

APHIS Workshop on Avocado Model

June 16, 2009

**4700 River Road
Riverdale, Maryland 20737**

Contents

1. APHIS Workshop on Avocado Model – Lecture Note
2. A Note on Avocado Supply Calibration
3. GAMS file: “*avocado_data.gms*”
4. GAMS file: “*avocado_model.gms*”
5. GAMS file: “*avocado_results.gms*”
6. Some References

APHIS Workshop on Avocado Model Lecture Note

Caesar B Cororaton, Virginia Tech
June 16, 2009

Objective

The goals of the workshop are: (a) to discuss and learn how to code the Mexican avocado model of Peterson and Orden (2008); and (b) to run the model under various model parameters and scenarios.

The avocado model is discussed in detail in three papers:

1. Peterson and Orden, 2006. "Linking Risk and Economic Assessments in the Analysis of Plant Pest Regulations: The Case of U.S. Imports of Mexican Avocados" (available at: <http://www.ers.usda.gov/publications/ccr25/ccr25.pdf>).
2. U.S. Department of Agriculture, Animal and Plant Health Inspection Services (USDA-APHIS, 2004). "Economic Analysis Final Rule: Allow Fresh Hass Avocados Grown in Approved Orchards in Approved Municipalities in Michoacan, Mexico, to be Imported Into all States Year-Round (APHIS Docket No. 03-022-5) (available at: http://www.aphis.usda.gov/plant_health/spotlights/downloads/avocado_03-022-5analysis.pdf).
3. Peterson, E., and D. Orden. 2008. "Avocado Pests and Avocado Trade." *American Journal of Agricultural Economics*. 90(2): 321-335.

The Peterson/Orden Avocado Model

The articles above describe the complete specification of the model. In brief, the major components of the model are the specification of the demand and supply of avocado. The specification of the demand for avocado is derived as one of the first order conditions for utility maximization, where the form of the utility function is constant elasticity of substitution. The price of avocado that enters into the derived function is affected by two factors: shippers' compliance cost and fixed wholesale profit margins. Shippers' compliance cost is affected by several factors such as packing plant investment; variable cost paid by packers to APHIS, inspectors' cost; number of processing plants; fixed cost paid by packers to reimburse APHIS; proportion of fruit cut in packing houses; and income.

The specification of the supply of avocado is derived as first order condition for revenue maximization subject to a constant elasticity transformation (CET) production possibility frontier. Supply of avocado is affected by several factors: growers' compliance cost (which in turn is affected by a number of items); expected cost of control in the importing region (which is also affected by several items, including the frequency of pest outbreak); productivity loss; and an aggregate of factor inputs.

Market equilibrium is attained where demand is equal to supply. Both the demand and supply were calibrated using base data. Comparative static analysis can be conducted by changing one of the exogenous variables. There are 4 demand regions in the U.S. There are 3 supply regions: California, Chile, and Mexico.

Software and Model Codes

The avocado model requires the GAMS¹ software. There are two versions of the software: (a) a student version which is downloadable for free (www.gams.com) but could only be used to solve model of very limited size and dimension; and (b) a professional version, which is costly, but the exact price varies depending upon the number of numerical solvers included in the package.

The avocado model requires the following files:

- (i) “*avocado_data.xls*” – This is an excel file that contains the data and parameters of the model. The data in this file are the base values. Updates of the data and parameters of the model can be incorporated in this file.
- (ii) “*avocado_data.gms*” – This is a set of GAMS codes² that retrieves data from “*avocado_data.xls*” and assigns them into various user-defined matrices and scalars for use in calibrating the model. This set of codes may only be used once, unless there are updates and changes in the base data and parameters.
- (iii) “*avocado_model.gms*” – This is a set of GAMS codes that calibrates the model to the base values. The codes also include the system of equations in the avocado model specified in different functional forms with various endogenous and exogenous variables. The codes also solve the model and compute the total change in welfare in the U.S. which consists of: consumer surplus; producer surplus; and cost pest mitigation.
- (iv) “*avocado_results.gms*” – This is a set of GAMS codes that writes the results into an excel file.
- (v) “*avocado_results.xls*” – This is an excel file that contains the base values and the simulated changes in the endogenous variables of the model.

Each of these files are explained below and discussed in detail during the workshop.

Excel Data File: *avocado_data.xls*

Open this file and go to worksheet “basic inputs”. The data in this file are base value of the following variables:

¹ General Algebraic Modeling System

² A GAMS file is annexed (suffixed) by “.gms”.

*Demand region in the U.S.	*Supply region	*Pests	* Production period
Region a - currently approved states	cal - California	ff - fruit flies	t1 - October 15 to April 15
Region b - southeast region	ch - Chile	sw - seed weevil	t2 - April 16 to October 14
Region c - southwest region and remainder of pacific	mex - Mexico	stw - stem weevil	
Region d - southern California		sm - seed moth	

The data are organized in matrixes that are enclosed by bold boxes. For example, matrix “data1” has the following values in the *avocado_data.xls*.

Various data in demand regions: data1				
	a	b	c	d
sig1	1.75	1.75	1.75	1.75
sig2	2.75	2.75	2.75	2.75
pop	149.27458650	58.33359650	59.85456950	21.92909800

The values in these boxes were taken from Peterson and Orden (2006 and 2008). These numbers may be changed directly in *avocado_data.xls* if updates are available.

Note: The dimension and the exact range of these boxes should not be altered because they are exactly referenced in “*avocado_data.gms*”. If the dimension of the boxes is changed, the reference in “*avocado_data.gms*” should also be changed. The row and column headings of each of the boxes/matrixes (e.g.in “data1” above, *sig1*... and *a*...) should not be altered as well because they are sets defined in “*avocado_data.gms*”. Modifying these row and column headings will also require corresponding change in the sets defined in “*avocado_data.gms*” and in the rest of the codes in the various files.

Outside of the area enclosed by the bold boxes are “free spaces” where one can write notes.

GAMS File: *avocado_data.gms*

The first few lines in *avocado_data.gms* are options in GAMS which are used to write a title, to control GAMS output, etc., all of which are explained in the GAMS manual, which is downloadable for free.

In GAMS there are two ways of writing notes which are not read in the program:

(a) putting ‘*’ in the first column and at the start of the note line. For example,

* *This is a note...*

(b) writing between two GAMS commands “*\$ontext*” and “*\$offtext*”. For example,

\$ontext

* *This is a note...*

\$offtext

The start of the codes defines the various set of the matrices that will contain base data and parameters for model calibration. For example, the avocado demand regions in the U.S. is indexed by $i = a, b, c, d$. After i is a line description of i , while the phrase after $a\dots$ are descriptions of regions. All these description are noted in GAMS and are displayed automatically in the presentation of results.

```
set i  avocado demand regions in U.S.
    / a  currently approved states
      b  southeast region
      c  southwest region plus remainder of pacific
      d  southern california /
```

To retrieve data from avocado_data.xls one needs to invoke a GAMS facility called GDX³. For example, data in avocado_data.xls within the range from a11 to e14 (which contains model parameters *sig1* and *sig2* - substitution elasticities in the utility function in each demand region in the U.S., and population data in each demand region) can be retrieved using the following commands:

```
$call gdxrw.exe avocado_data.xls par=data1 rng=a11:e14
parameter data1(*,i);
$gdxin avocado_data.gdx
$load data1
$gdxin
```

The data will be in a matrix form called “*data1*”. The “*” in *data1(*,i)* means that the row dimension of *data1* will depend on what is written in the excel file, but the column i means $i=a,b,c,d$. The rest of the relevant data and parameters in avocado_data.xls can be retrieved using similar commands as coded in avocado_data.gms.

After retrieving the data and parameters from the excel file, one needs to define variable and parameters names that will be used to store the base and parameters values. There is no rule in defining variable and parameters names in GAMS. The convention I used is to annex a variable name with a “0” if it stores base values (for example *exp0*). A model parameter not annexed with a “0” means parameter (for example *sig1*). For example, some of the variable names in the code are:

```
parameters4
sig1(i)      elasticity of substitution CES between avocados and all other goods (this is a
              parameter)
sig2(i)      elasticity of substitution CES between avocados (this is a parameter)
```

³ GDX – GAMS Data Exchange

⁴ Note: In GAMS, there are two commands for defining variables. “*parameters*” (or “*parameter*”) is used if the variable defined has fixed value and it is not intended for use as endogenous variable later in the model. Thus, I used “*parameters*” to define variables that will store base values and model parameters. The other command is “*variables*” (or “*variable*”). This command is used in model specification. As we shall see later in that this provides two options: it is suffixed by “.l” if it is an endogenous variable or “.fx” if it is an exogenous variable.

```

beta(j)      exponent for CET revenue function (this is a parameter)
exp0(i,t)    percapita avocado expenditure in period t1 and t2 (this is base value)
inc0(i,t)    base percapita income in period t1 and t2 (this is base value)
...
;

```

Note: As shown above, one has to end the “parameters” command with an “;” to tell GAMS that the command has ended. But after “;” one is free to start another set of variable definition with “parameters ...;”, so on and so forth.

The last part of avocado_data.gms assigns values from taken from the excel file to the define variables and parameters. For example,

```

sig1(i)      = data1('sig1',i);
sig2(i)      = data1('sig2',i);
beta(j)      = data6('beta',j);
exp0(i,'t1') = data11(i,'exp1');
exp0(i,'t2') = data11(i,'exp2');
...

```

“sig1(i) = data1('sig1',i);” means that the value of sig1(i) is taken from matrix data1(). ‘sig1’ tells GAMS to assign row ‘sig1’ in data1() to parameter sig1(i). One can check whether the values of the variables are consistent with the values in the excel file by using the GAMS command “display”. For example, after “sig1(i) = data1('sig1',i);”, one can write “display sig1;” The values of sig(i) are displayed at the end of file avocado_data.lst, which generated after running the codes.

Important Note: In the codes, one needs to save all information generated by avocado_data.gms for use in the next set of codes. This can be done by writing “s=save\dat”. “s” stands for save; “save” the directory where the information is saved; and “dat” the file that contains the information. This is will be illustrated during the workshop. This is important because it will be recalled in the next set of GAMS codes: avocado_model.gms.

GAMS File: avocado_model.gms

This set of GAMS codes calibrates and solves the avocado model. It recalls the results from avocado_data.gms through the following command “r=save\dat”. This will be illustrated during the workshop. “r” stands for recall; “save” the directory where the information is saved; and “dat” the file that contains all relevant information on base value data and parameters.

The putting a set of notes, there is a GAMS command “alias()”. The arguments of this command are the defined set index. For example, “alias(t,ta)” the set t and set ta have the same contents/elements which can be used interchangeably in the program.

The details of these set of GAMS codes will be discussed and illustrated in great detail during the workshop. However, few important points are noted below.

- (a) The idea in model calibration is to start with what is given in the base values and then start with calculating unknown parameters of an equation sequentially. An illustration is given below.
- (b) The avocado model has 5 major parts
 - i. Definition of variables
 - ii. Definition of equations
 - iii. Model Specification
 - iv. Assigning initial values to variables and declaring whether they are endogenous or exogenous.
 - v. Solve statement

Model Calibration

We shall illustrate how the demand function is calibrated. In the “Final Rule” (page 43), demand for avocado in region i is derived as one of first order conditions (FOCs) of utility maximization subject to an income constraint. The utility function is a CES function of avocado and an aggregate quantity of all goods in the consumption basket. One of the FOCs is the derived demand for avocado which is given by

$$(1) \quad x_{ij} = \frac{b_i a_{ij} (p_{ij} + m_{ij})^{-\sigma_2} PI_i^{(\sigma_2 - \sigma_1)} I_i}{b_i PI_i^{(1 - \sigma_1)} + (1 - b_i) PE_i^{(1 - \sigma_1)}}$$

Where the aggregate price PI_i is

$$(2) \quad PI_i = \left\{ \sum_j a_{ij} (p_{ij} + m_{ij})^{(1 - \sigma_2)} \right\}^{\frac{1}{(1 - \sigma_2)}}$$

Where x_{ij} is quantity of avocados from the j th supply region consumed in the i th demand region; p_{ij} is the producer price of avocados from the j th producer region in the i th demand region, m_{ij} is the fixed marketing margin for avocados from the j th producer region in the i th demand region; I_i is per capita income in demand region i ; PE_i is the aggregate price of the composite “all other goods” in region i , which is assumed 1⁵; b_i and a_{ij} are shift parameters (a_{ij} is share parameter, thus $\sum_j a_{ij} = 1$) and σ_s are elasticities of substitution.

⁵ Which is a reasonable assumption since expenditure on avocado is a tiny share in the overall consumption expenditure of households.

In calibrating these two equations, one needs initial values for x , p , m , PI , and I . The codes in avocado_model.gms show how these initial values are calculated. I have included several important notes to guide users.

In a CES⁶ utility function a_{ij} are share parameters (Rutherford 2002). These can be computed using initial values of x and PI . One can easily see this in the codes. The last remaining parameter to be calculated is b_i . This is calculated residually, after calculating all values of the variable and parameters in the equation. The idea here is that when all values of variables and parameters (except for x) are substituted back into the demand function in (1), the exact base value of x is generated. One important note: one has to check this process to make sure that there is a perfect fit in the calibrated demand equation. Otherwise, the model will not solve perfectly. I also indicated in the codes how the checking process can be done.

The calibration of the supply equation is somewhat involved. I have written a short note that accompanies this lecture note so the user can follow the codes in the model. This will be discussed in detail in the workshop.

Model Specification

As stated above, this part has 5 components. The first item is the definition of variable using the GAMS command “variables”. I also follow a convention in defining variables in the model. Unlike the variables that store the base values, variables in the model are not annexed by “0”. For example, the demand for avocado and the regional income are both variables in the model. However, the demand for avocado is an endogenous variable. Since the avocado model is partial equilibrium, the regional income variable is exogenous. To make the codes easier to follow and read, I indicated which variables are endogenous and which are exogenous. Thus, we write

```
variables
*endogenous
x(i,j,t)    demand for avocado
...
*exogenous
inc(i,t)    base percapita income in demand regions in period t1 and t2
...
;
```

The next step is to define equation names. Again, one may adopt a convention. The one I applied annexes variable names with “eq”. For example, if the variable name is “ x ”, then I define the corresponding equation as “ x_{eq} ”. That is,

⁶ CES means constant elasticity of substitution.

equations

$x_{eq}(i,j,t)$ demand for avocado
 $pcost_{eq}(j,t)$ compliance cost
 $cpeq(i,j,t)$ consumer price of avocado or wholesale price of avocado
 ...
 ;

The model specification is organized in three major blocks: the demand block, the supply demand, and market equilibrium model. The demand block specifies the demand function for avocado in i region by j supplier region in period t . In GAMS, an equation is specified in the following way

*demand for avocado
 $x_{eq}(i,j,t) \$cp0(i,j,t).. x(i,j,t) =e= (b(i,j,t)*a(i,j,t)*(cp(i,j,t)**(-sig2(i))) * (pi(i,t)**(sig2(i)-sig1(i)))*inc(i,t)*pop(i) / (b(i,j,t)*(pi(i,t)**(1-sig1(i))) + (1-b(i,j,t)))));$

This specification means the following:

- The name of the equation $x_{eq}(i,j,t)$.
- The option $\$cp0(i,j,t)$ implies that this equation is defined only, in cases where $cp0(i,j,t)$ is non-zero.
- The equation name is followed immediately by two dots ($..$), and at least one space between the equation name and variable name.
- The equation is written as in equation (1) above
- The equality in the equation is written as “ $=e=$ ”
- The equation is ended with a “;”

The rest of the demand block specifies an equation for the cost of compliance, the consumer price of avocado (or the wholesale price of avocado), and the demand price indices for avocados. The specifications of these equations are in the “Final Rule” or in Peterson/Orden (2006 and 2008).

