

Lecciones de Equilibrio General Computable con MatLab

Gonzalo Fernández de Córdoba Martos

Esta versión 28 de julio de 2002

1 Lección primera

1.1 Objetivo del curso

Muchos de los modelos que se utilizan en la actualidad para plantear y resolver problemas macroeconómicos son modelos de equilibrio general computables. Estos modelos deben ser resueltos con la ayuda de un ordenador puesto que salvo en raras excepciones estos modelos tienen soluciones analíticas. Por tanto, en la mayoría de las ocasiones nos encontraremos con complicados modelos en los que el lector no "ve" cómo han sido resueltos, es decir, estos modelos tienen una caja negra en las que se introducen un conjunto de condiciones y de las que sale un conjunto de soluciones. El objetivo fundamental de este curso es el de abrir la caja y ver qué sucede en su interior. Un ejemplo de modelo de equilibrio general dinámico computable y determinista con una enorme caja negra es el de Fernández de Córdoba y Kehoe (1999). En las siguientes lecciones empezaremos a construir y resolver modelos sencillos que vayan aproximándose al modelo de FdCK introduciendo y explicando los elementos que dieron lugar a este modelo.

Este curso de computación de modelos de equilibrio general dinámico deterministas está dirigido a estudiantes de doctorado que quieren aprender a modelizar y a resolver con la ayuda de un ordenador modelos computables. No se supone ningún conocimiento previo de computación y por ello las primeras lecciones se dedican a mostrar la lógica de los algoritmos de optimización basados en el método de Newton explicando cómo se confeccionan los programas partiendo de los comandos más básicos. Todos los códigos que a continuación se presentan están escritos en MatLab[®]. La razón es que MatLab es un lenguaje suficientemente flexible y fácil de aprender, permite construir nuestras propias funciones con sencillez y el soporte gráfico es excelente. Además son muchos los programas ya construidos en MatLab por economistas.

1.2 La lógica del algoritmo de Newton

Cualquier modelo que planteemos a lo largo del curso se reducirá después de unas transformaciones a un sistema de ecuaciones que estará formado por las condiciones de primer orden del problema y por las restricciones de factibilidad. Podemos describir este sistema de n ecuaciones en n incógnitas como $F : R^n \longrightarrow R^n$. Nuestro problema será entonces encontrar un vector

$\hat{x} = (\hat{x}_1, \dots, \hat{x}_n)$ de R^n tal que su imagen por $F : R^n \rightarrow R^n$ sea $F(\hat{x}) = 0$. El algoritmo basado en el método de Newton-Raphson pretende encontrar una solución a este sistema de la forma $F(\hat{x}) = 0$. Para encontrar el \hat{x} solución del sistema se aproxima la función a través de la primera expansión de Taylor a la función F .

$$F(x) = F(\bar{x}) + J(\bar{x})(x - \bar{x}), \quad (1)$$

donde $J(\bar{x})$ es la matriz jacobiana de F evaluada en \bar{x} , es decir,

$$J(\bar{x}) = \begin{bmatrix} F_{11}(\bar{x}) & F_{12}(\bar{x}) & F_{1n}(\bar{x}) \\ F_{21}(\bar{x}) & F_{22}(\bar{x}) & F_{2n}(\bar{x}) \\ F_{n1}(\bar{x}) & F_{n2}(\bar{x}) & F_{nn}(\bar{x}) \end{bmatrix},$$

donde $F_{ij}(\bar{x}) = \frac{\partial F_i(\bar{x})}{\partial x_j}$.

Dado que estamos buscando un cero de la ecuación $F(x)$, la ecuación 1 podemos evaluarla en \hat{x} y escribirla como,

$$\hat{x} = \bar{x} - J(\bar{x})^{-1}F(\bar{x}).$$

El algoritmo funcionaría de la siguiente manera:

1. Proponemos una semilla x_1 y evaluamos la función $F(x_1)$ y $J^{-1}(x_1)$, para calcular

$$x_2 = x_1 - J(x_1)^{-1}F(x_1)$$

2. Fijamos un nivel de tolerancia ε y calculamos alguna distancia entre x_2 y x_1 . Si la distancia es inferior a ε , entonces nos quedamos con x_2 como solución. En caso contrario volvemos al paso 1. y evaluamos la función $F(x_2)$ y $J^{-1}(x_2)$, para calcular

$$x_3 = x_2 - J(x_2)^{-1}F(x_2)$$

Realizamos estas operaciones tantas veces como sea necesario hasta que encontremos un x_t y x_{t+1} tales que la distancia sea menor que el nivel de tolerancia.

1.2.1 Ejemplo 1

Veamos ahora cómo funciona con un ejemplo sencillo. Queremos encontrar los ceros de la función

$$y = (x - 4)(x + 4).$$

La simple inspección visual muestra que los ceros de la función se encontrarán en $x = 4$ y $x = -4$. Ahora vamos a construir nuestro primer programa de MatLab para encontrar esas soluciones.

```
%ej1.m
%Este programa resuelve la ecuación sencilla del ejemplo 1
%La ecuación es:  $y = (x - 4)(x + 4)$ 
clear
%Punto inicial
x(1) = -10;
%Máximo número de iteraciones
maxit=1000;
    for s=1:maxit
        J(s) = 2*x(s);
        x(s+1) = x(s)-J(s)^(-1)*(x(s)-4)*(x(s)+4);
        if abs(x(s+1)-x(s))<1e-20; break; end
    end
if s>=maxit
sprintf('Atención: Número máximo de %g iteraciones alcanzado', maxit)
end
    sprintf('La solución es %g', x(s))
```

El programa ha empezado con una semilla $x_1 = -10$, y por tanto la solución que el programa nos dará es $x^* = -4$. Si hubiéramos introducido como semilla el valor $x_1 = 10$, el programa habría dado como solución $x^* = 4$. En este simple ejemplo vemos algo esencial del algoritmo de Newton: las semillas son de una importancia capital para encontrar la solución a un problema. La localización de la semilla en relación a la solución tiene un impacto en la respuesta del programa. Cuanto más cerca esté la semilla a la solución que buscamos tanto mejor. En este sencillo problema de hallar los ceros de una función de R en R hemos podido computar el jacobiano sin ningún problema, pero si la función hubiera sido de R^n en R^n con n muy grande entonces computar el jacobiano se puede convertir en un problema tan complicado como el de hallar los ceros del sistema. Para resolver este problema podemos computar las derivadas numéricas de la función. La definición de derivada nos dice que

$$\begin{aligned} J_1(x) &= \lim_{h_1 \rightarrow 0} \frac{F(x) - F(x_1 + h, x_2, \dots, x_n)}{h_1}, \\ J_2(x) &= \lim_{h_2 \rightarrow 0} \frac{F(x) - F(x_1, x_2 + h, \dots, x_n)}{h_2}, \\ &\vdots \\ J_n(x) &= \lim_{h_n \rightarrow 0} \frac{F(x) - F(x_1, x_2, \dots, x_n + h)}{h_n}, \end{aligned}$$

donde $J_i(x)$ es el vector columna con las n derivadas parciales con respecto a x_i . Por tanto $J(x) = [J_1(x), J_2(x), \dots, J_n(x)]$. Podemos fijar en un ordenador un incremento h suficientemente pequeño como para poder escribir

$$\begin{aligned} J_1(x) &\approx \frac{F(x) - F(x_1 + h_1, x_2, \dots, x_n)}{h_1}, \\ J_2(x) &\approx \frac{F(x) - F(x_1, x_2 + h_2, \dots, x_n)}{h_2}, \\ &\vdots \\ J_n(x) &\approx \frac{F(x) - F(x_1, x_2, \dots, x_n + h_n)}{h_n}, \end{aligned}$$

y de esta manera aproximar las derivadas parciales.

1.2.2 Ejemplo 2

Ahora queremos encontrar los ceros de la función del Ejemplo 1, pero haciendo uso de una aproximación numérica a la derivada de y con respecto a x . Para ello construimos el siguiente programa en el que definimos un cociente incremental para el jacobiano y un incremento que fijamos en $h = 1e-8$.

```
%ej2.m
%Este programa resuelve por el método de Newton-Raphson la
%ecuación trivial  $y = (x + 4) * (x - 4)$  haciendo uso de una
%aproximación numérica a la derivada de  $y$  con respecto a  $x$ 

clear
%Punto inicial
x0 = -10;
%Número máximo de iteraciones permitido
maxit = 1000;
%Incremento
h = 1e-8;
x(1) = x0;
    for s=1:maxit
        J(s) = 1/(2*h)*((x(s)+h+4)*(x(s)+h-4)-(x(s)-h+4)*(x(s)-h-4));
        x(s+1) = x(s)-J(s)^(-1)*(x(s)-4)*(x(s)+4);
        if abs(x(s+1)-x(s))<1e-20; break; end
    end
if s>=maxit
    sprintf('Atención: Numero máximo de %g iteraciones alcanzado', maxit)
end
    sprintf('La solución es %g', x(s))
```

2 Lección segunda

Una vez que hemos entendido la lógica del algoritmo de Newton-Raphson y la aproximación numérica a las derivadas parciales nos gustaría disponer de un programa más general que resolviera estos problemas con sólo especificar el sistema de ecuaciones que queremos resolver. Esto es posible hacerlo gracias a que disponemos de rutinas muy eficientes que nos permiten resolver el problema de encontrar las soluciones a un sistema de ecuaciones llamando a esas rutinas. A continuación vamos a presentar dos de esas rutinas Urrutia (1998), una para el método de Newton, llamada **newton.m** y otra para el método de aproximación numérica a las derivadas parciales llamada **secant.m**. No son las únicas y de hecho MatLab posee en el "toolbox optim" una rutina llamada **fsolve.m** que permite resolver sistemas de ecuaciones no lineales y que incorpora tanto al método de Newton como al de aproximación numérica de las derivadas, además de poder especificar criterios de tolerancia, máximo número de iteraciones etc. La razón por la que usaremos fundamentalmente el programa **secant.m** se debe a que es fácil intervenir en este programa para obtener información sobre el proceso de optimización. Pero para entender esto, lo mejor es presentar y discutir brevemente estos programas. La diferencia entre el programa **newton.m** y el programa **secant.m** es simplemente que en el primero nosotros especificamos una función en la que está definido el jacobiano del sistema y en el otro hacemos una aproximación numérica a las derivadas parciales, con lo que sólo es necesario especificar en el programa el sistema de ecuaciones. Normalmente utilizaremos el programa **secant.m** dado que a medida que los problemas que nos planteemos sean más complejos no dispondremos de información sobre el jacobiano o éste será demasiado engorroso como para introducirlo dentro del programa.

```

function x = newton(func, x0, param, crit, maxit)
%newton.m      Programa para resolver un sistema de ecuaciones
%simultáneas.
%x=newton(func, x0, param, crit, maxit) usa el método de
%Newton-Raphson para resolver el siguiente sistema:
%          f1(z1, z2, ..., zn)=0
%          f2(z1, z2, ..., zn)=0
%          :                               (*)
%          fn(z1, z2, ..., zn)=0
%en donde x=[z1, z2, ..., zn]' es el vector que resuelve (*).
%Para usar esta función, debe especificarse 'func' que es un "string"

%con el nombre del archivo .m que contiene la función f y su matriz
%jacobiana J. En caso necesario, puede usarse el vector
%param para incluir parámetros adicionales de dicha función.
%Los argumentos x0, crit, y maxit son la semilla inicial para x,
%el criterio de convergencia (tolerancia) y el número máximo de
%iteraciones permitidas respectivamente.
%Programa de Carlos Urrutia
for i=1:maxit
    [f, J]=feval(func, x0, param);
    x=x0-inv(J)*f;
    if norm(x-x0)<crit;      break;      end
    x0=x;
end
if i>=maxit
sprintf('Advertencia: Número máximo de %g iteraciones ...
alcanzado', maxit)
end

```


Lo primero que observamos al mirar al archivo **newton.m** es que comienza con el comando **function**. En MatLab hay dos tipos de archivo que tienen extensión **.m**, los primeros que hemos visto en los ejemplos 1 y 2, son archivos del tipo "script" y contienen una colección de comandos de MatLab que son ejecutados con sólo declarar el nombre del archivo sobre la ventana de comandos de MatLab tras el símbolo `>>`. Los segundos son los que estamos viendo ahora, son del tipo "function" y también contienen comandos de MatLab pero necesariamente tienen que empezar con el comando **function**. Su utilidad consiste en que dentro de ellos podemos definir nuevas funciones y lo que haremos será definir en su interior el sistema de ecuaciones (funciones) que queremos resolver. Para ver cómo funciona plantearemos un ejercicio sencillo.

2.0.3 Ejemplo3

Queremos encontrar el par (x^*, y^*) que resuelve el sistema de ecuaciones

$$\begin{aligned} ay - (x - b)(x + c) &= 0 \\ yx - d &= 0. \end{aligned}$$

Para resolver este problema usando la función **newton.m** debemos calcular el jacobiano del sistema.

$$J = \begin{bmatrix} -2x + (b - c) & a \\ y & x \end{bmatrix}$$

A continuación escribimos un programa similar a los que ya hemos escrito para resolver los ejemplos 1 y 2, con la salvedad de que ahora vamos a introducir el sistema de ecuaciones y el jacobiano en un archivo a parte y vamos a llamar a la rutina **newton.m** para que se encargue del asunto. Los pasos son los siguientes: primero construimos un archivo llamado **sistema.m** donde definimos nuestro problema, fijamos la semilla con la que el algoritmo debe empezar, fijamos el número máximo de iteraciones, fijamos el criterio de tolerancia, fijamos el valor de los parámetros y llamamos al programa **newton.m** indicando el nombre del archivo sobre el que tiene que operar y que vamos a llamar **aqui.m**, puesto que aquí es donde vamos a escribir el valor de la función y el valor del jacobiano.

```

%Archivo sistema.m
%Este programa resuelve por el método de Newton-Raphson
%el sistema de ecuaciones general
%
%ay-(x+b)*(x-c) = 0
%xy-d = 0
%
%para el caso particular
%
%y-(x+4)*(x-4) = 0
%xy-1 = 0
%
%Para ello hacemos uso de la función newton.m
clear
%Punto inicial
%x0 = [-0.001; 0.001];
%x0 = [-0.001; -2];
x0 = [-0.001; -0.001];
%Número máximo de iteraciones permitido
maxit = 1000;
%Criterio de convergencia (tolerancia)
crit = 1e-3;
%Vector de parámetros del sistema
param = [1 4 4 1];
%Llamamos a newton.m y especificamos el archivo sobre el que
debe actuar
sol = newton('aquí', x0, param, crit, maxit);
sprintf('x= %g', sol(1))
sprintf('y= %g', sol(2))

```

Ahora mostramos el contenido del archivo **aqui.m** que como podéis ver es un archivo del tipo function puesto que en él se define una función que debe ser evaluada.

```
function [f, J]=aqui(z, p)
%archivo aqui.m
x = z(1);
y = z(2);
a = p(1);
b = p(2);
c = p(3);
d = p(4);
f = [a*y-(x-b)*(x+c); y*x-d];
J = [-2*x+b-c a; y x];
```

El comando **function [f,J]=aqui(z,p)** dice que el resultado de los cálculos realizados son las matrices **f** (de dimensión 2x1) y **J** de dimensión (4x4), y que el archivo **aqui.m** tiene dos argumentos, el primero es **z** que se identifica con el valor de **x0** del archivo **sistema.m** y el segundo es **p**, que se identifica con el vector **param** del mismo archivo. Las líneas sucesivas asignan valores a las variables y a los parámetros.

Ahora podemos ver cómo funciona el programa **secant.m** en el mismo ejemplo.

2.0.4 Ejemplo 4

Las dos únicas diferencias con respecto a los programas del ejemplo 3 son que naturalmente en el programa **sistema.m** ya no vamos a llamar al programa **newton.m** sino al programa **secant.m**, y la otra diferencia es que el programa **aqui.m** ya no necesita tener especificado el jacobiano del sistema de ecuaciones puesto que la rutina **secant.m** hace una aproximación numérica al jacobiano. El programa **aqui.m** quedaría así:

```
function f=aqui(z, p)
%Programa aqui.m sin el jacobiano
x = z(1);
```

```
y = z(2);  
a = p(1);  
b = p(2);  
c = p(3);  
d = p(4);  
f = [a*y-(x-b)*(x+c); y*x-d];
```

Como vemos, el valor del jacobiano no es computado en esta versión de **aqui.m**, ya que en la rutina **secant.m** que mostramos a continuación se realiza una aproximación numérica al jacobiano.

```

function x=secant(func, x0, param, crit, maxit)
%secant.m Programa para resolver un sistema de ecuaciones
%simultaneas.
% x=secant(func, x0, param, crit, maxit) usa el método de secante
% para resolver el siguiente sistema:
%
% f1(z1,z2, ...,zn)=0
% f2(z1,z2, ...,zn)=0 (*)
% : :
% fn(z1,z2, ...,zn)=0
%
% donde x=[z1,z2, ...,zn]' es el vector que resuelve (*).
% Para usar esta función, debe especificarse 'func' que es un string
% con el nombre de un archivo .m que contiene la función f.
%En caso necesario, puede usarse el vector 'param' para incluir
%parámetros adicionales de dicha función.
% Los argumentos x0, crit y maxit son la semilla inicial para x,
% el criterio de convergencia (tolerancia) y el número máximo de
% iteraciones permitidas.
% Programa de Carlos Urrutia
del = diag(max(abs(x0)*1e-4, 1e-8));
n = length(x0);
    for i=1:maxit
        f=feval(func,x0,param);
        for j=1:n
            J(:,j)=(f-feval(func,x0-del(:,j),param))/del(j,j);
        end
        x=x0-inv(J)*f;
        if norm(x-x0)<crit;      break;      end
        x0=x;
    end
if i>=maxit
    sprintf('Advertencia: Número máximo de %g iteraciones ...
alcanzado', maxit)
end

```

2.0.5 Ejemplo 5

Antes de entrar en modelos dinámicos vamos a realizar un ejercicio sobre un modelo de equilibrio general computable estático en el que un consumidor representativo consumirá dos bienes producidos.

