# Macroeconomics II
# Resursive deterministic models

Prof. McCandless

June 14, 2022

# Optimal control

- Mathematics developed to control the movement of some dynamic system
  - Initially to control physical systems, rockets, machines, quantum mechanics
  - then applied to policy development for social systems
- Soviet version (1950s): variational methods: solve a dynamic problem
- US version (1950s): Bellmans equations: solving recursive systems
- developed as method to control long range rockets
- today: deterministic recursive systems
- next class: variational methods and stochastic recursive systems

# Recursive deterministic models

- What is a recursive problem
- Nature of the problem is the same independent of the period
  1. Same maximization problem
  2. Same budget constraints
  3. Initial values can be different since they can change over time
- Example
  1. A household maximizes its discounted utility stream subject to a budget constraint
  2. If each period the utility **function** is the same
  3. The **form** of the budget constraints are the same
     - The values in the budget constraint can change
     - Wealth can be different in different periods

# States and controls

- Three types of variables
  1. State variables (those that are given at the beginning of a period)
     - stochastic variables chosen by "nature"
     - predetermined variables, whose values were chosen in pervious periods
  2. Control variables (those that can be chosen inside a period)
  3. Other (jump) variables (other variables of interest)

- States variables are those predetermined at the beginning of a period
  1. Capital in a Solow growth model (came from previous period)
  2. Shock to technology (determined by **nature**)
  3. Money stock carried over from previous period
- Control variables are those chosen to maximize some objective function
  1. Investment in the period
  2. Labor supplied in the period
  3. Consumption in the period
  4. Capital to be carried over to the next period
  5. Money holding to be carried over to the next period
- Other variables: those determined by the states and the choices for the values of the controls
  1. Output is determined by capital (normally a state) and by labor supply (normally a control)
  2. If output is determined and investment is a control, consumption is determimed from budget constraints

# Policy Function

- The solution we look for is called a
- A policy function gives
  - The optimizing values for the time $t$ **Controls**
  - As a function of the values of the time $t$ **States**
  - A policy function tells what to do based on what is happening or has happened
    - was developed in the 1950's
    - separately in the US and in the Soviet Union
    - countries needed controls for their rockets
- In a recursive problem the policy **function** will be the same each period
- riding a bicycle is one such optimal control problem

# Infinite horizon problem with states and controls

- Robinson Crusoe want to maximize the infinite horizon discounted utility

$$\max \sum_{i=0}^{\infty} \beta^i u(c_{t+i}),$$

subject to the budget restrictions,

$$k_{t+1} = (1-\delta)k_t + i_t, \qquad y_t = f(k_t) = c_t + i_t.$$

- $k_t$ is the predetermined state variable (no stochastic variables)
- possible choices of controls
  1. $k_{t+1}$ (choosing $k_{t+1}$ determines the required $i_t$ and, from the second budget constraint, $c_t$
  2. $c_t$ (choosing $c_t$ determines $i_t$ and this determines $k_{t+1}$
  3. $i_t$ (choosing $i_t$ determines $c_t$ from one budget constraint and $k_{t+1}$ from the other
- As the problem is written, $c_t$ is the control

# Writing the problem with capital as the control

- Use the budget constraint to write

$$c_t = f(k_t) + (1-\delta)k_t - k_{t+1}$$

- Substitute this into the utility function to remove consumption and investment to get

$$\max \sum_{i=0}^{\infty} \beta^i u\left[f(k_{t+i}) + (1-\delta)k_{t+i} - k_{t+i+1})\right]$$

- In this format, $k_{t+i+1}$ is a control in period $t+i$ and will become the state in period $t+i+1$
  - $k_{t+i}$ is the state in period $t+i$

# The value function (explained for a solow type model)

**Definition**

For given values of the state variables at time t, the value function gives the value of the discounted objective function when that objective function has been maximized.

