestation action of the founder level.
  - *Horizon*: Finite (12 stages).
  - *Stage 1*:
    - *Stage length*: 30 years.
    - *States*: $s \in$\{*dummy*\}.
      - *Actions*: set of different silviculture strategies.
  - *Stage 2-4*:
    - *Stage length*: 10 years.
    - *States*: $s \in TV$.
      - *Actions*: $a \in$\{*thin,don't thin*\}.
  - *Stage 5-12*:
    - *Stage length*: 5 years.
    - *States*: $s \in TV \times P$.
      - *Actions*: $a \in$\{*fell,don't fell*\}.

Figure 3: The MLHMP model for a forest stand.

Stage 1 The first stage covers the first 30 years. Only one dummy state is defined. The action space is defined by the alternative silviculture strategies.

Stages 2-4 Each stage has duration of 10 years. The state $s \in TV$ is the present timber volume of the stand, where $TV = \{tv_1, ..., tv_q\}$ denote the discretized set of possible timber volumes which can be obtained during a rotation. The action space is defined as thinning versus no thinning.

Stages 8-12 Each stage has duration of 5 years. The state $s \in TV \times P$ is defined as the present timber volume of the stand and the price index, where $P = \{p_1, ..., p_l\}$ denote the set of discretized possible timber price levels. The action space is defined as felling versus no felling.

A compact representation of the MLHMP model is given in Figure 3.

4 Risk criteria for the forest stand model

Forestry is certainly exposed to risk: rot, insect attacks, wind throws, other production risk, price risk etc. Moreover, the owners are most likely to have different degrees of risk aversion. Finally, there exists different ways to quantify risk. In the following sections we introduce risk criteria which may be relevant for the forest owner.

4.1 The variance risk criterion

One way to consider risk is to consider the *variance risk criterion* (*VRC*) defined as the variance of the total discounted economic cost (1). i.e.

$$VRC_s^\delta = \mathbb{V}\left(TDC_s^\delta\right)$$

given policy δ and initial state s. The variance is the most common measure of how far the tails of a distribution extend, so it seems natural to use it as a risk measure. Note that for the forest stand model the VRC only provide us with risk information about the state variables included in the model, e.g. timber price. As concerns other risk variables, such as fire or insect attacks, the criterion only measures the effect of decreased timber volume as a consequence of those events.

Recursive equations for calculating VRC_s^δ are given in Nielsen and Kristensen [14]. Unfortunately, in the same paper it was shown that Bellmans principle of optimality does not hold for the VRC. That is, we cannot find the policy δ with minimal VRC_s^δ using standard dynamic programming methods.

4.2 The expected total consequence risk criterion

Due to the unfortunate properties for the VRC we choose to model risk of catastrophes differently inspired by studies on catastrophe avoidance for hazardous materials route planning, see Erkut and Ingolfsson [4].

Assume that we consider *catastrophe* measures $r \in R$ such as rot, insect attacks, wind throws etc and let the *consequence* of a catastrophe r denote a common measure for all catastrophes in R, e.g. number of trees affected or the cost of the catastrophe.

Consider process ρ^l. Given the decision maker observes state s at stage n and chooses action a the probability of catastrophe $r \in R$ is $\theta_n^l(s,a,r)$ and the consequence of the catastrophe is estimated to be $\kappa_n^l(s,a,r)$. Then the expected consequence of the catastrophes when the decision maker observes state s and chooses action a is

$$\eta_n^l(s,a) = \sum_{r\in R} \theta_n^l(s,a,r)\,\kappa_n^l(s,a,r)$$

Using value $\eta_n^l(s,a)$ we can define a random variable TCR_s^δ denoting the total consequence risk similar to the total discounted cost in (1).

$$TCR_s^\delta = \sum_{i=1}^{\infty} \eta_{n_i}^{l_i}\left(X_i, \delta\left(X_i\right)\right), \quad X_1 = s \tag{5}$$

and rank policies using the *expected total consequence risk criterion* (*ETCR criterion*)

$$ETCR_s^\delta = \mathbb{E}\left(TCR_s^\delta\right) \tag{6}$$

Note that we do not model catastrophes using the state space but model them implicit using $\eta_n^l(s,a)$. Hence the model does not provide us with any information about what happens when a catastrophe occur.

Since the ETCR criterion from a mathematical point of view is equivalent to the ETDC criterion (2), the policy minimizing $ETCR_s^\delta$ can be found using dynamic programming and policy iteration.

