November 18th 2009 – Lecture Seventeen

Today's Lecture Covers – Dynamic Programming Continued. Contraction Mapping Theorem. Finding the Euler's Equation. A fancy trick that gets the Envelope Condition to lead to the Bellman Equation Solution. The Neo Classical Growth Model with in Bellman Equation Form. Finding the Steady State and examining Transition Dynamics. Competitive Equilibrium & Dynamic Programming, first looking at Neo Classical Model first, then turning to full Recursive Competitive Equilibrium.

Contraction Mapping Theorem

Defined: If you have a particular function and you apply a contraction to it, it going generates a unique fixed point. Thus if you take a function and apply a contraction to it, you generate that unique fixed point.

For Example with the Bellman Equation

$$V(x) = \max_{y \in \Gamma(x)} F(x, y) + \beta\, V(y)$$

The Contraction Mapping theorem tells you that if there is a contraction then a solution to this equation exists. What are the sorts of conditions that we need? Before, on condition was that $\beta < 1$. There are a number of conditions to guarantee a solution.

Check out the Stokey Lucas, & Sargent Ljungquist books

Last Class – we left off on part four of solving a dynamic programming problem

4) We want assumptions on F, Γ so that there are nice properties on our optimal solution.
V should be monotone, concave (gives a unique maximize), and differentiable. Going one by one,

We've showed a) Monotonicity and b) Concavity. Now let's do c) Differentiability.

As we said back then, concavity was important because it shows our solution is single valued – there is a unique optimal policy.

We want differentiability because we'll be able to solve Bellman Equations analytically to find and Euler Equation. If our problem wasn't differentiable we'd have to find other solution methods.

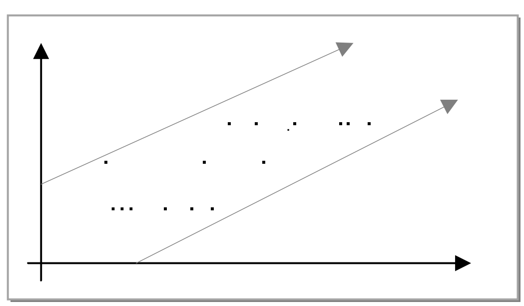
c) **Differentiability** – We now want to show that V is differentiable.

Suppose (for a solution to exit:)

i) Concavity conditions hold
ii) F is continuously differentiable at $\hat{x}$ ($\hat{x}$; at a point)
iii) $\hat{x}$ is in the interior of x —that is analogous to saying that $g(\hat{x})$ is in the interior of $\Gamma(x)$

If these conditions hold then V is differentiable at $\hat{x}$.

Condition two is something you certainly need. Condition iii) is a bit stronger than you'll need, what you're worried about is you want x to be in



You need your choice set to be inside a nice area like this one – one that allows you to move smoothly in your optimization.

Where g is the optimal policy function.

$define\ \{g_n\}$
$V_{n+1} = TV_n$
$g_n(x) = \arg\max_{y\in\Gamma(x)} F(x,y) + \beta\, V_n(y)$
Converges point-wise to the optimal function g.

Back in the concavity section, we wrapped it up by saying that $g(x)$ is single valued because it guarantees that the maximization problem has a unique (single) solution. We also said that if you do the value function iteration you're going to converge to the optimal g. We'll call the ideal sequence g_n. By doing the above over and over ($V_{n+1} = TV_n$) you'll get your unique solution ($g_n(x)$) -

And the rate of convergence of g is going to be slower than the rate of convergence of V because for g to converge it means the derivatives of V are converging. Even if you have convergence of a function - the values the derivatives of that function are still going to be further apart – so in general this g shall take longer. For programs that calculate this sort of thing, good programs will stop when g stops changing. A solution where g has converged tends to be a very good solution for V.

5) **Euler Equation**

The standard equation we get from the Euler Equation from social planner is

$$F_2(x_t, x_{t+1}) + \beta\, F_1(x_{t+1}, x_{t+2}) = 0$$

This is just taking derivatives of the Euler Equation that we we're given at the beginning of part one. (part one, 11/16).
We will write this down, solve the Bellman equation, do the neoclassical growth model as an example.

Working with the Following Bellman Equation
$$V(x) = \max_{y\in\Gamma(x)} F(x,y) + \beta\, V(y)$$

Solving the Bellman, (above) we get a FOC

$$F_y(x,y) + \beta\, V'(y) = 0$$

This is the Euler Equation for this problem.
We've just taken the derivative of the Bellman with respect to y $\left(or\ g(g(x))\right)$

Envelope Condition – that this is a function $V'(x)$

$$V'(x) = F_x(x,y)$$

We've just taken the derivative of the Bellman with respect to x

Now, we can shift this forward one period, (& the y should probably be a g(x))

$$V'(g(x)) = F_x\left(g(x), g(g(x))\right)$$

Doing the same to the Euler:

$$\Rightarrow F_y(x, g(x)) + \beta F_x\left(g(x), g(g(x))\right) = 0$$

Remember the Euler Equation is the "empirical content of the model" – very important
Since we have these generalized equations – when we do an example our solution via the old-fashion discrete time method or dynamic programming, either method should match the two equations above.
Let's do the NeoClassical model to confirm.

