SL Chp 2.1 model | Chp 3 math | Chp 4 apply Chp 3

LS Chp 3

ECON 202A: Introduction to Dynamic Programming

Bangyu He*

October 6, 2023

These notes are intended to introduce the basic concept of dynamic programming, the mathematical tool for solving dynamic economic problems in a recursive way. First, we discuss the sequential formulation of a dynamic problem, and then continue to the recursive formulation which we solve using value function iteration. We also solve for the steady states (which doesn't require solving the whole model).

1 The sequential problem and its solution

Let's first work through an example of how to solve a household's sequential problem. This boils down to figuring out the law of motion of capital (the saving decision today), which we get through the Euler equation, which in turn comes from first-order conditions.

A general example: Consider a simple deterministic neoclassical growth model. In this model, there is an infinitely-lived, representative household who chooses an infinite sequence for consumption, investment and next period's capital stock $\{c_t, i_t, k_{t+1}\}_{t=0}^{\infty}$ in order to maximize its lifetime utility, subject to a resource constraint and law of motion for the capital stock, given the initial capital stock k_0. That is, the representative solves the following maximization problem:

$$\max_{\{c_t, n_t, i_t, k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(c_t)$$

subject to

$$
\begin{aligned}
&(1) & c_t + i_t &= F(k_t, n_t), && \forall t, \\
&(2) & k_{t+1} &= i_t + (1-\delta)k_t, && \forall t, \\
& & c_t, k_t &> 0, && \forall t, \\
&(3) & n_t &\in [0, 1], && \forall t, \\
& & k_0 &\text{ given.}
\end{aligned}
$$

1

0% 100%

$k_{t+1} = i_t$

$n_t = 1$

$n_t < 1$ $n_t < n_t < 1$

$\frac{n_t + 1}{2}$

*I thank the former TAs for this course: Jesper Bojeryd, Paula Beltran, Santiago Justel, Emmanouil Chatzikonstantinou, Andrés Schneider, Adrien des Enffans d'Avernas, Roberto Fattal, Matt Luzzetti, Kyle Herkenhoff and Jinwook Hur. These notes are **heavily** based on their notes. All errors are my own. Email me at bangyuhe@g.ucla.edu for any errors, corrections, or suggestions.

①②③ $\Rightarrow$

$$C_t + i_t = F(k_t, n_t)$$

$$= F(k_t, 1) \equiv f(k_t)$$

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

$$C_t + k_{t+1} = f(k_t)$$

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

$$C_t = f(k_t) - k_{t+1}$$

$$\Rightarrow \max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1})$$

$$\text{s.to} \quad 0 \leq k_{t+1} \leq f(k_t)$$

$<$ $<$

$$\boxed{\lim_{c \to 0} u'(c) = \infty} \Rightarrow C_t \neq 0 \quad \forall t$$

$$k_{t+1} = f(k_t) \Rightarrow C_t = 0$$

$$k_{t+1} = 0 \Rightarrow f(k_{t+1}) = 0 \Rightarrow C_{t+1} = 0$$

$$\Rightarrow \max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1})$$

$$= \cdots + \beta^t U(f(k_t) - k_{t+1}) + \beta^{t+1} U(f(k_{t+1}) - k_{t+2})$$

k_{t+1}

$+ \cdots$

FOC:

k_{t+1}: $\beta^t U'(f(k_t) - k_{t+1})(-1) + \beta^{t+1} U'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1}) = 0$

MC MB

k_{t+1}

$+1$ $+1$

$$\underline{U'(f(k_t) - k_{t+1}) = \beta U'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1})}$$

Intertemporal Euler Eq.

$$\underbrace{U'(f(k_t)-k_{t+1})}_{\text{MC in utility}} = \underbrace{\beta}_{} \underbrace{U'(f(k_{t+1})-k_{t+2})}_{\text{value the additional resources}} \underbrace{f'(k_{t+1})}_{\text{how much more resources?}}$$

transform tomorrow's value to today's value

$u'(c)$

$u(c) = \log c$

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

$u(c_t)$

$$\frac{d\,u(c_t)}{d\,k_{t+1}} = \underbrace{\frac{d\,u(c_t)}{dc_t}}_{u'(c)} \underbrace{\frac{dc_t}{d\,k_{t+1}}}_{(-1)}$$

$$u'(f(k_t)-k_{t+1}) = \frac{1}{f(k_t)-k_{t+1}}$$

$$u'(f(k_{t+1})-k_{t+2}) = \frac{1}{f(k_{t+1})-k_{t+2}}$$

$$f(k) = k^\alpha$$

$$f'(k_{t+1}) = \alpha\, k_{t+1}^{\alpha-1}$$

$$\frac{1}{f(k_t)-k_{t+1}} = \beta \frac{1}{f(k_{t+1})-k_{t+2}} \alpha\, k_{t+1}^{\alpha-1}$$

$$\frac{1}{k_t^\alpha - k_{t+1}} = \beta \frac{1}{k_{t+1}^\alpha - k_{t+2}} \alpha\, k_{t+1}^{\alpha-1} \quad \leftarrow$$

$$\max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1}) \qquad k_0$$

$T \longrightarrow \infty$

$$\max_{\{c_t, k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t u(c_t)$$

$$\text{s.to} \quad c_t = f(k_t) - k_{t+1}$$

$$\mathcal{L} = \sum_{t=0}^{\infty} \left\{ \beta^t u(c_t) + \lambda_t [f(k_t) - k_{t+1} - c_t] \right\}$$