4.3 The catastrophe avoidance risk criterion

The ETCR criterion provided us with an easy way of ranking policies according to expected risk. However, the ETCR criterion is risk neutral in the sense that a decision maker is indifferent between two policies as long as their expected risk (6) is the same. Often this is not the case when dealing with catastrophes having high consequence but occurring with low probability. Here the human decision maker may exhibit risk aversion trying to avoid catastrophes with high consequences even though they occur with low probability. This kind of risk aversion can be modelled using the *catastrophe avoidance risk criterion* (*CAR criterion*)

$$CAR_s^\delta(r) = \max_{\substack{i=1,\dots,\infty \\ s_i \in \{s : P(X_i = s) > 0\}}} \left\{\kappa_{n_i}^{l_i}\left(s_i, \delta\left(s_i\right), r\right)\right\}$$

which simply specify the maximum possible consequence of catastrophe r given policy δ. We are not interested in finding the policy with minimal $CAR_s^\delta(r)$ (there may the many). Instead we assume that the decision maker put an upper bound ub_{CAR} on the maximum consequence acceptable, i.e. all policies δ under consideration most satisfy

$$CAR_s^\delta(r) < ub_{CAR}, \quad \forall r \in R \tag{7}$$

Of cause using a too low upper bound may result in that no management strategy exists.

5 Optimization model and possible solutions methods

In this section we consider optimization models for finding a good set of policies from where the decision maker can choose the management plan he finds best. We start by presenting the optimization model.

5.1 Optimization model formulation

In the previous sections a range of criteria which can be used to rank policies have been presented. The goal is the find a policy minimizing point

$$\hat{P}^{\delta} = \left(ETDC_s^{\delta}, VRC_s^{\delta}, ETCR_s^{\delta}\right)$$

under constraint (7). In general this is not possible to find a single minimal point $\hat{P}^{\delta}$ since given policy δ one criterion might be high while another criterion is low. Instead we are interested in finding *efficient policies*, i.e. policies where the weight of one criterion cannot be reduced without increasing the weight of another criterion. If δ is an efficient policy we call $\hat{P}^{\delta}$ a *nondominated point*. We get the following multi-criteria optimization problem

$$\begin{array}{ll} \min\limits_{\delta} & \left(ETDC_s^{\delta}, ETCR_s^{\delta}, VRC_s^{\delta}\right) \\ st. & CAR_s^{\delta}(r) < ub_{CAR}, \quad \forall r \in R \end{array} \tag{8}$$

Unfortunately, we cannot find the set of efficient policies to (8) due to the nonadditive properties of the variance risk criterion (see Section 4.1). Instead we will find an approximation of the set of efficient policies to (8) by solving the bicriterion optimization problem

$$\begin{array}{ll} \min\limits_{\delta} & \left(ETDC_s^{\delta}, ETCR_s^{\delta}\right) \\ st. & CAR_s^{\delta}(r) < ub_{CAR}, \quad \forall r \in R \end{array} \tag{9}$$

That is, the variance criterion is not considered directly. However, during the solution procedure for solving (9), we for each policy δ considered calculate VRC_s^{δ} and for efficient policies store point $\hat{P}^{\delta}$ instead of point $\left(ETDC_s^{\delta}, ETCR_s^{\delta}\right)$.

5.2 The bicriterion optimization problem

Consider bicriterion optimization problem (9). First note that each policy δ corresponds to a point $P^{\delta} = \left(ETDC_s^{\delta}, ETCR_s^{\delta}\right)$[1] in the *criterion space* and the goal is to find all efficient points. This provide us with a set of points from which the decision maker may pick the policy which fits best according to his risk adversity.

An example of points in the criterion space is shown in Figure 4 (right side) where $P_1^{\delta} = ETDC_s^{\delta}$ and $P_2^{\delta} = ETCR_s^{\delta}$. The points P^1, P^4 and P^5 are nondominated points on the border of the criterion space also denoted *extreme points*.

Note that the domain of (9) is discrete. As a result efficient points are not always only located on the border. Instead the extreme points define a set of triangles in which further nondominated points, such as P^8, may be found. Points outside the triangles are always dominated by one of the extreme points. In the following we shortly describe the two-phase approach for solving (9).

[1]We in the following only consider point P^{δ} instead of $\hat{P}^{\delta}$, however the variance VRC_s^{δ} can be stored as pointed out above.