Consumidores Existe un número muy grande de consumidores con idénticas preferencias y dotaciones de tiempo y capital, de modo que podemos agregarlos en un único agente representativo que resuelve el problema:

$$\begin{aligned} & \underset{\{c_1, c_2\}}{Max} \quad \varepsilon c_1^\rho + (1 - \varepsilon) c_2^\rho \\ s.a. \quad & p_1 c_1 + p_2 c_2 \leq w(l_1 + l_2) + r(k_1 + k_2) \\ & l_1 + l_2 \leq \bar{l} \\ & k_1 + k_2 \leq \bar{k} \\ & \bar{l}, \bar{k} \text{ dados} \end{aligned}$$

Hay que notar dos cosas importantes. La primera es que las condiciones de factibilidad sobre las dotaciones las podemos incorporar en la primera restricción, de modo que podemos plantear un problema más sencillo con sólo una restricción. La segunda cuestión importante es que sabemos que las ofertas totales de factores serán exactamente \bar{l} y \bar{k} . La razón es que ambos factores son ofertados inelásticamente. Por tanto el problema es:

$$\begin{aligned} & \underset{\{c_1, c_2\}}{Max} \quad \varepsilon c_1^\rho + (1 - \varepsilon) c_2^\rho \\ s.a. \quad & p_1 c_1 + p_2 c_2 \leq w\bar{l} + r\bar{k} \\ & \bar{l}, \bar{k} \text{ dados} \end{aligned}$$

De este problema obtenemos las demanda de las dos mercancías una vez que hemos calculado las condiciones de primer orden del problema y tenemos en cuenta la restricción presupuestaria, que son:

$$\varepsilon \rho c_1^{\rho-1} - \lambda p_1 = 0 \tag{2}$$

$$(1 - \varepsilon) \rho c_2^{\rho-1} - \lambda p_2 = 0 \tag{3}$$

$$p_1 c_1 + p_2 c_2 - w\bar{l} - r\bar{k} = 0 \tag{4}$$

Empresa 1 La primera empresa tiene una tecnología con la que produce el bien y_1 a partir de trabajo l_1 contratado y capital k_1 alquilado. La tecnología a su disposición es

$$y_1 = A_1 k_1^{\alpha_1} l_1^{1-\alpha_1}$$

que es una tecnología del tipo Coob-Douglas con parámetro de productividad agregada A_1 y parámetro distributivo α_1 (donde α_1 representa a la participación de las rentas del capital en el total de la renta generada por el primer sector, y $(1 - \alpha_1)$ representa la participación de las rentas laborales en la renta generada por el primer sector). El problema de la empresa propiedad del trabajador y consumidor y capitalista representativo es:

$$\underset{\{l_1, k_1\}}{\text{Min}} \quad w l_1 + r k_1$$

$$s.a. \quad y_1 \leq A_1 k_1^{\alpha_1} l_1^{1-\alpha_1},$$

y que

$$p_1 y_1 - w l_1 - r k_1 = 0. \quad (5)$$

De la resolución de este problema obtenemos las siguientes condiciones de primer orden:

$$w = p_1 A_1 (1 - \alpha_1) \left(\frac{k_1}{l_1} \right)^{\alpha_1}, \quad (6)$$

$$r = p_1 A_1 \alpha_1 \left(\frac{k_1}{l_1} \right)^{\alpha_1 - 1}. \quad (7)$$

Como vemos, la función de producción al ser homogénea de grado uno nos permite escribir las productividades marginales (o sea, los precios de los factores) como función de los ratios de capital por trabajador.

Empresa 2 La segunda empresa tiene una tecnología con la que produce el bien y_2 a partir de trabajo l_2 contratado y capital k_2 alquilado. La tecnología a su disposición es

$$y_2 = A_2 k_2^{\alpha_2} l_2^{1-\alpha_2}$$

que es una tecnología del tipo Coob-Douglas con parámetro de productividad agregada A_2 y parámetro distributivo α_2 . El problema de la empresa es:

$$\underset{\{l_2, k_2\}}{\text{Min}} \quad w l_2 + r k_2$$

$$s.a. y_2 \leq A_2 k_2^{\alpha_2} l_2^{1-\alpha_2},$$

y que

$$p_2 y_2 - w l_2 - r k_2 = 0. \quad (8)$$

De la resolución de este problema obtenemos las siguientes condiciones de primer orden:

$$w = p_2 A_2 (1 - \alpha_2) \left(\frac{k_2}{l_2} \right)^{\alpha_2}, \quad (9)$$

$$r = p_2 A_2 \alpha_2 \left(\frac{k_2}{l_2} \right)^{\alpha_2 - 1}. \quad (10)$$

Condiciones de factibilidad Además de las condiciones de primer orden y de las restricciones presupuestaria y tecnológicas tenemos un conjunto de condiciones de factibilidad que nos dicen que en los mercados no pueden entrar en transacción más cantidades de factores que aquellas disponibles a través de las dotaciones. Esto es decir:

$$l_1 + l_2 \leq \bar{l} \quad (11)$$

$$k_1 + k_2 \leq \bar{k} \quad (12)$$

Definición de equilibrio competitivo Una vez que tenemos a todos nuestros agentes caracterizados a través de las funciones que definen su comportamiento, y una vez que tenemos explícitamente escritas todas las restricciones presupuestarias, tecnológicas y de factibilidad, estamos en condiciones de definir un equilibrio competitivo para esta economía.

Definición Un equilibrio competitivo para esta economía artificial es un vector de precios (p_1, p_2, r, w) , una asignación de consumos (c_1, c_2) para el consumidor, y una utilización de factores productivos para las empresas (l_1, k_1, l_2, k_2) tal que:

- Dados los precios de las mercancías (p_1, p_2) y los precios de los factores de producción (r, w) , las cantidades de consumo (c_1, c_2) resuelven el problema de maximización de los consumidores.

- Dado el precio de la mercancía p_1 y dados los precios de los factores (r, w) la asignación de factores de producción (l_1, k_1) resuelve el problema de minimización de costes, con beneficios cero, de la primera empresa.
- Dado el precio de la mercancía p_2 y dados los precios de los factores (r, w) la asignación de factores de producción (l_2, k_2) resuelve el problema de minimización de costes, con beneficios cero, de la segunda empresa.
- Todos los mercados vacían y la asignación es factible.

Que todos los mercados vacían significa que las ofertas y las demandas en los mercados de los bienes c_1 y c_2 coinciden con las cantidades producidas y_1 , e y_2 . Lo mismo sucede en los mercados de factores de producción donde debe suceder que $l_i^s = l_i^d$ y $k_i^s = k_i^d$, $i = \{1, 2\}$. Además de vaciarse los mercados, las condiciones de factibilidad 11 y 12 deben cumplirse.

Solución del modelo Resolver este modelo es tan sencillo como resolver un sistema de ecuaciones. Para hacerlo primero tenemos que tener claro qué ecuaciones son parte de la determinación de la solución y qué ecuaciones no lo son. Una vez que tengamos determinado un número de ecuaciones que son linealmente independientes sabremos para cuántas variables podemos resolver el modelo. Así que contamos ecuaciones:

$$\begin{aligned}
\varepsilon \rho c_1^{\rho-1} - \lambda p_1 &= 0 \\
(1 - \varepsilon) \rho c_2^{\rho-1} - \lambda p_2 &= 0 \\
p_1 c_1 + p_2 c_2 - w \bar{l} - r \bar{k} &= 0 \\
p_1 A_1 (1 - \alpha_1) \left(\frac{k_1}{l_1} \right)^{\alpha_1} - w &= 0 \\
p_1 A_1 \alpha_1 \left(\frac{k_1}{l_1} \right)^{\alpha_1 - 1} - r &= 0 \\
p_2 A_2 (1 - \alpha_2) \left(\frac{k_2}{l_2} \right)^{\alpha_2} - w &= 0 \\
p_2 A_2 \alpha_2 \left(\frac{k_2}{l_2} \right)^{\alpha_2 - 1} - r &= 0 \\
l_1 + l_2 - \bar{l} &= 0
\end{aligned}$$

$$\begin{aligned}
k_1 + k_2 - \bar{k} &= 0 \\
A_1 k_1^{\alpha_1} l_1^{1-\alpha_1} - c_1 &= 0 \\
A_2 k_2^{\alpha_2} l_2^{1-\alpha_2} - c_2 &= 0
\end{aligned}$$

Aquí tenemos el listado completo. Hay que notar que algunas condiciones ya han sido incluidas en la forma en la que las ecuaciones están escritas. Por ejemplo, no hemos escrito ahora variables como l_i^d , o k_i^s , para referirnos a las ofertas y demandas de factores. Simplemente hemos escrito l_i y k_i para referirnos al trabajo y al capital de equilibrio. Es decir, la condición de equilibrio en el mercados de factores está incluida y como $l_i^s = l_i^d = l_i$, entonces nosotros simplemente escribimos la variable sin hacer referencia a ofertas y demandas. En el mercado de factores sí hemos escrito más explícitamente que la condición de equilibrio es $A_i k_i^{\alpha_i} l_i^{1-\alpha_i} - c_i = 0$, ($i = \{1, 2\}$). El sistema de más arriba lo podemos arreglar marginalmente eliminando una variable y deshaciéndonos de una ecuación. La variable sería el multiplicador de Lagrange del problema del consumidor y la ecuación que perdemos es una de las dos condiciones de primer orden del problema del consumidor. Dividiendo una por la otra nos quedaríamos al final con el siguiente sistema:

$$\frac{\varepsilon}{(1-\varepsilon)} \left(\frac{c_1}{c_2}\right)^{\rho-1} - \frac{p_1}{p_2} = 0 \quad (13)$$

$$p_1 c_1 + p_2 c_2 - w \bar{l} - r \bar{k} = 0 \quad (14)$$

$$p_1 A_1 (1 - \alpha_1) \left(\frac{k_1}{l_1}\right)^{\alpha_1} - w = 0 \quad (15)$$

$$p_1 A_1 \alpha_1 \left(\frac{k_1}{l_1}\right)^{\alpha_1-1} - r = 0 \quad (16)$$

$$p_2 A_2 (1 - \alpha_2) \left(\frac{k_2}{l_2}\right)^{\alpha_2} - w = 0 \quad (17)$$

$$p_2 A_2 \alpha_2 \left(\frac{k_2}{l_2}\right)^{\alpha_2-1} - r = 0 \quad (18)$$

$$l_1 + l_2 - \bar{l} = 0 \quad (19)$$

$$k_1 + k_2 - \bar{k} = 0 \quad (20)$$

$$A_1 k_1^{\alpha_1} l_1^{1-\alpha_1} - c_1 = 0 \quad (21)$$

$$A_2 k_2^{\alpha_2} l_2^{1-\alpha_2} - c_2 = 0 \quad (22)$$

Este es el sistema más pequeño que podemos obtener después de haber hecho uso de algunas relaciones triviales como $c_1 = y_1$, o $l_1^s = l_1^d$.

Pasamos a contar ecuaciones para descubrir que tenemos exactamente 10. Hacemos asimismo un recuento de las incógnitas y nos salen $(c_1, c_2, l_1, l_2, k_1, k_2, p_1, p_2, w, r)$ que son también 10. Podríamos pensar erróneamente que éste es un sistema con tantas ecuaciones como incógnitas y que por tanto podemos encontrar el valor de todas y cada una de ellas. Esto sería un error porque en realidad tenemos sólo 9 ecuaciones linealmente independientes. Para verlo escribamos

$$\begin{aligned}
 p_1c_1 + p_2c_2 &= \bar{w}l + \bar{r}k \\
 &= (wl_1 + rk_1) + (wl_2 + rk_2) \\
 &= p_1y_1 + p_2y_2 = p_1c_1 + p_2c_2
 \end{aligned}$$

Es decir, tenemos una ecuación que puede ser encontrada como combinación lineal de otras ecuaciones del modelo, y por tanto no es una ecuación que nos pueda dar información que no tuviéramos ya. Esto significa que una ecuación sobra [$p_1c_1 + p_2c_2 - \bar{w}l - \bar{r}k = 0$] y por tanto tenemos que eliminar una variable. Elegimos, por ejemplo $p_1 = 1$ (a la primera mercancía la consideramos como numerario), y ya tenemos el problema resuelto. Este problema no es más que una consecuencia de la Ley de Walras, que dice que una economía con n mercados de los cuales $n - 1$ están en equilibrio, entonces tiene todos los mercados en equilibrio.

Tenemos, por tanto, 9 ecuaciones linealmente independientes y 9 incógnitas cuyo valor tenemos que determinar. De acuerdo con lo que hemos visto en los anteriores ejercicios bastaría hacer un programa en el que proponemos al algoritmo de Newton-Raphson una solución en este espacio de dimensión 9, dejamos que el algoritmo itere y esperamos la solución. En tanto que esta lógica es completamente correcta es posible hacerlo, sin embargo podemos ayudar al algoritmo de Newton-Raphson disminuyendo la dimensionalidad del problema. Como vamos a ver ahora mismo será suficiente con proponer al algoritmo una semilla que en vez de tener dimensión 9 tendrá sólo dimensión 2. Vamos a verlo:

Imaginemos que conocemos (o proponemos como semilla) el valor de l_1 y k_1 . Podemos seguir a través de la siguiente tabla el razonamiento sin olvidar que $p_1 = 1$:

De la ecuación	Obtenemos	Y sabemos
		l_1, k_1
19	l_2	l_1, k_1, l_2
20	k_2	l_1, k_1, l_2, k_2
21	c_1	l_1, k_1, l_2, k_2, c_1
22	c_2	$l_1, k_1, l_2, k_2, c_1, c_2$
15	w	$l_1, k_1, l_2, k_2, c_1, c_2, w$
16	r	$l_1, k_1, l_2, k_2, c_1, c_2, w, r$
13	p_2	$l_1, k_1, l_2, k_2, c_1, c_2, w, r, p_2$

Hemos obtenido todos los valores de todas las variables del sistema salvo de dos de cuyo valor no estamos seguros pues son una semilla. Tampoco hemos verificado si dos de las ecuaciones son correctas, a saber, la ecuación 17 y la ecuación 18. Por tanto podemos hacer correr el algoritmo de Newton-Raphson sobre estas dos ecuaciones que no hemos comprobado. Si el algoritmo se detiene en un punto del espacio bidimensional (l_1, k_1) , entonces sabremos que esa es la solución ya que todas las demás ecuaciones se verán verificadas.

Para finalizar el ejercicio concluimos escribiendo un programa en MatLab que resuelve exáctamente este problema de dos sectores.

Programa en MatLab A continuación voy a escribir y comentar los programas de MatLab que nos ayudarían a resolver numéricamente este problema de Equilibrio General Competitivo. Para ello vamos a necesitar tres archivos de MatLab, dos de ellos serán del tipo *function* y uno será del tipo *script*. En el archivo de tipo *script* vamos a definir los parámetros del modelo y vamos a hacer una llamada a las funciones que contienen al algoritmo de Newton-Raphson y a las condiciones de primer orden.

Run4m.m

```

%Programa run4m.m, funciona conjuntamente con
%secant.m y cpo4m.m. Este programa resuelve un
%modelo de equilibrio general computable
%para una economía con dos sectores y dos factores de
%producción ofertados inelásticamente por un consumidor
%representativo.
%CONSUMIDORES
%Max epsilon*c1^rho+(1-epsilon)*c2^rho

```

```

%c1,c2
%s.a.  $p_1*c_1+p_2*c_2=w*\bar{l}+r*\bar{k}$ 
%
%EMPRESA 1
%Min  $w*l_1+r*k_1$ 
%k1,l1
%s.a.  $y_1=A_1*k_1^{\alpha_1}*l_1^{(1-\alpha_1)}$ 
%
%p1*y1-w*l1-r*k1=0
%
%EMPRESA 2
%Min  $w*l_2+r*k_2$ 
%k2,l2
%s.a.  $y_2=A_2*k_2^{\alpha_2}*l_2^{(1-\alpha_2)}$ 
%
%p2*y2-w*l2-r*k2=0
%
%lbar y kbar dados
clear
%Definición de parámetros del modelo
A1      = 10;
A2      = 10;
alpha1  = 0.35;
alpha2  = 0.4;
epsilon = 0.2;
rho     = -1;
lbar    = 2;
kbar    = 20;
param   = [A1 A2 alpha1 alpha2 epsilon rho lbar kbar];
%Definición de parámetros de programa
crit    = 1e-4;
maxit   = 1000;
x0      = [1.0687; 9.6207];
sol     = secant('cpo4m', x0, param, crit, maxit);
%Resultados

```

```

l1 = sol(1);
k1 = sol(2);
l2 = lbar-l1;
k2 = kbar-k1;
c1 = A1*k1^alpha1*l1^(1-alpha1);
c2 = A2*k2^alpha2*l2^(1-alpha2);
p2 = ((1-epsilon)/epsilon)*(c2/c1)^(rho-1);
r = A1*alpha1*(k1/l1)^(alpha1-1);
w = A1*(1-alpha1)*(k1/l1)^alpha1;

```

Este programa **run4m.m** comienza con unos comentarios y el comando **clear**. Después pasa a la definición de parámetros del modelo y a continuación se definen parámetros que tienen que ver con el funcionamiento del programa, que son una semilla (**x0**) que contiene el valor del par (l_1, k_1) , un nivel de tolerancia (**crit**) y un número máximo de iteraciones (**maxit**) para el algoritmo de Newton-Raphson.