- The value function is a function of the current **state variables**
- The sum of the discounted objective function is being optimized
- Example:

$$V(k_t) = \max_{\{k_s\}_{s=t+1}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i})$$

# The value function

- Let $k_t$ be the state variable.
- $V(k_t)$ is the value of

$$\sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i})$$

- When the sequence of $\{k_s\}_{s=t+1}^{\infty}$ has been chosen to maximize it
- $V(k_t)$ is a *function* of the current state variables and its value changes when the value of the state variables change (in this case, $k_0$)

# Recursive problems

- Robinson Crusoe's time $t$ problem

$$V(k_t) = \max_{\{k_s\}_{i=0}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i})$$

  is recursive

- In time $t+1$, Robinson Crusoe solves

$$V(k_{t+1}) = \max_{\{k_s\}_{i=0}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+1+i}) - k_{t+2+i} + (1-\delta)k_{t+1+i})$$

# Decomposing the time t problem

The time $t$ problem

$$V(k_t) = \max_{\{k_s\}_{i=0}^{\infty}} \sum_{i=0}^{\infty} \beta^t u(f(k_{t+i}) - k_{t+1+i} + (1-\delta)k_{t+i})$$

can be broken into two components and written as

$$\begin{aligned} V(k_t) &= \max_{k_{t+1}} \Big[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) \\ &\quad + \beta \max_{\{k_s\}_{i=0}^{\infty}} \sum_{i=0}^{\infty} \beta^i u(f(k_{t+1+i}) - k_{t+2+i} + (1-\delta)k_{t+1+i}) \Big] \end{aligned}$$

or, substituting for the second line what it equals, as

$$V(k_t) = \max_{k_{t+1}} \left[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

# The Bellman equation

- The recursive equation

$$V(k_t) = \max_{k_{t+1}} \left[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

  is called a Bellman equation
- It is recursive because value of the function $V(k_t)$ dependes on the value of the same function $V(\cdot)$ but evaluated at $k_{t+1}$
- It is a one period problem
  - One only chooses the value of $k_{t+1}$
  - Notice that the entire future utility is captured in $V(k_{t+1})$
  - Choice of $k_{t+1}$ will change the value of $V(k_{t+1})$
- Lots of systems work this way
  - riding a bicycle, driving a car, flying a glider

# First order conditions

- Take the derivative of $V(k_t)$ with respect to $k_{t+1}$
- Get

$$0 = -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V'(k_{t+1})$$

- Problem is that we do not know the function $V(k_{t+1})$ nor its derivative $V'(k_{t+1})$

# Benveniste - Scheinkman envelope theorem conditions

- Benveniste - Scheinkman give conditions under which one can find $V'(\cdot)$
- Take derivative of $V(k_t)$ in

$$V(k_t) = \max_{k_{t+1}} \left[ u(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

  with respect to $k_t$
- Get

$$V'(k_t) = u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \left( f'(k_t) + (1-\delta) \right)$$

  which we can evaluate at $k_{t+1}$
- The result is called an envelope theorem

# and envelope

- Combine first order conditions
- Get the Euler equation

$$\frac{u'(c_t)}{u'(c_{t+1})} = \beta\left(f'(k_{t+1}) + (1-\delta)\right).$$

- In a stationary state, where $c_t = c_{t+1}$, this is

$$\frac{1}{\beta} - (1-\delta) = f'(\overline{k}).$$

- This works out nicely, but let's look at a general version to see why.

# General version of problem

- Let $x_t$ be the state variables and $y_t$ the controls
- We want solve

$$V(x_t) = \max_{\{y_s\}_{s=t}^{\infty}} \sum_{s=t}^{\infty} \beta^{s-t} F(x_s, y_s)$$

subject to the set of budget constraints

$$x_{s+1} = G(x_s, y_s).$$

- The functions, $F(\cdot, \cdot)$ and $G(\cdot, \cdot)$, are the same for all periods
- Both time $t$ state variables and control variables can be in the objective function and the budget constraints at time $t$.
- This can be written as a Bellmans equation,

$$V(x_t) = \max_{y_t} \left[ F(x_s, y_s) + \beta V(x_{t+1}) \right],$$

subject to the budget constraints

$$x_{s+1} = G(x_s, y_s),$$

# General version of problem

- Using the budget constraint, the Bellmans equation can be written as

$$V(x_t) = \max_{y_t} \left[ F(x_t, y_t) + \beta V(G(x_t, y_t)) \right]$$

- We solve for a **policy function** of the form

$$y_t = H(x_t)$$

- which gives the time $t$ controls are functions of the time $t$ state variables
- Notice that the problem is a **functional equation** and that the solution is the **function** $y_t = H(x_t)$

# General version of problem: the first order conditions

- Taking the derivative of the Bellmans equation gives

$$0 = F_y(x_t, y_t) + \beta V'(G(x_t, y_t))G_y(x_t, y_t)$$

- As before we can find the Benveniste-Scheinkman envelope theorem

$$V'(x_t) = F_x(x_t, y_t) + \beta V'(G(x_t, y_t))G_x(x_t, y_t)$$

  - If $G_x(x_t, y_t) = 0$
  - The envelope condition is simply $V'(x_t) = F_x(x_t, y_t)$
  - The solution can be written as

$$0 = F_y(x_t, y_t) + \beta F_x(G(x_t, y_t), y_{t+1})G_y(x_t, y_t)$$

  - If the function, $F_x(G(x_t, y_t), y_{t+1})$, is independent of $y_{t+1}$,
  - This equation can be solved directly for, $y_t = H(x_t)$

