

Management of a Renewable Ressource. Deterministic and Robust Cases

Michel DE LARA & Luc DOYEN

October 10, 2017

Contents

1	Problem statement	1
2	Deterministic growth	1
2.1	Try your own strategy	2
2.2	Intertemporal discounted utility maximization	3
2.3	Minimal utility maximization or Rawls criterion	5
2.4	Tolerable window and guaranteed harvesting (viability)	6
2.5	Intertemporal utility maximization under tolerable window and guaranteed harvesting	7
2.6	Minimal utility maximization or Rawls criterion under tolerable window and guaranteed harvesting	8
3	Uncertain growth	9
3.1	Worst intertemporal utility maximization	9
3.2	Worst minimal utility maximization	9
3.3	Tolerable window and guaranteed harvesting (robust viability)	10

1 Problem statement

Consider a renewable ressource whose stock, measured through its biomass B , is considered globally, as one single unit, without any consideration for the structure population. Its growth is materialized through the linear *Schaefer model*

$$B(t + 1) = R(t)(B(t) - h(t)) , \quad 0 \leq h(t) \leq B(t) , \quad (1)$$

where $h(t)$ is the harvesting at time t and $R(t) = 1 + r(t)$ is the productivity of the resource.

Adapted from [1, p. 258], we shall illustrate the results with the Antartic baleen whale example as in Table 1.

	Antartic baleen whale
intrinsic growth	$r = 5\%$
carrying capacity	$K = 400\ 000$ BWH
catchability	$q = 0.0\ 016$ WCY
price	$p = 7\ 000$ \$ per BWU
cost	$c = 600\ 000$ \$ per WCY

Table 1: Antartic baleen whale data for a discrete time logistic model (adapted from [1, p. 258]). BWH: blue whale unit. WCY: whale-catcher year.

2 Deterministic growth

In this case, the productivity is stationary and deterministic:

$$R(t) = R = 1 + r \quad (r > 0). \quad (2)$$

2.1 Try your own strategy

Question 1 *You have to manage the whale population whose dynamics parameters are given in Table 1. Design your own catch strategy $\text{Catch}(t, B)$ (following the model of the random catch effort strategy in the SCILAB code below). Give the SCILAB code corresponding to your own catch strategy. Draw simulations of biomass and catch trajectories.*

```
R_whale=1.05; // per capita productivity
R=R_whale;
K_whale=400000; // carrying capacity
K=K_whale;

// LINEAR DYNAMICS
function y=linear(B) y=R*B, endfunction;

// RANDOM CATCH EFFORT
function h=rand_effort(t,B) h=rand()*B, endfunction;

Horizon=10;
years=1:Horizon;
yearss=1:(Horizon+1);
yearsss=1:(Horizon+2);

Binit=K/2;
// initial condition
trajectory_whale=zeros(yearss);
```

```

// vector will contain the trajectory B(1),...,B(Horizon+1)
catch_whale=zeros(years);
// vector will contain the catches h(1),...,h(Horizon)
trajectory_whale(1)=Binit;
// initialization of vector B(1),...,B(Horizon+1)
for t=years do
    catch_whale(t)=rand_effort(t,trajectory_whale(t));
    trajectory_whale(t+1)=linear(trajectory_whale(t)-catch_whale(t));
end;

// Graphics
xset("window",20+1);xbasc(20+1);
plot2d2(yearss,[trajectory_whale;[catch_whale,0]]');
xtitle('Trajectory under linear growth with R='+string(R_whale)+' and random effort', ..
        'year (t)', 'biomass (t)')
legends(['Biomass trajectory'; 'Catch trajectory'],[1,2], 'ur');

```

2.2 Intertemporal discounted utility maximization

We now assume that some planner aims at maximizing the sum of discounted utilities of the catches plus a final stock utility. The maximization problem is

$$\sup_{h(t_0), \dots, h(T-1)} \left(\sum_{t=t_0}^{T-1} \rho^t \text{Util}(h(t)) + \rho^T \text{Util}(B(T)) \right) \quad (3)$$

where $\rho = \frac{1}{1+r_f} \in [0, 1]$ is a discount factor and $\text{Util}(\cdot)$ is a utility function, concave and strictly increasing on \mathbb{R}_+ . Notice that the final term $\text{Util}(B(T))$ corresponds to an existence or inheritance value of the stock. We shall use a discount factor ρ with typical values ranging between 0.9 and 1. We restrict the study to the isoelastic utility case where

$$\text{Util}(h) = h^\eta \quad \text{with} \quad 0 < \eta < 1. \quad (4)$$

Analytic resolution by dynamic programming

The dynamic programming equation is:

$$\begin{cases} V(T, B) = \rho^T \text{Util}(B) \\ V(t, B) = \sup_{0 \leq h \leq B} \left(\rho^t \text{Util}(h) + V(t+1, R(B-h)) \right). \end{cases} \quad (5)$$