NeoClassical Growth Model

$$\max \sum_{t=0}^{\infty} \beta^t u(c_t)$$
$$s.t.\ \ k_{t+1} = f(k_t) + (1+\delta)k_t - c_t$$

Substituting this in

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

If you take the first order conditions with respect to capital,
In period t-1

$$\beta^{t-1} u'(c_{t-1}) = \beta^t u'(c_t)\,[f'(k_t) + (1-\delta)]$$

NeoClassical Growth Model with Bellman

The state variable is k' - substituting out c_t

$$V(k) = \max_{k'} u(f(k) + (1-\delta)k - k') + \beta V(k')$$

For Problem Set 7, think carefully what the state variables are. Write out the Law of Motion of the state variables explicitly. There are a couple of ways to solve these – you may want a feasibility constraint for the controls and a LOM of the states. Or you may want to substitute out your control variables and just solve for the state variables…. But be careful that you know what you are doing in setting up dynamic programming problems.

FOCs with respect to k'

$$u'(f(k) + (1-\delta)k - k') = \beta V'(k')$$

Envelope – derivative with respect to k

$$V'(k) = u'(f(k) + (1-\delta)k - k')[f'(k) + (1-\delta)]$$

Now we need to get rid of the value functions. Shifting all forward one period

$$\Rightarrow u'(c_{t-1}) = \beta\, u'(c_t)[f'(k_t) + (1-\delta)]$$

Or, depending on how you do your shifting,

$$\Rightarrow u'(c_t) = \beta\, u'(c_{t+1})[f'(k_{t+1}) + (1-\delta)]$$

$$\Rightarrow u'(c) = \beta\, u'(c')[f'(k') + (1-\delta)]$$

Inside the c' there is a k''

Summary - As long as the conditions hold such that V is differentiable, this is a method you can use to take the first order condition and the envelope condition from a dynamic programming problem. You are substituting out the Value Functions and you're getting the same Euler Equation you got from solve the model in any other way – the same empirical content in this model as you had solving it any other way.

You can now answer questions as to what is the steady state etc etc.

6) Look for Steady States and Examine Transition Dynamics.

If x^{ss} is a steady state, then $g(x^{ss}) = x^{ss}$

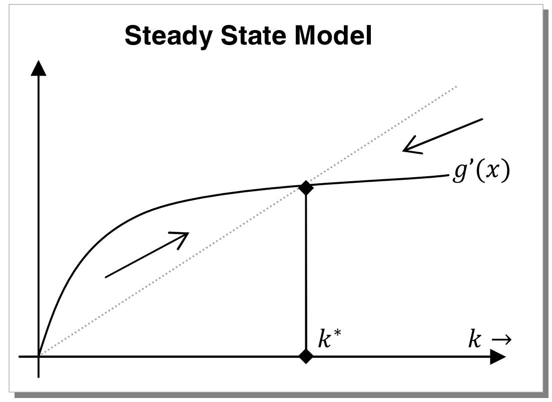
$$F_y(x^{ss}, x^{ss}) + \beta F_x(x^{ss}, x^{ss}) = 0$$

This will give you a steady state

Graphically the SS is defined by where $g(x)$ crosses the $45°$ line
That means if $g'(x) < 1$ then you have local stability.

Thus the question of local stability has only to do with the slope of $g'(x)$

You can log linearize the equation and get a quadratic equation. The roots of that problem are almost reciprocal pairs. It means at least one roots could be stable.



7) Ask How Changing Parameters Alters the Solution

We can ask how changing various parameters of the model alter the solution. There is not going to be one recipe for doing this (meaning, it will be hard). That is going to depend on the specifics of the model. In the past we've added leisure and then did comparative statics on that. We've added taxes and did sensitivity analysis on how taxes affect the steady state. The procedures you might use to think about the implications of various parameter changes is going to depend heavily on the exact structure of the model is and what you're looking at.

Thus, no example.

<break>

What we have done so far is think about how to set up and solve problems as dynamic problems.

If you notice the set-ups we've used, the dynamic problem is typically used to solve an individual's optimization problem – some individual who is maximizing their objective function based on their value today plus the discounted future value plus whatever their state is.

The question is how do we extend this methodology to think about multiple people in the economy?