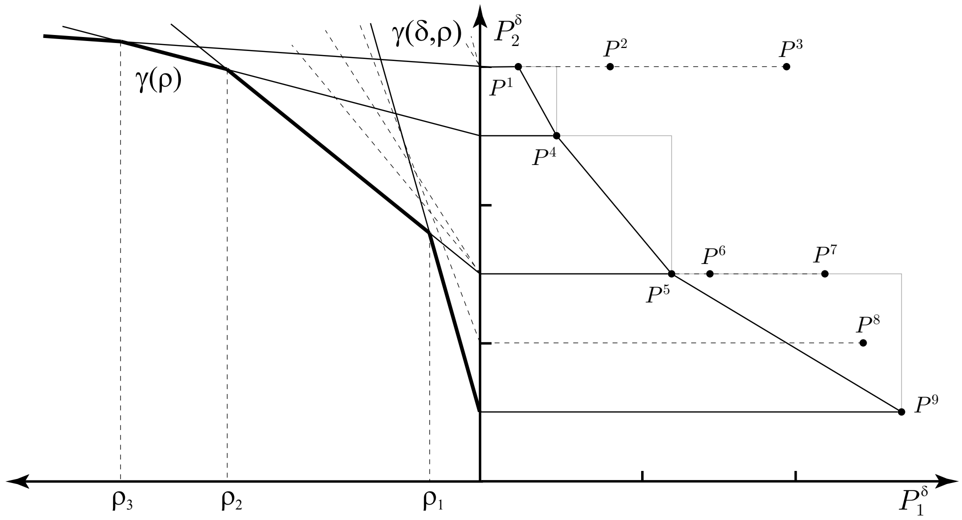

Figure 4: The criterion space (to the right) and its corresponding parametric space (to the left).

5.3 Solution method

The two-phase approach is used to solve (9) which is a general method for solving bicriterion combinatorial problems. As the name suggests, the two-phase approach splits the search of nondominated points into two phases. In phase one the extreme points are found. These extreme points define the triangles in which further nondominated points may be found. Phase two proceeds to search the triangles one by one. This is done parametrically. The approach is illustrated by the set of criterion points shown in Figure 4 .

Let γ denote the *parametric weight* of a policy δ.

$$\gamma\left(\delta,\rho\right)=P_1^\delta\rho+P_2^\delta \tag{10}$$

Given $\rho>0$, a *minimum parametric weight policy* $\delta\left(\rho\right)$, is a policy with minimal parametric weight (10) denoted $\gamma\left(\rho\right)$ (how to find $\delta\left(\rho\right)$ is described in Section 5.6).

5.4 Finding the extreme points - phase one

The criterion space and its corresponding parametric space are shown in Figure 4. For a given policy δ, each point P^δ corresponds to a line with slope P_1^δ and intersection P_2^δ in the parametric space. Given a fixed $\rho>0$, we have a line in the parametric space defined by some δ which minimizes $\gamma(\delta,\rho)$, see Figure 4. Moreover, the lower envelope of the lines in the parametric space defines $\gamma\left(\rho\right)$, which

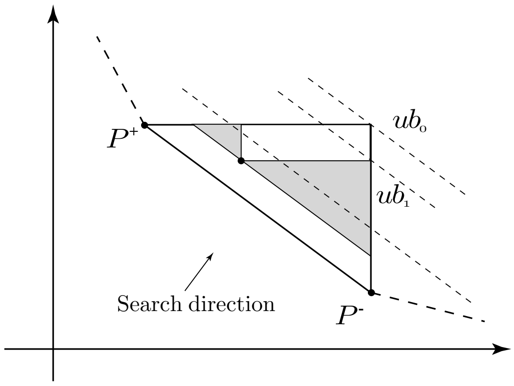

Figure 5: A triangle defined by P^+ and P^-.

is a non-decreasing piecewise linear function with break points ρ_i. Note that each breakpoint ρ_i corresponds to a value of ρ where two adjacent extreme points have the same minimal parametric weight. For instance for $\rho = \rho_2$ we have that P^4 and P^5 have the same minimal parametric weight, i.e. finding a minimal parametric weight policy $\delta(\rho_2)$ corresponds to searching in the direction of the normal between P^4 and P^5.

Obviously each piece of $\gamma(\rho)$ defines an extreme point. Hence the extreme points can be found by finding a minimal parametric weight policy $\delta(\rho)$ for different values of ρ. This is done by using a NISE[2] algorithm (see Cohen [3]) which first finds the *upper/left* and the *lower/right* point (P^1 and P^9 in Figure 4). The upper/left point is the nondominated point which has minimum weight using the second criterion when weight one is fixed to its minimum weight. Similarly, the lower/right point is the nondominated point which has minimum weight using the first criterion when weight two is fixed to its minimum weight. Given two nondominated points P^+ and P^-, we now calculate the search direction ρ defined by the slope of the line between the points and find the minimum parametric weight policy $\delta(\rho)$. If $P^{\delta(\rho)}$ corresponds to a new extreme point the points $P^+, P^{\delta(\rho)}$ and P^- define two new search directions which can be searched similarly. This step is repeated until no new extreme points are found. Since the number of policies are finite, we have that the number of lines defining $\gamma(\rho)$ in the parametric space is finite and hence the first phase will stop in a finite number of steps.