FOC:

c_t: $\quad \beta^t u'(c_t) + \lambda_t(-1) = 0$

$$\beta^t u'(c_t) = \lambda_t \quad (4)$$

k_{t+1}: $\quad \lambda_t(-1) + \lambda_{t+1} f'(k_{t+1}) = 0$

$$\lambda_t = \lambda_{t+1} f'(k_{t+1}) \quad (5)$$

λ_t: $\quad f(k_t) - k_{t+1} = c_t \quad (6)$

(4)(5) $\Rightarrow \quad \beta^t u'(c_t) = \beta^{t+1} u'(c_{t+1}) f'(k_{t+1})$

$$u'(c_t) = \beta u'(c_{t+1}) f'(k_{t+1})$$

(6) $\Rightarrow \quad u'(f(k_t) - k_{t+1}) = \beta u'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1})$

(2)

Here $\beta \in (0,1)$ denotes the household's subjective discount factor for the future utility, δ is the depreciation rate for capital, i_t is investment, and n_t is the hours worked by the household. $U(c_t)$ is the instantaneous utility function and $F(k_t, n_t)$ is the production function. Assume that those functions satisfy the properties discussed in lecture.[1] Since the marginal product of labor and the marginal utility of consumption are always positive according to the assumptions, we know that $n_t = 1$ for all t in any solution. Otherwise, the household will lose the opportunity to obtain more output from production, while there is no benefit from lower hours worked (because there is no disutility from work). Therefore, now $F(k_t, n_t) = F(k_t, 1)$ in any solution, so let me define

$$f(k_t) \equiv F(k_t, 1).$$

To simplify the problem even further, I assume that capital fully depreciates each period (i.e. $\delta = 1$) so that the law of motion of the capital stock is simplified to $i_t = k_{t+1}$. With these assumptions, we can substitute c_t and i_t with functions of the capital stock of today and tomorrow. Therefore, the problem is:

$$\max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1})$$

subject to

$$\begin{aligned} 0 \leq k_{t+1} \leq f(k_t) \quad \forall t, \\ k_0 \quad \text{given}. \end{aligned}$$

Now "all we have to do" is to solve the infinite sequence of capital stock only. The first-order condition of this problem with respect to k_{t+1} is called the Euler Equation. In general, any variable that carries over several periods can generate an Euler equation, or "an intertemporal first-order condition for a dynamic choice problem".

$$\begin{aligned}
0 &= \frac{\partial}{\partial k_{t+1}} \left(\sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1}) \right) = \\
&= \frac{\partial}{\partial k_{t+1}} (\beta^0 U(f(k_0) - k_1)) + \ldots + \frac{\partial}{\partial k_{t+1}} (\beta^t U(f(k_t) - k_{t+1})) \\
&\quad + \frac{\partial}{\partial k_{t+1}} (\beta^{t+1} U(f(k_{t+1}) - k_{t+2})) + \ldots \\
&= 0 + \ldots + \beta^t U'(f(k_t) - k_{t+1}) \times (-1) + \beta^{t+1} U'(f(k_{t+1}) - k_{t+2}) \times (f'(k_{t+1})) + 0 + \ldots \\
\Leftrightarrow \quad 0 &= -U'(f(k_t) - k_{t+1}) + \beta U'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1}).
\end{aligned}$$

The following form is what we usually think of as the Euler equation.

$$U'(f(k_t) - k_{t+1}) = \beta U'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1}). \tag{1}$$

This equation shows the intertemporal condition for optimality: the marginal cost of forgone consumption today by increasing k_{t+1} (LHS) must equal the present value of marginal benefit of the household tomorrow by the higher capital stock tomorrow (RHS). Increasing savings today leads to more productive capital tomorrow that enters into the production function, which is why f' appears in the expression.

[1] So I write the resource constraint and capital law of motion with equalities.

In order to obtain an analytic solution, let me specify the functional forms of the utility function and the production function as follows:

$k^\alpha n^{1-\alpha}$

$$U(c) = \log(c), \quad f(k) = k^\alpha, \quad \alpha \in (0,1).$$

That is, log utility and Cobb-Douglas technology are assumed.[2] Then, (1) becomes:

$$\begin{aligned}
& U(c) = \log(c) \Rightarrow U'(c) = \frac{1}{c} \\
\text{For LHS:} \quad & U'(f(k_t) - k_{t+1}) = \frac{1}{k_t^\alpha - k_{t+1}} \\
\text{For RHS:} \quad & f'(k) = \alpha k^{\alpha-1}
\end{aligned}$$

So,

$$\begin{aligned}
& U'(f(k_t) - k_{t+1}) = \beta U'(f(k_{t+1}) - k_{t+2}) f'(k_{t+1}), \\
\Rightarrow \quad & \frac{1}{k_t^\alpha - k_{t+1}} = \beta \frac{1}{k_{t+1}^\alpha - k_{t+2}} \alpha k_{t+1}^{\alpha-1}. \qquad (2) \\
\text{Equivalently,} \quad & \frac{1}{c_t} = \alpha\beta \frac{1}{c_{t+1}} k_{t+1}^{\alpha-1}. \qquad (3)
\end{aligned}$$

We have thus obtained a nonlinear second-order difference equation.

Equation (2) can be used to find the steady state of the capital stock and of consumption. What is the steady state?

The steady state: A steady state is a state so that if the optimal decision functions are applied to the (steady) state, the state in following period is the same. For now we denote steady states by a bar, so $\bar{k} = k' = g(k) = g(\bar{k})$ where $g(\cdot)$ is the optimal capital decision function and $\bar{k}$ is the steady state of the capital stock.