Competitive Equilibrium & Dynamic Programming

before we had a CE concept.
We had shown that the Social Planner in Bellman gets same answer as the Social Planner in NeoClassical Growth Model.
So our question is - How do we apply the Bellman Equation set up to a Competitive Equilibrium.?

Back in NeoClasssical

$$Consumer's\ Problem$$

$$\max \sum_{t=0}^{\infty} \beta^t \; u(c_t)$$

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

$$k_0 \;\; given$$

We are just assuming labor supply (h_t) is one

Can you formulate this problem recursively?

$$V(k_t) = \max_{c_t, k_{t+1}} u(c_t) + \beta V(k_{t+1})$$
$$s.t. \quad c_t + k_{t+1} = w_t + r_t k_t + (1-\delta)k_t$$

Is that recursive? NO – because the prices r_t & w_t are functions of time.
What's the problem with plugging in the firm's FOCs F_1' & F_2' ? $\Rightarrow$ $c_t + k_{t+1} = F_2' + F_1' \cdot k_t + (1-\delta)k_t$

In this set up the consumer is getting to choose k, that is, choose the prices. It's true that CE in non-dynamic programming problem worked out this way in the *end* (via equilibrium in a competitive market), but they could not be set up that way – it would imply consumer have a power over prices they do not.

With a CE model we are assuming we're dealing with a representative consumer – one who thinks prices are given.

But a recursive this problem cannot depend on time.
And if you plug in those values we have for w and r, it is like consumer's are choosing their prices – No Good.
We need a formulation for equilibrium in a recursive way that satisfies two things

- one, things cannot depend on time
- two, when the individual solves their max problem they cannot think their choices affect the aggregate values.

The Trick that people use is they define another variable
The big k / little k trick.
We'll define a new a new variable

Firm's – they care about the aggregate amount of capital, K
K → the aggregate capital stock
Firm's make their decision based on the state variable K

Consumers
k → the consumers choose. The state variable.
Consumer make their chooses based on the state variable k.

Many Consumers	One Consumer
$\sum_{i=1}^{N} k_i = K$	$k = K$

Now consumers are solving a problem where they don't believe they are choosing prices

■ **A Recursive Competitive Equilibrium** is a list of functions (**This Will Be Monday's Quiz**)

The solution to the consumer's problem is going to be a value function and a an optimal policy function

$V(k,K),\ k'(k,K),$

We need a law of motion for the aggregate stock that only depends on the current value.

$K'(K),$

And we need prices r & w that are functions of the aggregate capital stock.

$r(K),\ \ w(K)$

1) **Consumer's Maximization** - Taking $r(K),\ w(K),\ \&\ K'(K)$ as given
$V(k,K)$ solves:

$$V(k,\mathrm{K}) = \max_{\bar{k}} u\big(w(K) + r(K)k + (1-\delta)k - \bar{k}\big) + \beta\, V\left(\bar{k}, K'(K)\right)$$

$$s.t.\quad 0 \le \bar{k} \le w(K) + r(K)k + (1-\delta)k$$
$$\&\ \ k'(k,K)\ \ is\ an\ optimal\ decision\ rule$$

The bar is for those variables that you choose that period

2) Firm Maximization

$\forall K$
$r(K) = F_1(K,1)$
$w(K) = F_2(K,1)$

3) Markets Clear

Nothing specific to write here, but we still need it. This depends entirely on the model we're dealing with.
If you have multiple consumers then you'll need some adding up conditions
If we're using a representative consumer we might need to have the labor market or capital market to clear.
What we need is a consistency conditions

Consistency condition – it says the laws of motion for the large capital stock (K) and small capital stocks (k) are consistent.

$$\forall k \quad k'(K,K) = K'(K)$$

This is basically saying: rational expectations. People are expecting capital K will evolve in the way it's going to evolve given optimal decisions by everyone in the economy.

$k'(K,K) = K'(K)$ ⇒ that if an individual's capital k was ALL of the capital K, then the their decision would be consistent with the Law of Motion.

Summary

- ✓ With this notation we are ensuring that when the consumers make their decisions they don't think that they are affecting prices. Implicitly they are taking prices as given making optimization decisions.
- ✓ Firms are making their optimization decisions taking prices as given with the aggregate capital stock.
- ✓ Markets clear
- ✓ And finally, if there ere just one representative consumer making those decisions, it has to be that the aggregate level of capital evolves in the same way that the optimal decision rule for the individual agent causes the capital to evolve (that ties everything back together).

Paper to Read – posted on webct

Cooly and Prescott – on Business Cycles and Economic Growth

Gorry will not cover business cycles in class – he's guessing that the first thing we cover with Walsh next quarter will be business cycles. In the Cooly & Prescott paper there is a great deal of discussion about the data and measurement of business cycles (a good head start). The paper touches on an application for the growth model with CE, picking parameter values and sees what the implications are – it also touches on stochastic aspects in the model.