5.5 Finding points inside the triangles - phase two

Due to there may exist nondominated points inside the triangles, it is not in general possible to find all nondominated points during the first phase. This can be seen in Figure 4 where nondominated point P^8 cannot be found by the first phase since it corresponds to a dashed line lying above $\gamma(\rho)$. Points like P^8 are found in phase two which searches each triangle defined by the set of extreme points found in

[2]Non-inferior set estimation.

phase one. Consider the triangle defined by the extreme nondominated points P^+ and P^- (see Figure 5). The second phase searches each triangle using a K best policy procedure in the direction ρ defined by the slope between the two points defining the triangle. The procedure stops when an upper bound has been reached. At start the upper bound is $ub_0 = P_1^- \rho + P_2^+$. However, when a new unsupported nondominated point is found inside the triangle, the upper bound is updated to ub_1 (see Figure 5). Note that the procedure may find points outside the triangle because all points with parametric weight below ub_1 are found.

5.6 Complexity of the solution method

Since the number of efficient points may grow exponential with the size of the problem the complexity of a bicriterion optimization problem is in general $\mathcal{NP}$-hard. Moreover, the two-phase approach require that given $\rho > 0,$ a minimum parametric weight policy $\delta(\rho)$, with minimal parametric weight (10), can be found. This is done using the state-expanded directed hypergraph of the MLHMP for the forest stand. Here each hyperarc represent a specific stage, state and action. Now by assigning weight

$$c_n^l(s,a)\,\rho + \eta_n^l(s,a) \tag{11}$$

to the corresponding hyperarc, policy $\delta(\rho)$ can be found by finding a shortest hyperpath in the state-expanded hypergraph.

Bicriterion optimization techniques using the two-phase approach for directed hypergraphs modelling stochastic time-dependent networks have been addressed by Nielsen [10]. Similar, we can solve problem (9) using the two-phase approach on the state-expanded hypergraph. Here the constraints in (9) are kept simply by removing hyperarcs in the hypergraph. A detailed description on how to build the state expanded hypergraph is given in Appendix A.

Solution times for problem (9) are not known. However, since the structure of the state-expanded hypergraph resembles the structure of the hypergraph representing the stochastic time-dependent network, the solution time may be quite the same. In Nielsen [10] a good approximation of the efficient set could be found in reasonable time.

Finally, note also that efficient policies may be found in interaction with the decision maker. For instance the extreme points on the border, defining the triangles in phase two, may be found. Next, the decision maker may choose the triangle to search for further efficient policies.

5.7 Application of the solution method to the forest stand model

It is important in the optimization procedure to treat the forest management problem in a replacement framework. If the reward from future rotations is ignored, the optimal felling time determined will inevitably be too high. Thus, optimization must basically be performed under infinite planning horizon using the policy iteration technique as described in Section 2.1.

When it comes to risk assessment, on the other hand, it may be more realistic to evaluate the defined risk criteria inside a rotation, since the duration of a rotation is at least 60 years. It is therefore neither realistic that the forest owner is the same nor that the degree of risk aversion is the same during more than one rotation.

The following procedure for application of the solution method to the forest stand model is therefore suggested:

1. An optimal policy δ' according to the $ETDC_s^\delta$ criterion under infinite horizon is calculated.

2. In each of the child processes (Level 1) representing a rotation, the solution method described in the previous sections is applied. The $ETDC_s^\delta$ criterion is evaluated under infinite planning horizon assuming that the policy δ' is followed from *next* rotation, but during the present rotation any policy δ may be used. The VRC_s^δ, $ETCR_s\delta$ and $CAR_s^\delta(r)$ criteria are evaluated under the time horizon of the rotation.

6 Conclusions

In this paper we have presented a model for risk management of a forest stand. The stand is represented using a MLHMP which can be modelled using a state-expanded hypergraph. For a risk-neutral decision maker the overall goal is to minimize the expected total discounted cost. However, due to risks in forestry we introduce three risk criteria.

The variance risk criterion is used to provide us with risk information about the state variables included in the model, e.g. timber price and timber volume. It does not provide us with any information about other risks such as fire, insect attacks etc. Instead catastrophes are introduced implicitly into the model and the expected total consequence risk criterion is used to provide us with risk information about the risk of catastrophes. The expected total consequence risk criterion is risk neutral in the sense that a decision maker is indifferent between two policies as long as there expected risk is the same. Often this is not the case when dealing with catastrophes having high consequence but occurring with low probability. As a result we introduce the catastrophe avoidance risk criterion which is not optimized. However, the decision maker may put an upper bound on this criterion.