By backward induction, it can be proved that

$$\begin{cases} V(t, B) = \rho^t \beta_d(t)^{\eta-1} B^\eta, \\ \text{Catch}^*(t, B) = \beta_d(t) B, \end{cases} \quad (6)$$

where $\beta_d(t)$ is given by the backward equation

$$\beta_d(t) = \frac{\alpha_d \beta_d(t+1)}{1 + \alpha_d \beta_d(t+1)}, \quad \beta_d(T) = 1 \quad \text{with} \quad \alpha_d = (\rho R^\eta)^{\frac{1}{\eta-1}}. \quad (7)$$

It can be deduced that, along the optimal path given by

$$B^*(t+1) = R(1 - \beta_d(t))B^*(t), \quad h^*(t) = \beta_d(t)B^*(t), \quad (8)$$

we have

$$\frac{h^*(t+1)}{h^*(t)} = (\rho R)^{\frac{1}{1-\eta}}. \quad (9)$$

scilab simulations

Question 2 *Simulate optimal catches and biomass trajectories $(B^*(t), h^*(t))$. Change the value of the discount factor ρ to modify the location of ρR with respect to 1. Do at least three cases with $\rho R > 1$, $\rho R = 1$ and $\rho R < 1$. What do you observe?*

```

Binit=K/2;
eta=0.5;
rho=1/(1+0.03);
alpha=(rho*R^eta)^(1/(eta-1))

b=[];
b(Horizon+1)=1;
for t=Horizon:-1:1 do
    b(t)=alpha*b(t+1)/(1+alpha*b(t+1));
end;

// Optimal catches and biomass

Bopt=[];
hopt=[];
Bopt(1)=Binit;
for t=1:Horizon do
    hopt(t)=b(t)*Bopt(t);
    Bopt(t+1)=R*(Bopt(t)-hopt(t));
end;

// Graphic display
xbasc();xset('window',20+2);
plot2d2(yearss,[[hopt;Bopt($)],[Bopt]])
// plot2d2(yearsss, [[hopt; Bopt($); 0] [Bopt;0]])
xtitle('Intertemporal utility maximization '+'with rho*R = '+string(rho*R),'years', ...

```

```

    'biomass')
legends(['Optimal catches'; 'Optimal biomass'], [1,2], 'ul');

```

Question 3 What is the influence of η on the optimal paths $(B^*(t), h^*(t))$? In particular, what happens when $\eta \approx 1$? Comment knowing that $1 - \eta$ may be interpreted as the constant relative risk aversion of the decision-maker.

Question 4 Write a program which evaluates the criterion

$$\text{Crit}^{\text{Catch}}(t_0, B_0) := \sum_{t=t_0}^{T-1} \rho^t \text{Util}(h(t)) + \rho^T \text{Util}(B(T)) \quad (10)$$

for any catch decision rule $\text{Catch} : (t, B) \mapsto \text{Catch}(t, B)$ (admissible in the sense that $0 \leq \text{Catch}(t, B) \leq B$), and where

$$B(t+1) = R(B(t) - h(t)) \quad \text{with} \quad h(t) = \text{Catch}(t, B(t)) . \quad (11)$$

Compare the criterion values given by

- the random effort strategy,
- your test strategy in Question 1,
- the discounted utility optimal strategy (6).

Question 5 Write a program which computes for any harvesting strategy $\text{Catch}(t, B)$

- the catch effort

$$E(t) = \frac{h(t)}{qB(t)} \quad (12)$$

- the profit or rent of the exploitation

$$\text{profit}(t) = ph(t) - cE(t) . \quad (13)$$

Plot and compare the profit trajectories $t \mapsto \text{profit}(t)$ given by

- the random effort strategy,
- your test strategy in Question 1,
- the discounted utility optimal strategy (6).