We can find the steady states of k_t, c_t and implied decision functions by setting $\bar{k} \equiv k_t = k_{t+1}$ and $\bar{c} \equiv c_t = c_{t+1}$ in the Euler equation. I recommend that you start by solving for $\bar{k}$, and then figure out what $\bar{c}$ is. Substitute all ks in (2) by $\bar{k}$:

$$\begin{aligned}
& \frac{1}{\bar{k}^\alpha - \bar{k}} = \alpha\beta \frac{\bar{k}^{\alpha-1}}{\bar{k}^\alpha - \bar{k}} \\
\Leftrightarrow \quad & \bar{k} = \left(\frac{1}{\alpha\beta}\right)^{\frac{1}{\alpha-1}} = (\alpha\beta)^{\frac{1}{1-\alpha}}. \qquad (4)
\end{aligned}$$

To get $\bar{c}$ we use the resource constraint.

$$\bar{c} + \bar{i} = f(\bar{k}) \quad \Rightarrow \quad \bar{c} = \bar{k}^\alpha - \bar{k} = (\alpha\beta)^{\frac{\alpha}{1-\alpha}} - (\alpha\beta)^{\frac{1}{1-\alpha}}.$$

When the problem is more difficult to solve for, we'll ask you to only provide the equations determining the steady state. Then it should be enough to write down the first equations above (i.e., Euler equation and the resource constraints expressed in steady-state variables). Make sure to have the enough equations to solve for the sought number variables.

[2] This kind of specification, together with the assumption of full depreciation of capital stock, makes it possible to solve the optimal policy function in closed form by hand. Closed form solutions do not exist in most of the cases though.

Our second goal is to derive the optimal decision rule, a relationship between k_t and k_{t+1} for all t, so that we can solve for the entire sequence of capital stock with k_0 given.

To do this, I use the fact that *the limit ($t \to \infty$) of the solution to the finite horizon problem* is equivalent to the unique solution to *the infinite horizon problem.* Proving that this conjecture is correct 'involves establishing the legitimacy of interchanging the operators "max" and "$\lim_{T\to\infty}$"', which is more challenging than one might guess.[3] Assume T is the terminal period of the finite horizon problem. Since (2) is a second-order equation, we need not only an initial condition (k_0 given) but also a terminal condition to solve the problem. Since life ends in period T, the agent won't be saving for period $T+1$, and therefore $k_{T+1} = 0$ is the terminal condition.[4] Then, considering (2) at $t = T-1$ with using this terminal condition, we have:

$$\frac{1}{k_{T-1}^{\alpha} - k_T} = \beta \frac{1}{k_T^{\alpha} - 0} \alpha k_T^{\alpha-1} = \beta \frac{\alpha k_T^{\alpha-1}}{k_T^{\alpha}} = \alpha\beta k_T^{-1}. \tag{5}$$

Equivalently,

$$\begin{aligned} k_T &= \alpha\beta(k_{T-1}^{\alpha} - k_T) \\ \Leftrightarrow \quad (1+\alpha\beta)k_T &= \alpha\beta k_{T-1}^{\alpha} \\ \Leftrightarrow \quad k_T &= \frac{\alpha\beta}{1+\alpha\beta} k_{T-1}^{\alpha} \end{aligned}$$

Using this value for k_T, we can return to the Euler Equation (1), evaluated at $t = T-2$.

$$\begin{aligned} \frac{1}{k_{T-2}^{\alpha} - k_{T-1}} &= \frac{\alpha\beta}{k_{T-1}^{\alpha} - k_T} k_{T-1}^{\alpha-1} = \frac{\alpha\beta}{k_{T-1}^{\alpha} - \frac{\alpha\beta}{1+\alpha\beta} k_{T-1}^{\alpha}} k_{T-1}^{\alpha-1} \\ &= \frac{\alpha\beta}{1 - \frac{\alpha\beta}{1+\alpha\beta}} k_{T-1}^{-1} = \frac{\alpha\beta(1+\alpha\beta)}{1+\alpha\beta-\alpha\beta} k_{T-1}^{-1} = \alpha\beta(1+\alpha\beta) k_{T-1}^{-1} \\ \Leftrightarrow \quad k_{T-1} &= \alpha\beta(1+\alpha\beta)(k_{T-2}^{\alpha} - k_{T-1}) \\ \Leftrightarrow \quad (1+\alpha\beta(1+\alpha\beta))k_{T-1} &= \alpha\beta(1+\alpha\beta)k_{T-2}^{\alpha} \\ \Leftrightarrow \quad k_{T-1} &= \frac{\alpha\beta(1+\alpha\beta)}{1+\alpha\beta+(\alpha\beta)^2} k_{T-2}^{\alpha}. \end{aligned}$$

Continuing this procedure, we can find the expression for the relationship between k_t and k_{t+1} for an arbitrary $t < T$ as follows:[5]

$$k_{t+1} = \alpha\beta \frac{1 - (\alpha\beta)^{T-t}}{1 - (\alpha\beta)^{T-t+1}} k_t^{\alpha}. \tag{6}$$

Hence, we can solve for the optimal sequence of capital stock in an infinite-horizon model by taking the limit as $T \to \infty$. Since $\alpha, \beta \in (0,1)$, we find the solution to the Euler Equation:

$$k_{t+1} = \alpha\beta k_t^{\alpha}. \tag{7}$$

Given k_0, therefore, we can solve for the entire sequence of $\{k_{t+1}\}_{t=0}^{\infty}$. After doing so, we can solve for the sequence of consumption using the resource constraint as well.

[3]See Stokey and Lucas, page 12.

[4]It actually is $\beta^T U'(c_T)k_{T+1} = 0$ which corresponds to the transversality condition in an infinite-horizon framework. By strict monotonicity of $U(c) : \beta^T U'(c_T) > 0 \Rightarrow k_{T+1} = 0$

[5]See last section.

