



## Modelo Informático de la Seguridad Social (MISS)

Alvaro Forteza

### Ejercicios <sup>1</sup>

1	Control del flujo del programa usando completo.bat.....	2
2	Estimación del consumo óptimo.....	5
3	Historias laborales (senderos de ocio y de declaración de actividad al BPS).....	8
4	Algunos escenarios simples .....	13
5	La introducción del COFIS y la reducción de los aportes patronales.....	16
6	La agregación y los resultados macro.....	18

---

<sup>1</sup> Versión WEB



# 1 Control del flujo del programa usando completo.bat.

## 1.1 Motivación y objetivos

- a) Mostrar cómo se controla el flujo del programa con completo.bat.
- b) Mostrar el contenido de resgen.gms. Por ser relativamente pequeño, se puede abrir sin problemas. Mostrar que este archivo recoge los cálculos realizados por familias.gms para cada tipo de trabajador. Son los datos "micro".
- c) Mostrar el control de las generaciones en comunes.gms.
- d) En la tercera variante (ver más abajo), mostrar cómo activar agregados.gms e interpretar un mensaje de error.

## 1.2 Características de la corrida

Generaciones 1916 y 1917. Un sendero de ocio y declaración al BPS. Solvers desactivados.

## 1.3 Ejecución de la corrida

- a) Iniciar el proyecto: clickear en File, Project, New Project. Elegir el subdirectorio de trabajo (Taller MISS/Ejercicios MISS).<sup>2</sup> Bautizar el proyecto como Taller3.gpr.
- b) Abrir completo.bat: clickear en File, Open, Tipo de archivos: All Files, clickear en completo.bat. Comentamos su contenido.
- c) Apretar el botón de DOS en el extremo superior derecho del panel de control. Se pasa a ambiente DOS.
- d) En ambiente DOS, escribir COMPLETO y apretar Enter. Se ejecuta el programa. Al terminar el proceso, se observan mensajes como el siguiente:

\* call gams familias\_1303.gms  
Comando o nombre de archivo incorrecto

Comentario: El mensaje sugiere que hay un error, pero está todo bien. El DOS no entiende el asterisco (de allí el mensaje de comando no válido) y lo que hace es pasar a la línea siguiente, salteándose el comando que queríamos anular.

---

<sup>2</sup> Atención: no usar tildes al nombrar carpetas y tratar de usar nombres cortos compatibles con DOS. Los solver pueden no funcionar en caso contrario.



- e) Al completar el proceso escribir EXIT y apretar Enter. Se vuelve al ambiente GAMS.
- f) Si miramos en el subdirectorío de trabajo (Taller MISS/Ejercicios MISS), vemos que aparece un nuevo archivo llamado OCUPAYDECLARA.DAT.
- g) No tenemos cuadros.dat, porque anulamos la orden de ejecutar agregados.gms.
- h) Abrir Resgen.gms. Observar que tiene un montón de datos para las generaciones 1916 y 1917. Son los resultados de los cálculos hechos por familias.gms. La presentación de estos datos es poco clara para el usuario, pero el archivo agregados.gms los lee correctamente.
- i) Cerrar Resgen.gms.

## **1.4 Variantes**

### **1.4.1 Desactivando senderos.gms**

Una vez que se dispone de ocupa y declara.dat, no es necesario volver a estimarlo hasta que se quiera definir otro conjunto de senderos a evaluar.

- a) En completo.bat, poner:

```
* call gams senderos_1303
```

y guardar completo.bat modificado.

- b) Apretar el botón de DOS en el extremo superior derecho del panel de control.
- c) En ambiente DOS escribir: completo. Apretar Enter.
- d) escribir EXIT, apretar Enter. Se vuelve al ambiente GAMS.

### **1.4.2 Desactivando la generación 1916**

Anular la ejecución de la generación 1916, dejando sólo la generación 1917:

- a) En completo.bat, poner:

```
* echo run("1916") = yes; >generation.gms  
* call gams familias_1303
```

y guardar completo.bat modificado.



b) Abrir listas.inc.

b.1) Cambiar el contenido del conjunto g. Para ello, clickear el botón Search en el panel de control y luego Find o presionar Ctrl F. Escribir: generaciones y confirmar la búsqueda. El cursor se detiene en la siguiente expresión:

```
g    generaciones / 1916*1917 /
```

Cambiar por:

```
g    generaciones / 1917 /
```

b. 2) Cambiar la generación inicial. Buscar la expresión gfirst (usando Ctrl+F gfirst) y poner 1917:

```
gfirst generacion inicial /1917/
```

b. 3) Guardar listas.inc modificado.

c) Apretar el botón de DOS en el extremo superior derecho del panel de control.

d) En ambiente DOS escribir: completo. Apretar Enter.

e) Al terminar el proceso escribir EXIT, apretar Enter. Se vuelve al ambiente GAMS.

f) Volver a abrir resgen.gms y ver que ahora sólo recoge la generación 1917.

g) Cerrar resgen.gms.