2.3 Minimal utility maximization or Rawls criterion

We now assume that the planner aims at sustainability and intergenerational equity, and attempts at maximizing the utility of the “poorest” generation. The maximization problem is then

$$\sup_{h(t_0), \dots, h(T-1)} \min \left(\min_{t=t_0, \dots, T-1} \text{Util}(h(t)), \text{Util}(B(T)) \right) . \quad (14)$$

Analytic resolution by dynamic programming

The dynamic programming equation is:

$$\begin{cases} V(T, B) = \text{Util}(B) \\ V(t, B) = \sup_{0 \leq h \leq B} \min \left(\text{Util}(h), V(t+1, R(B-h)) \right) \end{cases} \quad (15)$$

By backward induction, it can be proved that

$$\begin{cases} V(t, B) = \text{Util}(\gamma_d(t)B) \\ \text{Catch}^*(t, B) = \gamma_d(t)B, \end{cases} \quad (16)$$

where $\gamma_d(t)$ is given by the backward equation

$$\gamma_d(t) = \frac{R\gamma_d(t+1)}{1 + R\gamma_d(t+1)}, \quad \gamma_d(T) = 1. \quad (17)$$

Consequently, along the maximin optimal path given by

$$B^*(t+1) = R(1 - \gamma_d(t))B^*(t), \quad h^*(t) = \gamma_d(t)B^*(t), \quad (18)$$

we have

$$\frac{h^*(t+1)}{h^*(t)} = 1. \quad (19)$$

scilab simulations

Question 6 Adapt the previous SCILAB code to obtain the maximin path $(B^*(t), h^*(t))$.

Question 7 Compare the trajectories of the maximin criterion, of the stock biomasses and of the catches between

- the random effort strategy,
- your test strategy in Question 1,
- the discounted utility optimal strategy (6) in the case $\rho R < 1$,
- the maximin optimal strategy (16).

2.4 Tolerable window and guaranteed harvesting (viability)

We assume that the policy is to constrain the biomass level within an ecological window, namely between conservation and maximal safety values:

$$0 < B_b \leq B(t) \leq B_\# . \quad (20)$$

A minimal catch is also required:

$$0 < h_b \leq h(t) . \quad (21)$$

Analytic resolution by dynamic programming

The dynamic programming equation for the viability kernels is:

$$\begin{cases} \text{Viab}(T) &= [B_b, B_\#], \\ \text{Viab}(t) &= \{B \in [B_b, B_\#] \mid \exists h \in [h_b, B], \quad R(B - h) \in \text{Viab}(t + 1)\}. \end{cases} \quad (22)$$

Question 8 Define by backward induction

$$\begin{cases} B_b(T) &= B_b \\ B_b(t) &= \max\{B_b, h_b + \frac{B_b(t + 1)}{R}\}. \end{cases}$$

Show that, whenever $B_b(t) \leq B_\#$, the viability kernels are intervals:

$$\text{Viab}(t) = [B_b(t), B_\#].$$

Show that the viable decision rules $\text{Catch}(t, B)$ are those which belong to

$$[h_b(t, B), h_\#(t, B)] := [B - \frac{B_\#}{R}, B - \frac{B_b(t + 1)}{R}].$$

scilab simulations

Question 9 Fix a guaranteed yield h_b as a fraction of carrying capacity K . Simulate trajectories with different viable decision rules.

2.5 Intertemporal utility maximization under tolerable window and guaranteed harvesting

The maximization problem is

$$\sup_{h(t_0), \dots, h(T-1)} \left(\sum_{t=t_0}^{T-1} \rho^t \text{Util}(h(t)) + \rho^T \text{Util}(B(T)) \right) \quad (23)$$

under the constraints

$$\begin{aligned} 0 < h_b &\leq h(t) \leq B(t) \\ 0 < B_b &\leq B(t) \leq B_\#. \end{aligned}$$

The dynamic programming equation is:

$$\begin{cases} V(T, B) &= \rho^T \text{Util}(B), \quad \forall B \in \text{Viab}(T) \\ V(t, B) &= \sup_{h_b(t, B) \leq h \leq h_\#(t, B)} \left(\rho^t \text{Util}(h) + V(t + 1, R(B - h)) \right), \quad \forall B \in \text{Viab}(t). \end{cases} \quad (24)$$

Question 10 Adapt the result and the associated scilab code to compute optimal viable paths $(B^*(t), h^*(t))$.