The supply block starts with the supply function of avocado

*supply of avocado
 $y_{eq}(j,t).. y(j,t) =e= (delta(j,t)*sp(j,t)**(beta(j)-1) * (sum(ta, delta(j,ta)*sp(j,ta)**beta(j)))**((1/beta(j)-1)*v(j)*(1-prd_loss))\$j1(j) + (delta(j,t)*sp(j,t)**(beta(j)-1) * (sum(ta, delta(j,ta)*sp(j,ta)**beta(j)))**((1/beta(j)-1)*v(j))\$j2(j);$

One can observe that the supply equation has two components: (a) one that is defined over the set “ $\$j1(j)$ ”, which is a subset of j that only includes California (see “*avocado_data.gms*”), and (b) another that is defined over “ $\$j2(j)$ ”, which is another subset of j that only includes Chile and Mexico. The reason for this is that it is only avocado production in California that is affected negatively if pest is able to enter

the U.S. The productivity loss factor is defined as $(1-prd_loss)$, where the prd_loss is a variable that depends on the frequency of pest outbreak. This productivity loss factor decreases with higher frequency of pest outbreak, which in turn scales down production in California. If prd_loss is one, then the productivity loss factor is zero and will wipe out completely avocado production in California.

The rest of the supply block specifies various equations in the supply of avocado, which include the factor endowment function; prd_loss ; the expected cost of control in the importing region for avocado for fruit fly and other pests (seed weevil, stem weevil, and seed moth); the supply price of avocado in period t1 and t2; the frequency of pest outbreak; and the average producer price of avocado. The specifications of these equations are also in the “Final Rule” or in Peterson/Orden (2006 and 2008).

The last block in the avocado model is the market equilibrium. This is specified as the equality between the sum of demand for avocado in the different regions (set i) in the U.S. and the supply of avocado from the different supply regions (set j). The market is in equilibrium in both periods, t1 and t2. This equation determines the value of the producer price (pp). Note that this is an endogenous variable which appears in the $cpeq(i,j,t)$. (in the demand block) and in $speq(j,t)$. (supply block). The producer price therefore does not have a specified equation. It is determined in the market equilibrium and which is written as

$$equi(j,t).. \quad y(j,t) = e = sum(i,x(i,j,t));$$

Note that there is no new variable in the market equilibrium equation. It only contains the demand and supply of avocado, each of which has its own defined equation in the model.

The next item is an dummy equation (*objec.*) and a dummy variable (*omega*). This dummy variable is set to a constant, the level of which will not affect the equilibrium solution of the model. One can try any other value, and it will not affect the equilibrium values generated by the model. The reason for this is that the problem of finding the solution of the model is set up as a non-linear optimization problem. The objective is a constant, while the constraints are the equations of the model. The solution of this optimization process is the market equilibrium. In the present case the model is solved using either of the two solvers of GAMS: MINO5 or CONOPT3. Both of these solvers are discussed in detail in GAMS reference manual.

Alternatively, one can solve the model in GAMS not as an optimization problem, but as a mixed complementarity problem (MCP) using either of two solvers: PATH or MILES. The technical discussion of MCP, PATH, and MILES are available in the website of GAMS. But this format requires pairing each equation with a unique variable in the model.

The next item in the model is to assign initial values to the variables. This step is important because it will put the search process for solution of the model within the neighborhood of the initial values. If the initial values are well calibrated, then they are the best initial values one can supply to the model so that the search process will be very short and efficient. One may assign initial values to the variable in the following way

```
* assign initial values to variables and put lower bound for prices
*endogenous
x.l(i,j,t)    = x0(i,j,t);
y.l(j,t)      = y0(j,t);
v.l(j)        = v0(j);
cp.l(i,j,t)   = cp0(i,j,t);          cp.lo(i,j,t)   = 0.0000000001;
...
```

Note that we assign values stored in variables annexed by “0” to the variables in the model. Furthermore, after “*cp.l(i,j,t) = cp0(i,j,t);*” I added a restriction “*cp.lo(i,j,t) = 0.0000000001;*”. This means that the value of *cp* (consumer price) cannot be lower than *0.0000000001*. Putting a very low price restriction will assure that prices cannot have negative values. Note that an endogenous variable is suffixed by “.l”, while its lower limit by “.lo”.⁷

The model also has a number of exogenous variables such as region income (*inc*). To set a variable as an exogenous one needs to suffix it with “.fx”. For example,

```
* exogenous
pop.fx(i)      = pop0(i) ;
inc.fx(i,t)    = inc0(i,t);
pinvest.fx(j,t) = pinvest0(j,t);
...
```

Note that similar to the endogenous variables, the exogenous variables have to be supplied with initial values as well.

The last major item in model specification is the solve statement. It is done through a number of lines in the code. That is,

```
* Model execution
model avocado_mod /all;
avocado_mod.holdfixed=1;
option decimals = 3;
options iterlim = 2000;
option nlp=minos5;
*option nlp=conopt3;
solve avocado_mod maximizing omega using nlp;
```

⁷ In GAMS, upper limit of a variable is suffixed by “.up”.

The explanation of each of the lines are given below:

- a) *model avocado_mod /all/;* - This line gives a name to the system of equations defined by GAMS command “*equations*” as “*avocado_mod*”. “*/all/*” implies that equations are considered in the system of equations.
- b) *avocado_mod.holdfixed=1;* This command counts the number of variables and equations in the model. A square model means the number of endogenous variables is the same as the number of equations. The result of this line is displayed in “*avocado_model.lst*”
- c) *option decimals = 3;* This is an option that limits the display of the results to 3 decimal places.
- d) *option nlp=minos5;*
**option nlp=conopt3;*

This is an option that allows one to solve the model using MINO5 or CONOPT 3

- e) *solve avocado_mod maximizing omega using nlp;* This command tells GAMS to solve the model (*avocado_mod*) by maximizing *omega using nlp* (non-linear program).

I inserted several lines of notes in the codes to help the user analyze the model. This completes the specification of the model.

In the next set of codes I included several lines which compute the total change in welfare in the U.S. These lines are not part of the model, but they are a set of calculations that utilize results generated by the model to compute the change in total welfare. The total change in welfare has components: equivalent variation (EV) as a measure of consumer surplus; an estimate of the change in producer surplus (PS); and the cost of pest mitigation (CM).

The idea in EV is to calculate what would be the change in income that is valued at initial prices and is welfare-equivalent to the observed change in prices. The computation of EV requires calculation of: (a) the indirect utility function at the initial price and after the price change, and (b) the cost of a unit of utility (Rutherford, 2002). The details of the computation of EV are shown in the “*avocado_model.gms.*”

In the codes, EV is computed as

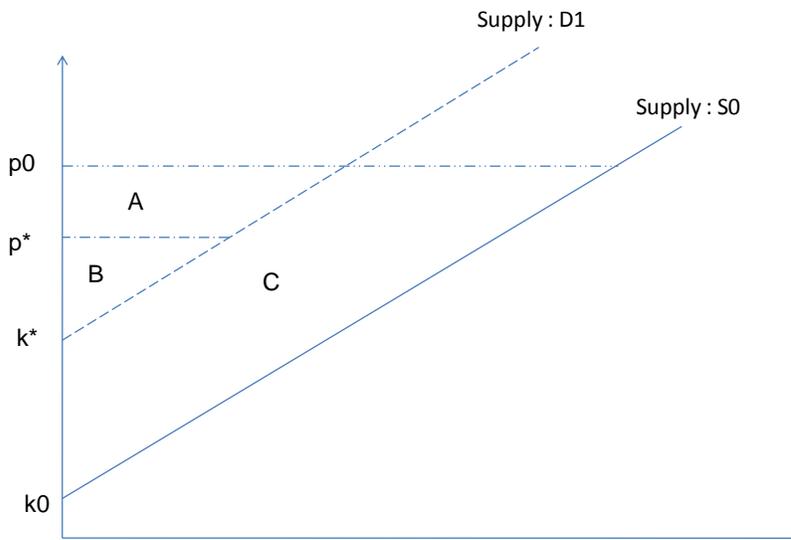
```

parameters
in_util0(i,t)  indirect utility function at the benchmark data
in_util1(i,t)  indirect utility function after the simulation
expend0(i,t)  unit cost of utility at the benchmark data
ev            equivalent variations
;
ev            = sum(t,sum(i,(expend0(i,t)*(in_util1(i,t) - in_util0(i,t)))*pop.l(i)));

```

Peterson/Orden 2006 shows how producer surplus is reduced by productivity losses (Figure 1, p 9). Their analysis is shown below. Area B is the producer surplus after the productivity loss. The sum of

the areas A, B, and C is the original producer surplus. The loss in producer surplus is the sum of areas A and C.



In the codes, the change in producer surplus for producers in California is approximated by the following equation:

```

* producer surplus
parameters
psur(j,t)
pr_surplus;
psur(j,t) = y.l(j,t)*sp.l(j,t)/2 - y0(j,t)*sp0(j,t)/2;
pr_surplus = sum(t,psur('cal',t));

```

where $y.l(i,t)$ is the quantity supply after the shock; $sp.l(i,t)$; $y0(i,t)$ is the quantity supply before the shock; and $sp0(i,t)$. Both terms are divided by 2 because they are triangles.

The control cost is calculated as

```

parameters
oth_cost
;
oth_cost = sum(t,sum(i,freq.l(i,'ff',t)))*controlc.l("ff");

```

The overall welfare change in the U.S. is calculated as

```

*net welfare
parameters

```

```
net_welfare;
net_welfare = ev + pr_surplus - oth_cost;
```

Before proceeding to the next section, one may observe that last line in `avocado_model.gms` is a GAMS “`$include avocado_results.gms`”. This command tells GAMS to call a set of codes written in the “`avocado_results.gms`”, which is outside of “`avocado_model.gms`”. These codes write the results in excel file.

GAMS File: `avocado_results.gms`

This is a set of GAMS codes that write the results generated in “`avocado_model.gms`” in excel file. The set of codes will write the results in “`avocado_results.xls`”

```
file tabs_1 /
avocado_results.xls/;
tabs_1.pc=6;
tabs_1.nd=4;
*
*-----
*Printing base values consistent with Table 4 in Peterson/Orden AJAE 2008
*-----
put tabs_1;
put /;
put '--- Base Values ---'/;
put 'Producer Prices in Season 1' put /;
loop(j, put j.te(j) put pp0(j,'t1')/) put /;
...
```

The above commands will write the initial values of the producer price $pp0(j,t)$ in period $t1$.

Excel Data File: `avocado_data.xls`

One this excel file after running the “`avocado_model.gms`”. One can see

--- Base Values ---		
Producer Prices in Season 1		
california	0.8712	
chile	0.5767	
mexico	0.54	
Producer Prices in Season 2		
california	1.1008	
chile	0.599	
mexico	0.54	
Wholesale price (weighted annual)		
california	1.6223	
chile	1.2677	
mexico	1.08	
Mexican compliance cost at the		
Growers -	0.0813	
Growers -	4.7348	
Shippers -	0.0265	
Shippers -	1.5412	

Test Run of the Model

Below are several suggested steps to run the avocado model

- Make a directory in your disk drive, e.g. “*avocado_model*”. Copy all relevant files into *_model* into this directory.
- Within this directory, create a subdirectory “*save*”.
- Open the GAMS software.
- In GAMS click “file” and then “project” and then “New project”. Go to the directory you just created (*avocado_model*) and assign a project name (e.g. *avocado_model*). Click “open” and then “yes”.
- Click “file” again and then open the file “*avocado_data.gms*”. Before running this program make sure that in the upper corner (box) of the GAMS software the following is written “*s=save\dat*”. If not, then write this statement. Run the program. It retrieves data from “*avocado_data.xls*” and saves the data in *dat.g00* in the subdirectory “*save*”. Note that this program will only have to be run **ONCE** if you do not intend to change the benchmark data.
- Click “file” and open file “*avocado_model.gms*”. Before running this program make sure that in the upper corner (box) of the GAMS software the following is written “*r=save\dat*”. If not, then write this statement. This statement recalls the data in *dat.g00*.

- To check whether the model replicates properly the benchmark data, run this program without any changes in the exogenous variables. Check the results in “avocado_results.xls”. Furthermore, in “avocado_model.gms” go to the part of the program where it says “model execution”. I inserted a small note with guidelines for you to follow to check whether the model is properly working or not. If you want to conduct a policy experiment, you need to change the values of the exogenous variables. I also inserted a small note in the program on how to conduct a policy experiment using the model.

- To conduct experiments, exogenous variables may be increased or decreased. For example, a 5% increase in per capita income in all demand regions (*a,b,c,d*) in periods t1 and t2 can be simulated using:

$$inc.fx(i,t) = inc0(i,t)*1.05;$$

A 5% increase in cost per hectare of field sanitation in Mexico can be simulated using:

$$fieldc.fx('mex') = fieldc0('mex')*1.05;$$

A Note on Avocado Supply Calibration

**A Note on Avocado Supply Calibration
(Appendix 2 in the Final Rule)**

The supply function is derived from the revenue function.

$$(1) \quad R = (\delta \cdot P_1^\beta + (1-\delta) \cdot P_2^\beta)^{1/\beta} \cdot V$$

Where R is revenue, P₁, P₂ producer prices in period t₁ and t₂, and V level of factor endowment used in avocado production.

Maximizing (1) over P₁ and P₂ yields the first order conditions which are the supply function.

$$(2) \quad \frac{\partial R}{\partial P_1} = \delta \cdot P_1^{\beta-1} \cdot (\delta \cdot P_1^\beta + (1-\delta) \cdot P_2^\beta)^{(1/\beta-1)} \cdot V = y_1$$

$$(3) \quad \frac{\partial R}{\partial P_2} = (1-\delta) \cdot P_2^{\beta-1} \cdot (\delta \cdot P_1^\beta + (1-\delta) \cdot P_2^\beta)^{(1/\beta-1)} \cdot V = y_2$$

If y₂ is not zero, take the ratio y₁/y₂ and solve for δ

$$(4) \quad \delta = \frac{\left[\frac{y_1 \cdot p_1 \cdot p_2^\beta}{y_2 \cdot p_2 \cdot p_1^\beta} \right]}{1 + \left[\frac{y_1 \cdot p_1 \cdot p_2^\beta}{y_2 \cdot p_2 \cdot p_1^\beta} \right]}$$

The value of δ can be solved given initial values of y₁, y₂, p₁, p₂, and β. Note that if y₂ is zero (as in the case of Mexico in the base), then δ is 1.