# Conditions for the envelope theorem (from Benveniste-Scheinkman)

- Conditions are (for our form of the model)
  - $x_t \in X$ where $X$ is convex and with non-empty interior
  - $y_t \in Y$ where $Y$ is convex and with non-empty interior
  - $F(x_t, y_t)$ is continuous and differentiable
  - $G(x_t, y_t)$ is continuous and differentiable and invertible in $y_t$
- This gives enough structure so the envelope theorem holds

Newer, more general results in Milgrom and Segel

# Approximation of the value function

- What happens if $G_x(x_t, y_t) \neq 0$ ?
- One can approximate the value function numerically
  - Great contribution of Bellman
- Choose some initial **function** $V_0(x_t)$
  - Most any function will do
  - a good one is $V_0(x_t) = c$
  - where $c$ is a constant (0, for example)
- Find (approximately) the **function** $V_1(x_t)$

$$V_1(x_t) = \max_{y_t} \left[ F(x_t, y_t) + \beta V_0(G(x_t, y_t)) \right]$$

over a sufficiently dense set of values from the domain of $x_t$

  - sufficient for the degree of accuracy that one needs
- One now has the function $V_1(x_t)$

# Approximation of the value function (continued)

- Using this function $V_1(x_t)$, find

$$V_2(x_t) = \max_{y_t} \left[ F(x_t, y_t) + \beta V_1(G(x_t, y_t)) \right]$$

  over a sufficiently dense set of values from the domain of $x_t$
  - one will need to interpolate the function $V_1(x_t)$
  - when the needed $G(x_t, y_t)$ is not part of the relatively dense set of $x_t$
  - linear interpolation is normally good enough
- Using $V_2(x_t)$ repreat the process
- Get a sequence $\{V_i(x_t)\}_{i=0}^{\infty}$
- Bellman showed that $\{V_i(x_t)\}_{i=0}^{\infty} \longrightarrow V(x_t)$
- Once you have $V(x_t)$ finding $y_t = H(x_t)$ is easy
  - Actually, one finds a sequence $\{H_i(x_t)\}_{i=0}^{\infty} \longrightarrow H(x_t)$
  - while finding $\{V_i(x_t)\}_{i=0}^{\infty} \longrightarrow V(x_t)$
- Why does this work? Answer $= \beta$

# Problems of dimensionality

How well do we choose to approximate the function
How many points in the domain of $x_t$
If $x_t \in \mathbb{R}^1$ we can choose lots of points, $M$ points
As dimensionality of $x_t$ grows (say to $\mathbb{R}^N$)
number of points needed is $M^N$ which can be very large

# Comparing example economy to general problem 1: using B-S

- The objective function is

$$F(x_t, y_t) = u(f(k_t) - k_{t+1} + (1-\delta)k_t)$$

- The budget constraint is

$$k_{t+1} = x_{t+1} = G(x_t, y_t) = y_t = k_{t+1}$$

or

$$k_{t+1} = k_{t+1}$$

- The first order condition is

$$\begin{aligned} 0 &= F_y(x_t, y_t) + \beta V'(G(x_t, y_t)) G_y(x_t, y_t) \\ &= -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) + \beta V'(G(x_t, y_t)) \cdot 1 \end{aligned}$$

- Because $\partial k_{t+1} / \partial k_t = 0$, the B-S condition is

$$V'(x_t) = F_x(x_t, y_t) = u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \left( f'(k_t) + (1-\delta) \right)$$

# Comparing example economy to general problem 1: using B-S

- Use this $V'(\cdot)$ in the first order conditions to get

$$\begin{aligned} 0 &= -u'(f(k_t) - k_{t+1} + (1-\delta)k_t) \\ &\quad +\beta\left[u'(f(k_{t+1}) - k_{t+2} + (1-\delta)k_{t+1})\left(f'(k_{t+1}) + (1-\delta)\right)\right]. \end{aligned}$$