El programa **cpo4m.m** es un archivo del tipo *function* en el que tenemos todas las condiciones de primer orden y las condiciones de factibilidad del problema.

Cpo4m.m

```

function f=cpo4m(z, param)
%Asignación de variables
l1 = z(1);
k1 = z(2);
%Asignación de parámetros
A1 = param(1);
A2 = param(2);
alpha1 = param(3);
alpha2 = param(4);
epsilon = param(5);
rho = param(6);
lbar = param(7);
kbar = param(8);
%Condiciones de factibilidad y vaciado

```

```

l2 = lbar-l1;
k2 = kbar-k1;
c1 = A1*k1^alpha1*l1^(1-alpha1);
c2 = A2*k2^alpha2*l2^(1-alpha2);
%Condiciones de primer orden
p2 = ((1-epsilon)/epsilon)*(c2/c1)^(rho-1);
r = A1*alpha1*(k1/l1)^(alpha1-1);
w = A1*(1-alpha1)*(k1/l1)^alpha1;
%Condiciones por comprobar
f(1) = r-p2*A2*alpha2*(k2/l2)^(alpha2-1);
f(2) = w-p2*A2*(1-alpha2)*(k2/l2)^alpha2;
f=f';

```

Finalmente la función **secant.m** que escribo aquí pero que ya tenéis más arriba.

Secant.m

```

function x=secant(func, x0, param, crit, maxit)
%secant.m Programa para resolver un sistema de ecuaciones simultaneas.
% x=secant(func, x0, param, crit, maxit) usa el metodo de secante
% para resolver el siguiente sistema:
%
% f1(z1,z2, ...,zn)=0
% f2(z1,z2, ...,zn)=0 (*)
% : :
% fn(z1,z2, ...,zn)=0
%
% donde x=[z1,z2, ...,zn]' es el vector que resuelve (*).
% Para usar esta función, debe especificarse 'func' que es un string
% con el nombre de un archivo .m que contiene la función f. En caso
% necesario, puede usarse el vector 'param' para incluir parámetros
% adicionales de dicha función.
% Los argumentos x0, crit y maxit son la semilla inicial para x,
% el criterio de convergencia (tolerancia) y el número máximo de
% iteraciones permitidas.

```

```

% Programa de Carlos Urrutia
del = diag(max(abs(x0)*1e-4, 1e-8));
n = length(x0);
for i=1:maxit
f=feval(func,x0,param);
for j=1:n
J(:,j)=(f-feval(func,x0-del(:,j),param))/del(j,j);
end
x=x0-inv(J)*f;
if norm(x-x0)<crit
break
end
x0=x;
end
if i>=maxit
sprintf('Advertencia: Número máximo de %g iteraciones alcanzado', maxit)
end

```

Con estos programas ya podéis calcular el equilibrio general competitivo para una economía con dos mercados de factores y dos mercados de bienes.

3 Lección tercera

Una vez que hemos entendido cómo funcionan los algoritmos básicos de computación podemos introducir otros nuevos basados en los conocidos. Para ilustrar los nuevos métodos que vamos a presentar lo haremos introduciendo el tipo de problemas que dieron lugar a estas soluciones. El problema básico será un modelo de crecimiento sobre el cual comenzaremos a introducir más y más elementos hasta alcanzar el grado de sofisticación al que ha llegado la literatura económica en nuestras fechas. En esta lección vamos a construir nuestro primer programa para resolver un modelo económico, pero antes tenemos que repasar un poco la teoría económica que hay detrás del programa.

3.1 El modelo básico de crecimiento

Estudiar macroeconomía es estudiar modelos de Equilibrio General Dinámico Aplicado. En estos modelos SIEMPRE deberemos especificar un entorno y una definición de equilibrio. La necesidad de hacerlo SIEMPRE no se debe únicamente a que deseamos tener construcciones formales elegantemente construidas. Además deseamos que estas construcciones estén claramente definidas en el contexto de nuestro lenguaje. Para hacernos entender es necesario respetar las reglas de la gramática de la economía puesto que sólo de esta manera podremos comparar los resultados de distintos modelos, es decir, comparar el significado de distintas frases, y así progresar en nuestro entendimiento de la realidad.

Vamos a empezar definiendo el entorno y luego pasaremos a la definición de equilibrio. Un entorno en un modelo económico es una especificación de *i)* las preferencias de los agentes, *ii)* las tecnologías, *iii)* las dotaciones de la economía y *iv)* la información. De estos cuatro elementos hay uno que no va a cambiar a lo largo del curso, y es la información. Vamos a suponer que los agentes tienen conocimiento perfecto de todo lo que ha sucedido, de lo que sucede y de lo que está por suceder, es decir, hay lo que en la literatura se conoce como "perfect foresight". Vamos ahora con aquellos elementos del entorno que sí irán cambiando en lo sucesivo

3.1.1 El entorno

Las preferencias

Vamos a considerar que hay un número muy grande de consumidores y que se dan las propiedades para poder agregarlos en un único agente representativo¹. Este agente representativo tiene como función de utilidad

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

donde c_t es el consumo realizado en el momento t y $\beta \in (0, 1)$ es el factor de descuento intertemporal de utilidad. Para muchos ejemplos definiremos

$$u(c_t) = \frac{c_t^{1-\eta} - 1}{1-\eta}$$

donde $\eta > 0$ es el coeficiente que indica el grado de concavidad de la función de utilidad. En entornos donde hay incertidumbre en la información este coeficiente se interpreta de modo natural como el coeficiente de aversión relativa al riesgo.

Las tecnologías

La tecnología agregada del producto interior del país la representaremos como una Coob-Douglas puesto que por un lado así lo recomiendan los datos, y por otro las propiedades de esta tecnología son compatibles con nuestra teoría de la empresa. Así, el producto total se escribe como:

$$y_t = Ak_t^\alpha l_t^{1-\alpha},$$

donde k_t y l_t son el stock de capital y la cantidad de trabajo en el momento t respectivamente. El parámetro A se conoce como el coeficiente de productividad agregada y el parámetro α es la participación de las rentas del capital en la renta nacional. Más adelante se explica por qué.

Las dotaciones

En cada unidad de tiempo la economía posee una unidad de trabajo $l_t = 1$ y al inicio del periodo $t = 0$, hay k_0 unidades de capital.

Información

Información perfecta.

¹Estas condiciones son que o bien los agentes tienen idénticas preferencias y las dotaciones de cada uno de ellos son arbitrarias, o bien las preferencias son distintas pero homotéticas y las dotaciones son colineales, es decir, que las dotaciones de un individuo son las mismas que las de otro multiplicadas por un número real positivo, y por tanto si las representásemos caerían todas en un línea.

El problema del planificador social El problema del planificador social es maximizar la función de utilidad del agente representativo sujeta a la restricción de factibilidad de la economía que dice que lo consumido y lo ahorrado no puede exceder a lo producido.

$$\max_{c_t} \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\eta} - 1}{1-\eta}$$

$$\begin{aligned} s.a. \quad c_t + i_t &\leq y_t \\ k_{t+1} &\leq i_t + (1-\delta)k_t \\ y_t &\leq Ak_t^\alpha \\ k_0 &\text{ dado} \end{aligned}$$

Podemos sustituir i_t de la segunda restricción e y_t de la tercera restricción en la primera para obtener el siguiente problema, equivalente y más sencillo como:

$$\max_{c_t} \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\eta} - 1}{1-\eta}$$

$$\begin{aligned} s.a. \quad c_t + k_{t+1} - (1-\delta)k_t &\leq Ak_t^\alpha \\ k_0 &\text{ dado} \end{aligned}$$

Escribimos la función auxiliar de Lagrange como:

$$L = \max_{\{c_t, k_t\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t \left(\frac{c_t^{1-\eta} - 1}{1-\eta} - \lambda_t (c_t + k_{t+1} - (1-\delta)k_t - Ak_t^\alpha) \right)$$

Para hallar las condiciones de primer orden primero escribimos los términos de la suma en los que nos encontremos con variables con subíndice t . No debemos olvidar que al derivar con respecto a k_t nos encontraremos ese término en $t-1$. Desarrollando la función auxiliar de Lagrange para los términos t y $t-1$:

$$\begin{aligned} L = \quad &\dots + \beta^{t-1} \left(\frac{c_{t-1}^{1-\eta} - 1}{1-\eta} - \lambda_{t-1} (c_{t-1} + k_t - (1-\delta)k_{t-1} - Ak_{t-1}^\alpha) \right) + \\ &\beta^t \left(\frac{c_t^{1-\eta} - 1}{1-\eta} - \lambda_t (c_t + k_{t+1} - (1-\delta)k_t - Ak_t^\alpha) \right) + \dots \end{aligned}$$

Ahora que los tenemos delante derivamos para obtener

$$\begin{aligned}\frac{\partial L}{\partial c_t} &= c_t^{-\eta} - \lambda_t = 0, \\ \frac{\partial L}{\partial k_t} &= \beta \lambda_t (A\alpha k_t^{\alpha-1} + (1 - \delta)) - \lambda_{t-1} = 0, \\ \frac{\partial L}{\partial \lambda_t} &= c_t + k_{t+1} - (1 - \delta)k_t - Ak_t^\alpha = 0.\end{aligned}$$

Más una condición de transversalidad. Sustituyendo el valor del multiplicador de Lagrange $\lambda_t = c_t^{-\eta}$ en la segunda condición obtenemos:

$$\begin{aligned}\beta c_t^{-\eta} (A\alpha k_t^{\alpha-1} + (1 - \delta)) - c_{t-1}^{-\eta} &= 0, \\ c_t + k_{t+1} - (1 - \delta)k_t - Ak_t^\alpha &= 0.\end{aligned}$$

Estas dos ecuaciones junto con el valor de k_0 dado, son el sistema que tenemos que resolver y para el que escribiremos un programa que a su vez usará o bien el programa **newton.m**, o bien el programa **secant.m**. Conviene no perder la perspectiva de lo que estamos haciendo y recordar que nuestra misión hoy es resolver exactamente este sistema de ecuaciones. No obstante es interesante entender que podríamos haber llegado a este sistema de otra manera. Podríamos haber resuelto el equilibrio competitivo para lo cual necesitamos definir el segundo aspecto esencial de una economía y es nuestra definición de equilibrio. En este caso plantearíamos un problema descentralizado en el que las empresas contratan trabajo y alquilan capital para desarrollar un plan de producción maximizador de beneficios y los consumidores maximizan su utilidad dados los ingresos que obtienen derivados del trabajo ofertado en la empresa y de las rentas del capital alquilado a la empresa.

3.1.2 El equilibrio

Un equilibrio para nuestra economía es una secuencia $\{c_t, l_t, k_t, r_t, w_t\}_{t=0}^\infty$, tal que

1. Dado k_0 y dados $\{r_t, w_t\}_{t=0}^\infty$, el consumidor representativo resuelve:

$$\begin{aligned}\max_{\{c_t, k_{st}\}_{t=0}^\infty} & \sum_{t=0}^{\infty} \beta^t \frac{c_t^{1-\eta} - 1}{1 - \eta} \\ \text{s.a. } & l_{st} = 1 \\ & c_t + k_{st+1} = w_t l_{st} + r_t k_{st}\end{aligned}$$

2. Dados $\{r_t, w_t\}_{t=0}^{\infty}$, la empresa representativa resuelve

$$\max_{k_t, l_t} Ak_{dt}^{\alpha} l_{dt}^{1-\alpha} + (1 - \delta)k_{dt} - w_t l_{dt} - r_t k_{dt}$$

3. Todos los mercados vacían

$$\begin{aligned} l_{st} &= l_{dt} = l_t \\ k_{st} &= k_{dt} = k_t \\ c_t + k_{t+1} - (1 - \delta)k_t &= Ak_t^{\alpha} l_t^{1-\alpha} \end{aligned}$$

El equilibrio competitivo Para resolver el equilibrio competitivo tenemos que tener en cuenta que una de las condiciones de vaciado de mercado sobra puesto que la ley de Walras asegura que si dos de estos mercados están en equilibrio, el restante también lo está. Es por esto por lo que hemos considerado al bien de consumo como numerario.

Del problema de la empresa representativa obtenemos, una vez que hemos igualado las ofertas a las demandas,

$$w_t = (1 - \alpha)A \left(\frac{k_t}{l_t} \right)^{\alpha}, \quad (23)$$

$$r_t = \alpha A \left(\frac{k_t}{l_t} \right)^{\alpha-1} + (1 - \delta). \quad (24)$$

Recordad que α era la participación de la renta del capital en la renta nacional. Ahora podemos ver por qué. Definimos $R_t = r_t - (1 - \delta)$ y multipliquemos en ambos miembros de la ecuación 24 por k_t obtenemos $R_t k_t = \alpha y_t$, de donde $\alpha = R_t k_t / y_t$, es decir, la fracción de la renta nacional que son rentas del capital. Por otro lado, el consumidor representativo resolvería independientemente su problema de optimización, para lo cual plantea el lagrangiano

$$L = \max_{\{c_t, k_t\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t \left(\frac{c_t^{1-\eta} - 1}{1-\eta} + \lambda_t (w_t + r_t k_t - c_t - k_{t+1}) \right),$$

de donde obtiene las condiciones de primer orden

$$\begin{aligned} \frac{\partial L}{\partial c_t} &= c_t^{-\eta} - \lambda_t = 0 \\ \frac{\partial L}{\partial k_t} &= \lambda_t \beta r_t - \lambda_{t-1} = 0 \\ \frac{\partial L}{\partial \lambda_t} &= w_t + r_t k_t - c_t - k_{t+1} = 0 \end{aligned} \quad (25)$$

y una condición de transversalidad. Sustituyendo λ_t de la primera condición en la segunda obtenemos

$$\beta c_t^{-\eta} r_t - c_{t-1}^{-\eta} = 0,$$

si sustituimos en la última expresión el valor de r_t obtenido en 24, teniendo en cuenta que $l_t = 1$, y luego sustituimos los valores de r_t y w_t en 25 obtenemos un nuevo sistema;

$$\begin{aligned} \beta c_t^{-\eta} (\alpha A k_t^{\alpha-1} + (1 - \delta)) - c_{t-1}^{-\eta} &= 0, \\ c_t + k_{t+1} - (1 - \delta)k_t - A k_t^\alpha &= 0, \end{aligned}$$

junto con el valor de k_0 dado. Es decir, EXACTAMENTE el mismo sistema que encontramos al resolver el problema del planificador social! Sabemos que el equilibrio competitivo es Pareto eficiente, además las condiciones del problema hacen que el equilibrio sea único, de modo que cuando resolvemos el problema del planificador estamos encontrando una asignación Pareto eficiente y por tanto el único equilibrio competitivo. Estos son los teoremas de la economía del bienestar.