The average price index of avocado is

$$(5) \quad APP = (\delta \cdot P_1^\beta + (1-\delta) \cdot P_2^\beta)^{1/\beta}$$

V is avocado factor endowment and is linear function of the average avocado producer price index

$$(6) \quad V = k + d \cdot APP$$

Where k is the intercept, and d the slope.

One page 49 in the “Final Rule” the aggregate supply elasticity (η) is

$$(7) \quad \eta = \frac{\partial V}{\partial APP} \frac{APP}{V} = d \frac{APP}{V} \quad \text{or} \quad d = \frac{\eta \cdot V}{APP}$$

Substitute (7) into (6) yields

$$(8) \quad V = \frac{k}{1-\eta}$$

Substitute (8) into (2) (or (3)) and solve for k .⁸

$$(9) \quad k = \frac{(1-\eta) \cdot y_1}{\delta \cdot P_1^{\beta-1} \cdot (\delta \cdot P_1^\beta + (1-\delta) \cdot P_2^\beta)^{(1/\beta-1)}}$$

The value of V in (6) can be solved given all the values of the parameters and initial values of variables.

8

Note: in the derivation of k , the expected productivity loss function in Peterson/Orden ERS 2006 (equation 19, page 21) will have to be taken into account especially in the case of California where there is information on productivity loss. The expected productivity loss is $(1-(N_{1D}+N_{2D}) \cdot PL)$, where PL is the percent reduction in productivity due to infestation and N_{1D} and N_{2D} are frequency of a pest outbreak in period t_1 and t_2 . In the GAMS program this has been taken into account in the calibration.

GAMS file: “*avocado_data.gms*”

```

$title Avocado model
$offsymxref
$offsymlist
option limrow=0,limcol=0,decimals=8,solprint=on;
* Caesar B. Cororaton, Virginia Tech. USDA-APHIS Workshop on Avocado Model, June 16, 2009
$ontext
*-----
-----
This model is based on three articles:
1. Peterson and Orden, 2006. "Linking Risk and Economic Assessments in the Analysis of Plant Pest
Regulations:
The Case of U.S. Imports of Mexican Avocados"
(available at: http://www.ers.usda.gov/publications/ccr25/ccr25.pdf)

2. U.S. Department of Agriculture, Animal and Plant Health Inspection Services (USDA-APHIS, 2004).
"Economic Analysis Final Rule: Allow Fresh Hass Avocados Grown in Approved Orchards in
Approved Municipalities in Michoacan, Mexico, to be Imported Into all
States Year-Round (APHIS Docket No. 03-022-5)
(available at: http://www.aphis.usda.gov/plant\_health/spotlights/downloads/avocado\_03-022-5analysis.pdf)

3 Peterson, E., and D. Orden. 2008. "Avocado Pests and Avocado Trade."
American Journal of Agricultural Economics. 90(2): 321-335.

The model was first coded in GAMS by Everett Peterson (May 2008). This is another set of GAMS codes
of the same model done by Caesar Cororaton (December 2008)

Data used in the calibration is stored in excel file: "avocado_data.xls".
Model results are written in excel file: "avocado_results.xls".
The model is coded in: "avocado_model.gms"

Note!!!
1. Run "avocado_data.gms" to retrieve base data and assign them to variables needed in model
calibration.
Run "avocado_data.gms" ONLY ONCE, unless you change the base data in "avocado_data.xls".
2. Run "avocado_model.gms" to solve the model. Different scenarios may be solved using
"avocado_model.gms" only
without having to run "avocado_data.gms" if there are no changes in the base data in
"avocado_data.xls".
*-----
$offtext
*
set i avocado demand regions in U.S.
/ a currently approved states
b southeast region
c southwest region plus remainder of pacific
d southern california /
j avocado supply
/ cal california
ch chile
mex mexico /
j1 avocado supply from california
/ cal california /
j2 avocado supply from Chile and Mexico
/ ch chile
mex mexico /
t production period
/ t1 period1 October15 to April15
t2 period2 April16 to October14 /
pest pests
/ ff fruit flies
sw seed weevil
stw stem weevil
sm seed moth /
avcdo_pest(pest) /
sw seed weevil
stw stem weevil
sm seed moth /
*
*-----
* Retrieving data from "avocado_data.xls".
*-----
*retrieving various data (data1)
$call gdxrw.exe avocado_data.xls par=data1 rng=all:e14
parameter data1(*,i);
$gdxin avocado_data.gdx
$load data1
$gdxin
*
*retrieving wholesale prices by demand and supply region in t1 (data2)
$call gdxrw.exe avocado_data.xls par=data2 rng=a22:d26

```

```

parameter data2(i,j);
$gdxin avocado_data.gdx
$load data2
$gdxin
*
*retrieving wholesale prices by demand and supply region in t2 (data3)
$call gdxrw.exe avocado_data.xls par=data3 rng=a30:d34
parameter data3(i,j);
$gdxin avocado_data.gdx
$load data3
$gdxin
*
*retrieving percapita quantity demand in period t1 in million lbs (data4)
$call gdxrw.exe avocado_data.xls par=data4 rng=a39:d43
parameter data4(i,j);
$gdxin avocado_data.gdx
$load data4
$gdxin
*
*retrieving percapita quantity demand in period t2 in million lbs (data5)
$call gdxrw.exe avocado_data.xls par=data5 rng=a47:d51
parameter data5(i,j);
$gdxin avocado_data.gdx
$load data5
$gdxin
*
*retrieving other various data (data6)
$call gdxrw.exe avocado_data.xls par=data6 rng=a55:d68
parameter data6(*,j);
$gdxin avocado_data.gdx
$load data6
$gdxin
*
*retrieving other various data (data7)
$call gdxrw.exe avocado_data.xls par=data7 rng=a75:e83
parameter data7(*,pest);
$gdxin avocado_data.gdx
$load data7
$gdxin
*
*retrieving fixed wholesale margins for period t1 (data8)
$call gdxrw.exe avocado_data.xls par=data8 rng=a87:d91
parameter data8(i,j);
$gdxin avocado_data.gdx
$load data8
$gdxin
*
*retrieving fixed wholesale margins for period t2 (data9)
$call gdxrw.exe avocado_data.xls par=data9 rng=a95:d99
parameter data9(i,j);
$gdxin avocado_data.gdx
$load data9
$gdxin
*
*retrieving proportion of population in susceptible areas (data10)
$call gdxrw.exe avocado_data.xls par=data10 rng=a103:e107
parameter data10(i,pest);
$gdxin avocado_data.gdx
$load data10
$gdxin
*
*retrieving other various data (data11)
$call gdxrw.exe avocado_data.xls par=data11 rng=a111:e115
parameter data11(i,*);
$gdxin avocado_data.gdx
$load data11
$gdxin
*
*retrieving other various data (data12)
$call gdxrw.exe avocado_data.xls par=data12 rng=a119:e122
parameter data12(j,*);
$gdxin avocado_data.gdx
$load data12
$gdxin
*
*-----
* Defining variables to hold base values and elasticities
*-----
parameters
sig1(i)          elasticity of substitution CES between avocados and all other goods
sig2(i)          elasticity of substitution CES between avocados

```

```

beta(j)          exponent for CET revenue function
exp0(i,t)        percapita avocado expenditure in period t1 and t2
inc0(i,t)        base percapita income in period t1 and t2
pop0(i)          initial population in demand regions in millions
wp0(i,j,t)       wholesale prices by demand and supply region in period t1 and t2
q0(i,j,t)        percapita quantity demand in period t1 and t2 in million lbs
pp0(j,t)         producer price of avocado $ per lb in period t1 and t2
y0(j,t)          base supply in time period t1 and t2 million lbs
plants0(j,t)     number of processing plants in export regions with systems
inspect0(j,t)    cost_$millions_of plant inspectors in exporting regions with systems
pinvest0(j,t)    pakcing plant investment cost per pound exported
paphisf0(j,t)    fixed cost_$millions_of fees paid by pakcer to reimburse APHIS
paphisv0(j,t)    variable cost per pound of fees paid by pakcer to APHIS
pfruit0(j,t)     proportion of fruit cut in packing house
ha0(j)           hectares in approved orchards
fieldc0(j)       cost per hectare of field sanitation_$millions
pestsurv0(j)     cost per hectare of pest surveys paid by growers_$millions
gfruit0(j)       proportion of fruit cut in field
yield0(j)        yield per acre in importing region million lbs
eta(j)           aggregate supply elasticity
prob10(pest)     probability that pest infects fruit
prob20(pest)     probability the pest is not detected during packing
prob30(pest)     probability that pest survives shipment
prob40(pest)     probability that pest is not detected at port-of-entry
prob50(pest)     probability that pest is able to become established
controlc0(pest)  fixed control cost of pest p in importing region j_$millions
ploss0(pest)     percent productivity loss from infestation in importing region
pcteff0(pest)    percent of crop in importing region affected by infestation
m0(i,j,t)        fixed wholesale margins for period t1 and t2
suscept0(i,pest) proportion of population in susceptible areas
;
*
*-----
* Assigning base values to variables
*-----
sig1(i)          = data1('sig1',i);
sig2(i)          = data1('sig2',i);
beta(j)          = data6('beta',j);
exp0(i,'t1')    = data11(i,'exp1');
exp0(i,'t2')    = data11(i,'exp2');
inc0(i,'t1')    = data11(i,'inc1b');
inc0(i,'t2')    = data11(i,'inc1b');
pop0(i)          = data1('pop',i);
wp0(i,j,'t1')   = data2(i,j);
wp0(i,j,'t2')   = data3(i,j);
q0(i,j,'t1')    = data4(i,j);
q0(i,j,'t2')    = data5(i,j);
pp0(j,'t1')     = data12(j,'pp1');
pp0(j,'t2')     = data12(j,'pp2');
y0(j,'t1')      = data12(j,'yb1');
y0(j,'t2')      = data12(j,'yb2');
plants0(j,'t1') = data6('plants',j);
plants0(j,'t2') = 0;
inspect0(j,'t1') = data6('inspect',j);
inspect0(j,'t2') = 0;
pinvest0(j,'t1') = data6('pinvest',j);
pinvest0(j,'t2') = 0;
paphisf0(j,'t1') = data6('paphisf',j);
paphisf0(j,'t2') = 0;
paphisv0(j,'t1') = data6('paphisv',j);
paphisv0(j,'t2') = 0;
pfruit0(j,'t1') = data6('pfruit',j);
pfruit0(j,'t2') = 0;
ha0(j)          = data6('ha',j);
fieldc0(j)      = data6('fieldc',j);
pestsurv0(j)    = data6('pestsurv',j);
gfruit0(j)      = data6('gfruit',j);
yield0(j)       = data6('yield',j);
eta(j)          = data6('eta',j);
prob10(pest)    = data7('prob1',pest);
prob20(pest)    = data7('prob2',pest);
prob30(pest)    = data7('prob3',pest);
prob40(pest)    = data7('prob4',pest);
prob50(pest)    = data7('prob5',pest);
controlc0(pest) = data7('controlc',pest);
ploss0(pest)    = data7('ploss',pest);
pcteff0(pest)   = data7('pcteff',pest);
m0(i,j,'t1')   = data8(i,j);
m0(i,j,'t2')   = data9(i,j);
suscept0(i,pest) = data10(i,pest);

```

GAMS file: “*avocado_model.gms*”

```

$title Avocado model
$offsymxref
$offsymlist
option limrow=0,limcol=0,decimals=8,solprint=on;
*
* Caesar B. Cororaton, Virginia Tech. USDA-APHIS Workshop on Avocado Model, June 16, 2009
*
$ontext
*-----
This model is based on three articles:
1. Peterson and Orden, 2006. "Linking Risk and Economic Assessments in the Analysis of Plant Pest
Regulations:
The Case of U.S. Imports of Mexican Avocados"
(available at: http://www.ers.usda.gov/publications/ccr25/ccr25.pdf)

2. U.S. Department of Agriculture, Animal and Plant Health Inspection Services (USDA-APHIS, 2004).
"Economic Analysis Final Rule: Allow Fresh Hass Avocados Grown in Approved Orchards in
Approved Municipalities in Michoacan, Mexico, to be Imported Into all
States Year-Round (APHIS Docket No. 03-022-5)
(available at: http://www.aphis.usda.gov/plant\_health/spotlights/downloads/avocado\_03-022-5analysis.pdf)

3 Peterson, E., and D. Orden. 2008. "Avocado Pests and Avocado Trade."
American Journal of Agricultural Economics. 90(2): 321-335.

The model was first coded in GAMS by Everett Peterson (May 2008). This is another set of GAMS codes
of the same model done by Caesar Cororaton (December 2008)

Data used in the calibration is stored in excel file: "avocado_data.xls".
Model results are written in excel file: "avocado_results.xls".
The model is coded in: "avocado_model.gms"

Note!!!
1. Run "avocado_data.gms" to retrieve base data and assign them to variables needed in model
calibration.
Run "avocado_data.gms" ONLY ONCE, unless you change the base data in "avocado_data.xls".
2. Run "avocado_model.gms" to solve the model. Different scenarios may be solved using
"avocado_model.gms" only
without having to run "avocado_data.gms" if there are no changes in the base data in
"avocado_data.xls".

set i avocado demand regions in U.S.
/ a currently approved states
b southeast region
c southwest region plus remainder of pacific
d southern california /
j avocado supply
/ cal california
ch chile
mex mexico /
j1 avocado supply from california
/ cal california /
j2 avocado supply from Chile and Mexico
/ ch chile
mex mexico /
t production period
/ t1 period1 October15 to April15
t2 period2 April16 to October14 /
pest pests
/ ff fruit flies
sw seed weevil
stw stem weevil
sm seed moth /
avcdo_pest(pest) /
sw seed weevil
stw stem weevil

$offtext
*-----
alias(t,ta), (j,ja), (i,ia) ;
*-----
*
*-----
* Start of Model Calibration
*-----
*
*-----
* calculate demand and supply of avocado using base data
*-----
parameters
x0(i,j,t) demand for avocado in US in regions a_b_c_d in period t1 and t2
y0(j,t) supply of avocado in US from California_Chile_Mexico in period t1 and t2

