Econ 204B: Section 2

Ryan Sherrard

University of California, Santa Barbara

18 January 2019

Conceptual Starting Point

- Recall from 204A the "social planner," an agent that would make choices for all other agents so as to maximize the utility for the whole economy
- This exercise illuminates the concept of Pareto optimality, the notion that any one agent cannot be made better off without making another worse off
- That is, we know which allocations are optimal; are there other ways (i.e. mechanisms) of achieving these allocations?
  - command (SP), markets (decentralized), etc.
- Much of the above can be tied together with the first and second welfare theorems

$$\text{Competitive Eq.} \implies \text{Pareto Optimal} \qquad \text{(first)}$$
$$\text{Pareto Optimal} \implies \text{Competitive Eq.} \qquad \text{(second)}$$

Some Definitions (from SLP)

Consumer i chooses $x_i \in X_i \subseteq S$ by evaluation according to a utility function $u_i : X_i \to \mathbb{R}$. Firm j chooses $y_j \in Y_j \subseteq S$ by evaluation according to (max. of) total profits. Let the continuous linear functional $\phi : S \to \mathbb{R}$ define a price system.

Definition. An allocation $[(x_i),(y_j)]$ is *Pareto optimal* if it is feasible and if there is no other feasible allocation $[(x_i'),(y_j')]$ such that $u_i(x_i') \geq u_i(x_i)$, all i; and $u_i(x_i') > u_i(x_i)$, some i.

Definition. An allocation $[(x_i^0),(y_j^0)]$ together with a continuous linear functional $\phi : S \to \mathbb{R}$ is a *competitive equilibrium* if

1. $[(x_i^0),(y_j^0)]$ is feasible (attainable)
2. for each i, $x \in X_i$ and $\phi(x) \leq \phi(x_i^0)$ implies $u_i(x) \leq u_i(x_i^0)$ (utility maxing)
3. for each j, $y \in Y_j$ implies $\phi(y) \leq \phi(y_j^0)$ (profit maxing)

First Welfare Theorem

Suppose that for each i and each $x \in X_i$, there exists a sequence $\{x_n\}$ in X_i converging to x, such that $u_i(x_n) > u_i(x)$, $n = 1, 2, \ldots$. If $[(x_i^0), (y_j^0), \phi]$ is a competitive equilibrium, then the allocation $[(x_i^0), (y_j^0)]$ is Pareto optimal.

Sketch of Proof.

- If $[(x_i^0), (y_j^0), \phi]$ is a competitive equilibrium, it must be that the allocation if feasible, agents are utility maximizing, and firms are profit maximizing
- Suppose $\exists$ another feasible allocation $[(x_i'), (y_j')]$ such that $u_i(x_i') \geq u_i(x_i^0)$ for all i (with strict inequality for some i); it must then be that $\phi(\sum_i x_i') > \phi(\sum_i x_i^0)$ (it costs more than x^0)
- Since it's feasible we'll also have that $\phi(\sum_j y_j') > \phi(\sum_j y_j^0)$, meaning that the profits under the "prime" allocation are higher (this contradicts the profit max condition of comp. eq.)
- Thus there cannot be another feasible allocation that makes anyone better off ($\therefore$ it's Pareto Optimal)

Second Welfare Theorem

Assume that the following hold.

1. For each i, X_i is convex
2. For each i, if $x, x' \in X_i$, $u_i(x) > u_i(x')$, and $\theta \in (0,1)$, then $u_i(\theta x + (1-\theta)x') > u_i(x')$
3. For each i, $u_i : X_i \to \mathbb{R}$ is continuous
4. The set $Y = \sum_j Y_j$ is convex
5. Either the set Y has an interior point, or S is finite dimensional

Next, let $[(x_i^0), (y_j^0)]$ be a Pareto optimal allocation, and assume that for some $h \in \{1, \dots, I\}$ there is $\hat{x}_h \in X_h$ with $u_h(\hat{x}_h) > u_h(x_h^0)$. Then there exists a continuous linear functional $\phi : S \to \mathbb{R}$, not identically zero on S, such that

- for each i, $x \in X_i$ and $u_i(x) \geq u_i(x_i^0)$ implies $\phi(x) \geq \phi(x_i^0)$; and
- for each j, $y \in Y_j$ implies $\phi(y) \leq \phi(y_j^0)$.