### 1.4.3 Activando agregados.gms

a) En completo.bat,

```
sustituir:  * call gams agregados_1303  
por:       call gams agregados_1303
```

y guardar completo.bat modificado.

b) Apretar el botón de DOS en el extremo superior derecho del panel de control.

c) En ambiente DOS escribir: completo. Apretar Enter.

d) Observar: mensaje de error. Escribir EXIT, apretar Enter. Se vuelve al ambiente GAMS.



e) Abrir Agregados.*lst*. Se obtiene mensaje:

```
--- 13167 !!! Cotizantes al BPS en 1980 no coincide con datos estimados !!!
```

```
**** Exec Error 20 at line 13167 .. User abort: execution halted
```

Comentario: Agregados.gms intentó hacer los cálculos macroeconómicos y obtuvo un resultado inaceptable: el número de cotizantes al BPS en 1980 no es correcto. En consecuencia, el programa abortó. Está bien que ocurra esto, porque con la generación de 1917 solamente no podemos reproducir el número de cotizantes que estimamos que tenía el BPS en 1980.

## 2 Estimación del consumo óptimo

### 2.1 Motivación y objetivos

- a) Cambios en tasas del IVA y COFIS inducen cambios del consumo. La recaudación puede variar menos que proporcionalmente con las tasas impositivas, debido a cambios en la base imponible.
- b) El efecto que la reforma de la seguridad social tiene en los trabajadores depende, entre otras cosas, de su situación patrimonial al iniciarse la reforma, que a su vez depende de su consumo previo.
- c) Introducir el comando "solve". Mostrar algunos mensajes comunes del solver.

### 2.2 Características de la corrida

- a) Generación 1970.
- b) El comando "solve" *activado*.

### 2.3 Ejecución de la corrida

- a) Abrir completo.bat y anular nuevamente la ejecución de agregados.gms. Anular la generación 1917 y activar la generación 1970:

```
...  
* echo run("1917") = yes; >generation.gms  
* call gams familias_1303  
...  
echo run("1970") = yes; >generation.gms  
call gams familias_1303  
...  
* call gams agregados_1303
```



b) Abrir listas.inc. Modificar el conjunto g:

g generaciones / 1970 /

y el parámetro gfirst:

gfirst generacion inicial /1970/

Guardar listas.inc modificado.

c) Ejecutar completo.bat.

d) Abrir Familias.lst. Buscar (Ctrl+F): cons. Observar el "display" del consumo:

```
INDEX 1      = siop8 INDEX 2      = H
          1996      1997      1998      1999      2000      2001      2002      2003
AA.1970 12694.261 12694.261 12694.261 12694.261 12694.261 12694.261 12694.261 12694.261
```

Comentarios:

d.1) La variable es Cons(r,s,i,g,t):

Set	Elementos del set	Interpretación
r	siop8	siop8 = optantes por el artículo 8
s	H	Hombres
i	AA	Curva salarial Alto-alto.
g	1970	Generación 1970
t	1996, 1997, ... , 2003, ...	Años

d.2) En esta corrida, el consumo es *constante* a lo largo de la vida del individuo.

Explicación: cuando mantenemos desactivado el comando "solve", el programa calcula el nivel de consumo constante que el trabajador puede permitirse a lo largo de toda su vida dados los ingresos netos de transferencias que tiene.

d.3) Problemas con el consumo constante: probablemente no sea una buena representación del consumo a lo largo de la vida y no refleja la sensibilidad del consumo ante cambios en las tasas de impuestos.

d.4) Cerrar familias.lst.

e) Abrir familias.gms, para habilitar los comandos "Solve" y "Abort".

e.1) Buscar el comando Solve. Para ello, clicar el botón Search en el panel de control y luego Find o presionar Ctrl F. Escribir solve y confirmar la búsqueda. El cursor se detiene en la siguiente expresión:

\* Solve pre maximizing auxiliar using nlp;

\* ABORT\$[(pre.MODELSTAT GT 2)]" !!! pre.MODELSTAT GT 2 !!! ", pre.MODELSTAT ;



Comentarios: La sentencia "solve pre..." indica que debe resolverse el modelo pre (previo a la reforma). El "ABORT..." dice que el proceso debe detenerse si el modelo pre no se resolvió satisfactoriamente.

e.2) Los asteriscos anulan los comandos indicados. Para habilitarlos hay que eliminar los asteriscos:

```
Solve pre maximizing auxiliar using nlp;  
ABORT$(pre.MODELSTAT GT 2)] !!! pre.MODELSTAT GT 2 !!! ", pre.MODELSTAT ;
```

e.3) Volver a buscar el comando solve (el primero corresponde al modelo pre y el segundo al modelo pos). Clickear el botón Find *again* o presionar F3. El cursor se detiene en la siguiente expresión:

```
* Solve pos maximizing auxiliar using nlp;  
* ABORT$(pos.MODELSTAT GT 2)] !!! pos.MODELSTAT GT 2 !!! ", pos.MODELSTAT ;
```

e.4) Eliminar los asteriscos para habilitar la solución del modelo pos.