```

```

;
*
*demand for avocado in the U.S. in region i from supply j in period t1 and t2
x0(i,j,t) = q0(i,j,t)*pop0(i);
*supply of avocado in US from California, Chile, and Mexico
y0(j,t) = sum(i,x0(i,j,t));
*
*-----
* calculate consumer price of avocado cp0
*-----
parameters
cp0(i,j,t)      consumer price of avocado
pcost0(j,t)     packer cost of compliance in exporting country
;
* for California and Chile, consumer price of avocado is equal to producer price plus wholesale margin
*
* for Mexico, consumer price of avocado is equal to producer price + wholesale margin + packer cost of
compliance which is the sum of:
* Packing plant investment cost per pound exported (pinvest)  +
* Variable cost per pound of fees paid by pakcer to APHIS (paphisv)  +
* [inspect*plants + paphisf + pfruit*(p1*y1+p2*y2)] / (y1 + y2);
* inspect = Cost ($ millions) of plant inspectors in exporting regions with systems
* plants = Number of processing plants in export regions with systems
* paphisf = Fixed cost ($ millions) of fees paid by packer to reimburse APHIS
* pfruit = Proportion of fruit cut in packing house
*
*Thus for California, consumer price of avocado is
cp0(i,'cal',t) = pp0('cal',t) + m0(i,'cal',t);
*
*Thus for Chile, consumer price of avocado is
cp0(i,'ch',t) = pp0('ch',t) + m0(i,'ch',t);
*
*Thus for Mexico, compute cost of compliance and then consumer price of avocado
* cost of compliance
pcost0('mex',t) = pinvest0('mex',t)+ paphisv0('mex',t)+
(inspect0('mex',t)*plants0('mex',t)+paphisf0('mex',t)+
pfruit0('mex',t)*(sum(ta,pp0('mex',ta)*y0('mex',ta))))
/(sum(ta,y0('mex',ta)));
*
*consumer price of avocado
cp0(i,'mex',t) = pp0('mex',t) + m0(i,'mex',t) + pcost0('mex',t);
*note cp0(i,j,t) goes into the demand for avocado
*
*-----
* calibration of demand for avocado
*-----
*On page 41 of "U.S. Department of Agriculture, Animal and Plant Health Inspection Services (USDA-
*APHIS, 2004)" we need to determine the values of: (1) alij and a2ij in the demand price equation; (2)
*blij and b2ij in the consumer demand for avocado.
*
* In Rutherford (2002) "Lecture Notes on Constant Elasticity Function",
* alij and a2ij are avocado demand shares in period t1 and t2
parameters
a(i,j,t)  avocado demand shares in region i from supply j in period t
;
a(i,j,t) = cp0(i,j,t)*x0(i,j,t);
a(i,j,t) = a(i,j,t)/sum(ja,a(i,ja,t));
*
*demand price indices
parameters
pi0(i,t)
;
pi0(i,t) = (sum(j,a(i,j,t)*cp0(i,j,t)**(1-sig2(i))))**(1/(1-sig2(i)));
* we will compute blij and b2ij so the equation for demand for avocado can be calibrated in such a way
* that it fits perfectly the equation on paper 41 of
* "U.S. Department of Agriculture, Animal and Plant Health Inspection Services (USDA-APHIS, 2004)"
*
parameters
b(i,j,t)      avocado share parameter in CES utility function
check_dem(i,j,t)  if demand is properly calibrated these are very small numbers
;
b(i,j,t)      = a(i,j,t)*(cp0(i,j,t)**(-sig2(i)))*(pi0(i,t)**(sig2(i) -
sig1(i)))*inc0(i,t)*pop0(i);
b(i,j,t)      = b(i,j,t) + x0(i,j,t) - x0(i,j,t)*pi0(i,t)**(1 - sig1(i));
b(i,j,t)$b(i,j,t) = x0(i,j,t)/b(i,j,t);
check_dem(i,j,t)$cp0(i,j,t)
= (b(i,j,t)*a(i,j,t)*(cp0(i,j,t)**(-sig2(i)))*
(pi0(i,t)**(sig2(i) - sig1(i)))*inc0(i,t)*pop0(i)/
(b(i,j,t)*(pi0(i,t)**(1 - sig1(i))) + (1 - b(i,j,t)))));
check_dem(i,j,t) = check_dem(i,j,t)- x0(i,j,t);
*note: check_dem(i,j,t) should be very small numbers

```

```

display check_dem;
* you may check the result of check_dem((i,j,t) in "avocado_model.lst".
*
*-----
* calibrate supply of avocado using benchmark data
*-----
parameters
y0(j,t)          supply of avocado in US from California_Chile_Mexico in period t1 and t2
ccost0(pest)     expected cost of control in importing region for avocado
freq0(i,pest,t)  frequency of pest outbreak in period t1 and t2
gcost0(j)        grower cost of compliance in exporting country
delta_1(j)       share parameter in CET supply of avocado factor endowment
delta(j,t)       share parameter in CET supply of avocado factor endowment
prd_loss0        productivity loss due to pest infestation
d(j)            slope of supply avocado factor endowment
v0(j)           supply of avocado factor endowment
k(j)            constant term in the slope of supply avocado factor endowment function
sp0(j,t)        supply price of avocado
av_sp0(j)        average producer price of avocado
;
* frequency of pest outbreak
freq0(i,pest,t) = prob10(pest)*prob20(pest)*prob30(pest)*prob40(pest)*
                  prob50(pest)*suscept0(i,pest)*x0(i,'mex',t)*40000;

*for fruit fly
ccost0('ff')    = (sum(t,freq0('d','ff',t)))*controlc0('ff')/(sum(t,y0('cal',t)));

*for avocado pests
ccost0(avcdo_pest) =
(controlc0(avcdo_pest)*pcteff0(avcdo_pest)*(sum(t,freq0('d',avcdo_pest,t)*y0('cal',t)))) /
(yield0('cal')*(1-ploss0(avcdo_pest))*sum(t,y0('cal',t)));

* calculate grower cost of compliance
gcost0(j)       = ((fielddc0(j) + pestsurv0(j))*ha0(j)+ gfruit0(j)*sum(t,pp0(j,t)*y0(j,t))
)/(sum(t,y0(j,t)));

*calculate supply price of avocado in period t1 and t2
sp0(j,t)        = pp0(j,t)- gcost0(j)- sum(pest,ccost0(pest));
*sp0('mex','t2') = 0;
* Note there is price for Mexican avocado in season 2 (t2) in Peterson/Orden AEJA_2008 (Table 4) even
though
* there is zero supply from Mexico in t2 Peterson/Orden ERS_2006 (Table 14, page 34)

* Note: if pest-control is zero then producer price is pp0. For California and Chile, there is no pest
* control cost but for Mexico there are pest control costs shown in Table 9 of Peterson/Orden, ERS
* 2006, page 23.

* Calibration of delta is quite involved. It involves maximization of revenue function over
* producer price in t1 and t2 subject to the level of factor endowment used in avocado production.
* For details, see my derivation ("A Note on Avocado Supply Calibration.docx").
*
* If output in second period is zero (the case of Mexico in the base value) then delta for the
* supplier is 1;
* otherwise delta is computed using equation 4 in my note "A Note on Avocado Supply Calibration.docx"
*
delta_1(j)       = 1$(y0(j,'t2') eq 0);
delta_1(j)$y0(j,'t2') =
(y0(j,'t1')*sp0(j,'t1')*sp0(j,'t2')**beta(j))/(y0(j,'t2')*sp0(j,'t2')*sp0(j,'t1')**beta(j));
delta_1(j)$y0(j,'t2') = delta_1(j)/(1+delta_1(j));
delta(j,'t1')    = delta_1(j);
delta(j,'t2')    = 1-delta_1(j);
*
*average producer price of avocado
av_sp0(j) = sum(t,delta(j,t)*sp0(j,t));
*
* calculate expected productivity loss due to pest infestation
* note: this specification is consistent with equations 20 and 21
* on page 22 of Everett/Orden (2006)
prd_loss0       = sum(pest,sum(t,freq0('d',pest,t))*pcteff0(pest)*ploss0(pest) );
*
k(j) = sum(t,delta(j,t)*sp0(j,t)**beta(j)**(1/beta(j) -1);
k(j) = (delta(j,'t1')*sp0(j,'t1')**(beta(j)-1))*k(j);
k(j) = ((y0(j,'t1')*(1-eta(j)))/(k(j)*(1-prd_loss0))$j1(j) +
((y0(j,'t1')*(1-eta(j)))/(k(j)))$j2(j);
*
* calculate initial value of avocado factor endowment, v0 (equation 6 in "my derivation");
v0(j) = k(j)/(1-eta(j));
*
* calculate the slope of the avocado factor endowment (equation 7 in "my derivation")
*
d(j) = eta(j)*v0(j)/av_sp0(j);

```

```

*
*check whether supply of avocado is properly calibrated
parameters
check_sup(j,t)  if supply is properly calibrated these are very small numbers
;
check_sup(j,t) = (sum(ta,delta(j,ta)*sp0(j,ta)**beta(j)))*(1/beta(j) -1);
check_sup(j,t) = (delta(j,t)*sp0(j,t)**(beta(j)-1)*check_sup(j,t)*v0(j)*(1-prd_loss0)-y0(j,t))$j1(j)+
                 (delta(j,t)*sp0(j,t)**(beta(j)-1)*check_sup(j,t)*v0(j)-y0(j,t))$j2(j);
*
display check_sup;
*note check_sup(j,t) should be very small number.
*you can see the result of check_sup in the in "avocado_model.lst" search "parameter check_sup"
*
*-----
* End of Model Calibration
*-----
*
* changes in parameters and probability of pest infestation used in simulation
*
*$include sim_param.gms
*
*-----
* Start of Avocado Model
*-----
*
* define variable names
variables
*endogenous
x(i,j,t)      demand for avocado
y(j,t)        supply of avocado
v(j)          supply of avocado factor endowment
prd_loss      productivity loss due to pest infestation
cp(i,j,t)     consumer price of avocado or wholesale price of avocado
pcost(j,t)    compliance cost
pi(i,t)       demand price index of avocado
pp(j,t)       producer price
ccost(pest)   expected cost of control in importing region for avocado
gcost(j)      grower cost of compliance in exporting country
sp(j,t)       supply price of avocado
freq(i,pest,t) frequency of pest outbreak in period t1 and t2
av_sp(j)      average producer price of avocado
omega         objective
*exogenous
pop(i)        initial population in demand regions in millions
inc(i,t)      base percapita income in demand regions in period t1 and t2
pinvest(j,t)  packing plant investment cost per pound exported
paphisv(j,t)  variable cost per pound of fees paid by packer to APHIS
inspect(j,t)  cost_$millions_of plant inspectors in exporting regions with systems
plants(j,t)   number of processing plants in export regions with systems
paphisf(j,t)  fixed cost_$millions_of fees paid by packer to reimburse APHIS
pfruit(j,t)   proportion of fruit cut in packing house
m(i,j,t)      fixed wholesale margins for period t1 and t2
controlc(pest) fixed control cost of pest p in importing region j_$millions
pcteff(pest)  percent of crop in importing region affected by infestation
ploss(pest)   percent productivity loss from infestation in importing region
yield(j)      yield per acre in importing region million lbs
fieldc(j)     cost per hectare of field sanitation_$millions
pestsurv(j)   cost per hectare of pest surveys paid by growers_$millions
ha(j)         hectares in approved orchards
gfruit(j)     proportion of fruit cut in field
suscept(i,pest) proportion of population in susceptible areas
prob1(pest)   probability that pest infects fruit
prob2(pest)   probability the pest is not detected during packing
prob3(pest)   probability that pest survives shipment
prob4(pest)   probability that pest is not detected at port-of-entry
prob5(pest)   probability that pest is able to become established
;
* define equation names
equations
xeq(i,j,t)    demand for avocado
pcosteq(j,t)  compliance cost
cpeq(i,j,t)   consumer price of avocado or wholesale price of avocado
pieq(i,t)     demand price index of avocado
yeq(j,t)      supply of avocado in t1 and t2
veq(j)        supply of avocado factor endowment
prd_losseq    productivity loss due to pest infestation
ccosteq1      expected cost of control in importing region for avocado
ccosteq2(avcd_pest) expected cost of control in importing region for avocado
gcosteq(j)    grower cost of compliance in exporting country
speq(j,t)     supply price of avocado
freq_eq(i,pest,t) frequency of pest outbreak in period t1 and t2

```

```

av_speq(j)          average producer price of avocoda
equi(j,t)           market equilibrium
objec              objective
;
*
* model specification
*
* ----- D E M A N D -----
*demand for avocado
xeq(i,j,t)$cp0(i,j,t).. x(i,j,t) =e= (b(i,j,t)*a(i,j,t)*(cp(i,j,t)**(-sig2(i)))*
                                     (pi(i,t)**(sig2(i)-sig1(i)))*inc(i,t)*pop(i)/
                                     (b(i,j,t)*(pi(i,t)**(1-sig1(i))) + (1-b(i,j,t))));

* cost of compliance
pcosteq(j,t)..      pcost(j,t) =e= pinvest(j,t)+ paphisv(j,t)+
                                     (inspect(j,t)*plants(j,t)+paphisf(j,t)+pfruit(j,t)*(sum(ta,pp(j,ta)*y(j,ta)))/
                                     / (sum(ta,y(j,ta))));

* consumer price of avocado or wholesale price of avocado
cpeq(i,j,t)..       cp(i,j,t) =e= pp(j,t) + m(i,j,t) + pcost(j,t);

* demand price indices of avocado
pieq(i,t)..         pi(i,t) =e= (sum(j,a(i,j,t)*cp(i,j,t)**(1-sig2(i))))**(1/(1-sig2(i)));

* ----- S U P P L Y -----
*supply of avocado
yeq(j,t)..          y(j,t) =e= (delta(j,t)*sp(j,t)**(beta(j)-1)*
                               (sum(ta, delta(j,ta)*sp(j,ta)**beta(j))**(1/beta(j)-1)*v(j)*(1 -
                               prd_loss))$j1(j) +
                               (delta(j,t)*sp(j,t)**(beta(j)-1)*
                               (sum(ta, delta(j,ta)*sp(j,ta)**beta(j))**(1/beta(j)-1)*v(j))$j2(j));

*supply avocado factor endowment
veq(j)..            v(j) =e= k(j) + d(j)*av_sp(j);

* productivity loss due to pest infestation
prd_losseq..       prd_loss =e= sum(pest,sum(t,freq('d',pest,t))*pcteff(pest)*ploss(pest)) ;

* expected cost of control in importing region for avocado
* fruit fly
ccosteq1..         ccost('ff') =e= (sum(t,freq('d','ff',t)))*controlc('ff')/(sum(t,y('cal',t)));

* avocado pests
ccosteq2(avcdo_pest).. ccost(avcdo_pest) =e= (controlc(avcdo_pest)*pcteff(avcdo_pest)*
                                               (sum(t,freq('d',avcdo_pest,t)*y('cal',t)))) /
                                               (yield('cal')*(1-ploss(avcdo_pest))*sum(t,y('cal',t)));

* grower cost of compliance
gcosteq(j)..       gcost(j) =e= ((fieldc(j) + pestsurv(j))*ha(j)+ gfruit(j)*sum(t,pp(j,t)*y(j,t)))/
                               / (sum(t,y(j,t)));

*supply price of avocado in period t1 and t2
*equations 21 and 20 (page 22) Everett/Orden (2006) - data of pest-control cost in Table 9, page 23
speq(j,t)..        sp(j,t) =e= pp(j,t)- gcost(j)- sum(pest,ccost(pest));

* frequency of pest outbreak
freq_eq(i,pest,t).. freq(i,pest,t) =e= prob1(pest)*prob2(pest)*prob3(pest)*prob4(pest)*
                                     prob5(pest)*suscept(i,pest)*x(i,'mex',t)*40000;

*average producer price of avocado
av_speq(j)..       av_sp(j) =e= sum(t,delta(j,t)*sp(j,t));

* ----- M A R K E T E Q U I L I B R I U M -----
equi(j,t)..        y(j,t) =e= sum(i,x(i,j,t));

* dummy objective
* note: changing the value of omega will not change the solution since it is just a dummy variable
objec..           omega =e= 10;
*
* assign initial values to variables and put lower bound for prices
*endogenous
x.l(i,j,t)        = x0(i,j,t);
y.l(j,t)          = y0(j,t);
v.l(j)            = v0(j);
cp.l(i,j,t)       = cp0(i,j,t);          cp.lo(i,j,t)      = 0.0000000001;
pcost.l(j,t)      = pcost0(j,t);
pi.l(i,t)         = pi0(i,t);           pi.lo(i,t)        = 0.0000000001;
pp.l(j,t)         = pp0(j,t);           pp.lo(j,t)        = 0.0000000001;
ccost.l(pest)     = ccost0(pest);
gcost.l(j)        = gcost0(j);
sp.l(j,t)         = sp0(j,t);           sp.lo(j,t)        = 0.0000000001;