Proof. see SLP pp.455-456

- To summarize, the *first welfare theorem* is pretty straightforward: competitive equilibria are Pareto optimal (this implicitly assumes that there are no externalities, of course)
- The *second welfare theorem* states that, under some assumptions / conditions, a Pareto optimal allocation can be achieved through a competitive equilibrium
- The proof of the second welfare theorem is a bit hairy, but you should be familiar with its structure
- On the other hand, I expect all of you to know the first welfare theorem or at least be able to give a very nice sketch as I have done
- The FWT is very useful; if you are looking for Pareto optima, we know that we can just solve a Planner's problem
    - recall, there are no prices in the Planner's problem

Sequential Formulation

Let's begin by writing down a very common / standard problem sequentially: the Cake eating problem. We will then move on to the same model recursively and see the benefits of doing so and learn new ways of solving it.

Suppose that the cake will go bad in $T = 20$ periods. Let k_t be the amount of cake available to eat at time t and let c_t denote how much cake you ate in t.

$$\max_{\{c_t, k_{t+1}\}_{t=1}^{T}} \sum_{t=1}^{T} \beta^t ln(c_t) \qquad \text{s.t.} \qquad k_{t+1} = k_t - c_t$$

$$k_0 \text{ given}, \;\; k_t \geq 0 \;\; \forall t$$

Let's go ahead and solve for the Euler Equation.

Now, note that we can solve this one of two ways. We could substitute our constraint in for c_t and maximize solely w.r.t. k_{t+1}, or set up a Lagrangian. I'll be doing the latter:

$$V = \max_{\{c_t, k_{t+1}\}_{t=1}^{20}} \left\{ \sum_{t=1}^{20} \beta^t u(c_t) + \sum_{t=1}^{20} \lambda_t \left[k_t - k_{t+1} - c_t\right] \right\}$$

Now, let's take FOCs ...

$$\frac{\partial V}{\partial c_t} = 0 : \qquad \lambda_t = \frac{\beta^t}{c_t} \quad \forall t$$

$$\frac{\partial V}{\partial k_{t+1}} = 0 : \qquad \lambda_{t+1} = \lambda_t \quad \forall t$$

Now notice that the we can push the first set of conditions forward by one period: $\lambda_{t+1} = \frac{\beta^{t+1}}{c_{t+1}}$.

We can plug both into the second set of conditions.

$$\frac{\beta^{t+1}}{c_{t+1}} = \frac{\beta^t}{c_t}$$

Which, simplifies to our Euler Equation:

$$c_{t+1} = \beta c_t$$

One thing that hasn't really been needed, though will prove to be quite useful, are these things we call *policy rules / functions*. They map the set of state variables into an action by agents. Now, suppose we want to find the policy function for k_t. One way of doing so is to begin at our terminal date, and to iterate backwards.

Policy Rules / Functions

First note that we can write $c_t = k_t - k_{t+1}$. We can plug this into the EE we found previously.

$$k_{t+1} - k_{t+2} = \beta\left(k_t - k_{t+1}\right)$$

From here we can start working backwards from the terminal date. Note that we have the terminal condition $k_{T+1} = 0$ because the cake goes bad. That is, on date T we know that we will just consume whatever is left of the cake. Now, go back one period to $T-1$.

$$c_T = \beta c_{T-1} \qquad \Longrightarrow \qquad k_T - k_{T+1} = \beta(k_{T-1} - k_T)$$

$$k_T - 0 = \beta(k_{T-1} - k_T) \qquad \text{(plug in for } k_{T+1}\text{)}$$

$$k_T = \frac{\beta}{1+\beta} k_{T-1} \qquad \text{(solve for } k_T\text{)}$$

This says that the best choice in $T-1$ (as the representative agent "chooses" next period's capital stock through savings today) for capital in T given some observed state k_{T-1} is given by the above relationship.