3.2 El programa

A la hora de resolver el problema de cómo construir un programa para esta economía se nos plantean dos dificultades de fácil solución. Para verlas vamos primero a inspeccionar visualmente el sistema de ecuaciones que queremos resolver:

$$\beta c_t^{-\eta} (\alpha A k_t^{\alpha-1} + (1 - \delta)) - c_{t-1}^{-\eta} = 0, \quad (26)$$

$$c_t + k_{t+1} - (1 - \delta)k_t - A k_t^\alpha = 0, \quad (27)$$

$$k_0 \quad \text{dado}$$

Si introducimos c_t de 27 en 26 llegamos a una nueva ecuación

$$\beta (A k_t^\alpha + (1 - \delta)k_t - k_{t+1})^{-\eta} (\alpha A k_t^{\alpha-1} + (1 - \delta)) - (A k_{t-1}^\alpha + (1 - \delta)k_{t-1} - k_t)^{-\eta} = 0, \quad (28)$$

que como se puede ver es una ecuación en diferencias de segundo orden, es decir, es del tipo $\varphi(k_t, k_{t+1}, k_{t+2}) = 0$. Nuestra primera dificultad es que

para resolver este tipo de ecuaciones necesitamos tener información sobre, al menos, dos puntos de la trayectoria que la ecuación describe². Uno lo conocemos y es k_0 que está dado al inicio de la economía, pero nos falta otro. El punto que nos falta lo podemos deducir de las condiciones del estado estacionario de la economía. Una economía está en un estado estacionario cuando a partir de un $\tau > T$, $V_\tau = V_{\tau+1}$, donde V_τ es cualquier variable del modelo. Esto significa que para calcular el estado estacionario es suficiente con eliminar los subíndices t del sistema de ecuaciones formado por 26 y 27, o bien eliminar los subíndices directamente de la ecuación 28. El resultado sería,

$$\beta (\alpha A k^{\alpha-1} + (1 - \delta)) - 1 = 0,$$

de donde podemos despejar el nivel de k para el cual la economía es estacionaria

$$k_{ss} = \left[\frac{1}{\alpha A} \left(\frac{1}{\beta} - (1 - \delta) \right) \right]^{\frac{1}{\alpha-1}} \quad (29)$$

Ahora ya tenemos dos valores para k_t , que son el par (k_0, k_{ss}) y que se corresponden con el stock de capital al inicio de la economía y el stock en el estado estacionario. La segunda dificultad reside en que estos modelos entran en el estado estacionario en tiempo infinito, de modo que el valor de $k_T = k_{ss}$ sólo cuando han pasado un número infinito de periodos (cuando $T \rightarrow \infty$). La solución a este problema es la de forzar a la ecuación a entrar en el estado estacionario en un T suficientemente grande fijado por nosotros ($T = 30$ suele ser suficiente). Ya tenemos todo lo que necesitamos para empezar a construir un programa que resuelva nuestro modelo. Antes de continuar es conveniente hacer notar la importancia de la ecuación 29. Esta ecuación dice que el stock de capital de estado estacionario es sólo función de parámetros tecnológicos, de la tasa de depreciación del capital y del parámetro de descuento intertemporal de utilidad, o sea, de la impaciencia de los individuos. Este modelo dice, por tanto, que si nosotros observamos diferencias de renta en el largo plazo entre países, es debido a que, o bien la tecnología utilizada por los distintos países es también distinta, o bien la tasa de depreciación es distinta, o bien la tasa de ahorro es distinta. Podemos comprobar como ejercicio que la diferencia que tiene que haber entre los parámetros de un país y de otro para explicar diferencias de renta del 1000% como se observa en

²En general, para resolver una ecuación en diferencias de orden n necesitaremos conocer al menos n puntos de la trayectoria.

algunos casos es absurda. Es decir, en este modelo falta algo que sea capaz de explicar las diferencias de renta entre países.

Nosotros ya sabemos resolver un sistema de ecuaciones usando el método de Newton. Resolver una ecuación en diferencias no es muy distinto como vamos a ver. La lógica va es la siguiente: construimos un vector de dimensión $T + 1$ que contenga los valores $K^1 = [k_0, k_{ss}, \dots, k_{ss}]$. A K^1 lo vamos a considerar la semilla de nuestro sistema (que por cierto no es muy buena) para resolver el sistema

$$\begin{aligned}\varphi(k_0, k_1, k_2) &= 0 \\ \varphi(k_1, k_2, k_3) &= 0 \\ \varphi(k_2, k_3, k_4) &= 0 \\ &\vdots \\ \varphi(k_{T-1}, k_T, k_{T+1}) &= 0.\end{aligned}$$

Como k_0 , y k_{T+1} son conocidos ($k_{T+1} = k_{ss}$) nos encontramos con un sistema con un número igual de ecuaciones que incógnitas y por tanto podemos resolverlo escribiendo un pequeño programa en el que se especifiquen las T ecuaciones, luego llamamos a **secant.m** y así obtenemos la solución.

3.2.1 Ejemplo 5 (Newton-Raphson)

Resolver la ecuación 28

```
%mbasico.m
```

```
%Este programa resuelve el modelo básico de crecimiento
```

```
% inf
```

```
%max sum beta^t (c(t)^(1-eta)-1)/(1-eta)
```

```
% t=0
```

```
% s.a. c(t)+k(t+1)-(1-delta)*k(t)<=A*k^alpha
```

```
% k(0) dado
```

```
clear
```

```
%Definición de parámetros del modelo
```

```
A = 10;
```

```
alpha = 0.35;
```

```
delta = 0.06;
```

```
eta = 0.99;
```

```
beta = 0.96;
```



```

%Definición de parámetros del programa
maxit = 1000;
crit = 1e-3;
T = 30;

%Definición de k0 y kss
kss = ((A*beta*alpha)/(1-(1-delta)*beta))^(1/(1-alpha));
k0 = 0.8*kss;

%Definición de la semilla
x0 = [k0 kss*ones(size(1:T-1))];

%Llamada a secant.m
param = [A alpha delta eta beta T k0 kss];
sol = secant('cpo', x0, param, crit, maxit)

%Resultados
k = [k0; sol; kss];
y = A*k.^alpha;
i = k(2:T+1)-(1-delta)*k(1:T);
c = y(1:T)-i;

subplot(2,2,1)
plot(k)
title('Capital')
subplot(2,2,2)
plot(y)
title('Producto')
subplot(2,2,3)
plot(i)
title('Inversión')
subplot(2,2,4)
plot(c)
title('Consumo')

```

En el programa **cpo.m** hemos introducido las condiciones de primer orden del programa. Como siempre el comando **function f=cpo(z, p)** evalúa **f** con los valores heredados de la semilla que fue definida en **mbasico.m** y

usa los parámetros definidos en **param** definidos en el mismo programa. Es importante hacer notar que el último comando en **cpo.m** transforma el vector fila de **f** evaluada en **z** con parámetros **p** en un vector columna, es decir **f=f'**, que asigna a **f** el valor de su traspuesta **f'**. Esto es así porque el programa **secant.m** exige que **f** sea un vector columna, en tanto que el vector de **f** generado en **cpo.m** es un vector fila, debemos hacer esta transformación. Si en vez de utilizar el programa **secant.m** usamos el programa **fsolve.m** incorporado en nuestra librería de programas de MatLab esta transformación no sería necesaria.

```
function f=cpo(z, p)
%función cpo.m donde se encuentran nuestras
%condiciones de primer orden
%Asignación de parámetros
A = p(1);
alpha = p(2);
delta = p(3);
eta = p(4);
beta = p(5);
T = p(6);
k0 = p(7);
kss = p(8);
%Asignación de variables
for t=1:T
k(t) = z(t);
end
k(T+1) = kss;
    f(1)=beta*(A*k(1)^alpha+(1-delta)*k(1)-k(2))^(eta)*...
    (alpha*A*k(1)^(alpha -1)+(1-delta))- ...
    (A*k0^alpha+(1-delta)*k0-k(1))^(eta);
    for t=2:T
    f(t)=beta*(A*k(t)^alpha+(1-delta)*k(t)-k(t+1))^(eta)*...
    (alpha*A*k(t)^(alpha -1)+(1-delta))- ...
    (A*k(t-1)^alpha+(1-delta)*k(t-1)-k(t))^(eta);
    end
f=f';
```

3.2.2 Ejemplo 6 (Gauss-Seidel)

En esta sección vamos a explicar y resolver el modelo básico de crecimiento que ya hemos estudiado y resuelto, pero ahora vamos a introducir otra técnica de resolución: el método de Gauss-Seidel. Volvamos a escribir el sistema de ecuaciones que queremos resolver, este era:

$$\begin{aligned}\varphi(k_0, k_1, k_2) &= 0 \\ \varphi(k_1, k_2, k_3) &= 0 \\ \varphi(k_2, k_3, k_4) &= 0 \\ &\vdots \\ \varphi(k_{T-1}, k_T, k_{T+1}) &= 0.\end{aligned}$$

Proponemos como semilla una secuencia completa para k_t a la que llamaremos $K^1 = [k_0, k_1^1, k_2^1, \dots, k_T^1, k_{ss}]$, donde cada elemento tiene como subíndice el periodo t correspondiente y como superíndice la iteración en la que nos encontramos. Vamos a la primera ecuación del sistema y la resolvemos independientemente del resto de ecuaciones, usando el método de Newton-Raphson, para calcular el k_1^2 que es solución de la ecuación $\varphi(k_0, k_1^2, k_2^1) = 0$, es decir, tomamos como dado k_0 y el k_2^1 de la semilla en la primera iteración. Guardamos el valor encontrado y resolvemos $\varphi(k_1^2, k_2^2, k_3^1) = 0$ para k_2^2 . Repetimos este procedimiento hasta llegar a la ecuación $\varphi(k_{T-1}^2, k_T^2, k_{ss}) = 0$. Una vez llegados a este punto tendremos un nuevo vector $K^2 = [k_0, k_1^2, k_2^2, \dots, k_T^2, k_{ss}]$. Evaluamos $|K^1 - K^2|$ y si esta diferencia es menor que nuestro criterio de convergencia tomamos a K^2 como la solución del problema, si no es así, tomamos a K^2 como la nueva semilla para iniciar de nuevo todo el proceso. A continuación aparecen los programas para resolver el modelo de crecimiento simple con la salvedad de que ahora la función de utilidad instantánea es $u(c_t) = \ln c_t$, que es el caso particular de $\eta = 1$ en el ejemplo 5.

```
%gseid.m Programa para resolver una versión sencilla del problema
% del planificador social usando el método de Gauss-Seidel.
% El problema a resolver es:
% inf
% max sum beta^t ln c(t)
% t=0
% s.a. c(t)+i(t)=A*k(t)^alpha
```

```

%  $k(t+1)=i(t)+(1-\delta)*k(t)$ 
% con  $k_0$  igual al 80% del stock de capital de estado estacionario.
%
% resultados: archivo gsout.mat con trayectoria para las variables
% Programa de Carlos Urrutia
clear
%Parámetros del modelo
alpha = 0.35;
beta = 0.99;
delta = 0.06;
A = 10;
%Parámetros del programa
maxit = 1000;
crit = 1e-4;
T = 60;
%Cálculo del stock de capital de estado estacionario
%del stock inicial  $k_0$ , y de la semilla
kss = ((A*beta*alpha)/(1-(1-delta)*beta))^(1/(1-alpha));
k0 = 0.8*kss;
for t=1:T+1
%  $kold(t)=k_0+(kss-k_0)*(t-1)/T$ ;
kold(t)=kss;
end
kold(1)=k0;
knew(1)=k0;
%Iteraciones usando Gauss-Seidel
for i=1:maxit
for t=2:T
param=[alpha; beta; delta; A; knew(t-1); kold(t+1)];
knew(t)=secant('gscpo', kold(t), param, crit, 100);
end

plot(kold); pause(0.1)
if norm(kold(2:T)-knew(2:T))<crit*mean(kold) break; end
kold=knew;
kold(T+1) = kss;
end

```

end

El programa **gscpo.m** contiene la ecuación de Euler del problema. Como se puede ver este programa es más simple que **cpo.m**, pero el procedimiento completo es bastante más lento

```
function y=gscpo(kt1, p)
%gscpo.m Función requerida por gseid que contiene la Ecuación de
Euler
y=zeros(1,1);
alpha = p(1);
beta = p(2);
delta = p(3);
A = p(4);
kt = p(5);
kt2 = p(6);
y = A*kt1^alpha+(1-delta)*kt1-beta*(A*kt1^alpha-kt1+ ...
(1-delta)*kt)*(A*alpha*kt1^(alpha-1)+(1-delta))-kt2;
```

El método de Gauss-Seidel tiene dos inconvenientes sobre el método de Newton-Raphson aplicado sobre todo el programa, en primer lugar es más lento como ya se ha comentado y en segundo lugar no tiene garantizada la convergencia cuando el sistema de ecuaciones que queremos resolver es más complejo.

Los resultados que obtenemos se pueden ver bien en los gráfico producidos por el programa **mbasico.m**.

FIGURA aquí.

La economía comienza en un periodo en el que el stock de capital es inferior al stock que la economía tendrá en el estado estacionario. El planificador social asigna una cantidad de consumo en el primer periodo inferior al consumo que habrá en el estado estacionario para que, a través de la inversión, se vaya acumulando capital. De este modo, la cantidad de capital disponible en segundo periodo será superior y por tanto la producción sea mayor. Con una mayor producción en el segundo periodo, la cantidad de consumo será también mayor y la inversión un poco menor que la que teníamos en el primer periodo. La inversión decrece pero sigue habiendo inversión neta en capital.

El proceso continúa hasta el punto en que la inversión sea $i_{ss} = \delta k_{ss}$, es decir, hasta que el esfuerzo en inversión sirva sólo al propósito de reemplazar el capital consumido en el proceso productivo.

4 Lección cuarta

En esta lección vamos a estudiar una economía muy similar a la estudiada en la lección anterior con la salvedad de que esta será una economía abierta. La economía abierta tiene, como vamos a ver, unas propiedades muy distintas a las de la economía cerrada. La propiedad fundamental es que si es posible importar bienes y capital, es posible que una economía con un stock inicial de capital bajo, quiera correr un déficit por cuenta corriente en los primeros periodos, mantener un alto nivel de consumo, y pagar más tarde al resto del mundo con un superávit. En una economía cerrada, para conseguir acumular un stock de capital hasta llegar al de estado estacionario, era necesario que el consumo en los primeros periodos fuera bajo y la inversión alta, acumular capital para el siguiente periodo y continuar con el plan de ahorro que llegara finalmente al stock de capital de estado estacionario. Los supuestos que vamos a realizar sobre esta economía son similares a los que ya hicieramos en el modelo de crecimiento básico. Existe un consumidor representativo que maximiza su utilidad intertemporal y que posee los medios de producción capital y trabajo. Hay también una empresa representativa que minimiza costes y tiene beneficios iguales a cero. Además es posible importar el bien de consumo, que también lo es de inversión para, o bien consumir, o bien construir el stock de capital, o ambas cosas a la vez. Podemos plantear el equilibrio competitivo o el problema del planificador social para obtener la solución al problema. El problema del planificador social es:

$$\begin{aligned} \max_{\{c_t, k_t, b_t\}} \quad & \sum_{t=0}^{\infty} \beta^t u(c_t) \\ \text{s.a.} \quad & c_t + i_t + b_{t+1} \leq y_t + (1+r)b_t \\ & k_{t+1} \leq i_t + (1-\delta)k_t \\ & y_t \leq Ak_t^\alpha \\ & k_0, b_0 \text{ dados} \end{aligned}$$

La primera restricción introduce dos elementos nuevos, por un lado tenemos un bono, o activo del exterior que recoge la posición acreedora o deudora

de nuestro país, y por otro lado tenemos r , que es el tipo de interés internacional. Vamos a considerar a r como un parámetro que está determinado en el resto del mundo y al que las decisiones de participar en el mercado de capitales internacional de nuestra economía no afectan. Es decir, estamos considerando que nuestra economía es una economía pequeña y abierta al resto del mundo. Su tamaño es tan pequeño en relación al resto del mundo que ninguna decisión de consumo o inversión de nuestra economía afecta al tipo de interés internacional. Vamos a suponer, como lo hicimos en lecciones anteriores que el capital es reversible, es decir, el bien de consumo se puede transformar en capital y el capital en bien de consumo sin necesidad de una tecnología que transforme uno en otro, o dicho de otra manera, los alimentos no consumidos son capital y el capital se puede comer. Como se puede ver, si la posición acreedora o deudora de nuestro país fuera siempre igual a cero $b_t = 0$ para todo t , estaríamos en la economía de la lección anterior. Vamos a suponer que la economía, al inicio del primer periodo, no tiene ninguna posición frente al resto del mundo, $b_0 = 0$. Queremos resolver el modelo y comprobar si este modelo tiene predicciones realistas en relación al comportamiento que observamos en las economías abiertas. Para resolver el modelo planteamos una versión más simple del problema sustituyendo la segunda y tercera restricción en la primera. Entonces tendríamos:

$$\begin{aligned} \max_{\{c_t, k_t, b_t\}} \sum_{t=0}^{\infty} \beta^t u(c_t) \\ \text{s.a. } c_t + k_{t+1} - (1 - \delta)k_t + b_{t+1} &\leq Ak_t^\alpha + (1 + r)b_t \\ &k_0, \text{ y } b_0 \text{ dados} \end{aligned}$$

Planteamos la función auxiliar de Lagrange como:

$$L = \max_{\{c_t, k_t, b_t\}} \sum_{t=0}^{\infty} \beta^t [u(c_t) + \lambda_t (Ak_t^\alpha + (1 + r)b_t - c_t - k_{t+1} + (1 - \delta)k_t - b_{t+1})]$$

para obtener las condiciones de primer orden del problema debemos computar las derivadas parciales de la función auxiliar de Lagrange con respecto a la variable de elección $\{c_t, k_t, b_t\}$, para lo cual es preciso darse cuenta de que en el momento $t - 1$ tenemos términos con subíndice t . Es decir, si desarrollamos la función de Lagrange encontraremos:

$$\begin{aligned} \dots + \beta^{t-1} [u(c_{t-1}) + \lambda_{t-1} (Ak_{t-1}^\alpha + (1 + r)b_{t-1} - c_{t-1} - k_t + (1 - \delta)k_{t-1} - b_t)] + \\ \beta^t [u(c_t) + \lambda_t (Ak_t^\alpha + (1 + r)b_t - c_t - k_{t+1} + (1 - \delta)k_t - b_{t+1})] + \dots \end{aligned}$$