```

```

freq.l(i,pest,t)      = freq0(i,pest,t);
av_sp.l(j)            = av_sp0(j);
prd_loss.l           = prd_loss0;
av_sp.lo(j)          = 0.0000000001;
*
* exogenous
pop.fx(i)             = pop0(i) ;
inc.fx(i,t)           = inc0(i,t);
pinvest.fx(j,t)      = pinvest0(j,t);
paphisv.fx(j,t)      = paphisv0(j,t);
inspect.fx(j,t)       = inspect0(j,t);
plants.fx(j,t)        = plants0(j,t);
paphisf.fx(j,t)       = paphisf0(j,t);
pfruit.fx(j,t)        = pfruit0(j,t);
m.fx(i,j,t)           = m0(i,j,t);
controlc.fx(pest)    = controlc0(pest);
pcteff.fx(pest)      = pcteff0(pest);
ploss.fx(pest)       = ploss0(pest);
yield.fx(j)           = yield0(j);
fieldc.fx(j)          = fieldc0(j);
pestsurv.fx(j)       = pestsurv0(j);
ha.fx(j)              = ha0(j);
gfruit.fx(j)          = gfruit0(j);
suscept.fx(i,pest)   = suscept0(i,pest);
probl.fx(pest)       = probl0(pest) ;
prob2.fx(pest)       = prob20(pest) ;
prob3.fx(pest)       = prob30(pest) ;
prob4.fx(pest)       = prob40(pest) ;
prob5.fx(pest)       = prob50(pest) ;

*
* to conduct experiments, exogenous variables may be increased or decreased. For example, a 5%
increase in
* percapita income in all demand regions (a,b,c,d) in periods t1 and t2 can be simulated using:
*
* inc.fx(i,t)         = inc0(i,t)*1.05;
*
* A 5% increase in cost per hectare of field sanitation in Mexico can be simulated using:
*
* fieldc.fx('mex')   = fieldc0('mex')*1.05;
*
*-----
* Model execution
*
* Note, if the model is calibrated properly, solving the model using CONOPT 3 solver without any
change in any
* of the exogenous variables should yield the solution in only 4 iterations. If the model
* is solved in MINOS5 solver, model should solve in 3 iterations. Otherwise, if the model is solved
* beyond these number of iterations, it would indicate that the model is not perfectly/properly
calibrated.
* However, solving the model with changes in any of the exogenous variables may take more iterations
* before the model yields the solution. In any case, the solution of the model should
* be the same irrespective of the solver used.
*
* The model can also be solved using MCP (mixed complementarity problem) using either PATH or MILES.
* Solving the model using MCP need not require a dummy objective function and variable. However,
* on needs to match each equation with each endogenous variable. The solution should not differ from
* the one solved through MINOS5 or CONOPT3.
*
* To check whether the model is square and to determine the number of equations and the number of
* endogenous variables in the model one needs to solve the model (with or without policy shocks).
* Solving the model will generate a list file, in the avocado model it will be
* "avocado_model.lst". Open this file and search for "block" (note: one can search this word using the
* icon which contains a "flash light
* with the letter A.")
*-----
model avocado_mod /all/;
avocado_mod.holdfixed=1;
option decimals = 3;
options iterlim = 2000;
option nlp=minos5;
*option nlp=conopt3;
solve avocado_mod maximizing omega using nlp;
*
*-----
* Calculate welfare effects
*-----
*
* consumer surplus: equivalent variations
*
* prices of all other goods are constant in the partial equilibrium model, and any change in the
* avocado price index represents a change in the relative prices.

```

```

*
* See Rutherford 2002 for the specification of indirect utility function from a CES utility function
*
parameters
in_util0(i,t)      indirect utility function at the benchmark data
in_util1(i,t)      indirect utility function after the simulation
expend0(i,t)       unit cost of utility at the benchmark data
ev                 equivalent variations
;
*
*
* note: cp0 is the consumer price of avocado at the benchmark data while cp is the consumer price of
avocado
* after simulation
in_util0(i,t)$cp.l(i,'cal',t) = inc0(i,t)*(b(i,'cal',t)**sig1(i)*cp0(i,'cal',t)**(1-sig1(i)) +
      b(i,'cal',t)**(1-sig1(i)))**(1/(sig1(i)-1));
*
in_util1(i,t)$cp.l(i,'cal',t) = inc.l(i,t)*(b(i,'cal',t)**sig1(i)*cp.l(i,'cal',t)**(1-sig1(i)) +
      b(i,'cal',t)**(1-sig1(i)))**(1/(sig1(i)-1));
*
* expenditure function
expend0(i,t)$cp.l(i,'cal',t) = (b(i,'cal',t)**sig1(i)*cp0(i,'cal',t)**(1-sig1(i)) +
      b(i,'cal',t)**(1-sig1(i)))**(1/(1-sig1(i)));
*
ev                 = sum(t,sum(i,(expend0(i,t)*(in_util1(i,t) - in_util0(i,t))*pop.l(i))));
*
* producer surplus
parameters
psur(j,t)
pr_surplus;
psur(j,t)          = y.l(j,t)*sp.l(j,t)/2 - y0(j,t)*sp0(j,t)/2;
pr_surplus         = sum(t,psur('cal',t));
*
parameters
oth_cost;
*
oth_cost = sum(t,sum(i,freq.l(i,'ff',t)))*controlc.l("ff");
*
*net welfare
parameters
net_welfare;
net_welfare = ev + pr_surplus - oth_cost;
*
*-----
*Write the results in excel: "avocado_results_original.xls"
* The Gams codes that write the results are in "avocado_results_original.gms"
* Note:
* 1. Go to the directory where the model is stored. Click "avocado_results_original.xls". Excel with
* prompt you a warning and please click "yes". The excel file contains values of the endogenous
* variables, welfare results (consumer surplus, producer surplus, and other cost). The excel file
* contains values of the exogenous variables also. You may save this as an excel file with a different
* filename.
*
* 2. Another important note is that before solving, the excel file "avocado_results_original.xls"
* should be closed.
* Otherwise, the GAMS program cannot write on this file with new solution if it is opened.
*-----
$include avocado_results.gms
*Note:
* When you are done with your experiment, go to the directory where the model is stored. Double click
"clean.bat". This will delete all
* the garbage (scratch files) that GAMS generates while solving the model.

display freq.l, y.l, delta;

```

GAMS file: “*avocado_results.gms*”

```

file tabs_1 /
avocado_results.xls/;
tabs_1.pc=6;
tabs_1.nd=4;
*
*-----
*Printing base values consistent with Table 4 in Peterson/Orden AJAE 2008
*-----
put tabs_1;
put /;
put '--- Base Values ---'/;
put 'Producer Prices in Season 1' put /;
loop(j, put j.te(j) put pp0(j,'t1') /) put /;

put 'Producer Prices in Season 2' put /;
loop(j, put j.te(j) put pp0(j,'t2') /) put /;

*-----
*computation of wholesale price (weighted annual average)
* Convert percapita quantity demand into quantity demand in each region/season
parameters
quan0(i,j,t)      quantity demand
tquan0(j,t)       total quantity demand
;
quan0(i,j,t)      = q0(i,j,t)*pop0(i);
tquan0(j,t)       = sum(i,quan0(i,j,t));

* weights and weighted annual average wholesale price
parameters
w_q0(i,j,t)       weights
w_wp0(j)          weighted annual average wholesale price
;
w_q0(i,j,t)       = (quan0(i,j,t)/sum(ta,tquan0(j,ta)));
w_wp0(j)          = sum(t,sum(i,w_q0(i,j,t)*wp0(i,j,t)));

put 'Wholesale price (weighted annual average)' put /;
loop(j, put j.te(j) put w_wp0(j) /) put /;
*-----

*--Mexican compliance cost at the base (growers)---
parameters
gcost0_1(j)       Mexican compliance cost at base in million dollars (growers)
;
gcost0_1(j) = gcost0(j)*sum(t,sum(i,quan0(i,'mex',t)));

put 'Mexican compliance cost at the base' put /;
put 'Growers - dollar per pound' put gcost0('mex') put /;
put 'Growers - million dollars' put gcost0_1('mex') put /;

*--Mexican compliance cost at the base (shippers)---
parameters
pcost0_1(j,t)     Mexican compliance cost at base in million dollars (shippers)
;
pcost0_1(j,t) = pcost0(j,t)*sum(ta,sum(i,quan0(i,'mex',ta)));
put 'Shippers - dollar per pound' put pcost0('mex','t1') put /;
put 'Shippers - million dollars' put pcost0_1('mex','t1') put /;
*-----
put ///;
*$ontext
parameters
tmp_a
tmp_b
tmp_c
tmp_d
shp_cost_mex1
shp_cost_mex2

pfruit0_1(j)
inspect0_1(j)
plants0_1(j)
paphisf0_1(j)
pinvest0_1(j)
paphisv0_1(j)
;

pfruit0_1(j) = pfruit0(j,'t1');
inspect0_1(j) = inspect0(j,'t1');
plants0_1(j) = plants0(j,'t1');
paphisf0_1(j) = paphisf0(j,'t1');
pinvest0_1(j) = pinvest0(j,'t1');
paphisv0_1(j) = paphisv0(j,'t1');

```

```

tmp_a = pfruit0_1('mex')*sum(t,sum(i,quan0(i,'mex',t)*pp0('mex',t))) ;
tmp_b = inspect0_1('mex')*plants0_1('mex');
tmp_c = tmp_a + tmp_b + paphisf0_1('mex') ;
tmp_d = tmp_c/sum(t,sum(i,quan0(i,'mex',t)));

shp_cost_mex1 = tmp_d + pinvest0_1('mex') +paphisv0_1('mex') ;
shp_cost_mex2 = shp_cost_mex1 * sum(t,sum(i,quan0(i,'mex',t)));

put 'Mexican compliance cost at the base' put /;
put ' Shippers dollar per pound' put shp_cost_mex1 put /;
put ' Shippers million dollars' put shp_cost_mex2 put /;

put tabs_1;
put /;
put '--- values of endogenous variables ---'/;
put /;
put 'avocado demand in t1' loop(j, put j.te(j) ) put /;
loop(i, put i.te(i) loop(j, put x.l(i,j,'t1')) put / );
put 'avocado supply in t1' put y.l('cal','t1') put y.l('ch','t1') put y.l('mex','t1') put /;
*
put /;
put tabs_1;
put 'avocado demand in t2' loop(j, put j.te(j) ) put /;
loop(i, put i.te(i) loop(j, put x.l(i,j,'t2')) put / );
put 'avocado supply in t2' put y.l('cal','t2') put y.l('ch','t2') put y.l('mex','t2') put /;
*
put /;
put tabs_1;
put 'wholesale prices in t1' loop(j, put j.te(j) ) put /;
put 'currently approved states' put cp.l('a','cal','t1') put cp.l('a','ch','t1') put
cp.l('a','mex','t1') put /;
put 'southeast region' put cp.l('b','cal','t1') put cp.l('b','ch','t1') put /;
put 'southwest region plus remainder of pacific'
put cp.l('c','cal','t1') put cp.l('c','ch','t1') put /;
put 'southern california' put cp.l('d','cal','t1') put cp.l('d','ch','t1') put /;
*
put /;
put tabs_1;
put 'wholesale prices in t2' loop(j, put j.te(j) ) put /;
put 'currently approved states' put cp.l('a','cal','t2') put cp.l('a','ch','t2') put /;
put 'southeast region' put cp.l('b','cal','t2') put cp.l('b','ch','t2') put /;
put 'southwest region plus remainder of pacific'
put cp.l('c','cal','t2') put cp.l('c','ch','t2') put /;
put 'southern california' put cp.l('d','cal','t2') put cp.l('d','ch','t2') put /;
*
put /;
put tabs_1;
put 'producer prices' loop(j, put j.te(j) ) put /;
put 'period t1' loop(j, put pp.l(j,'t1')) put /;
put 'period t2' put pp.l('cal','t2') put pp.l('ch','t2') put /;

put /
put ' ' loop(j, put j.te(j) ) put /;
put 'factor endowment' loop(j, put v.l(j)) put /;
put 'grower cost' loop(j, put gcost.l(j)) put /;
put 'average producer price' loop(j, put av_sp.l(j)) put /;
put 'productivity loss' put prd_loss.l put /;

put /;
put tabs_1;
put 'pcost' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put pcost.l(j,t)) put / );

put /;
put tabs_1;
put 'demand price index' put 't1' put 't2' put /;
loop(i, put i.te(i) loop(t, put pi.l(i,t)) put / );

put /;
put tabs_1;
put 'supply price' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put sp.l(j,t)) put / );

put /;
put 'frequency of pest outbreak in t1' loop(pest, put pest.te(pest) ) put /;
loop(i, put i.te(i) loop(pest, put freq.l(i,pest,'t1')) put / );

put /;
put 'frequency of pest outbreak in t2' loop(pest, put pest.te(pest) ) put /;
loop(i, put i.te(i) loop(pest, put freq.l(i,pest,'t2')) put / );