- we can find the stationary state where $k_t = k_{t+1} = k_{t+2} = \overline{k}$ as

$$f'(\overline{k}) = \frac{1}{\beta} - (1-\delta)$$

# Comparing example economy to general problem 2

- We can solve the problem a different way, with $c_t$ as control
- Let the objective function be

$$F(x_t, y_t) = u(c_t)$$

- The budget constraint is

$$k_{t+1} = x_{t+1} = G(x_t, y_t) = f(k_t) + (1-\delta)k_t - c_t$$

- The Bellmans equation is

$$V(k_t) = \max_{c_t} \left[ u(c_t) + \beta V \left( f(k_t) + (1-\delta)k_t - c_t \right) \right]$$

  - Notice that the budget constraint is already in $V(k_{t+1})$
- The derivative of the budget constraint is

$$\frac{\partial G(x_t, y_t)}{\partial x_t} = f'(k_t) + (1-\delta) \neq 0$$

  - Can't use B-S method

# Approximation of the Value function

- To approximate the value function need explicit functions for $u(c_t)$ and $f(k_t)$
- Let $f(k_t) = k_t^\theta$ and $u(c_t) = \ln(c_t)$
- Let $\delta = .1$, $\theta = .36$, and $\beta = .98$ (consistent with anual data for US)
- The Bellmans equation is

$$V(k_t) = \max_{k_{t+1}} \left[ \ln(k_t^\theta - k_{t+1} + (1-\delta)k_t) + \beta V(k_{t+1}) \right]$$

- Note: stationary state $\overline{k} = 5.537$ (how do you find this?)

# Approximation of the Value function

- Choose $V_0(\cdot) = 0$ (a constant initial guess for value function)
- Find $V_1(\cdot)$ using

$$
\begin{aligned}
V_1(k_t) &= \max_{k_{t+1}} \left[ \ln(k_t^{\theta} - k_{t+1} + (1-\delta)k_t) + \beta V_0(k_{t+1}) \right] \\
&= \max_{k_{t+1}} \left[ \ln(k_t^{.36} - k_{t+1} + .9k_t) + .98 \cdot 0 \right]
\end{aligned}
$$

  for a relatively dense set of $k_t$
- Find $V_2(\cdot)$ using

$$
V_2(k_t) = \max_{k_{t+1}} \left[ \ln(k_t^{.36} - k_{t+1} + .9k_t) + .98 \cdot V_1(k_{t+1}) \right]
$$

  for a relatively dense set of $k_t$. Use linear interpolation of $V_1(k_{t+1})$ between known points
- Repeat N times. Get approximate $V(k_t)$ function

# Computer program

## Main program

```
global vlast beta delta theta k0 kt
hold off
hold all
%set initial conditions
vlast=zeros(1,100);
k0=0.06:0.06:6;
beta=.98;
delta=.1;
theta=.36;
numits=240;
```

```
%begin the recursive calculations
for k=1:numits
     for j=1:100
         kt=j*.06;
         %find the maximum of the value function
         ktp1=fminbnd(@valfun,0.01,6.2);
         v(j)=-valfun(ktp1);
         kt1(j)=ktp1;
     end
 if k/48==round(k/48)
     %plot the steps in finding the value function
     plot(k0,v)
     drawnow
 end
 vlast=v;
end
hold off
% plot the final policy function
plot(k0,kt1)
```

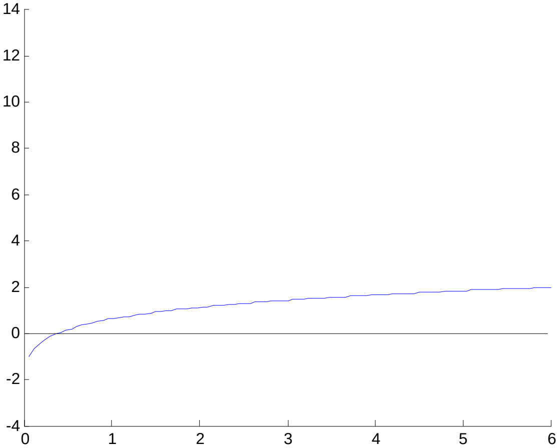
# Computer program

## Subroutine (valfun.m) to calculate value function

```
function val=valfun(k)
global vlast beta delta theta k0 kt
%smooth out the previous value function
g=interp1(k0,vlast,k,'linear');
%Calculate consumption with given parameters
kk=kt^theta-k+(1-delta)*kt;
if kk <= 0
     %to keep values from going negative
     val=-888-800*abs(kk);
else
     %calculate the value of the value function at k
     val=log(kk)+beta*g;
end
%change value to negative since "fminbnd" finds minimum
val=-val;
```