2 The recursive problem and its solution

2.1 The recursive formulation

This section introduces how to construct and solve the optimization a recursive formulation of the problem when $\delta = 1$ in the previous section. Define a real-valued function $V(k)$ as:

$$V(k_0) \equiv \max_{\{k_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1}) \tag{8}$$

subject to

$$0 \le k_{t+1} \le f(k_t) \quad \forall t,$$
$$k_0 \quad \text{given.}$$

Here, $V(\cdot)$ is called the value function. It represents the maximized value of the household's utility today and future as a function of the current capital stock k_0. Separating the infinite sum and distributing the max operator, (6) can be rewritten as:

$$\begin{aligned} V(k_0) &= \max_{k_1} \max_{\{k_{t+1}\}_{t=1}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1}) \\ &= \max_{k_1} \left[U(f(k_0) - k_1) + \beta \max_{\{k_{t+1}\}_{t=1}^{\infty}|k_1} \sum_{t=1}^{\infty} \beta^{t-1} U(f(k_t) - k_{t+1}) \right]. \end{aligned}$$

Note that the second maximum term has the same form as (8) with one period forwarded but with k_1 as the initial capital stock. Therefore, (8) can be written as the following recursive form:[6]

$$V(k_0) = \max_{k_1} \left[U(f(k_0) - k_1) + \beta V(k_1) \right]. \tag{9}$$

Hence, now the problem is reduced and k_1 is the only choice variable on the right-hand side. In order to underline the recursive structure of this expression, let k denote the capital stock today and k' denote the capital stock tomorrow. Then, we can rewrite (9) as:

state jump

$$\underbrace{V(k)}_{\text{value today}} = \max_{k'} \left[\underbrace{U(f(k) - k')}_{\text{utility today}} + \underbrace{\beta}_{\text{discount}} \underbrace{V(k')}_{\text{value tomorrow}} \right]. \tag{10}$$

The functional form of $V(\cdot)$ is what we want to solve for, so (10) is a functional equation. This functional equation is called the Bellman equation. The instantaneous utility $U(\cdot)$ is called the return function, and k is called the state variable. The state variable completely summarizes all information that is necessary for the household to make an optimal decision in the current period.

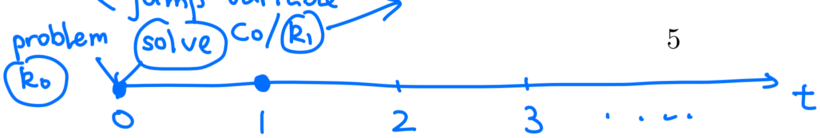
In order to confirm that a Bellman equation is well constructed, check to see:

1. All the right-hand side variables must be either state variables or choice variables. Otherwise, that is not a well-defined Bellman equation.

2. A sequential problem can have multiple number of Bellman equation representations that are equivalent.

[6] This separation and recursive formulation is what is called "Principle of Optimality"

state variable
jump variable c_0/k_1 → $c_0 + k_1 = f(k_0)$



$U'(f(k)-k')(-1) + \beta V'(k') = 0$

$U'(f(k)-k') = \beta V'(k')$

You may wonder why do we care about this function V if we actually care about the solution for k' as a function of k. The reason is the following, for the moment, assume that you know V and moreover, you know V is differentiable, so the first order condition for (10) defines k' implicitly:

$$U'\left(f(k) - k'\right) = \beta V'(k') \tag{11}$$

In particular, let's assume that $U(x) = \log(x)$, $f(k) = k^\alpha$ **and that you know** that $V(k) = A + \frac{\alpha}{1-\alpha\beta}\log(k)$, with A some constant, then (11) implies:

$$\frac{1}{k^\alpha - k'} = \frac{\alpha\beta}{1-\alpha\beta}\frac{1}{k'} \Rightarrow k' = \alpha\beta k^\alpha \tag{12}$$

Same as (7). So, as you can see, solving for V can be really helpful to solve for the policy function.

These are examples of recursive formulations:

Example 1: The Bellman equation (10) can be rewritten as follows:

$$V(k) = \max_{k',c}\left[U(c) + \beta V(k')\right] \tag{13}$$

subject to

$$c + k' = f(k).$$

Here, k is the state variable and c and k' are choice variables. (13) is a well-defined Bellman equation that represents the same problem as (10).

Example 2 – Habit Persistence: Consider a modified neoclassical growth model in which the household's utility is a function of not only its current consumption but also its consumption last period: that is, the utility function is $U(c_t, c_{t-1})$. This is what we mean by habit persistence. This specification implies that last period's consumption level directly influences the household's current utility. As a result, the household must know last period's consumption to optimally solve its problem today. In other words, c_{t-1} must be a state variable in this case. The corresponding Bellman equation can be written as:

$$V(k, c_{-1}) = \max_{c,k'}\Big[U(c, c_{-1}) + \beta V(k', c)\Big]$$

subject to

$$c + i = f(k),$$
$$k' = i + (1-\delta)k,$$

where c_{-1} denotes last period's consumption, with time subscript omitted.

Let's go back to the problem (10) and showing its solution is equivalent to that of the sequential problem. Unlike the sequential problem where the solution consists of the infinite sequences, the recursive formulation asks us to solve for the value function and the optimal policy function between two consecutive periods, which governs the evolution of the state. But how do we solve for $V(\cdot)$? That we do in the section following next section. But before, we derive the Euler equation from the recursive formulation.

$$V(k) = \max_{k'} \; u(f(k)-k') + \beta V(k')$$

$$V'(k) = \beta V'(k') f'(k)$$

LS

$$V(k) = \max_{c} \; u(c^*) + \beta V(k')$$

$$\text{s.to} \quad k' = f(k) - c^*$$

FOC:

c: $\quad u'(c) + \beta V'(k')(-1) = 0$

$$u'(c) = \beta V'(k') \quad (7)$$

p85 SL

ENV:

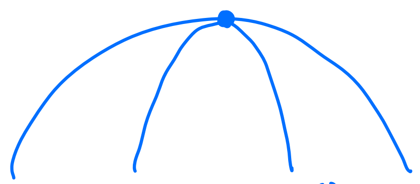
k: $\quad V'(k) = u'(c)\dfrac{dc}{dk} + \beta V'(k')\left(f'(k) - \dfrac{dc}{dk}\right)$

$$= \underbrace{\cancel{\left(u'(c) - \beta V'(k')\right)}}_{0}\frac{dc}{dk} + \beta V'(k') f'(k)$$

$$= \beta V'(k') f'(k) \quad (8)$$

↓ update

$$V'(k') = \beta V'(k'') f'(k') \quad (8')$$



$g(x) \le f(x)$

(7)(8') $\Rightarrow$ $\quad \dfrac{u'(c)}{\cancel{\beta}} = \beta \dfrac{u'(c')}{\cancel{\beta}} f'(k')$

$$u'(c) = \beta u'(c') f'(k')$$

$$u'(f(k)-k') = \beta u'(f(k')-k'') f'(k')$$

$$\Longrightarrow u'(f(k_t)-k_{t+1}) = \beta u'(f(k_{t+1})-k_{t+2}) f'(k_{t+1}) \quad (C11)$$

Step1: FOC w.r.t jump variables

Step2: ENV ("FOC" w.r.t state variables
given that jump variables
are constants)

Step3: update ENV by 1 period

Step 4: plug FOC into <u>updated</u> ENV

2.2 The Euler equation of an economy with full depreciation

The recursive formulation of the model we previously solved sequentially is

$$V(k) = \max_{k'} \ \log(k^\alpha - k') + \beta V(k'),$$
$$c = k' - k^\alpha.$$

The FOC w.r.t. the choice variable k' is

$$\frac{1}{k^\alpha - k'} = \beta V'(k').$$

But this contains the unknown value function. We get rid of it by using the envelope condition.

$$V'(k) = \frac{1}{k^\alpha - k'} \alpha k^{\alpha-1} \quad \Rightarrow \quad V'(k') = \alpha \frac{1}{k'^\alpha - k''}(k')^{\alpha-1},$$

where k'' is the capital stock two periods from today. Substitute into the FOC and we have the Euler equation.

$$\frac{1}{k^\alpha - k'} = \alpha\beta \frac{1}{k'^\alpha - k''}(k')^{\alpha-1}. \tag{14}$$

You can compare this to (2). To solve for the steady state you follow the same steps as for the sequential problem (substitute $k = k' = k'' \equiv \bar{k}$).

2.3 Contraction mapping theorem and Blackwell's sufficient conditions

Let $g(k)$ be the optimal policy function, i.e. the optimal choice of the capital stock next period given the current capital stock, or $k' = g(k)$.

To motivate the solution method that will come, let me define the Bellman operator $T(\cdot)$. T maps from functions to functions, and is defined as follows:

$$V_{m+1}(k) = T(V_m(k)) = \max_{k'} \left[U(f(k) - k') + \beta V_m(k') \right] \tag{15}$$

That is, given a function $V_m(\cdot)$, T returns another function $V_{m+1}(\cdot)$. Therefore, solving (10) is equivalent to finding a fixed point of the mapping T.

To solve the Bellman equation, we need to rely on some mathematical results, which are introduced in Stokey and Lucas chapter 3. Here are definitions that we need.

Definition: A metric space is a set X, together with a metric (distance function) $\rho : X \times X \to \mathbb{R}$, such that for all $x, y, z \in X$:

1. $\rho(x, y) \geq 0$ with equality iff $x = y$ (non-negativity)
2. $\rho(x, y) = \rho(y, x)$ (symmetry)
3. $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$ (triangular inequality)

Blackwell's sufficient conditions

Contraction mapping theorem (√) } SL 3.2

the theorem of the maximum } SL 3.3

Definition: A sequence $\{x_n\}_{n=0}^{\infty}$ in X is a Cauchy sequence if for each $\epsilon > 0$, there exists N_ϵ such that $\rho(x_n, x_m) < \epsilon$ for all $n, m \geq N_\epsilon$.

Definition: A metric space (X, ρ) is complete if every Cauchy sequence in X converges to some element in X.

Now we are ready to define a contraction mapping.

Definition: Let (X, ρ) be a metric space and $T : X \to X$ be a function mapping X onto itself. T is a contraction mapping (with modulus β) if for some β in $(0, 1)$, $\rho(T(x), T(y)) \leq \beta\rho(x, y)$, for all $x, y \in X$.

That is, a mapping T is a contraction mapping if, when you apply it to two elements in the metric space X, the distance between the transformed elements is reduced. Now I introduce the contraction mapping theorem.

Theorem: (Contraction mapping theorem) If (X, ρ) is a complete metric space and $T : X \to X$ is a contraction mapping with modulus β, then:

1. T has exactly one fixed point v in X, and,
2. for any $v_0 \in X$, $\rho(T^n(v_0), v) \leq \beta^n \rho(v_0, v)$.

Therefore, the contraction mapping theorem provides the conditions under which the unique fixed point of the mapping T in (15) exists. Instead of directly showing that T is a contraction mapping, we usually use Blackwell's sufficient conditions for a contraction mapping to examine whether the Bellman equation has a solution.

⇒ **Theorem:** (Blackwell's sufficient conditions for a contraction) Let $X \subset \mathbb{R}^l$, and let $B(X)$ be a space of bounded functions $f : X \to \mathbb{R}$. Let $T : B(X) \to B(X)$ be an operator satisfying:

1. (monotonicity) $f, g \in B(X)$ and $f(x) \leq g(x)$, for all $x \in X$, implies $T(f(x)) \leq T(g(x))$, for all $x \in X$, and
2. (discounting) there exists some $\beta \in (0, 1)$ such that $T(f(x) + a) \leq T(f(x)) + \beta a$ for all $f \in B(X)$, $a \geq 0$, and $x \in X$.

Then, T is a contraction mapping with modulus β.

Let me check if the operator T in (15) satisfies Blackwell's sufficient conditions to ensure that our operator is indeed a contraction mapping.