Risk aversion is introduced into the model in two different ways. First, by putting an upper bound on the catastrophe avoidance risk criterion, the decision maker sets a limit on how big a catastrophe he is willing to risk. Of course setting a too low limit will result in that no management strategy exists. Second, solving the bicriterion optimization problem (9) give us a set of efficient policies from which the decision maker may choose the policy which fits best according to his risk aversion.

The model in this paper is very simple and probably unrealistic. Some unrealistic assumptions are pointed out below.

- We assume that silviculture is done according to a silviculture strategy lasting 30 years, i.e no decisions are taken during the 30 years. A more realistic model would probably split this time period into smaller parts where the silviculture strategy may be revised. Moreover, intensity of thinning may vary, i.e. more than two actions at stage 2-4 at child level 1.
- In a more realistic model the actions "plant" and "natural regeneration" at the founder level could be replaced with alternative intensities of planting.
- We assume that, given the time from planting, the site quality is known. If the site quality should be learned from data, a second child level could be relevant for observation.
- In the current model we cannot harvest a stand before year 60 this might not always be realistic.
- The way timber prices are modelled are properly not correct. Furthermore if there is high variations in the prices of a tree bought for planting this price should be modelled similar to the timber price.
- We do not consider the climate at all. An increase in temperature may result in that other species have to be planted.

Although the model is simple it provides us with a good starting point for further research.

References

[1] G. Ausiello, P.G. Franciosa, and D. Frigioni. Directed hypergraphs: Problems, algorithmic results, and a novel decremental approach. *Lecture Notes in Computer Science*, 2202:312–328, 2001.

[2] C. Berge. *Graphs and hypergraphs*. North-Holland, 1973.

[3] J. Cohen. *Multiobjective Programming and Planning*. Academic Press, New York, 1978.

[4] E. Erkut and A. Ingolfsson. Catastrophe avoidance models for hazardous materials route planning. *Transportation Science*, 34(2):165–179, 2000.

[5] G. Gallo, G. Longo, S. Pallottino, and S. Nguyen. Directed hypergraphs and applications. *Discrete Applied Mathematics*, 42:177–201, 1993.

[6] R.A. Howard. *Dynamic Programming and Markov processes*. Cambridge, Massachusetts: The M.I.T. Press, 1960.

[7] A.R. Kristensen. Hierarchic Markov processes and their applications in replacement models. *European Journal of Operational Research*, 35:207–215, 1988.

[8] A.R. Kristensen and E. Jørgensen. Multi-level hierarchic Markov processes as a framework for herd management support. *Annals of Operations Research*, 94(1):69–90, 2000.

[9] S. Nguyen and S. Pallottino. Hyperpaths and shortest hyperpaths. In *Combinatorial optimization (Como, 1986)*, volume 1403 of *Lecture Notes in Math*, pages 258–271. Springer, 1989.

[10] L.R. Nielsen. *Route Choice in Stochastic Time-Dependent Networks*. PhD thesis, Department of Operations Research, University of Aarhus, 2004.

[11] L.R. Nielsen, K.A. Andersen, and D. Pretolani. Bicriterion shortest hyperpaths in random time-dependent networks. *IMA Journal of Management Mathematics*, 14(3):271–303, 2003.

[12] L.R. Nielsen, K.A. Andersen, and D. Pretolani. Finding the K shortest hyperpaths. *To appear in Computers & Operations Research*, 2004.

[13] L.R. Nielsen and A.R. Kristensen. Finding the K best policies in finite-horizon Markov decision processes. Submitted, August 2004.

[14] L.R. Nielsen and A.R. Kristensen. Risk aversion in Markov decision processes. In *Proceedings of the European Workshop for Decision Problems in Agriculture and Natural Resources (EWDA-04)*, Silsoe Research Institute, Silsoe, England, 27–28 September 2004.

[15] D. Pretolani. A directed hypergraph model for random time-dependent shortest paths. *European Journal of Operational Research*, 123:315–324, 2000.

[16] M.L. Puterman. *Markov Decision Processes*. Wiley Series in Probability and Mathematical Statistics. Wiley-Interscience, 1994.