$t$

function 1

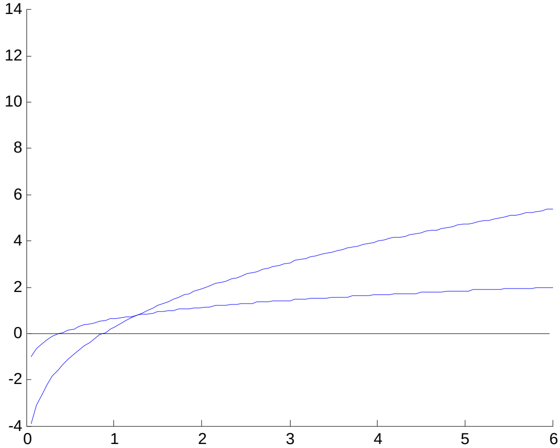



${}_1.pdf$

# after ten iterations

*t*

function 10

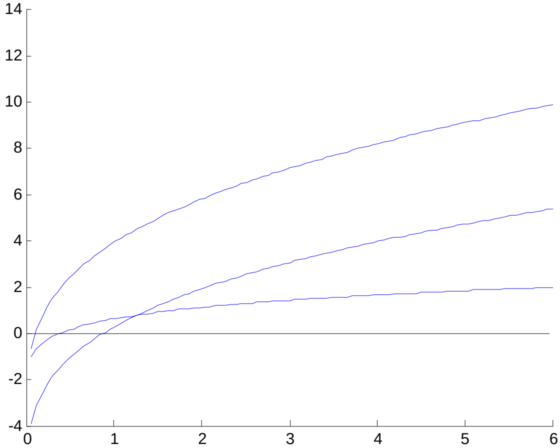



$_2.pdf$

# after 50 iterations

$t$

function 50

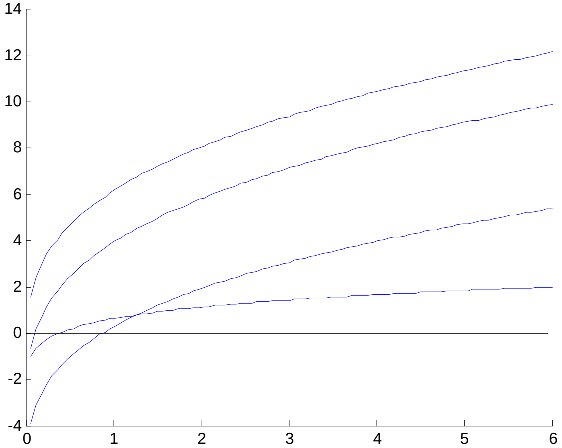



$_3.pdf$

# after 100 iterations

*t*

function 100

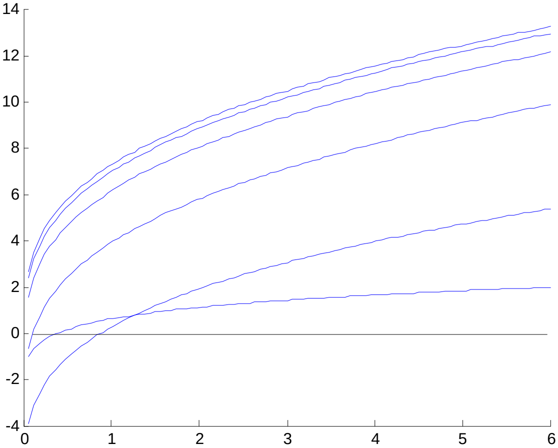



$_4.pdf$

# after 200 iterations

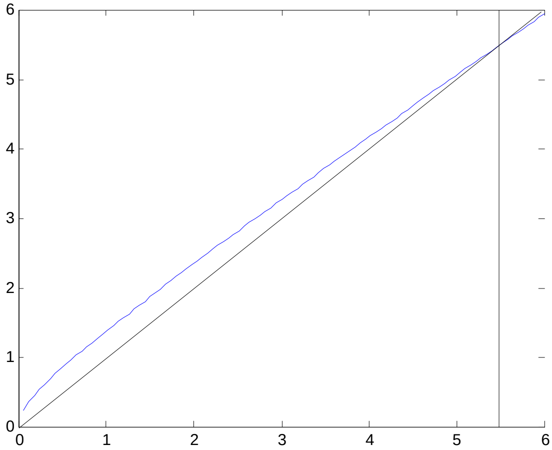
*t*

function 200




$_5.pdf$

function 200




$_6.pdf$