Derivando con respecto a nuestras variables de elección encontramos:

$$\begin{aligned}\frac{\partial L}{\partial c_t} &= u'(c_t) - \lambda_t = 0 \\ \frac{\partial L}{\partial k_t} &= \beta \lambda_t (A \alpha k_t^{\alpha-1} + (1 - \delta)) - \lambda_{t-1} = 0 \\ \frac{\partial L}{\partial b_t} &= \beta \lambda_t (1 + r) - \lambda_{t-1} = 0\end{aligned}$$

y dos condiciones de transversalidad. Despejando λ_t de la primera condición y sustituyendo en la tercera obtenemos

$$\beta(1 + r)u'(c_t) - u'(c_{t-1}) = 0$$

Vemos que en un estado estacionario se tiene que cumplir que $u'(c_\tau) = u'(c_{\tau-1})$ o bien que $\beta(1 + r) = 1$. Es decir, $\beta(1 + r) = 1$ es una condición necesaria para que la economía pueda ser estacionaria. La razón está bastante clara: si la tasa a la que descontamos el futuro es más alta que el tipo de interés internacional querría decir que el futuro es más importante para el agente representativo que para el resto del mundo, que descuenta el futuro a una tasa $\frac{1}{(1+r)}$ más baja. En este caso nuestra economía acumularía de forma ilimitada. En caso contrario se endeudaría sin límites. No parece que esta sea una propiedad que queramos tener en el modelo. Por tanto asumimos la condición necesaria de estacionariedad y fijamos para nuestro modelo $\beta = \frac{1}{(1+r)}$. Pero ahora surge un pequeño problema. En el estado estacionario esta ecuación no nos proporciona información alguna sobre ninguna de las variables que queremos determinar, sólo proporciona información sobre el valor de algunos parámetros del modelo. La segunda condición nos dice que en el estado estacionario

$$k_{ss} = \left[\frac{A * \alpha}{r + \delta} \right]^{\frac{1}{1-\alpha}}. \quad (30)$$

Este resultado llevado a la restricción del problema de optimización nos dice que

$$c_{ss} + \delta k_{ss} = A k_{ss}^\alpha + r b_{ss} \quad (31)$$

No es mucha la información que obtenemos del sistema formado por las condiciones de primer orden. Para poder determinar las trayectorias de equilibrio tenemos que conjeturar una solución y después comprobar que efectivamente

nuestra conjetura era correcta. Podemos imaginar una economía escasa en capital en relación al resto del mundo. Si esta economía se abre al comercio internacional podría realizar en el primer periodo todas las importaciones que necesita para construir su stock de capital y además situar su patrón de consumo en el nivel de estado estacionario c_{ss} . Si esta economía opta por esta solución, está claro que en el primer periodo correría un déficit por cuenta corriente y un déficit comercial que luego saldaría a la tasa r a lo largo del resto de su vida, generando un pequeño superávit en cada periodo. Para ver que esta es la única solución vamos a escribir la restricción presupuestaria para los periodos $t = \{1, 2\}$, ya que a partir del periodo 2 la ecuación 33 se repite para siempre. La secuencia de los eventos es la siguiente: al inicio del primer periodo se dispone de una cantidad de capital k_0 y de unos activos frente al resto del mundo de b_0 , entonces se inicia el proceso productivo del primer periodo, y al final del primer periodo se toman las decisiones de consumo e inversión. La inversión se incorpora al capital con el que empieza la economía en el segundo periodo, $k_1 = i_1 + (1 - \delta)k_0$, se pone en marcha el proceso productivo y al final del segundo periodo se toman unas nuevas decisiones de inversión y consumo, y así para siempre. Por tanto las dos restricciones para el primer y segundo periodo son:

$$b_{ss} + c_{ss} + k_{ss} - (1 - \delta)k_0 = Ak_0^\alpha + (1 + r)b_0 \quad (32)$$

$$b_{ss} + c_{ss} + \delta k_{ss} = Ak_{ss}^\alpha + (1 + r)b_{ss} \quad (33)$$

Si computamos la diferencia entre 33 y 32, obtenemos:

$$0 = A(k_{ss}^\alpha - k_0^\alpha) + (1 - \delta)(k_{ss} - k_0) + (1 + r)(b_{ss} - b_0) \quad (34)$$

Donde k_{ss} es el encontrado en la ecuación 30 y el par (k_0, b_0) es conocido. De la ecuación 34 podemos obtener el valor de b_{ss} , que sería compatible con el plan propuesto. Este valor viene dado por

$$b_{ss} = b_0 - \frac{A * (k_{ss}^\alpha - k_0^\alpha) + (1 - \delta) * (k_{ss} - k_0)}{(1 + r)}$$

Como se puede ver, cuanto más parecido fuera el estado inicial del stock k_0 en relación al estado final del stock k_{ss} , menor sería la cantidad de capital importado por esta economía. Además, cuanto mayor fuera la cantidad de activos frente al resto del mundo poseídos por este país frente al resto del mundo también sería la menor la necesidad de financiación por parte del

resto del mundo. De la ecuación 31, obtenemos la cantidad de consumo de estado estacionario compatible con el nivel de deuda encontrado, esta sería

$$c_{ss} = A * k_{ss}^\alpha - \delta k_{ss} + r b_{ss}.$$

Como es natural, cuanto mayor fuera la necesidad de financiación menor será la cantidad de consumo de estado estacionario, dado que éste tiene que compensar las devoluciones al interés r que se producirán para siempre. El superávit comercial que debe producirse en cada periodo para mantener la cuenta corriente equilibrada será,

$$t d_{ss} = A k_{ss}^\alpha - c_{ss} - \delta k_{ss} = -r b_{ss}.$$

Veamos cuales son la predicciones del modelo para España tras la apertura en 1986 al mercado de la Unión Europea. Para ello vamos a tomar algunos hechos de la contabilidad nacional. Sabemos de las Penn World Tables de Summers y Heston que

$$\frac{k_{1992}}{y_{1992}} = 3.5$$

Por otro lado sabemos que la participación de las rentas del capital en la renta nacional fue

$$\frac{(r + \delta)k_{1992}}{y_{1992}} = 0.35$$

Si arbitrariamente hacemos $y_{1992} = 100$, entonces $k_{1992} = 350$ y podemos resolver el sistema de ecuaciones

$$y_{1992} = A k_{1992}^\alpha \tag{35}$$

$$(r + \delta)k_{1992}^{(1-\alpha)} = A\alpha \tag{36}$$

para A y α . La ecuación 35 es simplemente la función de producción y la ecuación 36 se deriva de la condición de primer orden. Para calcular los valores de A y α , simplemente fijamos los valores de $r = 0.04$, y $\delta = 0.06$. El valor de r viene dado por el tipo de interés de largo plazo en Alemania (que es la economía grande de mayor importancia para España) y el valor de δ viene de la contabilidad nacional. Una vez que hemos fijado estos dos parámetros sólo tenemos que escribir un pequeño programa con el método de newton para encontrar que

$$A = 12.8699$$

$$\alpha = 0.3500$$

Con estos valores podemos iniciar un programa muy simple para calcular los valores de las variables en el estado estacionario. Únicamente tenemos que decidir cuál será el valor de k_0 . Parece razonable pensar que España estaba a un 80% del stock de capital de estado estacionario. Además, España tenía en 1986 una posición acreedora frente al resto del mundo de un 2% del PIB, por tanto fijamos $b_0 = 2$. Con estos valores podemos calcular la cuenta corriente, el déficit comercial, el stock de capital de estado estacionario y la producción a lo largo del tiempo. Un gráfico ilustra a qué se debería parece la economía española en caso de que el modelo fuera una representación correcta de la realidad. En la Figura 1, a la izquierda obtenemos la trayectoria de equilibrio para el PIB de la economía española después de su apertura en 1986. En sólo un año debería haber alcanzado el nivel de la economía estacionaria. Esto habría sido posible si la trayectoria del stock de capital hubiera sido la que aparece en la Figura 1 derecha, en la que vemos la trayectoria del stock del capital. En un sólo año la economía española habría construido su stock de capital de estado estacionario. El plan financiero compatible con el plan de producción lo vemos en la Figura 2.

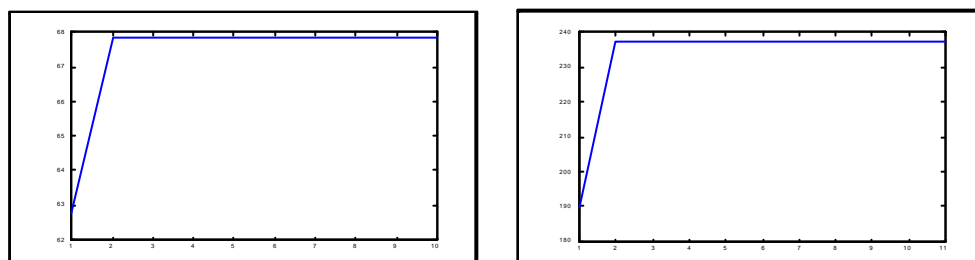


FIGURA 1

Lo más sorprendente del modelo es que predice que el déficit por cuenta corriente debería haber alcanzado un 68% del producto interior bruto el primer año para cerrarse completamente al año siguiente. Además, en este corto periodo de tiempo España debería haber construido completamente su stock de capital hasta situarlo en el estado estacionario. Ya con su stock de capital de estado estacionario habría producido un superávit por cuenta corriente cada año para ir cancelando su deuda pagando una anualidad del 2.73% del producto interior bruto. Parece que las cosas no fueron exactamente así, tal y como se puede apreciar en la Figura 3.

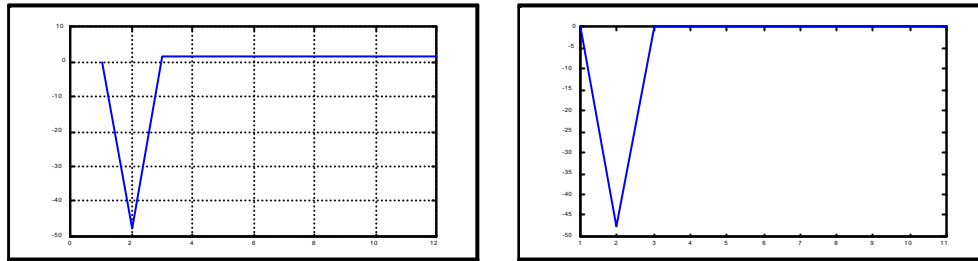


FIGURA 2

El déficit por cuenta corriente no llegó a ser el 4% del PIB. Esta cantidad no es poca pero está un poco alejada del 68% del modelo. No sólo el mínimo es mayor en los datos, sino que sucedió cerca de 7 años después.

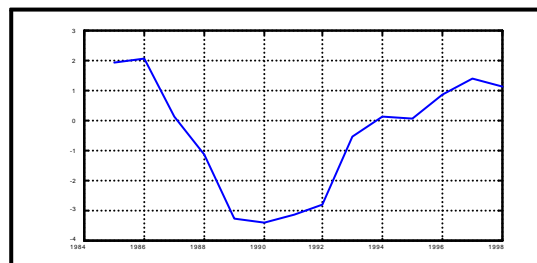


FIGURA 3

¿Qué es lo que está mal en el modelo?, o mejor expresado, ¿qué es lo que está mal en el mundo?. Son varias las cosas que nos deberían hacer pensar que todo sucede demasiado intensamente en el modelo y que todo sucede demasiado rápido. Por un lado, un país no fija un acuerdo de préstamo con un único prestamista con el que pueda negociar una anualidad eterna. Por otro lado, en caso de que ese prestamista existiera, no estaría dispuesto a realizar semejante contrato de préstamo, porque suponiendo que lo hiciera, quién garantiza al prestamista que recibirá su dinero. Una vez que el país a construido su stock de capital de estado estacionario, ¿que castigo puede recibir del prestamista en caso de no pagar?. pero no sólo eso, ¿quién puede levantar un stock de capital de estado estacionario en un sólo año?. Construir el capital lleva tiempo, y más tiempo cuanto más pobre fuera el país. Tenemos varias razones para cambiar la especificación del modelo e ir introduciendo nuevos elementos que dificulten el rapidísimo ajuste que hemos observado en nuestro primer modelo de una economía abierta. Tenemos una razón adicional para introducir un nuevo elemento en nuestro modelo, y esta es que el tipo de cambio real varió de un modo sustancial a lo largo de todo el periodo

que abarca 1986-1996. Sufrió una apreciación real de un 19%, alcanzando la máxima apreciación en 1992, para luego sufrir una depreciación real que lo llevo de nuevo a los niveles de 1986. En nuestro modelo con un único bien no podemos definir el tipo de cambio real puesto que éste es simplemente el precio del bien que se comercializa en los mercados y por la ley de un precio éste es igual en todas partes. Por tanto lo primero que nos gustaría introducir es la existencia de dos tipos de bienes distintos. Consideraremos que un bien es comercializado y su precio es por tanto el precio internacional, ya que si fuera distinto en un lugar y en otro se realizaría una operación de arbitraje beneficiosa comprando donde es barato y vendiendo donde es caro, igualando finalmente su precio. Por otro lado tendremos un bien que por su naturaleza no puede ser comercializado, como sería el caso de los edificios, las carreteras, los bienes inmuebles y en general los servicios de la economía. Estos bienes tendrán un precio distinto en cada país puesto que por su naturaleza, la ley de un precio no puede operar ya que las operaciones de arbitraje no son posibles. Los bienes transables los podremos identificar con los bienes agrícolas y los bienes industriales como manufacturas y recursos primarios. Este modelo lo definiremos y lo resolveremos en el siguiente capítulo, pero antes vamos a ver el código de MatLab con el que hemos realizado los gráficos para que podamos experimentar con tres países, otras especificaciones de los parámetros y otras funciones de producción para comprobar que efectivamente esta alta velocidad de ajuste y la alta intensidad del mismo son una propiedad del modelo y no de una especificación particular del mismo.

```
%open.m
%Este programa resuelve el modelo básico de crecimiento
% inf
%max sum beta^t (c(t)^(1-eta)-1)/(1-eta)
% t=0
% s.a. b(t+1)+c(t)+k(t+1)-(1-delta)*k(t)<=A*k^alpha+(1+r)*b(t)
% k(0) dado, b(0) dado
clear
%Definición de parámetros del modelo
A = 10;
```

```

alpha = 0.35;
delta = 0.06;
eta = 0.99;
r = 0.04;
beta = 1/(1+r);
%Definición de parámetros del programa
maxit = 1000;
crit = 1e-3;
T = 10;
%Definición de k0, b0 y kss
kss = ((r+delta)/(A*alpha))^(1/(alpha-1));
k0 = 0.8*kss;
b0 = 0;
bss = -(A*(kss^alpha-k0^alpha)+(1-delta)*(kss-k0))/(1+r)-b0;
css = A*kss^alpha-delta*kss+r*bss;
%Definición de la semilla
kg = [k0 kss*ones(size(1:T-1))];
b = [b0 bss*ones(size(1:T+1))];
%Llamada a secant.m
param = [A alpha delta eta beta T k0 kss css b];
sol = secant('opencpo', kg, param, crit, maxit);
k = [k0; sol];
ca = [0 b(2:T+1)-b(1:T)];
y(1) = A*k0^alpha;
y = [y(1); A*k(2:T).^alpha];
td(1) = 0;
td(2) = A*k0^alpha-css+(1-delta)*k0-kss;
td = [td(1); td(2); A*k(2:T+1).^alpha-css-delta*kss];
figure(1)
plot(y)
title('Producto')
figure(2)
plot(k)
title('Capital')
figure(3)
plot(ca)
title('Cuenta corriente')
figure(4)

```

```
plot(td)
title('déficit comercial')
```

Este problema podríamos haberlo resuelto sin necesidad de recurrir al programa **secant.m** ya que de las ecuaciones expuestas en el texto podemos derivar los valores de equilibrio para todas las variables en cualquier momento del tiempo. La razón por la que lo hemos hecho de este modo es para mostrar que efectivamente la semilla propuesta, que es la solución, no se ve modificada después de introducirla en las condiciones de equilibrio. Es decir, la semilla es la solución al problema de optimización. El programa **opencpo.m** es como sigue:

```
function f=opencpo(z, p)
    %función cpo.m donde se encuentran nuestras
    %condiciones de primer orden
    %Asignación de parámetros
    A = p(1);
    alpha = p(2);
    delta = p(3);
    eta = p(4);
    beta = p(5);
    T = p(6);
    k0 = p(7);
    kss = p(8);
    css = p(9);
    b0 = p(10);
    r = 1/beta-1;
    for i=11:length(p)
        b(i-10)=p(i);
    end
```

```

%Asignación de variables
for t=1:T
k(t) = z(t);
end
k(T+1) = kss;
f(1)=b(1)+css+k(1)-(1-delta)*k0-A*k0^alpha-(1+r)*b0;
for t=1:T
f(t)=b(t+1)+css+k(t+1)-(1-delta)*k(t)-A*k(t)^alpha-(1+r)*b(t);
end
f=f';

```


5 Lección quinta

En esta lección vamos a cambiar la definición del entorno con la finalidad doble de por un lado poder definir el tipo de cambio real y por otro tratar de frenar el ajuste de nuestra economía desde un estado inicial al estado estacionario final. Como hemos visto en la lección anterior el ajuste de nuestra economía abierta es demasiado intenso y demasiado veloz. Tanto que debemos pensar que existe una diferencia radical entre la manera en la que hemos concebido nuestra economía y el comportamiento en la realidad de las variables modelizadas. Para disminuir la intensidad y la velocidad del ajuste vamos a introducir en este capítulo bienes comercializados y bienes no comercializados. Las variables contables que se asocian a unos y otros serán productos agrícolas e industriales en el primer grupo y construcción y servicios en el segundo. Es evidente que los edificios, los puentes, las estaciones de ferrocarril, los embalses, los aeropuertos etc, no pueden ser transportados de un país a otro. Los servicios de peluquería y manicura, de consultoría fiscal y de agencia en general, ni los turísticos ni hosteleros y un largo etcétera pueden ser llevados de un lugar a otro y por ellos los vamos a identificar en nuestro modelo con los bienes no comercializados. Esto nos permitirá dos cosas, por un lado definir el tipo de cambio real y así comprobar si nuestro modelo de una economía abierta tiene algo razonable que decirnos en relación con la otra gran variable de una economía abierta; el tipo de cambio real. Por otro lado podemos albergar la esperanza de que al tener algunos bienes que no son comercializados, la velocidad de ajuste al estado estacionario sea menor, ya que ahora no podremos importar tantos bienes como deseemos corriendo un déficit por cuenta corriente tan grande como queramos. Ahora tendremos dos tecnologías en nuestra economía con la que se producirán bienes comercializados con una de ellas y bienes no comercializados con la otra.