2.6 Minimal utility maximization or Rawls criterion under tolerable window and guaranteed harvesting

The maximization problem is

$$\sup_{h(t_0), \dots, h(T-1)} \min \left(\min_{t=t_0, \dots, T-1} \text{Util}(h(t)), \text{Util}(B(T)) \right), \quad (25)$$

under the constraints

$$\begin{aligned} 0 < h_b &\leq h(t) \leq B(t) \\ 0 < B_b &\leq B(t) \leq B_\sharp. \end{aligned}$$

The dynamic programming equation is:

$$\left\{ \begin{array}{l} V(T, B) = \text{Util}(B), \quad \forall B \in \text{Viab}(T) \\ V(t, B) = \sup_{h_b(t, B) \leq h \leq h_\sharp(t, B)} \min \left(\text{Util}(h), V(t+1, R(B-h)) \right), \quad \forall B \in \text{Viab}(t). \end{array} \right. \quad (26)$$

Question 11 *Adapt the result and the associated scilab code to compute maximin viable paths $(B^*(t), h^*(t))$.*

3 Uncertain growth

The $R(t)$ vary in an interval $[R_b, R_\#]$.

3.1 Worst intertemporal utility maximization

The maximization problem is

$$\sup_{h(t_0), \dots, h(T-1)} \inf_{R(\cdot) \in [R_b, R_\#]^{T+1}} \left(\sum_{t=t_0}^{T-1} \rho^t \text{Util}(h(t)) + \rho^T \text{Util}(B(T)) \right). \quad (27)$$

Analytic resolution by dynamic programming

The dynamic programming equation is:

$$\begin{cases} V(T, B) = \rho^T \text{Util}(B), \\ V(t, B) = \sup_{0 \leq h \leq B} \inf_{R \in [R_b, R_\#]} \left(\rho^t \text{Util}(h) + V(t+1, R(B-h)) \right). \end{cases} \quad (28)$$

We restrict the study to the isoelastic case (4).

Question 12 *Show that, as for the corresponding deterministic case in subsection 2.2, (6)–(11) hold true with R replaced by R_b .*

scilab simulations

Question 13 *Generate trajectories with robust optimal decision rule. Adapt the code in subsection 2.2 to draw trajectories. Compare the criterion values given by the random effort strategy, by your test strategy in Question 1, and by the optimal strategy.*

3.2 Worst minimal utility maximization

The maximization problem is

$$\sup_{h(t_0), \dots, h(T-1)} \inf_{R(\cdot) \in [R_b, R_\#]^{T+1}} \min \left(\min_{t=t_0, \dots, T-1} \text{Util}(h(t)), \text{Util}(B(T)) \right). \quad (29)$$

Analytic resolution by dynamic programming

The dynamic programming equation is:

$$\begin{cases} V(T, B) = \text{Util}(B) \\ V(t, B) = \sup_{0 \leq h \leq B} \inf_{R \in [R_b, R_\#]} \min \left(\text{Util}(h), V(t+1, R(B-h)) \right). \end{cases} \quad (30)$$

Question 14 *Show that, as for the corresponding deterministic case in subsection 2.3, (16)–(17) hold true with R replaced by R_b .*

scilab simulations

Question 15 *Generate trajectories with robust viable decision rules. Adapt the code in subsection 2.3 to draw trajectories. Compare the criterion values given by the random effort strategy, by your test strategy in Question 1, and by the optimal strategy.*

3.3 Tolerable window and guaranteed harvesting (robust viability)

We assume that the policy is to constrain the biomass level within an ecological window, namely between conservation and maximal safety values, *whatever the scenario*:

$$0 < B_b \leq B(t) \leq B_\# . \quad (31)$$

A minimal catch is also required:

$$0 < h_b \leq h(t) . \quad (32)$$

The dynamic programming equation for the robust viability kernels is:

$$\begin{cases} \text{Viab}_1(T) &= [B_b, B_\#] , \\ \text{Viab}_1(t) &= \{B \in [B_b, B_\#] \mid \exists h \in [h_b, B] , \forall R \in [R_b, R_\#] , R(B - h) \in \text{Viab}_1(t + 1)\} . \end{cases} \quad (33)$$

Question 16 *Define by backward induction*

$$\begin{cases} B_b(T) &= B_b \\ B_b(t) &= \max\{B_b, h_b + \frac{B_b(t + 1)}{R_b}\} . \end{cases}$$

Show that, whenever

$$R_\# B_b(t) \leq R_b B_\# ,$$

the viability kernels are intervals:

$$\text{Viab}(t) = [B_b(t), B_\#] .$$

Show that the viable decision rules $\text{Catch}(t, B)$ are those which belong to

$$[h_b(t, B), h_\#(t, B)] := [B - \frac{B_\#}{R_\#}, B - \frac{B_b(t + 1)}{R_b}] .$$

References

- [1] M. Kot. *Elements of Mathematical Ecology*. Cambridge University Press, Cambridge, 2001.