```

```

put ///;
put '--- welfare results ---'/;
put 'equivalent variations' put ev put/;
put 'producer surplus' put pr_surplus put/;
put 'other costs' put oth_cost put/;
put 'net welfare' put net_welfare put/;

put ///;
put '--- values of exogenous variables ---'/;
put /;
put tabs_1;
put 'population' put /;
loop(i, put i.te(i) put pop.l(i) put / );

put /;
put tabs_1;
put 'percapita income' put 't1' put 't2' put /;
loop(i, put i.te(i) loop(t, put inc.l(i,t)) put / );

put /;
put tabs_1;
put 'pinvest' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put pinvest.l(j,t)) put / );

put /;
put tabs_1;
put 'paphisv' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put paphisv.l(j,t)) put / );

put /;
put tabs_1;
put 'inspect' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put inspect.l(j,t)) put / );

put /;
put tabs_1;
put 'plants' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put plants.l(j,t)) put / );

put /;
put tabs_1;
put 'paphisf' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put paphisf.l(j,t)) put / );

put /;
put tabs_1;
put 'pfruit' put 't1' put 't2' put /;
loop(j, put j.te(j) loop(t, put pfruit.l(j,t)) put / );

put /;
put tabs_1;
put 'wholesale margins in t1' loop(j, put j.te(j) ) put /;
loop(i, put i.te(i) loop(j, put m.l(i,j,'t1')) put / );

put /;
put tabs_1;
put 'wholesale margins in t2' loop(j, put j.te(j) ) put /;
loop(i, put i.te(i) loop(j, put m.l(i,j,'t2')) put / );

put /;
put tabs_1;
put ' ' loop(pest, put pest.te(pest) ) put /;
put 'control cost of pest' loop(pest, put controlc.l(pest)) put /;
put 'pcteff' loop(pest, put pcteff.l(pest)) put /;
put 'ploss' loop(pest, put ploss.l(pest)) put /;
put 'probl' loop(pest, put probl.l(pest)) put /;
put 'prob2' loop(pest, put prob2.l(pest)) put /;
put 'prob3' loop(pest, put prob3.l(pest)) put /;
put 'prob4' loop(pest, put prob4.l(pest)) put /;
put 'prob5' loop(pest, put prob5.l(pest)) put /;

put /;
put tabs_1;
put ' ' loop(j, put j.te(j) ) put /;
put 'yield' loop(j, put yield.l(j)) put /;
put 'fieldc' loop(j, put fieldc.l(j)) put /;
put 'pestsurv' loop(j, put pestsurv.l(j)) put /;
put 'ha' loop(j, put ha.l(j)) put /;
put 'gfruit' loop(j, put gfruit.l(j)) put /;

```

```
put /;
put tabs_1;
put 'susceptible areas ' loop(pest, put pest.te(pest) ) put /;
loop(i, put i.te(i) loop(pest, put suscept.l(i,pest) ) put / );

*$offtext
```

References

Lecture Notes on Constant Elasticity Functions

Thomas F. Rutherford
University of Colorado

November, 2002

1 CES Utility

In many economic textbooks the constant-elasticity-of-substitution (CES) utility function is defined as:

$$U(x, y) = (\alpha x^\rho + (1 - \alpha)y^\rho)^{1/\rho}$$

It is a tedious but straight-forward application of Lagrangian calculus to demonstrate that the associated demand functions are:

$$x(p_x, p_y, M) = \left(\frac{\alpha}{p_x}\right)^\sigma \frac{M}{\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma}}$$

and

$$y(p_x, p_y, M) = \left(\frac{1 - \alpha}{p_y}\right)^\sigma \frac{M}{\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma}}.$$

The corresponding indirect utility function has is:

$$V(p_x, p_y, M) = M (\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma})^{\frac{1}{\sigma-1}}$$

Note that $U(x, y)$ is linearly homogeneous:

$$U(\lambda x, \lambda y) = \lambda U(x, y)$$

This is a convenient cardinalization of utility, because percentage changes in U are equivalent to percentage Hicksian equivalent variations in income. Because U is linearly homogeneous, V is homogeneous of degree one in M :

$$V(p_x, p_y, \lambda M) = \lambda V(p_x, p_y, M)$$

and V is homogeneous of degree -1 in p :

$$V(\lambda p_x, \lambda p_y, M) = \frac{V(p_x, p_y, M)}{\lambda}.$$

Furthermore, linear homogeneity permits us to form an exact price index corresponding to the cost of a unit of utility:

$$e(p_x, p_y) = (\alpha^\sigma p_x^{1-\sigma} + (1 - \alpha)^\sigma p_y^{1-\sigma})^{\frac{1}{1-\sigma}}$$

The indirect utility function can then be written:

$$V(p_x, p_y, M) = \frac{M}{e(p_x, p_y)}$$

Conceptually, this equation states that the utility which can be realized with income M and prices p_x and p_y is equal to the income level divided by the unit cost of utility. The key idea is that when the underlying is linearly homogeneous, utility can be represented like any other good in the economy. Put another way, without loss of generality, we can think of each consumer demanding only one good.

2 CES Technology

In the representation of technology, we have a set of relationships which are directly analogous to the CES utility function. These relationships are based on the cost and compensated demand functions. If we have a CES production function of the form:

$$y(K, L) = \phi (\beta K^\rho + (1 - \beta)L^\rho)^{1/\rho}$$

the unit cost function then has the form:

$$c(p_K, p_L) = \frac{1}{\phi} (\beta^\sigma p_K^{1-\sigma} + (1 - \beta)^\sigma p_L^{1-\sigma})^{\frac{1}{1-\sigma}}$$

and associated demand functions are:

$$K(p_K, p_L, y) = \left(\frac{y}{\phi}\right) \left(\frac{\beta \phi c(p_K, p_L)}{p_K}\right)^\sigma$$

and

$$L(p_K, p_L, y) = \left(\frac{y}{\phi}\right) \left(\frac{(1 - \beta) \phi c(p_K, p_L)}{p_L}\right)^\sigma.$$

In most large-scale applied general equilibrium models, we have many function parameters to specify with relatively few observations. The conventional approach is to *calibrate* functional parameters to a single benchmark equilibrium. For example, if we have benchmark estimates for output, labor, capital inputs and factor prices, we calibrate function coefficients by inverting the factor demand functions:¹

$$\theta = \frac{\bar{p}_K \bar{K}}{\bar{p}_K \bar{K} + \bar{p}_L \bar{L}}, \quad \rho = \frac{\sigma - 1}{\sigma}, \quad \beta = \frac{\bar{p}_K \bar{K}^{1/\sigma}}{\bar{p}_K \bar{K}^{1/\sigma} + \bar{p}_L \bar{L}^{1/\sigma}}$$

and

$$\phi = \bar{y} [\beta \bar{K}^\rho + (1 - \beta) \bar{L}^\rho]^{-1/\rho}$$

¹I wish to thank Professor Olivier de La Grandville for correcting an error in the expression for β in an earlier version of these notes.

Exercises

1. Mikki once lived in Boulder and spent 30% of her income for rent, 10% for food and 60% for skiing. She then moved to Georgetown where rent and food prices are identical to Boulder. In Georgetown, however, Mikki discovered that the quality-adjusted cost of skiing was ten-times the cost of skiing in Boulder. She adopted a lifestyle in which she spend only 30% of her income on skiing. Suppose that her preferences are characterized by a CES utility function. What values of α and σ describe Mikki's utility function?
2. What fraction of Mikki's income does she spend on rent in Georgetown?
3. How much larger would Mikki's income need to be to compensate for the higher cost of skiing such that she would be indifferent between living in Boulder or Georgetown.

3 The Calibrated Share Form

Calibration formulae for CES functions are messy and difficult to remember. Consequently, the specification of function coefficients is complicated and error-prone. For applied work using calibrated functions, it is much easier to use the “calibrated share form” of the CES function. In the calibrated form, the cost and demand functions explicitly incorporate

- benchmark factor *demands*
- benchmark factor *prices*
- the elasticity of substitution
- benchmark *cost*
- benchmark *output*
- benchmark *value shares*

In this form, the production function is written:

$$y = \bar{y} \left[\theta \left(\frac{K}{\bar{K}} \right)^\rho + (1 - \theta) \left(\frac{L}{\bar{L}} \right)^\rho \right]^{1/\rho}$$

The only *calibrated* parameter, θ , represents the value share of capital at the benchmark point, i.e.

$$\theta = \frac{\bar{p}_K \bar{K}}{\bar{p}_K \bar{K} + \bar{p}_L \bar{L}}$$

The corresponding cost functions in the calibrated form is written:

$$c(p_K, p_L) = \bar{c} \left[\theta \left(\frac{p_K}{\bar{p}_K} \right)^{1-\sigma} + (1 - \theta) \left(\frac{p_L}{\bar{p}_L} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}$$

where

$$\bar{c} = \bar{p}_L \bar{L} + \bar{p}_K \bar{K}$$

and the compensated demand functions are:

$$K(p_K, p_L, y) = \bar{K} \frac{y}{\bar{y}} \left(\frac{\bar{p}_K c}{p_K \bar{c}} \right)^\sigma$$

and

$$L(p_K, p_L, y) = \bar{L} \frac{y}{\bar{y}} \left(\frac{c \bar{p}_L}{\bar{c} p_L} \right)^\sigma$$

Normalizing the benchmark utility index to unity, the utility function in calibrated share form is written:

$$U(x, y) = \left[\theta \left(\frac{x}{\bar{x}} \right)^\rho + (1 - \theta) \left(\frac{y}{\bar{y}} \right)^\rho \right]^{1/\rho}$$

The unit expenditure function can be written:

$$e(p_x, p_y) = \left[\theta \left(\frac{p_x}{\bar{p}_x} \right)^{1-\sigma} + (1 - \theta) \left(\frac{p_y}{\bar{p}_y} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}},$$

the indirect utility function is:

$$V(p_x, p_y, M) = \frac{M}{\bar{M}e(p_x, p_y)},$$

and the demand functions are:

$$x(p_x, p_y, M) = \bar{x} V(p_x, p_y, M) \left(\frac{e(p_x, p_y)\bar{p}_x}{p_x} \right)^\sigma$$

and

$$y(p_x, p_y, M) = \bar{y} V(p_x, p_y, M) \left(\frac{e(p_x, p_y)\bar{p}_y}{p_y} \right)^\sigma.$$

The calibrated form extends directly to the n -factor case. An n -factor production function is written:

$$y = f(x) = \bar{y} \left[\sum_i \theta_i \left(\frac{x_i}{\bar{x}_i} \right)^\rho \right]^{1/\rho}$$

and has unit cost function:

$$C(p) = \bar{C} \left[\sum_i \theta_i \left(\frac{p_i}{\bar{p}_i} \right)^{1-\sigma} \right]^{\frac{1}{1-\sigma}}$$

and compensated factor demands:

$$x_i = \bar{x}_i \frac{y}{\bar{y}} \left(\frac{C}{\bar{C}} \frac{\bar{p}_i}{p_i} \right)^\sigma$$

Exercises

1. Show that given a generic CES utility function:

$$U(x, y) = (\alpha^\rho + (1 - \alpha)y^\rho)^{1/\rho}$$

can be represented in share form using:

$$\bar{x} = 1, \bar{y} = 1, \bar{p}_x = t\alpha, \bar{p}_y = t(1 - \alpha), \bar{M} = t.$$

for any value of $t > 0$.

2. Consider the utility function defined:

$$U(x, y) = (x - a)^\alpha (y - b)^{1-\alpha}$$

A benchmark demand point with both prices equal and demand for y equal to twice the demand for x . Find values for which are consistent with optimal choice at the benchmark. Select these parameters so that the income elasticity of demand for x at the benchmark point equals 1.1.

3. Consider the utility function:

$$U(x, L) = (\alpha L^\rho + (1 - \alpha)x^\rho)^{1/\rho}$$

which is maximized subject to the budget constraint:

$$p_x x = M + w(\bar{L} - L)$$

in which M is interpreted as non-wage income, w is the market wage rate. Assume a benchmark equilibrium in which prices for x and L are equal, demands for x and L are equal, and non-wage income equals one-half of expenditure on x . Find values of α and ρ consistent with these choices and for which the price elasticity of labor supply equals 0.2.

4. Consider a consumer with CES preferences over two goods. A price change makes the benchmark consumption bundle unaffordable, yet the consumer is indifferent. Graph the choice. Find an equation which determines the elasticity of substitution as a function of the benchmark value shares. (You can write down the equation, but it cannot be solved in closed form.)
5. Consider a model with three commodities, x , y and z . Preferences are CES. Benchmark demands and prices are equal for all goods. Find demands for x , y and z for a doubling in the price of x as a function of the elasticity of substitution.
6. Consider the same model in the immediately preceding question, except assume that preferences are instead given by:

$$U(x, y, z) = (\beta \min(x, y)^\rho + (1 - \beta)z^\rho)^{1/\rho}$$

Determine β from the benchmark, and find demands for x , y and z if the price of x doubles.

7. Consider a two-period model in which consumers maximizes the discounted present value of utility:

$$U(c_1, c_2) = \frac{c_1^{1-\theta}}{1-\theta} + \beta \frac{c_2^{1-\theta}}{1-\theta}$$

subject to the budget constraint:

$$c_1 + \frac{c_2}{1+r} = 1 + \frac{1}{1+r}$$

in which β is the discount factor, θ is the intertemporal elasticity parameter and r is the given interest rate.

Use the calibrated share formulation to show (on inspection) that the equivalent variation of a change in the interest rate from \bar{r} to r is equal to:

$$EV = M/\bar{M} - 1 = \left(\frac{2+r}{2+\bar{r}}\right) \left(\frac{1+\bar{r}}{1+r}\right) \left(\frac{1+\beta^{1/\theta}(1+r)^{1/\theta-1}}{1+\beta^{1/\theta}(1+\bar{r})^{1/\theta-1}}\right)^{\theta/(1-\theta)} - 1$$

4 Flexibility and Non-Separable CES

We let π_i denote the user price of the i th input, and let $x_i(\pi)$ be the cost-minimizing demand for the i th input. The reference price and quantities are $\bar{\pi}_i$ and \bar{x}_i . One can think of set i as $\{K, L, E, M\}$ but the methods we employ may be applied to any number of inputs. Define the reference cost, and reference value share for i th input by \bar{C} and θ_i , where

$$\bar{C} \equiv \sum_i \bar{\pi}_i \bar{x}_i$$

and

$$\theta_i \equiv \frac{\pi_i \bar{x}_i}{\bar{C}}$$

The single-level constant elasticity of substitution cost function in calibrated form is written:

$$C(\pi) = \bar{C} \left(\sum_i \theta_i \left(\frac{\pi_i}{\bar{\pi}_i} \right)^{1-\sigma} \right)^{\frac{1}{1-\sigma}}$$

Compensated demands may be obtained from Shephard's lemma:

$$x_i(\pi) = \frac{\partial C}{\partial \pi_i} \equiv C_i = \bar{x}_i \left(\frac{C(\pi)}{\bar{C}} \frac{\bar{\pi}_i}{\pi_i} \right)^\sigma$$

Cross-price Allen-Uzawa elasticities of substitution (AUES) are defined as:

$$\sigma_{ij} \equiv \frac{C_{ij} C}{C_i C_j}$$

where

$$C_{ij} \equiv \frac{\partial^2 C(\pi)}{\partial \pi_i \partial \pi_j} = \frac{\partial x_i}{\partial \pi_j} = \frac{\partial x_j}{\partial \pi_i}$$

For single-level CES functions:

$$\sigma_{ij} = \sigma \quad \forall i \neq j$$

The CES cost function exhibits homogeneity of degree one, hence Euler's condition applies to the second derivatives of the cost function (the Slutsky matrix):

$$\sum_j C_{ij}(\pi) \pi_j = 0$$

or, equivalently:

$$\sum_j \sigma_{ij} \theta_j = 0$$

The Euler condition provides a simple formula for the diagonal AUES values:

$$\sigma_{ii} = \frac{-\sum_{j \neq i} \sigma_{ij} \theta_j}{\theta_i}$$

As an aside, note that convexity of the cost function implies that all minors of order 1 are negative, i.e. $\sigma_{ii} < 0 \quad \forall i$. Hence, there must be *at least one* positive off-diagonal element in each row of the AUES or Slutsky matrices. When there are only two factors, then the off-diagonals must be negative. When there are three factors, then only one pair of negative goods may be complements.