Los bienes comercializados serán denotados por $c_{T,t}$ y serán producidos con el capital disponible en ese sector, por tanto $c_{T,t} = F_T(k_{T,t})$. Análogamente los bienes no comercializados serán, denotados por $c_{N,t}$ y producidos con la tecnología $c_{N,t} = F_N(k_{N,t})$. Adicionalmente a estas dos novedades de nuestro modelo tendremos que incluir una restricción de factibilidad que nos impida usar más capital en un sector y otro del que hay disponible en la economía, por tanto debe suceder que $k_t \leq k_{T,t} + k_{N,t}$. A continuación vamos a definir el tipo de cambio real y después pasaremos a describir el modelo del planificador social, encontrar las condiciones de primer orden, y resolver

el modelo escribiendo un código de ordenador que nos permita computar las trayectorias de equilibrio y compararlas con la realización del mundo.

5.1 El tipo de cambio real

Al igual que hiciéramos en la lección cuarta vamos a asumir que la economía que estamos estudiando es una economía pequeña. Es decir, no estamos estudiando ni la economía de los Estados Unidos ni la economía de Alemania que son las dos únicas economías consideradas en la profesión como grandes. Para nuestra economía sólo existe otra economía gigante a la que denominaremos resto del mundo. En el resto del mundo también se producen bienes transables y bienes no transables, y cada uno de estos bienes tiene un precio. Vamos a aplicar la ley de un precio para el bien comercializable, ya que si existieran diferencias de precio entre nuestra economía pequeña y el resto del mundo una operación de arbitraje los acabaría igualando. Con el bien no comercializable no sucederá lo mismo puesto que por nuestra construcción no será posible realizar esas operaciones de arbitraje que aseguran la operatividad de la ley de un precio.

El tipo de cambio real se define como:

$$RER_t = NER_t \frac{p_{row}}{p_{esp}}$$

El término NER_t será el tipo de cambio nominal entre la peseta y el resto del mundo. Como resto del mundo podemos considerar Alemania, una cesta de tipos de cambio nominales de los principales países europeos, una cesta que incluya a los Estados Unidos, etc. p_{row} es un índice de precios del resto del mundo que al igual que el tipo de cambio nominal puede corresponderse con Alemania, con los principales países europeos, etc, siempre en concordancia con la definición de tipo de cambio nominal. p_{esp} será un índice de precios para España (si es España el país que estamos estudiando). Las unidades en las que el tipo de cambio real RER_t está expresado es:

$$\text{unidades} = \frac{\text{pesetas}}{\text{moneda}_{row}} \times \frac{\text{moneda}_{row}/\text{cesta}_{row}}{\text{pesetas}/\text{cesta}_{española}} = \frac{\text{cesta}_{española}}{\text{cesta}_{row}}$$

Una disminución de RER_t se corresponde con una apreciación real de la peseta frente al resto del mundo, ya que son menos cestas españolas las necesarias para comprar un cesta del resto del mundo. Supongamos que la

ley de un precio se sostiene para los bienes comercializados, de modo que

$$p_{T,esp} = NER \times p_{T,row}$$

Sustituyendo esta expresión en la definición de tipo de cambio real obtenemos

$$\widehat{REER}_t = \frac{p_{T,esp}}{p_{T,row}} \times \frac{p_{row}}{p_{esp}} = \frac{(p_{row}/p_{T,row})}{(p_{esp}/p_{T,esp})}$$

Esta expresión nos indica que las fluctuaciones en el tipo de cambio real pueden ser explicadas a través de las fluctuaciones en los precios de los bienes que no son comercializados.

5.2 El modelo con dos bienes

La novedad incorporada al modelo tendrá un reflejo en la función de utilidad, que ahora dependerá de los dos bienes de la economía. Habrá dos tecnologías con las que se producirán cada uno de los bienes, y tendremos una condición de factibilidad sobre la cantidad de capital disponible. Vamos a considerar al bien comercializable como el numerario de la economía, y tanto el capital físico k_t y la inversión i_t como la posición neta de activos frente al extranjero b_t estarán expresadas en términos del numerario. El bien no comercializado tendrá, sin embargo, su propio precio. Introduciendo estas modificaciones sobre nuestro modelo de la lección cuarta tenemos un nuevo problema para el planificador central:

$$\begin{aligned} & \max_{\{c_{Tt}, c_{Nt}, b_t, k_{Tt}, k_{Nt}\}} \sum_{t=0}^{\infty} \beta^t u(c_{Tt}, c_{Nt}) \\ \text{s.a. } & c_{Tt} + i_t + b_{t+1} \leq y_{Tt} + (1+r)b_t \\ & c_{Nt} \leq y_{Nt} \\ & y_{Tt} \leq F_T(k_{Tt}) \\ & y_{Nt} \leq F_N(k_{Nt}) \\ & k_{t+1} \leq i_t + (1-\delta)k_t \\ & k_{Tt} + k_{Nt} \leq k_t \\ & k_0, b_0 \text{ dados} \end{aligned}$$

Nótese que una particularidad de este modelo es que para construir nuevo capital basta con ahorrar parte del bien comercializado, o dicho de otra manera, para tener inversión que se transforme en nuevo capital ésta puede ser

obtenida del extranjero con la sola condición de que se corra el correspondiente déficit por cuenta corriente. No es necesario, por tanto, ahorrar nada del bien no comercializado para incorporar este ahorro al stock de capital.

Para resolver este problema primero especializamos el modelo introduciendo unas formas funcionales específicas para la función de utilidad y las dos funciones de producción. Después sustituimos estas formas funcionales de las restricciones dos y tres en la primera restricción para obtener finalmente:

$$\begin{aligned} & \max_{\{c_{Tt}, c_{Nt}, b_t, k_{Tt}, k_{Nt}\}} \sum_{t=0}^{\infty} \beta^t \frac{1}{\rho} (\varepsilon c_{Tt}^\rho + (1 - \varepsilon) c_{Nt}^\rho)^{\frac{\phi}{\rho}} \\ \text{s.a. } & c_{Tt} + k_{t+1} - (1 - \delta)k_t + b_{t+1} \leq A_T k_{Tt}^{\alpha_T} + (1 + r)b_t \\ & c_{Nt} \leq A_N k_{Nt}^{\alpha_N} \\ & k_{Tt} + k_{Nt} \leq k_t \\ & k_0, b_0 \text{ dados} \end{aligned}$$

Podemos simplificar más aún el problema y sustituir el valor de k_t en la primera restricción por su descomposición en $k_{Tt} + k_{Nt}$. Para resolver este problema construimos la función auxiliar de Lagrange como ya hemos hecho en el capítulo anterior para obtener

$$\begin{aligned} L = & \sum_{t=0}^{\infty} \beta^t \left[\frac{1}{\rho} (\varepsilon c_{Tt}^\rho + (1 - \varepsilon) c_{Nt}^\rho)^{\frac{\phi}{\rho}} + \dots \right. \\ & \dots + p_{Tt} (A_T k_{Tt}^{\alpha_T} + (1 + r)b_t - c_{Tt} - k_{Tt+1} - k_{Nt+1} + (1 - \delta)(k_{Tt} + k_{Nt}) - b_{t+1}) + \dots \\ & \left. \dots + p_{Nt} (A_N k_{Nt}^{\alpha_N} - c_{Nt}) \right] \end{aligned}$$

Como vemos, a la primera restricción le hemos asociado un multiplicador de Lagrange p_{Tt} que es el precio de los bienes transables, y a la segunda restricción le hemos asociado el multiplicador de Lagrange p_{Nt} que es el precio de los bienes no transables. Si realizamos las correspondientes derivadas parciales para obtener las condiciones de primer orden nos encontramos con el siguiente conjunto de ecuaciones:

$$\begin{aligned} \frac{\partial L}{\partial c_{Tt}} &= \phi (\varepsilon c_{Tt}^\rho + (1 - \varepsilon) c_{Nt}^\rho)^{\frac{\phi}{\rho} - 1} \rho \varepsilon c_{Tt}^{\rho - 1} - p_{Tt} = 0 \\ \frac{\partial L}{\partial c_{Nt}} &= \phi (\varepsilon c_{Tt}^\rho + (1 - \varepsilon) c_{Nt}^\rho)^{\frac{\phi}{\rho} - 1} \rho (1 - \varepsilon) c_{Nt}^{\rho - 1} - p_{Nt} = 0 \\ \frac{\partial L}{\partial k_{Tt}} &= p_{Tt} A_T \alpha_T k_{Tt}^{\alpha_T - 1} + p_{Tt} (1 - \delta) k_{Tt} - \beta^{-1} p_{Tt-1} = 0 \end{aligned}$$

$$\frac{\partial L}{\partial k_{Nt}} = p_{Nt}A_N\alpha_N k_{Nt}^{\alpha_N-1} + p_{Tt}(1-\delta)k_{Nt} - \beta^{-1}p_{Tt-1} = 0$$

$$\frac{\partial L}{\partial k_{Nt}} = \beta(1+r)p_{Tt} - p_{Tt-1} = 0$$

y dos condiciones de transversalidad. Tras inspeccionar las condiciones de primer orden observamos que en un estado estacionario el número de variables que tenemos que determinar es $(c_T, c_N, k_T, k_N, b, p_T, p_N, r)$ que hacen un total de 8 incógnitas. Al contar el número de ecuaciones vemos que disponemos de 4 condiciones de primer orden y 2 restricciones de factibilidad, es decir, de 6 ecuaciones. Esto no debe resultar sorprendente puesto que hay dos precios (r, p_T) que quedan determinados fuera de nuestro modelo. El tipo de interés internacional r está dado por la cantidad de capital existente en el resto del mundo, así como el precio del bien comercializado p_T , que viene dado por el resto del mundo y la operatividad de la ley de un precio. Por tanto el tipo de interés internacional será función de la cantidad medida de capital en nuestra definición de resto del mundo y es por tanto una variable que podemos observar. El precio del bien comercializado podemos suponerlo arbitrariamente igual a 1. Puestas las cosas de esta manera ya estamos en condiciones de realizar un nuevo programa que nos permita resolver este modelo ampliado. Con él seremos capaces de generar secuencias artificiales para la cuenta corriente y para el tipo de cambio real, compararlas con las secuencias que fueron observadas en el periodo de tiempo de nuestro interés y estudiar las posibles paradojas que surjan de esta comparación.

Lo primero que tenemos que observar antes de plantearnos hacer un programa para este modelo con dos bienes es que difiere demasiado poco del modelo estudiado en el capítulo anterior. El déficit por cuenta corriente nos permite introducir tanto capital como deseamos en esta economía. Por tanto el argumento ya utilizado que nos sugiere que el déficit por cuenta corriente será tan alto como fuera necesario para introducir la cantidad de capital de estado estacionario sigue siendo válida. Para ver que el argumento funciona del mismo modo vamos a escribir las ecuaciones de factibilidad y las condiciones de primer orden en el estado estacionario (denotando las variables en estado estacionario por v_{ss}), así como las condiciones de factibilidad en el primer periodo como ya lo hiciéramos en la lección anterior:

$$c_{Tss} + k_{ss} - (1-\delta)k_0 + b_{ss} = A_T k_{T0}^{\alpha_T} \quad (37)$$

$$c_{N0} = A_N k_{N0}^{\alpha_N} \quad (38)$$

$$\frac{\varepsilon}{1 - \varepsilon} \left(\frac{c_{Tss}}{c_{N0}} \right)^{\rho-1} = \frac{1}{p_{N0}} \quad (39)$$

$$A_T \alpha_T k_{T0}^{\alpha_T-1} = p_{N0} A_N \alpha_N k_{N0}^{\alpha_N-1} \quad (40)$$

$$k_{T0} + k_{N0} = k_0 \quad (41)$$

sería la primera condición de factibilidad en el momento $t = 0$. Las condiciones de factibilidad y las condiciones de primer orden en el estado estacionario serían:

$$c_{Tss} + \delta k_{ss} = A_T k_{Tss}^{\alpha_T} + r b_{ss} \quad (42)$$

$$c_{Nss} = A_N k_{Nss}^{\alpha_N} \quad (43)$$

$$k_{Tss} + k_{Nss} = k_{ss} \quad (44)$$

$$\frac{\varepsilon}{1 - \varepsilon} \left(\frac{c_{Tss}}{c_{Nss}} \right)^{\rho-1} = \frac{1}{p_{Nss}} \quad (45)$$

$$A_T \alpha_T k_{Tss}^{\alpha_T-1} + (1 - \delta) = (1 + r) \quad (46)$$

$$p_{Nss} A_N \alpha_N k_{Nss}^{\alpha_N-1} + (1 - \delta) = (1 + r) \quad (47)$$

Los valores de k_0 , k_{T0} y b_0 son conocidos. Imaginemos momentáneamente que el valor del stock de estado estacionario también es conocido, entonces podríamos realizar la siguiente operación: de la ecuación 46 podemos obtener el valor de estado estacionario para k_T , con él y dado que estamos suponiendo k conocido podríamos obtener k_N de la ecuación 44. Con este valor podríamos acudir a la ecuación 47 y a la ecuación 43 para obtener los valores de p_N y c_N respectivamente. Con estos valores calculados podríamos acudir a la ecuación 45 para hallar el valor de c_T y finalmente, acudiendo a la ecuación 42 obtener el valor de b . Para verificar si nuestro valor de k que se asumió conocido era correcto vamos a la ecuación 37 y comprobamos si se produce la igualdad. En caso de no ser iguales acudimos al método de Newton-Raphson y obtenemos un nuevo valor para k .

Como se puede ver, la trayectoria de equilibrio de esta economía artificial no va a ser muy distinta de la ya vista en la anterior lección. En un sólo periodo podemos construir todo el stock de capital corriendo un déficit por cuenta corriente adecuado en el primer periodo y cancelándolo en el siguiente con un superávit comercial. En relación al tipo de cambio real las cosas no van a ser muy distintas. Para verlo podemos combinar las ecuaciones 46, 47

y 40 para obtener:

$$\frac{p_{Nss}}{p_{N0}} = \frac{\left(\frac{c_{Tss}}{c_{N0}}\right)^{\rho-1}}{\left(\frac{c_{Tss}}{c_{Nss}}\right)^{\rho-1}} = \left(\frac{c_{Nss}}{c_{N0}}\right)^{\rho-1} = \left(\frac{k_{Nss}}{k_{N0}}\right)^{\alpha_N(\rho-1)}$$

Es decir, la relación de precios entre bienes transables y no transables cambiará en la medida en la que el stock de capital de bienes no comercializados en el periodo de la apertura difiera del stock de capital de estado estacionario de los bienes no comercializados. Lo importante de esta expresión es que la dinámica del tipo de cambio real no será muy distinta de la dinámica de la cuenta corriente en el sentido de que en un sólo periodo, después de un impacto inicial se llega al tipo de cambio real de estado estacionario.

5.2.1 La necesidad de introducir la inversión de un modo distinto

Como hemos visto nuestro intento de obtener series artificiales para nuestras variables con la única introducción de bienes comercializados y no comercializados ha sido un fracaso. No obstante hemos aprendido algo y es que en la medida en la que el stock de capital pueda ser construido en tan sólo un periodo nos encontraremos en dificultades para explicar de una manera satisfactoria la variación en la cuenta corriente y en el tipo de cambio real de una economía pequeña que súbitamente se abre al mercado de capitales. Para remediar esta dificultad podríamos introducir una función de inversión en la que para construir nuevo capital sea necesaria la concurrencia de ambos bienes, el comercializable y el no comercializable. Pensemos en esta posibilidad un momento. Si la incorporación de la inversión al stock de capital requiere la participación de un bien no comercializable entonces es seguro que el proceso de acumulación debe frenarse. La razón es simple: el país sólo puede importar bienes comercializados, pero no puede hacerlo con los bienes no comercializados, entonces si la incorporación de una unidad de capital al proceso productivo requiere que se transformen una unidad de bien comercializado y una unidad de bien no comercializado para obtener esa unidad de capital incorporada, entonces no será posible construir el stock de capital en un sólo periodo puesto que las cantidades de bien no comercializado existen en cantidades limitadas y el defecto no puede ser importado sin límites. En la siguiente lección exploramos esta nueva posibilidad y veremos que podemos aprender.