e.5) Salvar familias.gms.

f) Ejecutar completo.bat. Volver a ambiente GAMS.

g) Abrir familias.*lst*.

g.1) Buscar MODEL STATUS, usando Ctrl F. Obtenemos lo siguiente:

```
      SOLVE SUMMARY  
  
MODEL pre          OBJECTIVE AUXILIAR  
TYPE NLP           DIRECTION MAXIMIZE  
SOLVER MINOS      FROM LINE 3501  
  
**** SOLVER STATUS 1 NORMAL COMPLETION  
**** MODEL STATUS  2 LOCALLY OPTIMAL
```

Comentarios:

- Estamos viendo un resumen del estatus de la resolución del modelo pre.
- El resultado es el buscado. En particular, el MODEL STATUS 2 LOCALLY OPTIMAL es lo que buscamos en un proceso de programación no lineal.
- Notar que la orden ABORT que dimos no operó porque el model status es precisamente 2.

g.2) Volver a buscar MODEL STATUS presionando F3 (o eligiendo Find Again en el panel de control). Obtenemos un display similar al anterior, pero para el modelo pos.



g.3) Buscar el parámetro cons (consumo), haciendo Ctrl F : cons  
Observar el "display" del consumo:

INDEX 1	= siop8	INDEX 2	= H					
	1996	1997	1998	1999	2000	2001	2002	2003
AA.1970	11009.777	11009.777	11009.777	11009.777	11009.777	10955.949	10955.949	10955.949

Comentarios: El consumo disminuye a partir de 2001, que es el año en que se introduce el COFIS.

### 3 Historias laborales (senderos de ocio y de declaración de actividad al BPS)

#### 3.1 Motivación y objetivos

- Mostrar la modelización de la *historia laboral efectiva*, es decir de la verdadera situación laboral del trabajador a lo largo de su vida. Cómo representar distintas historias laborales efectivas.
- Mostrar la modelización de la *historia laboral declarada* o historia laboral a secas. Cómo representar distintas historias laborales.
- Mostrar cómo el programa selecciona el sendero o historia laboral preferido por el trabajador entre un conjunto de opciones posibles. Las decisiones de trabajar, retirarse y evadir.

#### 3.2 Características de la corrida

- a) Generación 1970.
- b) El comando "solve" habilitado.
- c) *Varios* senderos k.

Nota: El tiempo estimado de esta corrida con 9 senderos, en un PC con 384 Mb de memoria RAM, es *7 minutos*. Puede ser bastante mayor en equipos con poca memoria. Quienes estén trabajando en el taller con equipos de menos de 256 Mb hagan este ejercicio con los senderos 1 y 2 solamente (ver detalles más abajo, punto f).

#### 3.3 Ejecución de la corrida

- a) Abrir completo.bat. Verificar que agregados.gms está desactivado. Verificar que sólo está activada la generación 1970. Salvar.



b) Abrir ocupa y declara.dat. Extracto del contenido:

kcounter =9 ;

Table opciono(s,a,k)

```

      1  2  3  4  5  6  7  8 ...
H.18 24 24 24 24 24 24 24 24 ...
H.19 24 24 24 24 24 24 24 24
H.20 24 24 24 24 24 24 24 24
...
    
```

Comentarios:

b.1) kcounter es un parámetro que cuenta el número de senderos de trabajo y declaración al BPS (historias laborales) que serán evaluadas en la corrida.

b.2) Table opciono(s,a,k) contiene las historias laborales *efectivas*, presentadas como "senderos de ocio". Cada columna es un sendero de ocio (o historia) e indica las horas diarias de ocio del trabajador a la edad que corresponde a la fila. En la versión actual, trabajamos con dos opciones: 24 y 16 horas de ocio, es decir 0 y 8 hs diarias de trabajo. El comando Table produce la variable (parámetro en la jerga de GAMS) opciono(s,a,k):

Set	Elementos del set	Interpretación
s	H	Hombres
a	18, 19, 20, ...	Edad (a por "age")
k	1, 2, 3, ..., 8, ...	Senderos de ocio a ser evaluados

c) Seguimos en ocupa y declara.dat. Buscar la palabra *opciond* (presionar Ctrl F o clicar en el panel de control y escribir opciond).