Let:

k index a second-level nest

s_{ik} denote the fraction of good i inputs assigned to the k th nest

ω_k denote the benchmark value share of total cost which enters through the k th nest

γ denote the top-level elasticity of substitution

σ^k denote the elasticity of substitution in the k th aggregate

$p_k(\pi)$ denote the price index associated with aggregate k , normalized to equal unity in the benchmark, i.e.:

$$p_k(\pi) = \left[\sum_i \frac{s_{ik} \theta_i \pi_i}{\omega_k \bar{\pi}_i} \right]^{1-\sigma^k}$$

The two-level nested, nonseparable constant-elasticity-of-substitution (NNCES) cost function is then defined as:

$$C(\pi) = \bar{C} \left(\sum_k \omega_k p_k(\pi)^{1-\gamma} \right)^{\frac{1}{1-\gamma}}$$

Demand indices for second-level aggregates are needed to express demand functions in a compact form. Let $z_k(\pi)$ denote the demand index for aggregate k , normalized to unity in the benchmark; i.e.

$$z_k(\pi) = \left(\frac{C(\pi)}{\bar{C}} \frac{1}{p_k(\pi)} \right)^\gamma$$

Compensated demand functions are obtained by differentiating $C(\pi)$. In this derivative, one term arise for each nest in which the commodity enters, so:

$$x_i(\pi) = \bar{x}_i \sum_k z_k(\pi) \left(\frac{p_k(\pi) \bar{\pi}_i}{\pi_i} \right)^{\sigma^k} = \bar{x}_i \sum_k \left(\frac{C(\pi)}{\bar{C}} \frac{1}{p_k(\pi)} \right)^\gamma \left(\frac{p_k(\pi) \bar{\pi}_i}{\pi_i} \right)^{\sigma^k}$$

Simple differentiation shows that benchmark cross-elasticities of substitution have the form:

$$\sigma_{ij} = \gamma + \sum_k \frac{(\sigma^k - \gamma) s_{ik} s_{jk}}{\omega_k}$$

Given the benchmark value shares θ_i and the benchmark cross-price elasticities of substitution, σ_{ij} , we can solve for values of s_{ik} , ω_k , σ^k and γ . A closed-form solution of the calibration problem is not always practical, so it is convenient to compute these parameters using a constrained nonlinear programming algorithm, CONOPT, which is available through GAMS, the same programming environment in which the equilibrium model is specified. Perroni and Rutherford [1995] prove that calibration of the NNCES form is possible for arbitrary dimensions whenever the given Slutsky matrix is negative semi-definite. The two-level ($N \times N$) function is flexible for three inputs; and although we have not proven that it is flexible for 4 inputs, the only difficulties we have encountered have resulted from indefinite calibration data points.

Two GAMS programs are listed below. The first illustrates two analytic calibrations of the three-factor cost function. The second illustrates the use of nonlinear programming to calibrate a four-factor cost function. (See Rutherford [1999] for an introduction to MPSGE.)

```

$title Two nonseparable CES calibrations for a 3-input cost function.

*      Model-specific data defined here:

SET    i Production input aggregates / A,B,C /; ALIAS (i,j);

PARAMETER

    theta(i)    Benchmark value shares /A 0.2, B 0.5, C 0.3/

    aues(i,j)   Benchmark cross-elasticities (off-diagonals) /
                A.B      2
                A.C     -0.05
                B.C      0.5 /;

*      Use an analytic calibration of the three-factor CES cost
*      function:

ABORT$(CARD(i) <> 3) "Error: not a three-factor model!";

*      Fill in off-diagonals:

aues(i,j)$aues(j,i) = aues(j,i);

*      Verify that the cross elasticities are symmetric:

ABORT$SUM((i,j), ABS(aues(i,j)-aues(j,i))) " AUES values non-symmetric?";

*      Check that all value shares are positive:

ABORT$(SMIN(i, theta(i)) <= 0) " Zero value shares are not valid:",THETA;

*      Fill in the elasticity matrices:

aues(i,i) = 0; aues(i,i) = -SUM(j, aues(i,j)*theta(j))/theta(i); DISPLAY aues;

SET    n        Potential nesting /N1*N3/
    k(n)       Nesting aggregates used in the model
    i1(i)      Good fully assigned to first nest
    i2(i)      Good fully assigned to second nest
    i3(i)      Good split between nests;

SCALAR assigned /0/;

PARAMETER

    esub(*,*)   Alternative calibrated elasticities
    shr(*,i,n)  Alternative calibrated shares
    sigma(n)    Second level elasticities
    s(i,n)      Nesting assignments (in model)
    gamma       Top level elasticity (in model);

*      First the Leontief structure:

esub("LTF","GAMMA") = SMAX((i,j), aues(i,j));
esub("LTF",n) = 0;
LOOP((i,j)$((aues(i,j) = esub("LTF","GAMMA"))*(NOT assigned)),

```

```

        i1(i) = YES;
        i2(j) = YES;
        assigned = 1;
    );

i3(i) = YES$((NOT i1(i))*(NOT i2(i)));
DISPLAY i1,i2,i3;
LOOP((i1,i2,i3),
    shr("LTF",i1,"N1") = 1;
    shr("LTF",i2,"N2") = 1;
    shr("LTF",i3,"N1") = theta(i1)*(1-aues(i1,i3)/aues(i1,i2)) /
        ( 1 - theta(i3) * (1-aues(i1,i3)/aues(i1,i2)) );
    shr("LTF",i3,"N2") = theta(i2)*(1-aues(i2,i3)/aues(i1,i2)) /
        ( 1 - theta(i3) * (1-aues(i2,i3)/aues(i1,i2)) );
    shr("LTF",i3,"N3") = 1 - shr("LTF",i3,"N1") - shr("LTF",i3,"N2");
);
ABORT$(SMIN((i,n), shr("LTF",i,n)) < 0) "Benchmark AUES is indefinite.";

*      Now specify the two-level CES function:

esub("CES","GAMMA") = SMAX((i,j), aues(i,j));
ESUB("CES","N1") = 0;
LOOP((i1,i2,i3),
    shr("CES",i1,"N1") = 1;
    shr("CES",i2,"N2") = 1;
    esub("CES","N2") = (aues(i1,i2)*aues(i1,i3)-aues(i2,i3)*aues(i1,i1)) /
        (aues(i1,i3)-aues(i1,i1));
    shr("CES",i3,"N1") =
        (aues(i1,i2)-aues(i1,i3)) / (aues(i1,i2)-aues(i1,i1));
    shr("CES",i3,"N2") = 1 - shr("CES",i3,"N1");
);
ABORT$(SMIN(n, esub("CES",n)) < 0) "Benchmark AUES is indefinite?";
ABORT$(SMIN((i,n), shr("CES",i,n)) < 0) "Benchmark AUES is indefinite?";

PARAMETER      price(i)      Price indices used to verify calibration,
                aueschk(*,i,j) Check of benchmark AUES values;

price(i) = 1;

$ontext

$MODEL:CHKCALIB

$SECTORS:
    Y          ! PRODUCTION FUNCTION
    D(i)

$COMMODITIES:
    PY        ! PRODUCTION FUNCTION OUTPUT
    P(i)      ! FACTORS OF PRODUCTION
    PFX       ! AGGREGATE PRICE LEVEL

$CONSUMERS:
    RA

$PROD:Y s:gamma k.tl:sigma(k)

```

```

O:PY          Q:1
I:P(i)#(k)    Q:(theta(i)*s(i,k))    k.TL:

$PROD:D(i)
O:P(i)  Q:theta(i)
I:PFX   Q:(theta(i)*price(i))

$DEMAND:RA
D:PFX
E:PFX   Q:2
E:PY    Q:-1

$OFFTEXT
$SYSINCLUDE mpsgeset CHKCALIB

SCALAR delta /1.E-5/;
SET      function /ltf, ces/;
alias (i,ii);

LOOP(function,
  k(n) = YES$SUM(i, shr(function,i,n));
  gamma = esub(function,"GAMMA");
  sigma(k) = esub(function,k);
  s(i,k) = shr(function,i,k);
  loop(ii,
    price(j) = 1; price(ii) = 1 + delta;
$INCLUDE CHKCALIB.GEN
    SOLVE CHKCALIB USING MCP;
    aueschk(function,j,ii) = (D.L(j)-1) / (delta*theta(ii));
));
aueschk(function,i,j) = aueschk(function,i,j) - aues(i,j);
DISPLAY aueschk;

*      Evaluate the demand functions:

$LIBINCLUDE plot

SET pr Alternative price levels /pr0*pr10/;

PARAMETER
  demand(function,i,pr)  Demand functions
  dplot(pr,function)     Demand function comparison

loop(ii,
  LOOP(function,
    k(n) = YES$SUM(i, shr(function,i,n));
    gamma = esub(function,"GAMMA");
    sigma(k) = esub(function,k);
    s(i,k) = shr(function,i,k);
    LOOP(pr,
      price(j) = 1;
      price(ii) = 0.2 * ORD(pr);
$INCLUDE CHKCALIB.GEN
      SOLVE CHKCALIB USING MCP;
      demand(function,ii,pr) = D.L(ii);
      dplot(pr,function) = D.L(ii);
    );
  );

```

```
);  
  
*      Display the comparisons:  
  
$LIBINCLUDE PLOT dplot  
);  
  
DISPLAY demand;
```

```

$title Numerical calibration of Nested CES from KLEM elasticities
set i Production input aggregates / K, L, E, M/; ALIAS (i,j);

* Model-specific data defined here:

PARAMETER
  theta(i) Benchmark value shares /K 0.2, L 0.4, E 0.05, M 0.35/

  aues(i,j) Benchmark cross-elasticities (off-diagonals) /
              K.L 1
              K.E -0.1
              K.M 0
              L.E 0.3
              L.M 0
              E.M 0.1 /;

SCALAR epsilon Minimum value share tolerance /0.001/;

* Fill in off-diagonals:
aues(i,j)$aues(j,i) = aues(j,i);

* Verify that the cross elasticities are symmetric:
ABORT$(SUM((i,j), ABS(aues(i,j)-aues(j,i))) > 0) " AUES values non-symmetric?";

* Check that all value shares are positive:
ABORT$(SMIN(i, theta(i)) <= 0) " Zero value shares are not valid:",theta;

* Fill in the elasticity matrices:
aues(i,i) = 0; aues(i,i) = -SUM(j, aues(i,j)*theta(j))/theta(i); DISPLAY aues;

* Define variables and equations for NNCES calibration:

SET n Nests within the two-level NNCES function /N1*N4/,
    k(n) Nests which are in use;

VARIABLES
  S(i,n) Fraction of good I which enters through nest N,
  SHARE(n) Value share of nest N,
  SIGMA(n) Elasticity of substitution within nest N,
  GAMMA Elasticity of substitution at the top level,
  OBJ Objective function;

POSITIVE VARIABLES S, SHARE, SIGMA, GAMMA;

EQUATIONS
  SDEF(i) Nest shares must sum to one,
  TDEF(n) Nest share in total cost,
  ELAST(i,j) Consistency with given AUES values,
  OBJDEF Maximize concentration;

ELAST(i,j)$ (ORD(i) > ORD(j))..

```

```

        aues(i,j) =E=  GAMMA +
                SUM(k, (SIGMA(k)-GAMMA)*S(i,k)*S(j,k)/SHARE(k));
TDEF(k)..      SHARE(k) =E= SUM(i, theta(i) * S(i,k));
SDEF(i)..      SUM(n, S(i,n)) =E= 1;

*      Maximize concentration at the same time keeping the elasticities
*      to be reasonable:

OBJDEF..      OBJ =E= SUM((i,k),S(i,k)*S(i,k))
                - SQR(GAMMA) - SUM(k, SQR(sigma(k)));

MODEL  CESCALIB /ELAST, TDEF, SDEF, OBJDEF/;

*      Apply some bounds to avoid divide by zero:

SHARE.LO(n) = epsilon;

SCALAR  solved  Flag for having solved the calibration problem /0/
        minshr  Minimum share in candidate calibration;

SET     tries   Counter on the number of attempted calibrations /T1*T10/;

OPTION SEED=0;

LOOP(tries$(NOT solved),

*      Initialize the set of active nests and the bounds:

        k(n) = YES;
        S.LO(i,n) = 0;          S.UP(i,n) = 1;
        SHARE.LO(n) = epsilon;  SHARE.UP(n) = 1;
        SIGMA.LO(n) = 0;        SIGMA.UP(n) = +INF;

*      Install a starting point:

        SHARE.L(k) = MAX(UNIFORM(0,1), epsilon);
        S.L(i,k)   = UNIFORM(0,1);
        GAMMA.L    = UNIFORM(0,1);
        SIGMA.L(k) = UNIFORM(0,1);
        SDEF.M(i)  = 0; TDEF.M(k)   = 0; ELAST.M(i,j) = 0;

        SOLVE CESCALIB USING NLP MAXIMIZING OBJ;

        solved = 1$(CESCALIB.MODELSTAT LE 2);
        IF (solved,
            minshr = SMIN(k, SHARE.L(k)) - epsilon;
            IF (minshr EQ 0,
                k(n)$(SHARE.L(n) = epsilon) = NO;
                S.FX(i,n)$(not k(n)) = 0;
                SHARE.FX(n)$(not k(n)) = 0;
                SIGMA.FX(n)$(not k(n)) = 0;

```

```
        DISPLAY "Recalibrating with the following nests:",k;
        SOLVE CESCALIB USING NLP MAXIMIZING OBJ;
        IF (cescalib.modelstat gt 2, solved = 0;);
        minshr = SMIN(k, SHARE.L(k)) - epsilon;
        IF (minshr=0, solved = 0;);
    );
);
);
IF (solved, DISPLAY "Function calibrated:",GAMMA.L,SIGMA.L,SHARE.L,S.L;
ELSE      DISPLAY "Function calibration fails!";
);
```

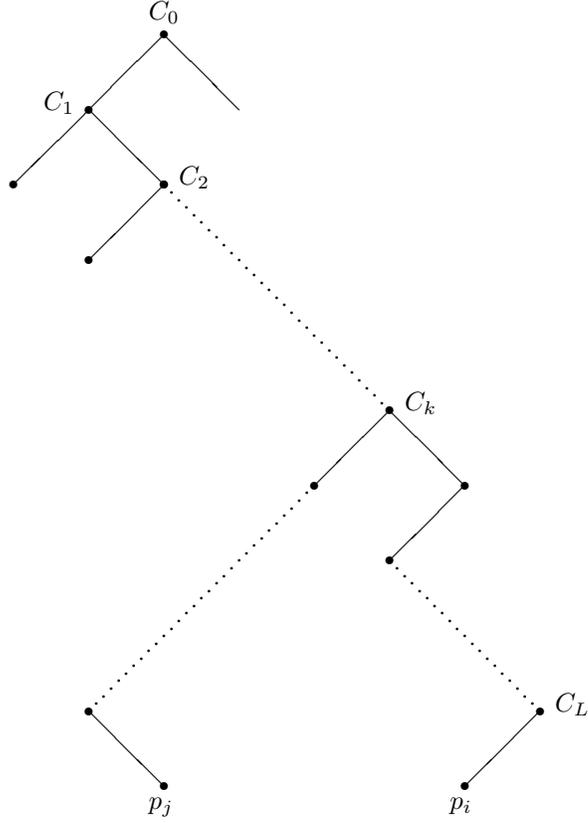


Figure 1: A Multi-level Nested CES Cost Function

5 Price Elasticities in Nested CES Functions

Suppose that we have a nested cost function of arbitrary depth and complexity. The prices for goods i and j , p_i and p_j are arguments of $C(p)$. Assume that the cost function $C(p)$ is *nested*. In the simplest two level case, we would have:

$$C(p) = \left(\sum_k \theta_k c_k(p)^{1-\sigma_0} \right)^{1/1-\sigma_0}$$

in which:

$$c_k(p) = \left(\sum_{i \in I_k} \alpha_{ik} p_i^{1-\sigma_k} \right)^{1/1-\sigma_k}$$

where I_k indicates the set of commodities entering nest k .