6 Lección sexta

En esta lección vamos a introducir una función de inversión por primera vez. Hasta este momento siempre hemos considerado que $k_{t+1} \leq i_t + (1 - \delta)k_t$ y que la inversión era simplemente la parte no consumida del producto. Imaginemos ahora que las cosas no son tan sencillas y que es necesaria una ulterior manipulación de la inversión para que ésta pueda pasar a formar parte del stock de capital. Vamos a imaginar además que en esa transformación intervienen tanto los bienes transables como los no transables del capítulo anterior. Si denotamos por x_{Tt} y x_{Nt} a la inversión realizada en ambos bienes entonces podríamos escribir $k_{t+1} \leq I(x_{Tt}, x_{Nt}) + (1 - \delta)k_t$. Si asumimos que la tecnología que transforma la inversión en nuevo capital es del tipo Coob-Douglas, nuestra nueva ecuación tendría el siguiente aspecto: $k_{t+1} \leq Gx_{Tt}^\gamma x_{Nt}^{1-\gamma} + (1 - \delta)k_t$.

Con esta especificación podemos estar seguros de que el proceso debe frenarse, y esto es así porque es necesario que concurra la participación de bienes no transables en la formación de nuevo capital, pero como sólo los bienes transables pueden ser importados sin límite, habrá que esperar a que la industria de construcción y servicios (no transables) tenga esas cantidades disponibles para invertir.

6.1 Las ecuaciones

El modelo que estamos planteando ahora sería:

$$\begin{aligned} \max \sum_{t=0}^{\infty} \beta^t \frac{1}{\rho} (\varepsilon c_{Tt}^\rho + (1 - \varepsilon) c_{Nt}^\rho)^{\frac{\rho}{\rho-1}} \\ \text{s.a. } c_{Tt} + x_{Tt} + b_{t+1} &\leq A_T k_{Tt}^{\alpha_T} + (1 + r)b_t \\ c_{Nt} + x_{Nt} &\leq A_N k_{Nt}^{\alpha_N} \\ k_{Tt+1} + k_{Nt+1} - (1 - \delta)(k_{Tt} + k_{Nt}) &\leq Gx_{Tt}^\gamma x_{Nt}^{1-\gamma} \\ &k_0, b_0 \text{ dados} \end{aligned}$$

A la primera restricción le vamos a asociar el multiplicador de Lagrange P_{Tt} que sabemos que será igual a 1. A la segunda restricción le asociaremos p_{Nt} y a la tercera le asociaremos el multiplicador q_{t+1} . Nótese que ahora el capital va a tener su propio precio, ahora va a ser un bien producido en la economía que será distinto de los otros bienes y que por tanto tendrá su propio precio.

Planteamos la ecuación auxiliar de Lagrange como:

$$L = \sum_{t=0}^{\infty} \beta^t \left[\frac{1}{\rho} (\varepsilon c_{Tt}^{\rho} + (1 - \varepsilon) c_{Nt}^{\rho})^{\frac{\rho}{\rho-1}} + \dots \right. \\ \left. \dots + p_{Tt} (A_T k_{Tt}^{\alpha_T} + (1 + r) b_t - c_{Tt} - x_{Tt} - b_{t+1}) + \dots \right. \\ \left. \dots + p_{Nt} (A_N k_{Nt}^{\alpha_N} - c_{Nt} - x_{Nt}) + q_{t+1} (G x_{Tt}^{\gamma} x_{Nt}^{1-\gamma} - k_{Tt+1} - k_{Nt+1} + (1 - \delta)(k_{Tt} + k_{Nt})) \right]$$

De donde extraemos las condiciones de primer orden y que formaran parte, junto con las restricciones de factibilidad, del sistema de ecuaciones que tenemos que resolver. Estas condiciones de primer orden son:

$$\begin{aligned} u_{Tt} \varepsilon \rho c_{Tt}^{\rho-1} - p_{Tt} &= 0 \\ u_{Nt} (1 - \varepsilon) \rho c_{Nt}^{\rho-1} - p_{Nt} &= 0 \\ -p_{Tt-1} + \beta(1 + r)p_{Tt} &= 0 \\ -p_{Tt} - q_{t+1} G \gamma x_{Tt}^{\gamma-1} x_{Nt}^{1-\gamma} &= 0 \\ -p_{Nt} - q_{t+1} G (1 - \gamma) x_{Tt}^{\gamma} x_{Nt}^{-\gamma} &= 0 \\ p_{Tt} A_T \alpha_T k_{Tt}^{\alpha_T-1} + q_{t+1} (1 - \delta) - q_t (1 + r) &= 0 \\ p_{Nt} A_N \alpha_N k_{Nt}^{\alpha_N-1} + q_{t+1} (1 - \delta) - q_t (1 + r) &= 0 \end{aligned}$$

y dos condiciones de transversalidad. Las ecuaciones de vaciado y de factibilidad que completan el sistema son:

$$\begin{aligned} c_{Tt} + x_{Tt} + b_{t+1} &\leq A_T k_{Tt}^{\alpha_T} + (1 + r) b_t \\ c_{Nt} + x_{Nt} &\leq A_N k_{Nt}^{\alpha_N} \\ k_{Tt+1} + k_{Nt+1} - (1 - \delta)(k_{Tt} + k_{Nt}) &\leq G x_{Tt}^{\gamma} x_{Nt}^{1-\gamma} \\ &k_0, b_0 \text{ dados} \end{aligned}$$

6.2 El programa

El programa tendrá tres partes La primera parte contendrá la definición de parámetros del modelo, los parámetros del programa y la semilla inicial que consistirá en toda una trayectoria para cada variable. La semilla inicial será el estado estacionario, para lo cual necesitaremos un pequeño programa que nos compute el estado estacionario y que sirve de introducción al tercer programa en el que estarán todas las condiciones de primer orden y las

condiciones de factibilidad del problema. El primer programa efectuará una llamada a nuestro programa estándar **secant.m** que se encargará de iterar sobre nuestro segundo programa. Una vez que el estado estacionario esté computado volveremos a llamar a **secant.m** para que itere sobre el programa que contiene la dinámica. La lógica es por tanto la misma de siempre. Dado que este es el primer programa con cierta complejidad que vamos a realizar comentaré detalladamente qué hace cada línea de comando que tengamos en uno y otro programa. Veamos el programa principal **inver.m**

```

%programa inver.m
%Este programa resuelve el modelo con dos bienes e inversión que
%requiere la participación de los bienes transables y no transables
% inf
%max sum beta^t (c(t)^(1-eta)-1)/(1-eta)
% t=0
% s.a. b(t+1)+cT(t)+xT(t)<=aT*kT^alphaT+(1+r)*b(t)
% cN(t)+xN(t)<=aN*kN(t)^alphaN
% iT(t)+iN(t)<=GxT(t)^gamma*xN(t)^(1-gamma)
% kT(t+1)<=iT(t)+(1-delta)*kT(t)
% kN(t+1)<=iN(t)+(1-delta)*kN(t)
% k(0) dado, b(0) dado
clear
%Definición de los parámetros del modelo
aT = 0.96612775*34.3541;
aN = 0.90461754*35.4459;
alphaT = 0.3071665;
alphaN = 0.35370848;
gamma = 0.39681632;
G = 1.95756169;
rho = -1;
theta = -1;
epsilon = (0.852600116^(1/(1-rho)))/(1+0.852600116^(1/(1-rho)));
delta = 0.0576;
beta = 0.9463;
r = 1/beta-1;
bss = 0;
%Definición de parámetros del programa
maxit = 1000;

```

```

crit = 1e-8;
T = 40;
%Llamada a secant.m para computar el estado estacionario
x0 = [896; 981];
param = [aT aN alphaT alphaN gamma G rho theta epsilon delta
beta bss];
sol = secant('sscpo', x0, param, crit, maxit);
%Resultados del estado estacionario
kTss = sol(1);
kNss = sol(2);
rss = 1/beta-1;
kss = kTss+kNss;
mpkTss = aT*alphaT*kTss^(alphaT-1);
mpkNss = aN*alphaN*kNss^(alphaN-1);
pNss = mpkTss/mpkNss;
ratxss = gamma*pNss/(1-gamma);
xNss = delta*kss/(G*ratxss^gamma);
xTss = ratxss*xNss;
qss = ratxss^(1-gamma)/(G*gamma);
yTss = aT*kTss^alphaT;
yNss = aN*kNss^alphaN;
cNss = yNss-xNss;
cTss = ((1-epsilon)/(epsilon*pNss))^(1/(rho-1))*cNss;
%Estado estacionario computado
%Construcción de la semilla
kTguess = kTss*ones(size(1:T-1));
kNguess = kNss*ones(size(1:T-1));
bguess = bss*ones(size(1:T-1));
Mguess = [kTguess kNguess bguess];
guess = [kTss kNss bss];
for t=2:T-1
c = Mguess(t,:);
guess = [guess c];
end
guess(3*t+1) = kTss;
guess(3*t+2) = kNss;
%LLamada a secant.m para computar la dinámica
kT0 = kTss*0.8;

```

```

kN0 = kNss*0.8;
param = [aT aN alphaT alphaN gamma G rho theta epsilon delta
beta bss T kT0 kN0];
sol = secant('dincpo', guess', param, crit, maxit);
    for t=1:T
        kT(t) = sol(3*t-2);
        kN(t) = sol(3*t-1);
    end
    for t=1:T-1
        b(t) = sol(3*t)
    end
%Reconstrucción de las trayectorias de equilibrio
mpkT(1) = aT*alphaT*kT(1)^(alphaT-1);
mpkN(1) = aN*alphaN*kN(1)^(alphaN-1);
pN(1) = mpkT(1)/mpkN(1);
ratx(1) = gamma*pN(1)/(1-gamma);
xN(1) = (kT(1)+kN(1)-(1-delta)*(kT0+kN0))/(G*ratx(1)^gamma);
xT(1) = ratx(1)*xN(1);
q(1) = ratx(1)^(1-gamma)/(G*gamma);
yT(1) = aT*kT(1)^alphaT;
yN(1) = aN*kN(1)^alphaN;
cN(1) = yN(1)-xN(1);
cT(1) = ((1-epsilon)/(epsilon*pN(1)))^(1/(rho-1))*cN(1);
u(1) = theta*(epsilon*cT(1)^rho+(1-epsilon)*cN(1)^rho)^(theta/rho-1);
for t=2:T
    mpkT(t) = aT*alphaT*kT(t)^(alphaT-1);
    mpkN(t) = aN*alphaN*kN(t)^(alphaN-1);
    pN(t) = mpkT(t)/mpkN(t);
    ratx(t) = gamma*pN(t)/(1-gamma);
    xN(t) = (kT(t)+kN(t)-(1-delta)*(kT(t-1)+kN(t-1)))/(G*ratx(t)^gamma);
    xT(t) = ratx(t)*xN(t);
    q(t) = ratx(t)^(1-gamma)/(G*gamma);
    yT(t) = aT*kT(t)^alphaT;
    yN(t) = aN*kN(t)^alphaN;
    cN(t) = yN(t)-xN(t);
    cT(t) = ((1-epsilon)/(epsilon*pN(t)))^(1/(rho-1))*cN(t);

```

```

u(t) = theta*(epsilon*cT(t)^rho+(1-epsilon)*cN(t)^rho)^(theta/rho-
1);
end
yT0 = aT*kT0^alphaT;
yT = [yT0 aT*kT(1:T).^alphaT];
yN0 = aN*kN0^alphaN;
yN = [yN0 aN*kN(1:T).^alphaN];
b0 = 0;
bT = [b0 b];
ca = [0 bT(2:T)-bT(1:T-1)];
td0 = 0;
td1 = aT*kT(1:T-1).^alphaT-cT(1:T-1)-xT(1:T-1);
td = [td0 td1];
figure(1)
plot(yT)
title('Producto')
figure(2)
plot(kT)
title('Capital')
figure(3)
plot(ca)
title('Cuenta corriente')
figure(4)
plot(td)
title('Déficit comercial')

```

El segundo programa llamado **sscpo.m** computa el estado estacionario del sistema.

```

function f=sscpo(z, p)
%sscpo.m Función que contiene las condiciones de primer orden
del estado estacionario
%Asignación de variables
kTss = z(1);
kNss = z(2);
%Asignación de parámetros
aT = p(1);
aN = p(2);

```

```

alphaT = p(3);
alphaN = p(4);
gamma = p(5);
G = p(6);
rho = p(7);
theta = p(8);
epsilon = p(9);
delta = p(10);
beta = p(11);
bss = p(12);
rss = 1/beta-1;
kss = kTss+kNss;
mpkTss = aT*alphaT*kTss^(alphaT-1);
mpkNss = aN*alphaN*kNss^(alphaN-1);
pNss = mpkTss/mpkNss;
ratxss = gamma*pNss/(1-gamma);
xNss = delta*kss/(G*ratxss^gamma);
xTss = ratxss*xNss;
qss = ratxss^(1-gamma)/(G*gamma);
yTss = aT*kTss^alphaT;
yNss = aN*kNss^alphaN;
cNss = yNss-xNss;
cTss = ((1-epsilon)/(epsilon*pNss))^(1/(rho-1))*cNss;
f(1) = aT*alphaT*kTss^(alphaT-1)-qss*(rss+delta);
f(2) = yTss-cTss-xTss+rss*bss;
f=f';

```

Y finalmente el programa **dincpo.m** computa la dinámica.

```

function f=dincpo(z, p)
%Asignación de parámetros
aT = p(1);
aN = p(2);
alphaT = p(3);
alphaN = p(4);
gamma = p(5);
G = p(6);
rho = p(7);

```

```

theta = p(8);
epsilon = p(9);
delta = p(10);
beta = p(11);
bss = p(12);
T = p(13);
kT0 = p(14);
kN0 = p(15);
%Asignación de variables
for t=1:T-1
kT(t) = z(3*t-2);
kN(t) = z(3*t-1);
b(t+1) = z(3*t);
end
kT(T) = z(3*t+1);
kN(T) = z(3*t+2);
r = 1/beta-1;
mpkT(1) = aT*alphaT*kT(1)^(alphaT-1);
mpkN(1) = aN*alphaN*kN(1)^(alphaN-1);
pN(1) = mpkT(1)/mpkN(1);
ratx(1) = gamma*pN(1)/(1-gamma);
xN(1) = (kT(1)+kN(1)-(1-delta)*(kT0+kN0))/(G*ratx(1)^gamma);
xT(1) = ratx(1)*xN(1);
q(1) = ratx(1)^(1-gamma)/(G*gamma);
yT(1) = aT*kT(1)^alphaT;
yN(1) = aN*kN(1)^alphaN;
cN(1) = yN(1)-xN(1);
cT(1) = ((1-epsilon)/(epsilon*pN(1)))^(1/(rho-1))*cN(1);
u(1) = theta*(epsilon*cT(1)^rho+(1-epsilon)*cN(1)^rho)^(theta/rho-1);
for t=2:T-1
mpkT(t) = aT*alphaT*kT(t)^(alphaT-1);
mpkN(t) = aN*alphaN*kN(t)^(alphaN-1);
pN(t) = mpkT(t)/mpkN(t);
ratx(t) = gamma*pN(t)/(1-gamma);
xN(t) = (kT(t)+kN(t)-(1-delta)*(kT(t-1)+kN(t-1)))/(G*ratx(t)^gamma);
xT(t) = ratx(t)*xN(t);
q(t) = ratx(t)^(1-gamma)/(G*gamma);

```

```

yT(t) = aT*kT(t)^alphaT;
yN(t) = aN*kN(t)^alphaN;
cN(t) = yN(t)-xN(t);
cT(t) = ((1-epsilon)/(epsilon*pN(t)))^(1/(rho-1))*cN(t);
u(t) = theta*(epsilon*cT(t)^rho+(1-epsilon)*cN(t)^rho)^(theta/rho-
1);
end
kss = kT(T)+kN(T);
mpkTss = aT*alphaT*kT(T)^(alphaT-1);
mpkNss = aN*alphaN*kN(T)^(alphaN-1);
pNss = mpkTss/mpkNss;
ratxss = gamma*pNss/(1-gamma);
xNss = delta*kss/(G*ratxss^gamma);
xTss = ratxss*xNss;
yTss = aT*kT(T)^alphaT;
yNss = aN*kN(T)^alphaN;
cNss = yNss-xNss;
cTss = ((1-epsilon)/(epsilon*pNss))^(1/(rho-1))*cNss;
cT(T) = cTss;
cN(T) = cNss;
u(T) = theta*(epsilon*cT(T)^rho+(1-epsilon)*cN(T)^rho)^(theta/rho-
1);
q(T) = qss;
f(1) = aT*alphaT*kT(1)^(alphaT-1)+q(2)*(1-delta)-q(1)*(1+r);
f(2) = cT(1)+xT(1)+b(2)-aT*kT(1)^alphaT;
f(3) = u(1)*epsilon*rho*cT(1)^(rho-1)-u(2)*epsilon*rho*cT(2)^(rho-
1);
for t=2:T-1
f(3*t-2) = aT*alphaT*kT(t)^(alphaT-1)+q(t+1)*(1-delta)-q(t)*(1+r);
f(3*t-1) = cT(t)+xT(t)+b(t+1)-aT*kT(t)^alphaT-(1+r)*b(t);
f(3*t) = u(t)*epsilon*rho*cT(t)^(rho-1)-u(t+1)*epsilon*rho*cT(t+1)^(rho-
1);
end
f(3*t+1) = mpkTss-qss*(r+delta);
f(3*t+2) = yTss+r*b(T)-cT(T)-xTss;
f=f';

```


En el programa **inver.m** nos encontramos, después de haber definido los parámetros del modelo y del programa, el conocido comando **sol = secant('sscpo', x0, param, crit, maxit);**. Este comando dispara a **secant.m** para operar sobre el sistema de ecuaciones introducido en **sscpo.m**. En este programa la primera línea **function f=sscpo(z, p)** dice que el programa es una función que depende de la variable **z** y del parámetro **p**. La variable **z** tiene dos componentes que son la semilla contenida en **x0 = [896; 981];** de modo que las líneas de comando **kTss = z(1);** y **kNss = z(2);** asignan como primer valor **kTss = 896** y **kNss = 981**. El listado que sigue a **%Asignación de parámetros** es la asignación apropiada a cada parámetro del valor que fue definido en **inver.m** en la línea de comando **param = [aT aN alphaT alphaN gamma G rho theta epsilon delta beta bss];** y cada uno de estos parámetros fue definido en la sección de definición de parámetros de modelo. Respetar el orden es absolutamente esencial. Una vez que el proceso haya terminado el resultado será guardado en la variable **sol** definida más arriba. Una vez que tenemos la **solución** al estado estacionario reconstruimos el valor de las variables. Esto tenemos que hacerlo en el programa **inver.m** porque las variables computadas dentro de **sscpo.m** no son variables globales, es decir, no son reconocidas en ningún lugar más que dentro de la función que las generó. Una vez que hemos computado el estado estacionario llegamos finalmente a la línea **%Estado estacionario computado**. Si detenemos el programa escribiendo un **pause** justo después de **sol = secant('sscpo', x0, param, crit, maxit)** y escribimos en la pantalla de comandos de MatLab el nombre de alguna variable que estuviera dentro de **sscpo.m** la respuesta será que la variable es desconocida. La razón es que las variables dentro de las funciones sólo son recordadas mientras son usadas, por eso tenemos que repetir en **inver.m** prácticamente todo el código que hay dentro de **sscpo.m**, que son básicamente las condiciones de primer orden y las condiciones de factibilidad en el estado estacionario.