Appendix

A Modelling the MLHMP using a directed hypergraph

In this appendix we present results on how to build the state-expanded hypergraph representing the MLHMP. We start by giving some formal definitions about directed hypergraphs

A.1 Directed hypergraphs

A *directed hypergraph* is a pair $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = (v_1, ..., v_{|\mathcal{V}|})$ is the set of *nodes*, and $\mathcal{E} = (e_1, ..., e_{|\mathcal{E}|})$ is the set of *hyperarcs*. A hyperarc $e \in \mathcal{E}$ is a pair $e = (T(e), h(e))$, where $T(e) \subset \mathcal{V}$ denotes the set of *tail* nodes and $h(e) \in \mathcal{V} \setminus T(e)$ denotes the *head* node. Note that a hyperarc has exactly one node in the head, and possibly several nodes in the tail. We denote by

$$FS(v) = \{e \in \mathcal{E} \mid v \in T(e)\}, \quad BS(v) = \{e \in \mathcal{E} \mid v = h(e)\}$$

the *forward star* and the *backward star* of node v, respectively. A directed hypergraph $\tilde{\mathcal{H}} = (\tilde{\mathcal{V}}, \tilde{\mathcal{E}})$ is a *subhypergraph* of $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, if $\tilde{\mathcal{V}} \subseteq \mathcal{V}$ and $\tilde{\mathcal{E}} \subseteq \mathcal{E}$. A subhypergraph is *proper* if at least one of the inclusions is strict.

Definition 2 A hyperpath $\pi_{ot} = (\mathcal{V}_\pi, \mathcal{E}_\pi)$ from *origin* o to *target* t, is a subhypergraph of $\mathcal{H}$ satisfying that, if $t = o$, then $\mathcal{E}_\pi = \varnothing$; otherwise the $q \geq 1$ hyperarcs in $\mathcal{E}_\pi$ can be ordered in a sequence $(e_1, ..., e_q)$ such that

1. $t = h(e_q)$.
2. $T(e_i) \subseteq \{o\} \cup \{h(e_1), ..., h(e_{i-1})\}, \quad \forall e_i \in \mathcal{E}_\pi$.
3. No proper subhypergraph of π_{ot} is an o-t hyperpath.

Condition 3 implies that, for each $u \in \mathcal{V}_\pi \setminus \{o\}$, there exists a unique hyperarc $e \in \mathcal{E}_\pi$, such that $h(e) = u$. We denote hyperarc e as the *predecessor* of u in π_{ot}. As a result π_{ot} can be described by a *predecessor function* $\mathfrak{g} : \mathcal{V}_\pi \setminus \{o\} \to \mathcal{E}_\pi$; $\mathfrak{g}(u)$ is the unique hyperarc in π_{ot} which has node u as the head.

The weighting function of a hyperpath is defined as follows. Assume that each hyperarc e is assigned a nonnegative real weight vector $w(e) = (w_1(e), ..., w_K(e))$. Given hyperpath π defined by $\mathfrak{g}$, a weighting function W is a node function assigning real weights $W(u)$ to all nodes in $\mathcal{H}$. The weight of π is $W(t)$ (or $W(\pi)$). We shall restrict ourselves to *additive weighting functions* introduced by Gallo et al. [5], defined by the recursive equations:

$$W(v) = \begin{cases} 0 & v = o \\ \mathfrak{l}(w(\mathfrak{g}(v))) + \mathfrak{f}(\mathfrak{g}(v)) & v \in \mathcal{V}_\pi \setminus \{o\} \end{cases} \tag{12}$$