1. **Monotonicity**: Assume that $v(k) \geq w(k)$ for all $k \in \mathbb{R}_+$, then

$$
\begin{aligned}
T(v) &= \max_{k'} \left[U(f(k) - k') + \beta v(k') \right] \\
&\geq \max_{k'} \left[U(f(k) - k') + \beta w(k') \right] \\
&= T(w)
\end{aligned}
$$

output = T(input)

output = $\max_{k'} U(\cdot) + \beta [\ ] \leftarrow$ input

$T(v(k')) = \max_{k'} U + \beta \underline{v(k')}$

$T(w(k')) = \max_{k'} U + \beta \underline{w(k')}$

$v(k') \geq w(k')$

$T(v(k')) \geq T(w(k'))$

2. **Discounting**: Let a be an arbitrary constant function. Then,

$$\begin{aligned}
T(v+a) &= \max_{k'} \Big[U(f(k)-k') + \beta \big(v(k') + a \big) \Big] \\
&= \max_{k'} \Big[U(f(k)-k') + \beta v(k') + \beta a \Big] \\
&= \max_{k'} \Big[U(f(k)-k') + \beta v(k') \Big] + \beta a \\
&= T(v) + \beta a
\end{aligned}$$

Therefore, our operator T satisfies Blackwell's sufficient conditions, and thus it is a contraction mapping. This guarantees that the Bellman equation has a unique fixed point which can be reached by starting with any initial guess that is a bounded continuous function. Moreover, the theorem indicates that we can arrive at the fixed point by iteration.

2.4 Solving for the value function

Now I present the method of value function iterations, to find the value function $V(\cdot)$ that solves the Bellman equation (9). The method generalizes for all the problems we deal with here.

2.4.1 Using value function iteration

The steps of value function iteration are:

1. Define the operator

$$T(V(k)) = \max_{0 \le k' \le f(k)} \Big[U(f(k)-k') + \beta V(k') \Big]$$

2. Check if the operator T satisfies Blackwell's sufficient conditions. If so, T is a contraction mapping and it has a unique fixed point.

3. Make an initial guess of the value function, say V_0. Any bounded continuous function works, e.g. $V_0(k) = 0$ for all k.[7]

4. Iterate on

$$V_{m+1}(k) = T(V_m(k)) = \max_{0 \le k' \le f(k)} \Big[U(f(k)-k') + \beta V_m(k') \Big] \tag{16}$$

until the distance between V_{m+1} and V_m is negligible. The sup norm is often used to measure the distance between the two functions. Then, the fixed point is $V(k) = \lim_{m\to\infty} V_m(k)$

As mentioned in lecture, the value function $V(k)$ does not have a closed-form representation in most cases. However, there are some cases where the value function has a clear closed form. In this case, we can solve for the functional form using the next procedure.

[7] On a computer, you need to define a grid for k.

2.4.2 Using the method of undetermined coefficients

1–3. Same as above.

4. Iterate on (16). Confirm that a certain functional form appears repeatedly.

5. Once the functional form has converged, use general version of that functional form (with unknown coefficients) and solve the problem as a function of these coefficients and the structural parameters of the model. You will get a 'new' value function.

6. Finally, match your unknown coefficients with the ones of the new value function and solve for them as a function of the known parameters.

<u>**Solving for Example 1:**</u> Let me apply this method to the problem in Example 1, which was

$$V(k) = \max_{k',c} \left[U(c) + \beta V(k')\right]$$

subject to

$$c + k' = f(k), \quad f(k) = k^{\alpha}, \quad U(c) = \log(c).$$

So, first I guess[8] $V_0(k) = 0$ for all k. Then,

$$\begin{aligned} V_1(k) = T(V_0(k)) &= \max_{0 \le k' \le f(k)} \left[\log(k^{\alpha} - k') + \beta V_0(k')\right] \\ &= \max_{0 \le k' \le f(k)} \log(k^{\alpha} - k') + 0. \end{aligned}$$

The right-hand side is maximized when $k' = 0$ (an edge case where FOC won't help us find the optimum). Plugging this into the objective function, V_1 has of the form:

$$V_1(k) = \max_{0 \le k' \le f(k)} \log(k^{\alpha} - k') = \log(k^{\alpha} - 0) = \alpha \log(k).$$

Using this new guess, we have:

$$\begin{aligned} V_2(k) = T(V_1(k)) &= \max_{0 \le k' \le f(k)} \left[\log(k^{\alpha} - k') + \beta V_1(k')\right] \\ &= \max_{0 \le k' \le f(k)} \left[\log(k^{\alpha} - k') + \beta\alpha \log(k')\right] \end{aligned}$$

This time we can't directly see what the optimal k' is, so we compute the first-order condition:

$$\frac{1}{k^{\alpha} - k'} = \frac{\alpha\beta}{k'} \quad \Rightarrow \quad k' = \alpha\beta(k^{\alpha} - k') \quad \Rightarrow \quad k' = \frac{\alpha\beta}{1+\alpha\beta} k^{\alpha}.$$

Plugging this into the objective function.

$$\begin{aligned} V_2(k) = \max_{0 \le k' \le f(k)} \left[\log(k^{\alpha} - k') + \beta\alpha \log(k')\right] &= \left[\log\left(k^{\alpha} - \frac{\alpha\beta}{1+\alpha\beta} k^{\alpha}\right) + \beta\alpha \log\left(\frac{\alpha\beta}{1+\alpha\beta} k^{\alpha}\right)\right] \\ &= \log k^{\alpha} + \log\left(1 - \frac{\alpha\beta}{1+\alpha\beta}\right) + \beta\alpha\left(\log(\frac{\alpha\beta}{1+\alpha\beta}) + \log(k^{\alpha})\right) \\ &= \underbrace{\alpha\beta \log(\alpha\beta) - (1+\alpha\beta)\log(1+\alpha\beta)}_{C} + \underbrace{\alpha(1+\alpha\beta)}_{F} \log(k) \\ &\equiv C + F \log(k). \end{aligned}$$

[8] This is a bad, illustrative, guess; it increases the number of iterations.