6.2.1 Estado estacionario

El sistema de ecuaciones en el estado estacionario viene dado por:

$$\begin{aligned}
 c_{Tss} + x_{Tss} - A_T k_{Tss}^{\alpha_T} - r b_{ss} &= 0 \quad \mathbf{I} \\
 c_{Nss} + x_{Nss} - A_N k_{Nss}^{\alpha_N} &= 0 \quad \mathbf{II}
 \end{aligned}$$

$$\begin{aligned}
\delta(k_{Tss} + k_{Nss}) - Gx_{Tss}^\gamma x_{Nss}^{1-\gamma} &= 0 \quad \text{III} \\
u_{Tss}\varepsilon\rho c_{Tss}^{\rho-1} - 1 &= 0 \quad \text{IV} \\
u_{Nss}(1 - \varepsilon)\rho c_{Nss}^{\rho-1} - p_{Nss} &= 0 \quad \text{V} \\
-1 - q_{ss}G\gamma x_{Tss}^{\gamma-1} x_{Nss}^{1-\gamma} &= 0 \quad \text{VI} \\
-p_{Nss} - q_{ss}G(1 - \gamma)x_{Tss}^\gamma x_{Nss}^{-\gamma} &= 0 \quad \text{VII} \\
A_T\alpha_T k_{Tss}^{\alpha_T-1} - q_{ss}(\delta + r) &= 0 \quad \text{VIII} \\
p_{Nss}A_N\alpha_N k_{Nss}^{\alpha_N-1} - q_t(\delta + r) &= 0 \quad \text{IX}
\end{aligned}$$

El sistema de ecuaciones que determina el estado estacionario tiene 9 ecuaciones y 9 incógnitas, que son $(c_{Tss}, x_{Tss}, k_{Tss}, c_{Nss}, x_{Nss}, k_{Nss}, b_{ss}, p_{Nss}, q_{ss})$. En principio nosotros podríamos introducir estas ecuaciones e incógnitas en un programa de MatLab, llamar a **secant.m** y terminar. No obstante es mejor proceder de otro modo. La dimensión del problema puede ser reducida muy fácilmente de un sistema de 9 ecuaciones en 9 incógnitas a uno de 2 ecuaciones en 2 incógnitas. Trabajar sobre un sistema más pequeño tiene la ventaja de que el algoritmo será más, rápido, más preciso y más estable. Para ver cómo podemos reducir la dimensión del problema empezamos imaginando que conocemos dos variables, como por ejemplo k_{Tss} y k_{Nss} . Si estas dos variables fueran conocidas, de las ecuaciones **VIII** y **IX** podríamos sacar el valor de p_{Nss} . Una vez conocido el valor de p_{Nss} , de las ecuaciones **VI** y **VII** podríamos extraer el valor de x_{Tss}/x_{Nss} . Conocido el valor de este ratio y nuestro supuesto conocimiento de k_{Tss} y k_{Nss} , podríamos extraer el valor de x_{Tss} de la ecuación **III**. Conocidos los valores de x_{Tss} y x_{Nss} , podemos averiguar sin dificultad el valor de q_{ss} de la ecuación **VI**. De la ecuación **II** extraeríamos el valor de c_{Nss} , y una vez conocido éste, del ratio entre las ecuaciones **IV** y **V** obtendríamos el valor de c_{Tss} . Nos quedarían sin comprobar dos ecuaciones que son la **I** y la **VIII**. Como son dos los valores que hemos supuesto, podemos iterar en los valores de k_{Tss} y k_{Nss} , de modo que si las dos ecuaciones finales no se cumplen probamos con otro par de valores de k_{Tss} y k_{Nss} , y así sucesivamente hasta que todas la ecuaciones se cumplen. Como se puede observar esto es exactamente lo que hace el programa **sscpo.m**.

6.2.2 Dinámica

La dinámica esta programada en **dincpo.m** cuya construcción es idéntica a la de **sscpo.m**. Comienza realizando una asignación de parámetros que en

este caso contiene algunos adicionales muy importantes, como son \mathbf{T} , que es la dimensión de la truncación y el par de valores de stock de capital inicial $\mathbf{kT0}$ y $\mathbf{kN0}$. Luego se realiza la asignación de la semilla a cada una de las variables. La semilla fue cuidadosamente construida en el programa **inver.m**. La lógica sobre el funcionamiento de este algoritmo es idéntica al de los ya presentados y por tanto no nos extenderemos más sobre su contenido.

Llegados a este punto ya sólo queda resolver el modelo calibrado y comparar los resultados con los generados por la economía real. En este momento comienzan otros campos de investigación relacionados con el modelo básico. Restricción de crédito, posibilidades de fallar en la devolución de la deuda, estudiar los efectos del ciclo económico, y un largo etcétera es lo que faltaría para que nuestro estudio sobre este tipo de economías fuera completamente exhaustivo. Pero esto es investigación futura en la que vosotros posiblemente queráis participar.

7 Lección séptima

En esta última lección vamos a estudiar de nuevo el modelo simple de crecimiento pero con una pequeña variante. Vamos a introducir perturbaciones estocásticas sobre el parámetro de productividad agregada en la función de producción. La técnica que vamos a emplear para resolver el problema es la de las Expectativas Parametrizadas de Albert Marcet. Esta técnica, a diferencia de otras como la loglinearización en torno al estado estacionario utilizada por King, Plosser y Rebelo o la de programación dinámica permite realizar el cálculo de toda la trayectoria óptima a lo largo del proceso de acumulación de capital. Por esta razón comenzamos con ella.

7.0.3 El entorno

Las preferencias

Vamos a considerar que hay un número muy grande de consumidores y que se dan las propiedades para poder agregarlos en un único agente representativo. Este agente representativo tiene como función de utilidad

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

donde c_t es el consumo realizado en el momento t y $\beta \in (0, 1)$ es el factor de descuento intertemporal de utilidad. Para este ejemplo definiremos

$$u(c_t) = \log(c_t)$$

Las tecnologías

La tecnología agregada del producto interior del país la representaremos como una Coob-Douglas puesto que por un lado así lo recomiendan los datos, y por otro las propiedades de esta tecnología son compatibles con nuestra teoría de la empresa. Así, el producto total se escribe como:

$$y_t = A_t k_t^\alpha l_t^{1-\alpha},$$

donde k_t es el stock de capital en el momento t . El parámetro $\{A_t\}_{t=0}^{\infty}$ sigue un proceso estocástico con la siguiente estructura:

$$A_t = \bar{A} e^{\phi_t} \tag{48}$$

$$\phi_t = \varphi \phi_{t-1} + \varepsilon_t, \quad \varphi < 1, \quad \text{y} \quad \varepsilon_t \sim N(0, \sigma^2). \tag{49}$$

el parámetro \bar{A} es la media del proceso que sigue el parámetro de productividad agregada, en tanto que ϕ_t sigue un proceso autorregresivo de orden 1 con parámetro φ . La perturbación estocástica que sigue el proceso autorregresivo ε_t , es normal con esperanza 0, y varianza σ^2 .

Las dotaciones

En cada periodo la economía posee una unidad de trabajo $l_t = 1$ y al inicio del periodo $t = 0$, hay k_0 unidades de capital.

Información

Hay incertidumbre en el proceso productivo. En el momento de iniciarse el proceso productivo el agente representativo tiene que decidir cuanta inversión debe incorporar al stock de capital para iniciar el proceso productivo, sin embargo, en el momento de tomar la decisión el agente representativo no sabe cuál será el valor de A_{t+1} . Toda la información se conoce con certeza hasta el periodo t .

El problema del agente representativo se formaliza:

$$\underset{\{c_t, k_t\}_{t=0}^{\infty}}{\text{Max}} \quad E_t \left\{ \sum_{t=0}^{\infty} \beta^t \log(c_t) \right\} \quad (50)$$

sujeto a las siguientes restricciones:

La primera es la ley de acumulación de capital

$$k_{t+1} = I_t + (1 - \delta)k_t \quad (51)$$

La segunda es la condición de factibilidad, según la cual lo consumido y lo invertido en un periodo no puede exceder a la capacidad productiva en ese periodo.

$$c_t + I_t = A_t k_t^\alpha l_t^{1-\alpha}$$

La tercera es el proceso estocástico que sigue el parámetro de productividad agregada.

$$A_t = \bar{A} e^{\phi_t} \quad (52)$$

$$\phi_t = \varphi \phi_{t-1} + \varepsilon_t, \quad \varphi < 1, \quad \text{and} \quad \varepsilon_t \sim N(0, \sigma^2). \quad (53)$$

La cuarta nos da las dotaciones iniciales.

$$k_0 \text{ dado y } l_t = 1, \forall t$$

Haciendo algunas sustituciones obvias para simplificar el problema obtenemos:

$$\underset{\{c_t, k_t\}_{t=0}^{\infty}}{Max} \quad E_t \left\{ \sum_{t=0}^{\infty} \beta^t \log(c_t) \right\}$$

$$s.a. \quad c_t + k_{t+1} - (1 - \delta)k_t = A_t k_t^{\alpha}$$

$$A_t = \bar{A} e^{\phi_t}$$

$$\varphi \phi_{t-1} + \varepsilon_t = \phi_t$$

$$k_0 \text{ dado y } l_t = 1, \forall t$$

Despejando c_t de la primera restricción y sustituyéndolo en la función objetivo encontramos un problema de maximización sin restricciones, del cual es muy fácil obtener la ecuación de Euler como:

$$1/c_t = \beta E_t \left\{ 1/c_{t+1} \left[\alpha A_{t+1} k_{t+1}^{\alpha-1} + (1 - \delta) \right] \right\} \quad (54)$$

El operador de esperanzas opera desde el momento $t + 1$ en adelante, ya que toda la información hasta el momento t es conocida. El problema que tenemos con la ecuación de Euler es que no sabemos cómo se distribuye el término sobre el que opera el operador de esperanzas, y por tanto no podemos calcularlo. La idea de la parametrización de las expectativas consiste en aproximar el lado derecho de la ecuación de Euler a un polinomio que sólo dependa del valor de variables de estado contemporáneas. La forma funcional del polinomio debería asemejarse a la forma funcional que queremos aproximar, por tanto un posible candidato podría ser:

$$E \left\{ 1/c_{t+1} \left[\alpha A_{t+1} k_{t+1}^{\alpha-1} + (1 - \delta) \right] \right\} \sim e^{\alpha_0} A_t^{\alpha_1} k_t^{\alpha_2}. \quad (55)$$

Una vez que tenemos un candidato deberemos proceder de acuerdo con el siguiente programa:

1. Generamos una secuencia completa hasta T , $\{A_t\}_{t=0}^T$.
2. Conjeturamos un vector de parámetros $(\alpha_0, \alpha_1, \alpha_2)$.
3. Con la aproximación

$$1/c_t = \beta e^{\alpha_0} A_t^{\alpha_1} k_t^{\alpha_2}. \quad (56)$$

y

$$k_{t+1} - (1 - \delta)k_t + c_t = A_t k_t^\alpha, \quad (57)$$

y el valor conocido de k_0 podemos generar las series $\{c_t\}_{t=0}^T$, y $\{k_t\}_{t=0}^T$. El procedimiento es simple: con k_0 y A_0 , podemos obtener c_0 de la ecuación 56, y después obtener el valor de k_1 de la ecuación 57. Ahora, con los valores de A_1 y k_1 podemos computar c_1 , y luego k_2 , y así sucesivamente.

4. Calculamos $\{y_t\}_{t=0}^T$ como

$$y_t = 1/c_{t+1} \left[\alpha A_{t+1} k_{t+1}^{\alpha-1} + (1 - \delta) \right]$$

y regresamos

$$\log y_t = \alpha_0 + \alpha_1 \log A_t + \alpha_2 \log k_t$$

para encontrar un nuevo vector $(\alpha_0, \alpha_1, \alpha_2)$.

5. Si el nuevo vector de parámetros MCO fuera sustancialmente diferente del vector propuesto como semilla entonces iteramos volviendo al paso 2, en caso contrario detenemos el proceso de iteración.

Un programa de MatLab que hace este proceso es el siguiente:

```
clear
%Valor de los parámetros
delta=0.1;
var=0.1;
prodconst=20;
betar=0.901;
alpha=0.3;
psi=0.7;
T=100;
phi(1)=0;
maxit=1000;
%La ecuación de Euler es:
%U'(Ct)=betar*E{U'(Ct+1)[alpha*At+1*kapt+1^(alpha-1)+(1-
delta)
%Polinomio de ajuste: exp(alpha0)*At^alpha1*kapt^alpha2
%Generamos una secuencia de longitud T
for i=1:T
epsilon(i)=randn;
phi(i+1)=psi*phi(i)+epsilon(i)*var;
aletpc(i)=prodconst*exp(phi(i));
```

```

end
%Estado estacionario
constant1=1-betar*(1-delta);
constant2=betar*alpha*prodconst;
kapss=(constant1/constant2)^(1/(alpha-1));
css=prodconst*kapss^alpha-delta*kapss;
kap(1)=kapss
%Semilla inicial para los parámetros del polinomio
alpha0(1)=log(1/css*(alpha*prodconst*kapss^(alpha-1)+(1-delta)));
alpha1(1)=0.1;
alpha2(1)=0.1;
for l=1:maxit      % Albert's Loop
for j=1:T          % Upper's Loop
%Calculamos el consumo
cons(j)=1/(betar*exp(alpha0(l))*aletpc(j)^alpha1(l)*kap(j)^alpha2(l));
%Calculamos el capital
kap(j+1)=aletpc(j)*kap(j)^alpha-cons(j)+(1-delta)*kap(j);
end                %End of Upper's loop
kap=kap(1:T);
%Cálculo del lado derecho del plinomio
for j=1:T-1
pol(j)=1/cons(j+1)*(alpha*aletpc(j+1)*kap(j+1)^(alpha-1)+(1-
delta));
end
expl=[exp(ones(size(1:T)))', log(aletpc)', log(kap)'];
b=expl(1:T-1,:)'\log(pol')
finalb=[b]';
coef=[alpha0(1), alpha1(1), alpha2(1)];
maxdifcoef=max(abs(coef-finalb'));
coefl(1,:)=[alpha0(1), alpha1(1), alpha2(1)];
if maxdifcoef<0.0001
break
else
alpha0(l+1)=(finalb(1));
alpha1(l+1)=finalb(2);
alpha2(l+1)=finalb(3);
end
end                %Albert's Loop

```



```

subplot(2,2,1);plot([log(pol)' (expl(1:T-1,:)*b)] ); title('Fit')
subplot(2,2,2);plot(kap); title('Kapital')
subplot(2,2,3);plot(cons);title('Consump')
subplot(2,2,4);plot(coef1);title('Param')

```

Con este programa surgen los siguientes gráficos: Figure 1 muestra la secuencia cuando el valor de k_0 es menor que el de estado estacionario. El gráfico titulado "Fit" muestra cómo el polinomio es ajustado con los valores de los parámetros mostrados en "Param". Se puede ver cómo la convergencia se consigue con bastante rapidez.