Table opciond(s,a,k)

```

      1  2  3  4  5  6  7  8
H.18 1  1  1  1  1  1  1  1
H.19 1  1  1  1  1  1  1  1
H.20 1  1  1  1  1  1  1  1
    
```

Comentarios:

c.1) Table opciond(s,a,k) contiene las historias laborales *declaradas*, o senderos de declaración de actividad al BPS. Cada columna es un sendero. Adopta el valor 1 si el trabajador se declara inactivo a la edad correspondiente a la fila y 0 si se declara activo.



c.2) Representación de la evasión. Si miramos el caso de un hombre ("H") en el sendero "9" a las edades 63 a 66 tenemos lo siguiente:

Edad	opciono("H",a,"9")	opciond("H",a,"9")
63	16	0
64	16	0
65	16	0
66	24	1

En este sendero o historia laboral, el trabajador está declarando verazmente. A los 63, 64 y 65 años está trabajando (opciono = 16, trabaja 8 hs) y declara hacerlo (opciond = 0). A los 66 años se retira (opciono = 24, trabaja 0 hs) y se declara inactivo (opciond = 1).

Consideremos ahora el sendero "5":

Edad	opciono("H",a,"5")	opciond("H",a,"5")
63	16	0
64	16	<i>1</i>
65	16	<i>1</i>
66	24	1

En este sendero, el trabajador está declarando verazmente a los 63 y 66 años, pero está evadiendo a los 64 y 65, ya que está trabajando pero se declara inactivo.

c.3) El programa MISS evalúa cuál de estos dos senderos es preferible para el trabajador y con ello determina si al trabajador que ha decidido dejar de trabajar efectivamente a los 66, le conviene o no declararse inactivo a partir de los 64.

d) Abrir listas.lst. Buscar k2 (Ctrl+F, k2). Obtenemos:

k2(k) subset para correr distintos senderos

Estamos declarando el subconjunto k2(k), que es el subconjunto de los senderos k que vamos a evaluar en la corrida.

e) Abrir familias.gms. Buscar k2 (Ctrl+F, k2). Obtenemos:

\$ontext

-----  
 Subset considerado para hacer el loop en senderos. Hay que elegir \$(ord(k)<=kcounter) para evaluar todos los senderos generados en Senderos\_0411.gms. Elegir \$(ord(k)<=n) para evaluar los senderos con k<=n, y \$(ord(k)=n) para evaluar solo el sendero n; siendo n<=kcounter.  
 -----

\$offtext

k2(k)=no;



```
k2(k)=yes $(ord(k)=5);  
*k2(k)=yes $(ord(k)<=9);
```

Comentarios:

e.1)  $k2(k)=no$ ; establece que el conjunto  $k2(k)$  está vacío.

e.2)  $k2(k)=yes \$(ord(k)=5)$ ; establece que el conjunto  $k2(k)$  contiene al sendero 5.

e.3)  $k2(k)=yes \$(ord(k)<=9)$ ; establece que el conjunto  $k2(k)$  contiene los senderos 1 a 9.

e.4) La tercera orden está desactivada por el asterisco, por lo cual el programa está corriendo sólo para el sendero 5.

f) Vamos a evaluar los primeros 9 senderos activando la orden correspondiente:

```
k2(k)=yes $(ord(k)<=9);
```

(eliminamos el asterisco).

*Nota:* quienes tengan equipos con menos de 256 Mb de RAM pongan 2 en lugar de 9.

g) Volvemos a buscar de nuevo (presionamos F3 dos o tres veces), hasta llegar a lo siguiente:

```
loop (k $k2(k) ,
```

Este comando indica que se inicia un loop en  $k$  que debe correr para todos los elementos de  $k$  que están también en  $k2(k)$ . Hasta ahora estuvimos estimando sólo el sendero 5, ahora vamos a estimar los senderos 1 a 9 (1 a 2 los que tengan restricciones de capacidad de equipo).

h) Miramos rápidamente lo que sigue. Aparecen fórmulas para variables tales como:  $serv(rpre,s,i,g,t)$ ,  $cj(rpre,s,i,g,t)$ ,  $tjub(rpre,s,i,g)$ , etc.

i) Volvemos a archivo listas.inc. Buscamos  $serv($  , haciendo Ctrl + F,  $serv($  . Obtenemos la *declaración del parámetro*:

```
serv(r,s,i,g,t) numero de años trabajados hasta el fin del año t
```

j) Volvemos a familias.gms. Buscamos  $serv($  y volvemos a las fórmulas que vimos en el punto h), es decir a la *asignación del parámetro*:

```
serv(rpre,s,i,g,t) = Sum (tt $ (gd(s,g,tt) and (year(tt) < year(t))),  
(1-d(rpre,s,i,g,tt))) $gd(s,g,t);
```



Esta fórmula calcula los años de servicio del trabajador en el sendero  $k$ , teniendo en cuenta lo que el trabajador está declarando al BPS. El año en que el trabajador se declara activo,  $d()=0$ ,  $1-d()=1$  y  $\text{serv}()$  aumenta en uno.

k) Bajamos con la flecha. Nos encontramos con:

\* Inclusion de archivo que calcula las jubilaciones por regimen anterior

\*-----

\$include prestaciones1\_1303.inc

l) Bajamos un poco más y volvemos a encontrar el comando solve.