We could do one more iteration of this but we would return to the same function form (a constant + a log function). Therefore we continue on to step 6. above, and identify the coefficients by matching.

Make the general guess that $V(k) = A + B\log(k)$, for some A and B, then we have:

$$V(k) = A + B\log(k) = \max_{0\le k'\le f(k)} \Big[\log\left(k^\alpha - k'\right) + \beta\left(A + B\log(k')\right) \Big]$$

FOC w.r.t. k':

$$0 = -\frac{1}{k^\alpha - k'} + \frac{\beta B}{k'} \quad \Rightarrow \quad k' = \frac{\beta B}{1+\beta B}k^\alpha.$$

Plug this back into (13) and arranging terms we have

$$\begin{aligned}
V(k) = A + B\log(k) &= \max_{0\le k'\le f(k)} \Big[\log\left(k^\alpha - k'\right) + \beta\left(A + B\log(k')\right) \Big] \\
&= \log\left(k^\alpha - \frac{\beta B}{1+\beta B}k^\alpha\right) + \beta\left(A + B\log(\frac{\beta B}{1+\beta B}k^\alpha)\right) \\
&= \log(k^\alpha) + \log(\frac{1}{1+\beta B}) + \beta A + \beta B\log(\frac{\beta B}{1+\beta B}) + \beta B\log(k^\alpha) \\
&= \Big[\beta A - \log(1+\beta B) + \beta B\log\beta B - \beta B\log(1+\beta B)\Big] + \Big[\alpha + \alpha\beta B\Big]\log(k).
\end{aligned}$$

By matching coefficients, we see that $A = \beta A + \log\left(\frac{1}{1+\beta B}\right) + \beta B\log\left(\frac{\beta B}{1+\beta B}\right)$ and most importantly, $B = \alpha + \alpha\beta B$. Then, $B = \frac{\alpha}{1-\alpha\beta}$, with that we can solve for A.[9] Thus,

$$V(k) = A + \frac{\alpha}{1-\alpha\beta}\log(k). \tag{17}$$

Finally, using (11) or the fact that $k' = \frac{\beta B}{1+\beta B}k^\alpha$, the policy function is:

$$k' = \frac{\beta\frac{\alpha}{1-\alpha\beta}}{1+\beta\frac{\alpha}{1-\alpha\beta}}k^\alpha = \frac{\alpha\beta}{1-\alpha\beta+\beta\alpha}k^\alpha = \alpha\beta k^\alpha.$$

This expression is the same as (7), which we derived from the sequential problem. We have therefore shown how to solve a dynamic programming problem and that the solution to this reformulated problem is identical to the solution to the original sequential problem.[10]

Steady state – revisited: By doing value function iteration we can circumvent deriving the Euler equation to solve for the Euler equation.

Take the above decision function and set $\bar{k} = k' = k$. Then,

$$\begin{aligned}
\bar{k} &= \alpha\beta(\bar{k})^\alpha, \\
\Leftrightarrow \quad \bar{k}^{1-\alpha} &= \alpha\beta \\
\Leftrightarrow \quad \bar{k} &= (\alpha\beta)^{\frac{1}{1-\alpha}}.
\end{aligned}$$

This is equal to the steady state derived for the sequential problem in (4).

[9] See end of this note.

[10] It is worth to mention that value function iteration (almost) always works when solving computationally, but it is quite slow because of the max operator. There are other numerical methods that give faster solutions e.g. policy function iteration, Howard's improvement, endogenous grid method, etc.

3 Clever guesses for the method of undetermined coefficients

Deterministic

States	Return	Good Guess	Notes
(k)	$\log(c)$	$E + F\log(k)$	
(k, c_{-1})	$\log(c) + A\log(c_{-1})$	$E + F\log(k) + G\log(c_{-1})$	
(k)	$\log(c) - Bh$	$E + F\log(k)$	don't include intratemporal variables[11]
(k, c_{-1})	$\log(c) + A\log(c_{-1}) - Bh$	$E + F\log(k) + G\log(c_{-1})$	

Stochastic

States	Return	Good Guess	Notes
$(k, z) - z\text{AR}(1)$	$\log(c)$	$E + F\log(k) + Gz$	The objective will look like $\log(e^z k^\theta - k')$, so the guess is linear in z
$(k, z) - z\text{AR}(1)$	$\log(c) - Bh$	$E + F\log(k) + Gz$	
$(k, z) - z$ 2-State Markov	$\log(c)$	$v_0^L(k) = E_L + F\log(k)$ and $v_0^H(k) = E_H + B\log(k)$	Split guess over high/low states
$(k, z) - z$ 2-State Markov	$\log(c) - Bh$	$v_0^L(k) = E_L + F\log(k)$ and $v_0^H(k) = E_H + B\log(k)$	

Math skipped in sequential problem

I skipped how we iterate the Euler equation from

$$k_{T-1} = \alpha\beta \frac{1 + \alpha\beta}{1 + \alpha\beta + (\alpha\beta)^2} k_{T-2}^\alpha \tag{18}$$

to get

$$k_{t+1} = \alpha\beta \frac{1 - (\alpha\beta)^{T-t}}{1 - (\alpha\beta)^{T-t+1}} k_t^\alpha.$$

It is purely algebraic manipulation. Do one more iteration. I.e., plug (18) into the Euler equation and solve for k_{T-2} expressed in k_{T-3}.