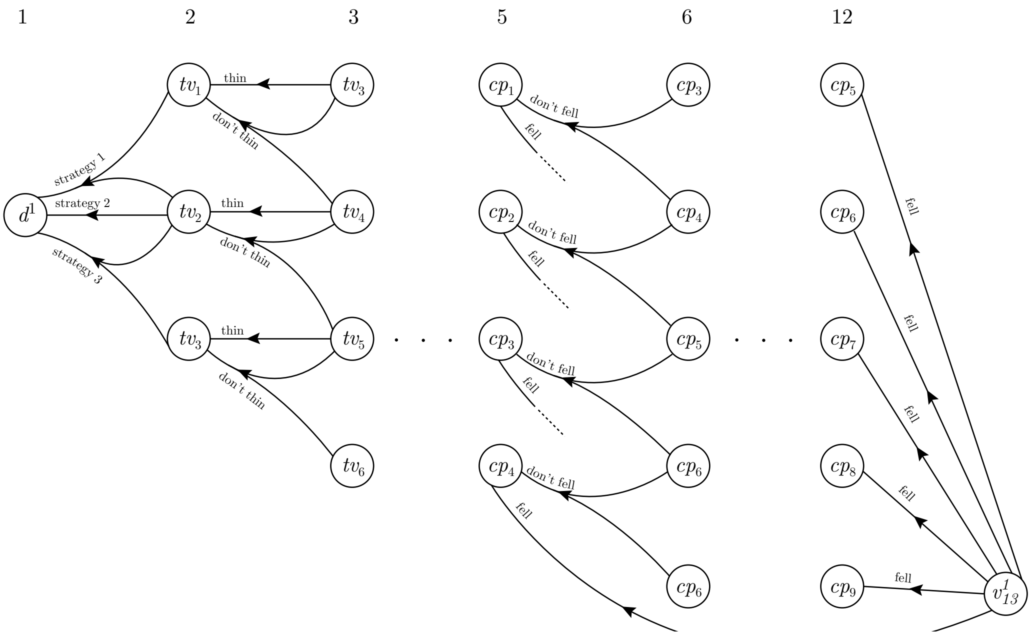

Figure 6: The finite-horizon child process $\rho^1\left(a^0\right)$ at level 1 given action a^0 for the founder process.

Here $\mathfrak{l}\left(\cdot\right)$ denote a non-decreasing function of $w\left(e\right)$ and $\mathfrak{f}(\cdot)$ a non-decreasing function of the weights in the nodes of $T(e)$.

Finding a shortest hyperpath can be viewed as a natural generalization of the shortest path problem and consists in finding the minimum weight for all nodes in $\mathcal{H}$. If $\mathcal{H}$ is acyclic (which is the case here) and the weighting function is additive a fast polynomial algorithm exist (see [5]).

A.2 Building the state-expanded hypergraph

We illustrate how to build the state-expanded hypergraph for the MLHMP by considering the MLHMP model for the forest stand given in Section 3. First, note that an example on how to build a state-expanded hypergraph for a finite-horizon MDP is given in Nielsen and Kristensen [14] and Nielsen and Kristensen [13]. Moreover, given the action chosen for the founder process, the child process is a finite-horizon MDP. That is, each child process $\rho^1\left(a^0\right)$ with $N = 12$ stages is modelled using a state-expanded hypergraph with nodes and hyperarcs defined as follows

$$\mathcal{V}^1 = \left\{v^1_{s,n} \mid n = 1, ..., N, s \in S^1_n\right\} \cup \left\{v^1_{N+1}\right\}$$

$$\mathcal{E}^1 = \left\{e^1_{a,s,n} \mid n = 1, ..., N-1, s \in S^1_n, a \in A^1_{s,n}\right\} \cup \left\{e^1_{s,N} \mid s \in S^1_N\right\}$$

with

$$e_{a,s,n}^{1}=\left(\left\{v_{s',n+1}^{1}\mid s'\in S_{n+1}^{1},\,p_{n}^{1}\left(s'\mid s,a\right)>0\right\},v_{s,n}^{1}\right),\quad e_{s,N}^{1}=\left(\left\{v_{N+1}^{1}\right\},v_{s,N}^{1}\right)$$

The state-expanded hypergraph for $\rho^1\left(a^0\right)$ is illustrated in Figure 6. Here stage 1 has a single dummy state and a hyperarc in its backward star for each possible silviculture strategy (assume 3 strategies possible). At stage 2-4 we have state $tv_i \in TV$ defining the present timber volume of the stand and two possible actions (hyperarcs) for each state (node). For stage 5-12 state $cp_i \in TV \times P$ define the present timber volume of the stand and the price index. Two actions are possible, namely, fell or don't fell. Note that if action fell is chosen then the process finishes, i.e. it enters the dummy node v_{13}^1 representing the finished process.

Observe that there is a one to one correspondence between a policy δ and a *predecessor function* $\mathfrak{g} : \mathcal{V}^1 \setminus \left\{v_{13}^1\right\} \rightarrow \mathcal{E}^1$. Indeed, choosing $\mathfrak{g}\left(v_{s,n}^1\right) = e_{a,s,n}^1$ is equivalent to choosing action a. Moreover, $\mathfrak{g}\left(v_{s,12}^1\right) = e_{s,12}^1$ is the only possible predecessor for node $v_{s,12}^1$ indicating that only the deterministic action *fell* is possible at stage 12.

According to Definition 2 predecessor function $\mathfrak{g}$ define a *hyperpath* with root v_{13}^1 and target $v_{s,1}^1$. That is, choosing a hyperpath defined by predecessor function $\mathfrak{g}$ in the state-expanded hypergraph is equivalent to choosing a specific policy in the MDP.