Comentario: para cada valor de  $k$  incluido en  $k2(k)$ , estamos calculando los ingresos del trabajador, las transferencias  $a$  y desde la seguridad social, los impuestos, etc. y estimamos su consumo óptimo. Luego evaluamos su bienestar ( $\text{Welfare}(r,pre,s,i,g)$ ) y volvemos a empezar con otro sendero en  $k2(k)$ . Finalmente, elegimos el sendero que da mayor bienestar.

m) Vamos al final del archivo y activamos el display siguiente:

Display kposop;

Para saber qué es esto, buscamos la declaración del parámetro en el archivo listas.inc (Find kposop) y encontramos:

kposop(r,s,i,g) sendero de ocio y declaracion a la SS optimo despues de la reforma

Esto nos permite ver cuál es el sendero preferido por el trabajador  $(r,s,i,g)$  en el escenario que estamos considerando, entre los 9 senderos que evaluamos.

o) ¡Salvar familias.gms!

p) Presionar botón DOS. Ejecutar completo.bat. Al terminar la corrida en DOS, digitar EXIT, presionar Enter. Volvemos al ambiente de Gams.

q) Abrir Familias.lst. Buscar kposop. Obtenemos:<sup>3</sup>

---

<sup>3</sup> Quienes corrieron solamente dos senderos, obtienen una salida diferente.



---- 4300 PARAMETER kposop sendero de ocio y declaracion a la SS optimo despues de la reforma

INDEX 1 = mxop8

1970  
H.AA 5.000  
H.AM 5.000  
H.MM 5.000  
H.MB 5.000  
H.BB 5.000  
M.AA 5.000  
M.AM 5.000  
M.MM 5.000  
M.MB 5.000  
M.BB 6.000

Comentarios:

q.1) Estamos mirando trabajadores amparados por el régimen mixto que han optado por el artículo 8 (mxop8), hombres y mujeres (H y M), las cinco curvas de ingreso laboral (AA, AM, MM, MB, BB) y generación 1970. Casi todos tienen preferencia por el sendero 5, salvo las mujeres de la curva más baja, que prefieren el 6. Más abajo aparecen otros trabajadores que hacen otras opciones.

q.2) No verificamos el MODEL STATUS, pero el hecho de que el programa no abortó indica que se obtuvo la solución óptima.

q.3) En general, no vamos a estar interesados por identificar el sendero en sí, sino por ver cuáles son los valores de variables tales como los aportes a la seguridad social, las prestaciones, año de jubilación, etc. y este es el tipo de información que almacenamos en resgen.gms. Notar que ahora vamos a estar teniendo una gran diversidad de historias laborales y de perfiles de consumo, dependiendo de las condiciones particulares de los distintos tipos de trabajadores.

## 4 Algunos escenarios simples

### 4.1 Motivación y objetivos

Mostrar cómo pueden simularse algunos escenarios alternativos sencillos modificando valores de algunos parámetros como la tasa de interés. Es completamente análogo para la tasa de crecimiento de la productividad del trabajo.<sup>4</sup>

---

<sup>4</sup> En este modelo, la tasa de crecimiento de la productividad del trabajo incide directamente en la tasa de crecimiento de los salarios reales y del PBI. Por lo tanto, al simular distintas tasas de crecimiento de la productividad del trabajo, se están simulando distintas tasas de crecimiento de los salarios reales y del PBI.



## 4.2 Características de la corrida

Generación 1970. Un sendero de ocio y declaración al BPS. Solvers activos.

Comentario: Ponemos un solo sendero de ocio y declaración para ahorrar tiempo.

## 4.3 Ejecución de la corrida

a) Abrir completo.bat. Verificar que la única generación activa es 1970 y que Agregados.gms está desactivado. Guardar.

b) Abrir listas.inc. Verificar que el set g sólo tiene el elemento 1970 y que gfirst es 1970. Guardar.

c) En listas.inc, buscar: Tasa de interes. Se obtiene:

```
rr          tasa de interes anual / .038 /
```

Modificar a:

```
rr          tasa de interes anual / .020 /
```

d) Salvar listas.inc.

e) Abrir familias.gms.

e.1) Verificar que los solvers están activos (buscar solver)

e.2) Verificar que corremos sólo el sendero 5. Buscar k2. Poner lo siguiente:

```
k2(k)=no;  
k2(k)=yes $(ord(k)=5);  
*k2(k)=yes $(ord(k)<=9);
```

f) Correr completo.bat desde DOS. Al terminar, escribir EXIT y volver al ambiente GAMS.

g) Abrir familias.lst. Buscar: rr. Se obtiene:

```
---- 4542 PARAMETER rr          =      0.038  tasa de interes anual
```

h) En familias.lst, buscar: ASSETS.L . Se obtiene:



---- 4542 VARIABLE ASSETS.L activos

INDEX 1 = ra

	1988	1989	1990	1991	1992	1993	1994
H.AA	-6819.020	-13897.163	-21244.275	-28870.578	-36786.680	-45003.594	-49882.688

i) En listas.inc, buscar: Tasa de interes. Se obtiene:

rr tasa de interes anual / .038 /

Modificar a:

rr tasa de interes anual / .020 /

j) Salvar listas.inc.

k) Correr completo.bat desde DOS. Volver a GAMS.

l) Abrir nuevamente (Reopen) familias.lst. Buscar: rr. Se obtiene:

---- 4542 PARAMETER rr = 0.020 tasa de interes anual

m) En familias.lst, buscar: ASSETS.L . Se obtiene:

---- 4542 VARIABLE ASSETS.L activos

INDEX 1 = ra

	1988	1989	1990	1991	1992	1993	1994
H.AA	-7579.668	-15310.929	-23196.815	-31240.419	-39444.895	-47813.461	-53293.793

Comentarios:

m.1) Los activos iniciales son más negativos con una tasa de interés de 2 por ciento que con una tasa de 3.8 por ciento. El trabajador estará más dispuesto a endeudarse inicialmente (cuando es joven) y tendrá mayor capacidad de hacerlo si la tasa de interés es menor.

m.2) Es más probable que estemos interesados en observar cómo evolucionan los fondos acumulados en las AFAP o la deuda pública con distintas tasas de interés. No lo hacemos ahora, porque para eso necesitamos correr el programa para todas las generaciones y eso lleva mucho tiempo.

o) Abrir listas.inc. Volver a modificar la tasa de interés. Buscar: Tasa de interes. Poner:

rr tasa de interes anual / .038 /

Salvar listas.inc.



## 5 La introducción del COFIS y la reducción de los aportes patronales

### 5.1 Motivación y objetivos

Mostrar cómo se simulan efectos de *paquetes* de política económica. En particular, se simulan los efectos de la introducción del COFIS y de la reducción de los aportes patronales.

### 5.2 Características de la corrida

Generación 1970. Un sendero de ocio y declaración al BPS. Solvers activos.

Comentario: Ponemos un solo sendero de ocio y declaración para ahorrar tiempo.

### 5.3 Ejecución de la corrida

- Abrir completo.bat. Verificar que la única generación activa es 1970 y que Agregados.gms está desactivado. Guardar.
- Abrir listas.inc. Verificar que el set g sólo tiene el elemento 1970 y que gfirst es 1970. Guardar.
- Abrir comunes.gms. Buscar: Tasas de aportes. Obtenemos:

\* Tasas de aportes

\*-----

```
per(r)= Sum(raux $(regimen(r)=regimena(raux)),seriepatper(raux,"per"));  
pat(r,t)= Sum(raux $(regimen(r)=regimena(raux)),seriepatper(raux,"pat"));
```

\* Supongo ahora que la tasa patronal se reduce en dos puntos porcentuales a partir

\* de 2001 en el regimen nuevo, coincidiendo con la introduccion del cofis

```
pat(r,t) $[(year(t)>=2001) and ((regimen(r)=titV) or (regimen(r)=rt) or  
(regimen(r)=siop8) or (regimen(r)=sino8) or (regimen(r)=mxop8) or  
(regimen(r)=mxno8)))] = pat(r,t)- 0.02;
```

Comentarios:

- Las primeras dos sentencias asignan a los parámetros per(t) y pat(r,t) las tasas de aportes contenidas en el archivo de datos datos.inc en la tabla seriepatper().



c.2) La tercera sentencia reduce la tasa de aportes patronales en dos puntos porcentuales a partir del 2001 en los pilares del régimen nuevo. Si desactivamos esta sentencia, simulamos lo que habría ocurrido sin reducción de aportes patronales.

d) En comunes.gms. Buscar: Tasas de iva afectado. Obtenemos:

\* Tasas de iva afectado a la seguridad social y cofis

\* Escenario con cofis

```
TimpBPS(t)= [TimpBPSa("1990") $(year(t)>=1990)and (year(t)<=1992)]  
+ TimpBPSa("1993") $(year(t)>=1993)]*ponbas  
+ [TimpBPSa("2001") $(year(t)>=2001)]*poncof;
```

\* Escenario sin cofis

```
* TimpBPS(t)= [TimpBPSa("1990") $(year(t)>=1990)and (year(t)<=1992)]  
+ TimpBPSa("1993") $(year(t)>=1993)]*ponbas ;
```

Comentarios:

d.1) Escenario con cofis. El parámetro TimpBPS(t) recoge los impuestos afectados al BPS, incluyendo el COFIS.

d.2) Escenario sin cofis. La fórmula es similar a la anterior, pero no incluye el cofis. Si queremos simular un escenario sin Cofis, tenemos que desactivar la primera fórmula y activar la segunda.

e) En comunes.gms, buscar: Tasas medias de iva y cofis. Obtenemos:

\* Tasas medias de iva y cofis

\$ontext

-----  
A continuacion se plantean dos formulas alternativas para simular  
la evolucion del iva y del cofis. Hay que desactivar la formula que no se  
quiere utilizar en la simulacion en curso.  
-----

\$offtext

\* Tasa media calculada a partir de sus componentes

```
tbasicaiva(t)= 0.23 ;  
tminimaiva(t)= 0.14 ;  
tcofis(t) $(year(t)>=2001) = 0.03 ;  
* tcofis(t)=0;  
ivacof(t)= ponbas*tbasicaiva(t) + ponmin*tminimaiva(t) + poncof*tcofis(t) ;
```

e.1) El parámetro ivacof(t) contiene una tasa media de impuestos indirectos calculada a partir de las tasas básica y mínima de iva y de cofis.

e.2) Si se quiere simular sin cofis, hay activar la sentencia siguiente:



tcofis(t)=0;

f) Una vez elegida la opción deseada - con cofis y reducción aportes patronales o sin cofis ni reducción aportes patronales -, se salva comunes.gms.

g) No vamos a hacer durante el taller lo que sigue, por falta de tiempo. En resumidas cuentas, habría que:

g.1) Volver a DOS y correr completo.

g.2) Cambiar el nombre del archivo resgen.gms. Poner, por ejemplo, resgenconcofis.gms, si la corrida fue con cofis, y resgensincofis.gms, si la corrida fue sin cofis. Al hacer esto, evitamos que los resultados archivados en resgen.gms sean borrados en la corrida siguiente.

g.3) Abrir agregados.gms. Buscar resgen.gms. Para procesar el escenario con cofis poner:

```
$include resgenconcofis.gms  
* $include resgen
```

Para procesar el escenario sin cofis poner:

```
$include resgensincofis.gms  
* $include resgen
```

g.4) Los cuadros.dat recogerán los resultados del escenario simulado. Los efectos del paquete compuesto por introducción del cofis y reducción de aportes patronales surgen por comparación de las trayectorias de las variables contenidas en los cuadros de salida en ambos escenarios.

## 6 La agregación y los resultados macro

Advertencia: debido a problemas de tamaño, la base de datos simulados necesaria para realizar este ejercicio no está disponible en la WEB. El ejercicio puede realizarse después de hacer una corrida completa del modelo y generar una nueva base de datos.

### 6.1 Motivación y objetivos

a) Mostrar cómo generamos las variables agregadas a partir de los resultados individuales.

b) Ver cómo se extraen resultados para ser recogidos luego en planilla electrónica.

c) Mostrar algunos de los tests que diseñamos para verificar que los resultados son "correctos" (en el sentido de ser coherentes y de reproducir la información estadística disponible para algún año inicial).



## 6.2 Características de la corrida

Trabajamos con un archivo resgen.gms obtenido previamente en una corrida completa y que hemos llamado Resgenconcofis1632\_3010.gms.

## 6.3 Ejecución de la corrida

a) Iniciar un nuevo proyecto. Clickear en File, Project, New Project. Elegir el subdirectorio de trabajo (Taller MISS/Sesion4). Bautizar el proyecto como Taller4.gpr.

b) Abrir completo.bat. Desactivar todo (con \*) salvo:

```
if exist resgen.gms del resgen.gms
...
call gams agregados_1303
```

Nota: En este ejercicio desactivamos el programa familias.gms para todas las generaciones porque vamos a usar una base de datos simulados generada previamente (Resgenconcofis1632\_3010.gms). En una simulación típica habría que *activar* todos los programas contenidos en completo.bat.

c) Guardar completo.bat.

d) Abrir listas.inc. Verificar generaciones:

```
g    generaciones / 1916*2032 /
gfirst generacion inicial /1916/
```

e) Guardar listas.inc.

f) Abrir agregados.gms. Buscar: Include resgen (Ctrl+F Include resgen). Obtenemos:

```
$include resgenconcofis1632_3010
*$include resgen
```

Notar: está desactivado el comando que incluye resgen.gms y, en cambio, tenemos un comando include del archivo resgenconcofis1632\_3010.gms.

g) Ejecutar completo.bat en DOS. (Apretar el botón de DOS en el extremo superior derecho del panel de control. Se pasa a ambiente DOS. En ambiente DOS, escribir COMPLETO y apretar Enter. Se ejecuta el programa. Al completar el proceso escribir EXIT y apretar Enter. Se vuelve al ambiente GAMS).

h) Recogemos los cuadros.dat.



h.1) Ir al explorador de Windows. Ir al subdirectorío de trabajo (Taller MISS/Sesion4). Vemos que están los archivos Cuadro1.dat a Cuadro5.dat.

h.2) Abrir una planilla excel. Clickear en: Archivo, Abrir. Ir al directorío de trabajo (Taller MISS/Sesion4). En Tipo de archivo:, elegir: Todos los archivos (\*.\*). Abrir Cuadro1.dat.