$$\begin{aligned}
\frac{1}{k_{T-3}^\alpha - k_{T-2}} &= \beta \frac{1}{k_{T-2}^\alpha - k_{T-1}} \alpha k_{T-2}^{\alpha-1} \\
\Leftrightarrow \quad k_{T-2}^\alpha - k_{T-1} &= \alpha\beta(k_{T-3}^\alpha - k_{T-2}) k_{T-2}^{\alpha-1} \\
\Leftrightarrow \quad k_{T-2}^\alpha - \alpha\beta \frac{1 + \alpha\beta}{1 + \alpha\beta + (\alpha\beta)^2} k_{T-2}^\alpha &= \alpha\beta(k_{T-3}^\alpha - k_{T-2}) k_{T-2}^{\alpha-1} \\
k_{T-2}\left(1 - \alpha\beta \frac{1 + \alpha\beta}{1 + \alpha\beta + (\alpha\beta)^2}\right) &= \alpha\beta(k_{T-3}^\alpha - k_{T-2}) \\
k_{T-2}\left(1 + \alpha\beta - \alpha\beta \frac{1 + \alpha\beta}{1 + \alpha\beta + (\alpha\beta)^2}\right) &= \alpha\beta k_{T-3}^\alpha \\
k_{T-2} &= \alpha\beta \frac{1 + \alpha\beta + (\alpha\beta)^2}{1 + \alpha\beta + (\alpha\beta)^2 + (\alpha\beta)^3} k_{T-3}^\alpha.
\end{aligned}$$

Do you see the pattern? To prove this we need a proof by induction which I leave as an exercise for the reader. If we keep on iterating we get

$$k_{T-\tau} = \alpha\beta \frac{\sum_{t'=0}^{\tau} (\alpha\beta)^{t'}}{\sum_{t'=0}^{\tau+1} (\alpha\beta)^{t'}} k_{T-\tau-1}^\alpha = \alpha\beta \frac{\frac{1-(\alpha\beta)^{\tau-1}}{1-\alpha\beta}}{\frac{1-(\alpha\beta)^{\tau}}{1-\alpha\beta}} k_{T-\tau-1}^\alpha = \alpha\beta \frac{1 - (\alpha\beta)^{\tau-1}}{1 - (\alpha\beta)^{\tau}} k_{T-\tau-1}^\alpha.$$

[11] Choice variables that are static and don't carry forward.

Substitute the index so $T-\tau = t+1 \;\Leftrightarrow\; \tau = T-t-1$. Then

$$k_{t+1} = \alpha\beta \frac{1-(\alpha\beta)^{T-t-2}}{1-(\alpha\beta)^{T-t-1}} k_t^{\alpha}. \tag{19}$$

This is the sought expression.

Solving for A in value function, recursive formulation

Since A doesn't matter for the decision functions (when taking derivatives, it always drops out) I didn't derive it in the note. But for completeness, its derivation is included here.

$$\begin{aligned}
A &= \beta A - \log(1+\beta B) + \beta B \log \beta B - \beta B \log(1+\beta B) \\
A &= \frac{1}{1-\beta}\left(\beta B \log \beta B - (1+\beta B)\log(1+\beta B)\right)
\end{aligned}$$

Plug the solution for B from above into this expression.

$$\begin{aligned}
A &= \frac{1}{1-\beta}\left(\beta B \log \beta B - (1+\beta B)\log(1+\beta B)\right) \\
&= \frac{1}{1-\beta}\left(\beta\frac{\alpha}{1-\alpha\beta}\log\beta\frac{\alpha}{1-\alpha\beta} - (1+\beta\frac{\alpha}{1-\alpha\beta})\log(1+\beta\frac{\alpha}{1-\alpha\beta})\right) \\
&= \frac{1}{1-\beta}\left(\frac{\alpha\beta}{1-\alpha\beta}\log(\frac{\alpha\beta}{1-\alpha\beta}) - \frac{1}{1-\alpha\beta}\log(\frac{1}{1-\alpha\beta})\right) \\
&= \frac{1}{(1-\beta)(1-\alpha\beta)}\left(\alpha\beta(\log\alpha\beta - \log(1-\alpha\beta)) + \log(1-\alpha\beta)\right) \\
&= \frac{\alpha\beta\log\alpha\beta + (1-\alpha\beta)\log(1-\alpha\beta)}{(1-\beta)(1-\alpha\beta)}.
\end{aligned}$$

This is the solution of A. But because it doesn't appear in any formula of interest, we usually don't derive it.

More on the envelope condition

This section expands on the idea of taking the envelope condition. Imagine you know the optimal decision function $k' = g(k)$. Then,

$$\begin{aligned}
V(k) &= \max_{k'} U(f(k)-k') + \beta V(k') \\
&= U(f(k)-g(k)) + \beta V(g(k)).
\end{aligned}$$

When we take the envelope condition, we differentiate this expression on both sides w.r.t. k. The interpretation is: what is the marginal change in total value of instant utility (reflected in U) and discounted future utility (reflected in $\beta V(\cdot)$) for a marginal change in today's capital stock k? Mathematically,

$$\begin{aligned}
V'(k) &= U'(f(k)-g(k))(f'(k)-g'(k)) + \beta V'(g(k))g'(k) \\
&= U'(f(k)-k')f'(k) + g'(k)\left(\beta V'(k') - U'(f(k)-k')\right).
\end{aligned}$$

(Nothing deep so far, just brute force chain rule and rearranging). The FOC of the original maximization problem w.r.t. the decision k' is

$$\begin{aligned} 0 &= -U'(f(k) - k') + \beta V'(k') \\ U'(f(k) - k') &= \beta V'(k'). \end{aligned}$$

If we substitute the result of the optimality decision into the envelope expression above, the last term cancels.

$$V'(k) = U'(f(k) - k')f'(k) + g'(k)\left(\beta V'(k') - \beta V'(k')\right) = U'(f(k) - k')f'(k).$$

This is the same as what we got before, but a bit more detailed. We actually don't know what $g(k)$ is throughout these steps, but we can still get rid of it by combining optimallity conditions and the recursive nature of the problem.