Now, take it back once more to period $T-2$ (we'll hopefully see a pattern).

$$c_{T-1} = \beta c_{T-2} \qquad \Longrightarrow \qquad k_{T-1} - k_T = \beta(k_{T-2} - k_{T-1})$$

$$k_{T-1} - \frac{\beta}{1+\beta} k_{T-1} = \beta(k_{T-2} - k_{T-1}) \qquad \text{(plug in for } k_T\text{)}$$

$$k_{T-1} = \frac{\beta + \beta^2}{1 + \beta + \beta^2} k_{T-2} \qquad \text{(solve for } k_{T-1}\text{)}$$

We can continue the above procedure all the way back to the initial period (period $T-T$, if you will).

$$c_1 = \beta c_0 \qquad \Longrightarrow \qquad k_1 - k_2 = \beta(k_0 - k_1)$$

$$k_1 = \frac{\beta + \cdots + \beta^T}{1 + \beta + \cdots + \beta^T} k_0 \qquad \text{(using the pattern)}$$

From here it helps to know the following trick: Let's consider the series

$$S = 1 + \beta + (\beta)^2 + \cdots + (\beta)^T$$

Now multiply S by β:

$$\beta S = \beta + (\beta)^2 + \cdots + (\beta)^{T+1}$$

Now subtract the two:

$$(1-\beta)S = 1 - (\beta)^{T+1}$$

We are left with $S = \frac{1-(\beta)^{T+1}}{1-\beta}$. Compare this to what we found before and we can see that:

$$k_1 = \frac{\beta + \cdots + \beta^T}{1 + \beta + \cdots + \beta^T} k_0$$

can be rewritten as:

$$k_1 = \frac{(S-1)}{S} k_0$$

which, plugging in for S, gives us:

$$k_1 = \frac{\beta(1-\beta^T)}{1-\beta^{T+1}} k_0$$

Let's now think about some other arbitrary period (let's call it $t+1$) using the above results:

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

Which is our policy function/rule for this problem! Given the state in period t, the agent sets k_{t+1} according to some rule (function) $k_{t+1} = \kappa(k_t)$.

Note that with the policy function, given some initial k_0 and some terminal period T, we can recover the full path of c, k, and $u(c)$. I will post some example code on my website which does exactly this, but the following pseudo-code outlines the basics:

Pseudo-code
$\beta = .8$
$k_0 = 1$
$T = 20$
for t in range 1 to 20:
$\quad k_t = \frac{\beta(1-\beta^{T-t})}{1-\beta^{T-t+1}} k_{t-1}$
$\quad c_t = k_t - k_{t+1}$
$\quad u_t = ln(c_t)$
Plot k, c, and u

Sequential Markets Equilibrium

Now, having walked through a very simple problem sequentially, let's get specific about what equilibrium conditions are in a sequential market:
Definition. A *SME* is prices $\{\widehat{r}_t\}_{t=0}^{\infty}$ and allocations $\{(\widehat{c}_t^i, \widehat{a}_{t+1}^i)\}_{t=0}^{\infty}$ (i indexes agents) such that

1. Given $\{\widehat{r}_t\}_{t=0}^{\infty}$, $\forall i$ we have that $\{(\widehat{c}_t^i, \widehat{a}_{t+1}^i)\}_{t=0}^{\infty}$ solves the agent's maximization problem
2. The allocation is feasible and the amount of debt in the economy equals the amount of assets. That is $\sum_i \widehat{c}_t^i = \sum_i e_t^i \;\; \forall t$ and $\sum_i \widehat{a}_{t+1}^i = 0 \;\; \forall t$.

Letting e_t^i denote "income" (and thus is more general than an "endowment"). Further note that, in the market clearing statement, we have assets and debt netting out in the aggregate. This must be true (any borrowings must come from some other agent, of course) but also requires a no-Ponzi condition to ensure people don't borrow an infinite amount (and attain infinite utility).