To build the state-expanded hypergraph for the whole MLHMP, we only need to link the state-expanded hypergraphs at level 1 together by defining nodes and hyperarcs representing the founder process ρ^0. Define the following node and hyperarc set

$$\mathcal{V}^0=\left\{v_{s,1}^0\mid s\in S^0\right\}\cup\left\{v_{s,2}^0\mid s\in S^0\right\}$$
$$\mathcal{E}^0=\left\{e_{a,s}^0\mid s\in S^0,a\in A_s^0\right\}\cup\left\{e_{s,2}^0\mid s\in S^0\right\}$$

with

$$e_{a,s}^{0}=\left(\left\{v_{s',1}^{1}\mid s'\in S_{1}^{1},\,p^{0}\left(s'\mid s,a\right)>0\right\},v_{s,1}^{0}\right),\quad e_{s,2}^{0}=\left(\left\{v_{s,2}^{0}\right\},v_{13}^{1}\right)$$

A compact representation of the state-expanded hypergraph $\mathcal{H} = \left(\mathcal{V}^0 \cup \mathcal{V}^1, \mathcal{E}^0 \cup \mathcal{E}^1\right)$ is shown in Figure 7. Note that we only need to represent two stages of the fonder process since it runs over a infinite time-horizon.

A.3 Finding the optimal policy

Since a policy δ for the MLHMP is equivalent to a hyperpath π in $\mathcal{H}$ the optimal policy for all child processes with respect to a specific criterion can be found by finding the shortest hyperpath π with origin $v_{d,2}^0$ and target $v_{d,1}^0$ using a specific weighting function. That is, the optimal action to choose in (4) of the policy iteration procedure can be found solving a shortest hyperpath problem. For instance if we consider the expected total discounted cost each node v in $\mathcal{H}$ denote

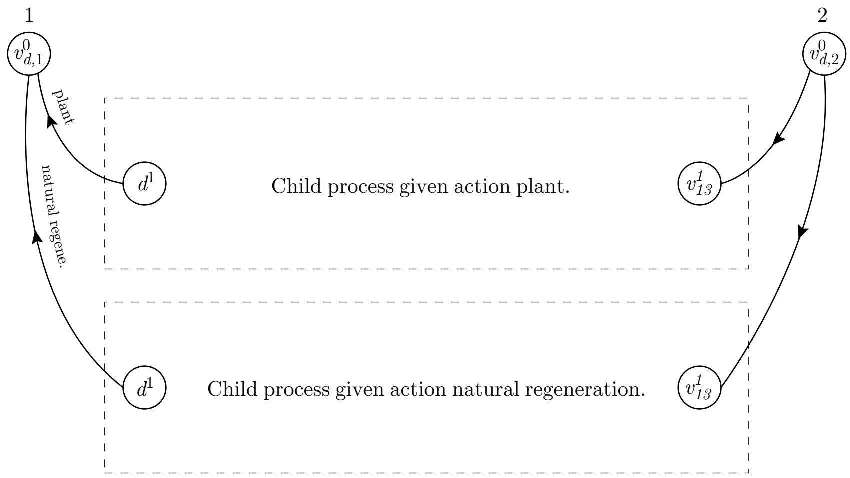

Figure 7: The state-expanded hypergraph for the MLHMP.

a state in a process and each hyperarc $e \in FS(v)$ corresponds to an action given state $v = h(e)$. Assign cost $c(e)$ to each hyperarc $e \in \mathcal{E}$ where $c(e)$ denote the economic cost of choosing action a given state $h(e)$. Similar let $\lambda(e)$ denote the discount rate when choosing action a given state $h(e)$. Finally, let multiplier $m_e(v)$, $v \in T(e)$ be equal to the transition probability of obtaining the state v when choosing action a given state $h(e)$. Then the best policy can be found by finding the shortest hyperpath when using the following weighting function

$$W(v) = \begin{cases} 0 & v = o \\ w(\mathfrak{g}(v)) + \lambda(\mathfrak{g}(v)) \sum_{u \in T(\mathfrak{g}(v))} m_{\mathfrak{g}(v)}(u) W(u) & v \in \mathcal{V}_\pi \setminus \{o\} \end{cases}$$

Note that, value iteration could have been used to find the optimal policy above. However, modelling the MLHMP using the state-expanded hypergraph provide us with efficient ways to calculate the optimal policy and to store the MLHMP. More important, specialized algorithms for directed hypergraphs can now be used on the state-expanded hypergraph. That is, we can now find the K best policies ranking the policies in nondecreasing order of e.g. the expected total cost, Nielsen and Kristensen [13] and use bicriterion optimization techniques for directed hypergraphs to find the trade-off between two different criteria.

Dina Notat No. 111 · January 2005

Danish Informatics Network in the Agriculture Sciences
The Royal Veterinary and Agricultural University
Thorvaldsensvej 40
1871 Frederiksberg C
Denmark