h.3) En el asistente para importar texto elegir De ancho fijo. Presionar Siguiete.

h.4) En la vista previa, deben aparecer las columnas separadas. Presionar Siguiete y Finalizar.

h.5) Si el PC está configurado para trabajar con coma decimal (en lugar de punto decimal), hay que sustituir los puntos por comas. Esto puede hacerse con el comando Reemplazar. Donde dice Buscar:, poner . y donde dice Reemplazar con:, poner , . Luego elegir Reemplazar todas. Salvar el archivo.

i) Identificación de las variables de los Cuadros.dat. Las variables tienen nombres que casi se explican solos. De todos modos, la descripción se encuentra en listas.inc. Buscar asisBPSpbi. Obtenemos la siguiente declaración:

```
AsisBPSpbi(t)  Asistencia financiera al BPS en proporcion del PBI
```

j) El comando Display en Agregados.gms. El diseño de los cuadros.dat da trabajo. Podemos querer mirar alguna variable antes de decidir incorporarla en un cuadro.dat usando el Display. A modo de ejemplo, pusimos en Agregados.gms que hiciera el display de JubBPSpbi. Vamos a Agregados.lst y buscamos este parámetro, haciendo Ctrl+F JubBPSpbi. Obtenemos:

```
---2989279 PARAMETER JubBPSpbi  Jubilaciones pagadas por el BPS en porcentaje del pbi  
1995 7.676,  1996 7.816,  1997 7.718,  1998 7.623,  1999 7.548,  2000 7.064,  2001 6.819,
```

Otro ejemplo. En agregados.lst, buscar exjub. Obtenemos:

```
---2989132 PARAMETER exjub          =    0.208 Exceso de jubilaciones o pagos a jee en proporcion  
del gasto total del  BPS
```

k) Tests usando el comando ABORT. Vamos a agregados.gms.

k.1) Buscamos ABORT. Obtenemos:

```
testjubil(r,s,i,g,t)= 1 $[jubil(r,s,i,g,t)>0]  
- 1 $[gd(s,g,t) and (ojala(r,s,i,g,t,"fintjub")<=year(t))and (ojala(r,s,i,g,t,"fintjub")>=1)] ;
```

```
ABORT$[Sum((r,s,i,g,t), Abs(testjubil(r,s,i,g,t)))> 1.E-3]  
" !!! No se verifica que: jubil(>)>0 si y solo si individuo vive y esta jubilado !!! ";
```



Comentario: el parámetro `testjubil()` es distinto de cero si se verifica una y sólo una de las condiciones escritas entre paréntesis rectos. Si eso ocurre, el proceso aborta. La primera de estas condiciones dice que el trabajador  $(r,s,i,g)$  en el año  $t$  recibe una jubilación ( $\text{jubil()} > 0$ ). La segunda dice que el trabajador está jubilado (está vivo, el año corriente es posterior al año de jubilación de ese trabajador y el trabajador se jubila). Habría una inconsistencia si el trabajador recibiera una jubilación sin estar jubilado o estuviera jubilado, pero no recibiera una jubilación.

k.2) Buscar de nuevo. Obtenemos:

```
genNOpro(g) = Sum(t $(year(t)=(gen(g)+18))and (ojala("BPSsr","M","MM",g,t,"fino")=0)),1);
```

```
numgenNOpro = Sum(g,genNOpro(g));
```

```
ABORT$[numgenNOpro > 1.E-3] " !!! Faltan generaciones !!! ";
```

Comentarios:

a) El parámetro `genNOpro(g)` (= generación NO procesada) asume valor 1 si la generación  $g$  no ha sido recogida en `resgen.gms`, a pesar de ser un elemento del set  $g$  (de acuerdo con lo especificado en `listas.inc`). Esto ocurre si, por ejemplo, `familias.gms` abortó para alguna generación.

b) `Agregados.gms` aborta si hay una o más generaciones para las que `genNOpro(g)=1` y nos da el mensaje de error: `!!! Faltan generaciones !!!`. A su vez, el parámetro `genNOpro(g)` nos dice cuál fue la o las generaciones con problemas.

k.3) Mirar el `jubtest` (Ctrl+F `jubtest`):

```
ABORT$[sum((ai,s,i,g),ABS(jubtest(ai,s,i,g)-posvacop(ai,s,i,g)))GT 1.E-3]
" !!! Jubilacion por ahorro individual no agota el fondo !!! ";
```

Comentario: el valor presente de las jubilaciones pagadas por ahorro individual debe igualar al fondo acumulado al momento de retirarse (descontadas las comisiones). `Agregados.gms` aborta si esta condición no se verifica para algún trabajador.

k.4) Hay varios tests que verifican que los resultados agregados coinciden con los datos observados para algún período base. A modo de ejemplo:

```
ABORT$[Abs(JPBPSpi95-9.56)>0.3]
"Egresos por jubilaciones y pensiones en 1995 difiere de lo estimado" ;
```