In practice, unless we are studying a situation where agents are constrained by borrowing limits, what we do put in a constraint that is high enough so that it never binds, but is present simply to stop individuals from attempting to borrow infinitely.

An aside on the no-Ponzi...

Sometimes you'll find out that the no-Ponzi condition is explicit. This might look something like

$$a_{t+1}^i \geq -\overline{A}^i \quad \forall t.$$

On the other hand, sometimes it's implicit or simply unstated. In these scenarios, it is useful to note that we typically endow our economic agents with a little bit of common sense, enough so that they can tell if one agent is trying to borrow infinitely.

It turns out, there is sometimes a *natural debt limit*, pinned down by rational agents' understanding of the ability of others to pay back debt given a stream of (expected) income. For example, in a stochastic income problem with some lower limit y_{min} and a constant interest rate r, the debt limit is

$$a_{t+1}^i \geq -\frac{y_{min}}{r} \quad \forall t.$$

Moving Forward

In the coming weeks we will dive into the different ways of solving recursive formulations of problems, in addition to the mathematical preliminaries underlying everything. But first, it is worth explicitly examining the link between sequential and recursive formulations.

Recall we have seen how to re-write a problem recursively:

Sequential Problem:

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

Recursive Problem:

$$V(k) = \max_c \{u(c) + \beta V(k')\} \qquad s.t. \qquad c + k' = f(k) + (1-\delta)k$$

SP solves the RP

Theorem: Solution to sequential problem also solves the recursive problem

$$v^*(k_0) = \overbrace{\max_{\{0 \le k_{t+1} \le f(k_t)\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t U(f(k_t) - k_{t+1})}^{sequential\ \ problem}$$

$$= \max_{\{0 \le k_{t+1} \le f(k_t)\}_{t=0}^{\infty}} [U(f(k_0) - k_1) + \beta \sum_{t=1}^{\infty} \beta^{t-1} U(f(k_t) - k_{t+1})]$$

$$= \max_{0 \le k_1 \le f(k_0)} [U(f(k_0) - k_1) + \beta \max_{\{0 \le k_{t+1} \le f(k_t)\}_{t=1}^{\infty}} \sum_{t=1}^{\infty} \beta^{t-1} U(f(k_t) - k_{t+1})]$$

$$= \underbrace{\max_{0 \le k_1 \le f(k_0)} [U(f(k_0) - k_1) + \beta v^*(k_1)]}_{recursive\ \ problem}$$

RP solves the SP

Theorem: If $\lim_{T \to \infty} \beta^{T+1} v(k_{T+1}) = 0$ for all $\{k_{T+1}\}$ s.t. $0 \leq k_{T+1} \leq f(k_{T+1})$, then v satisfies SP

$$v(k_0) = \overbrace{\max_{0 \leq k_1 \leq f(k_0)} [U(f(k_0) - k_1) + \beta v(k_1)]}^{recursive\ \ problem}$$

$$= \max_{0 \leq k_1 \leq f(k_0)} \{U(f(k_0) - k_1) + \beta \max_{0 \leq k_2 \leq f(k_1)} [U(f(k_1) - k_2) + \beta v(k_2)]\}$$

$$= \max_{\{0 \leq k_{t+1} \leq f(k_t)\}_{t=0}^{1}} \{\sum_{t=0}^{1} \beta^t [U(f(k_t) - k_{t+1}) + \beta^2 v(k_2)\}$$

$$\vdots$$

$$= \max_{\{0 \leq k_{t+1} \leq f(k_t)\}_{t=0}^{T}} \{\sum_{t=0}^{T} \underbrace{\beta^t [U(f(k_t) - k_{t+1})}_{sequential\ \ problem} + \beta^{T+1} v(k_{T+1})\}$$

Relationship between Policy Functions

Theorem: If $\{x_{t+1^*}\}$ attains the maximum of SP, then $x^*_{t+1} = g(x^*_t) \ \forall \ t \geq 0$

Theorem: If

$$\lim_{T\to\infty} \beta^{T+1} v(x_{T+1}) \leq 0$$

then $\{g(x_0), g(g(x_0)), \ldots\}$ attains the maximum of SP