In a more general case, we could have cost aggregates as arguments to other cost aggregates. The figure shown below displays a graph of the nested CES cost function in which we number the nested cost functions which lead from the top level to nest containing good i as C_0 (top level), C_1, \dots, C_L .

If we construct the cost function from a calibrated benchmark in which input prices and total cost are unity, we can scale the benchmark values of the subaggregate cost functions as

unity and express the demand for good i as:

$$x_i = \bar{x}_i \left(\frac{C_L}{p_i} \right)^{\sigma_L} \left(\frac{C_{L-1}}{C_L} \right)^{\sigma_{L-1}} \dots \left(\frac{C_0}{C_1} \right)^{\sigma_0} = \bar{x}_i p_i^{-\sigma_L} C_0^{\sigma_0} \prod_{n=1}^L C_n^{\sigma_n - \sigma_{n-1}}$$

5.1 The Own-Price Elasticity of Demand

By Shephard's lemma the derivative of C_n with respect to p_i equals the demand for good i per unit of aggregate n . Recalling that all prices are scaled to unity, the benchmark "quantity" of aggregate n equals the sum of the inputs which enter directly or indirectly into that cost function:

$$\bar{X}_n = \sum_{j \in I_n} \bar{x}_j$$

and

$$\left. \frac{\partial C_n}{\partial p_i} \right|_{p=1} = \begin{cases} 0 & i \notin I_n \\ \frac{\bar{x}_i}{\bar{X}_n} & i \in I_n \end{cases}$$

We then can compute the compensated own-price elasticity of demand for good i :

$$\eta_i \equiv \left. \frac{\partial x_i}{\partial p_i} \right|_{p=1} = -\sigma_L + \bar{x}_i \left(\sigma_0 + \sum_{n=1}^L \frac{\sigma_n - \sigma_{n-1}}{\bar{X}_n} \right)$$

5.2 The Cross-Price Elasticity of Demand

When we evaluate the elasticity of demand for i with respect to a change in the price of good j , we can let k denote the deepest price aggregate which contains both p_i and p_j . (See Figure 1).

The cross derivative can then be computed using the demand function for x_i , taking into account the impact of p_j on C_k, C_{k-1}, \dots, C_0 :

$$\left. \frac{\partial x_i}{\partial p_j} \right|_{p=1} = \bar{x}_i \left[\sigma_0 \frac{\partial C_0}{\partial p_j} + \sum_{n=1}^k (\sigma_n - \sigma_{n-1}) \frac{\partial C_n}{\partial p_j} \right]$$

One means of representing the dependence of x_i on p_j is with the Allen-Uzawa elasticity-of-substitution which:

$$\sigma_{ij} \equiv \frac{\partial x_i}{\partial p_j} \frac{C_0}{x_i x_j} = \sigma_0 + \sum_{n=1}^k \frac{\sigma_n - \sigma_{n-1}}{X_n}$$

As a logical check on this elasticity, consider two special cases:

1. $\sigma_n = \sigma_0 \quad \forall n$

This then leads to single level CES, implying a constant cross elasticity of substitution between all input pairs.

2. $\sigma_n = 0 \quad \forall n < k$

We then have Leontief demand for aggregate k , implying that the elasticity of substitution between i and j is given by

$$\sigma_{ij} = \sigma_k$$

N.B. The cross elasticity between i and j is independent of the subnest elasticity for all nests $n > k$.

6 Benchmarking Supply Functions

This section describes how calibrate the fixed factor input for a constant returns to scale CES technology and obtain an arbitrary price elasticity of supply at a reference point. For concreteness, consider output as a function of labor and capital inputs. Consider the labor input to be variable and the capital input to be fixed. We then have a CES cost function which in equilibrium defines the price of output:

$$p = c(r, w)$$

in which w is the exogenous wage rate and r is the residual return to the sector's fixed factor. Because this factor is fixed, by Shepard's lemma we have the following relationship between output, the supply of the fixed factor and the return to the fixed factor:

$$y \frac{\partial c(r, w)}{\partial r} = \bar{R}$$

If we use the calibrated CES cost function of the form:

$$c(r, w) = (\theta r^{1-\sigma} + (1-\theta)w^{1-\sigma})^{\frac{1}{1-\sigma}}$$

then the calibration problem consists of finding a values for θ and σ for which:

$$\frac{\partial y}{\partial(p/w)} \frac{(p/w)}{y} = \eta$$

at the benchmark point.

Note that we are free to choose units of the specific factor such that its benchmark price is unity. Hence, when we calibrate the share parameter, we are also determining the supply of the fixed factor:

$$\bar{R} = \theta \bar{y}$$

in which we scale the benchmark price of output to unity.

If the relative price of output and the variable factor depart from their benchmark values, the supply constraint for sector-specific can be inverted to obtain an explicit expression for the return:

$$r = p \left(\frac{\theta y}{\bar{R}} \right)^{1/\sigma}$$

where we have substituted the equilibrium price for the cost function. Substituting back into the cost function, we have

$$p^{1-\sigma} = \theta p^{1-\sigma} \left(\frac{\theta y}{\bar{R}} \right)^{\frac{1-\sigma}{\sigma}} + (1-\theta)w^{1-\sigma}$$

or

$$y = \bar{R} \theta^{\frac{1}{\sigma-1}} \left[1 - (1-\theta) \left(\frac{w}{p} \right)^{1-\sigma} \right]^{\frac{\sigma}{1-\sigma}}$$

Differentiating this expression with respect the relative price of output, and setting all prices equal to unit, we have:

$$\eta = \frac{\sigma(1-\theta)}{\theta}$$

This equation can be used in a variety of ways to calibrate the supply function. One approach would be to choose the value share of the fixed factor θ to match the base year profits, and then assign the elasticity according to:

$$\sigma = \frac{\theta \eta}{(1-\theta)}$$

Alternatively, one choose to use a Cobb-Douglas function and set the specific factor value share accordingly:

$$\theta = \frac{1}{1 + \eta}$$

7 Calibration of Short- and Long-Run Elasticities

In a dynamic model it may be helpful to introduce two notions of the elasticity of supply: short-run and long-run. A simple way to introduce this distinction into a numerical model is to work with a three factor production function:

$$y = f(L, K, R)$$

where L is labor, a production factor which is variable in both the short and long run, K is capital, a quasi-fixed production factor which is variable in the long run but fixed in the short run, and R is a sector-specific resource which is fixed in both the short and long run.

We can write the long-run production function as:

$$y = \bar{y} \left[\theta_L \left(\frac{L}{\bar{L}} \right)^\rho + \theta_K \left(\frac{K}{\bar{K}} \right)^\rho + (1 - \theta_L - \theta_K) \right]^{1/\rho}$$

The R input does not appear in the calibrated production function because we have assumed that $R = \bar{R}$. In the short-run model, we have:

$$y = \bar{y} \left[\theta_L \left(\frac{L}{\bar{L}} \right)^\rho + (1 - \theta_L) \right]^{1/\rho}$$

If the short-run elasticity of supply is given by η_S , and the labor value share (θ_L) is given in the benchmark data, we can then calibrate the elasticity of substitution to match these inputs:

$$\sigma = \eta_S \frac{1 - \theta_L}{\theta_L}$$

Let us assume that while the labor value share is observed (employment statistics are commonly available for many sectors and regions), but the allocation of the remaining value added is not known with certainty. We can take advantage of this uncertainty to calibrate the long-run supply response by choosing the capital value as:

$$\theta_K = \frac{\eta_L}{\sigma + \eta_L} - \theta_L$$

given the value of σ previously calibrated. (It can be shown that when $\eta_L > \eta_S$, we find that $\theta_K < 1 - \theta_L$.)

8 The GEMTAP Final Demand System

Following Ballard, Fullerton, Shoven and Whalley (BFSW), we consider a representative agent whose utility is based upon current consumption, future consumption and current leisure. Changes in *future consumption*; in this static framework are associated with changes in the level of savings. There are three prices which jointly determine the price index for future consumption. These are:

P_I the composite price index for investment goods

P_K the composite rental price for capital services

P_C the composite price of current consumption.

All of these prices equal unity in the benchmark equilibrium.

Capital income in each future year finances future consumption, which is expected to cost the same as in the current period, P_C (static expectations). The consumer demand for savings therefore depends not only on P_I , but also on P_K and P_C , namely:

$$P_S = \frac{P_I P_C}{P_K}$$

The price index for savings is unity in the benchmark period. In a counter-factual equilibrium, however, we would expect generally that

$$P_S \neq P_I$$

. When these price indices are not equal, there is a *virtual tax payment*; associated with savings demand.

Following BFSW, we adopt a nested CES function to represent preferences. In this function, at the top level demand for savings (future consumption) trades off with a second CES aggregate of leisure and current consumption. These preferences can be summarized with the following expenditure function:

$$P_U = [\alpha P_H^{1-\sigma_S} + (1-\alpha) P_S^{1-\sigma_S}]^{\frac{1}{1-\sigma_S}}$$

Preferences are homothetic, so we have defined P_U as a linearly homogeneous cost index for a unit of utility. We conveniently scale this price index to equal unity in the benchmark. In this definition, α is the benchmark value share for current consumption (goods and leisure). P_H is a composite price for current consumption defined as:

$$P_H = [\beta P_\ell^{1-\sigma_L} + (1-\beta) P_C^{1-\sigma_L}]^{\frac{1}{1-\sigma_L}}$$

in which β is the benchmark value share for leisure within current consumption.

Demand functions are:

$$S = S_0 \left(\frac{P_U}{P_F} \right)^{\sigma_S} \frac{I}{I_0 P_U},$$

$$C = C_0 \left(\frac{P_H}{P_C} \right)^{\sigma_L} \left(\frac{P_U}{P_H} \right)^{\sigma_S} \frac{I}{I_0 P_U},$$

and

$$\ell = \ell_0 \left(\frac{P_H}{P_L} \right)^{\sigma_L} \left(\frac{P_U}{P_H} \right)^{\sigma_S} \frac{I}{I_0 P_U}.$$

Demands are written here in terms of their benchmark values (S_0 , C_0 and ℓ_0) and current and benchmark income (I and I_0).

There are four components in income. The first is the value of labor endowment (E), defined inclusive of leisure. The second is the value of capital endowment (K). The third is all other income (M). The fourth is the value of *virtual tax revenue*; associated with differences between the shadow price of savings and the cost of investment.

$$I = P_L E + P_K K + M + (P_S - P_I) S$$

The following parameter values are specified exogenously:

$\zeta = 1.75$ is the ratio of labor endowment to labor supply, $\zeta \equiv \frac{E}{L_0}$, where L_0 is the benchmark labor supply. Labor supply and ζ also define benchmark leisure demand, $\ell_0 = L_0(\zeta - 1)$.

$\xi = 0.15$ is the uncompensated elasticity of labor supply with respect to the net of tax wage, i.e.

$$\xi = \frac{\partial L}{\partial P_L} \frac{P_L}{L} = \frac{\partial(E - \ell)}{\partial P_L} \frac{P_L}{L} = -\frac{\partial \ell}{\partial P_L} \frac{P_L}{L}$$

$\eta = 0.4$ is the elasticity of savings with respect to the return to capital:

$$\eta \equiv \frac{\partial S}{\partial P_K} \frac{P_K}{S}$$

Shephard's lemma applied at benchmark prices provides the following identities which are helpful in deriving expressions for η and ξ :

$$\frac{\partial P_U}{\partial P_H} = \alpha, \quad \frac{\partial P_U}{\partial P_S} = 1 - \alpha, \quad \frac{\partial P_H}{\partial P_L} = \beta, \quad \frac{\partial P_H}{\partial P_C} = 1 - \beta$$

It is then a relatively routine application of the chain rule to show that:

$$\xi = (\zeta - 1) \left[\sigma_L + \beta(\sigma_S - \sigma_L) - \alpha\beta(\sigma_S - 1) - \frac{E}{I_0} \right]$$

and

$$\eta = \sigma_S \alpha + \frac{K}{I_0}$$

The expression for η does not involve σ_L , so we may first solve for σ_S and use this value in determining σ_L :

$$\sigma_S = \frac{\eta - \frac{K}{I_0}}{\alpha}$$

and

$$\alpha_L = \frac{\frac{\xi}{\zeta - 1} - \sigma_S \beta (1 - \alpha) - \alpha \beta + \frac{E}{I_0}}{1 - \beta}$$

\$TITLE A Maquette Illustrating Labor Supply / Savings Demand Calibration

* Exogenous elasticity:

SCALAR XI UNCOMPENSATED ELASTICITY OF LABOR SUPPLY /0.15/,
ETA ELASTICITY OF SAVINGS WRT RATE OF RETURN /0.40/,
ZETA RATIO OF LABOR ENDOWMENT TO LABOR SUPPLY /1.75/;

* Benchmark data:

SCALAR CO CONSUMPTION /2.998845E+2/,
SO SAVINGS /70.02698974/,
LSO LABOR SUPPLY / 2.317271E+2/,
KO CAPITAL INCOME /93.46960577/,
PLO MARGINAL WAGE /0.60000000/;

* Calibrated parameters:

SCALAR ELO LABOR ENDOWMENT
LO LEISURE DEMAND
MO NON-WAGE INCOME
I EXTENDED GROSS INCOME
ETAMIN SMALLEST PERMISSIBLE VALUE FOR ETA,
XIMIN SMALLEST PERMISSIBLE VALUE FOR XI,
ALPHA CURRENT CONSUMPTION VALUE SHARE
BETA LEISURE VALUE SHARE IN CURRENT CONSUMPTION
SIGMA_L ELASTICITY OF SUBSTITUTION WITHIN CURRENT CONSUMPTION
SIGMA_S ELASTICITY OF SUBSTITUTION - SAVINGS VS CURRENT CONSUMPTION
TS SAVINGS PRICE ADJUSTMENT;

LSO = LSO * PLO;

ELO = ZETA * LSO;

LO = ELO - LSO;

MO = CO + SO - LSO - KO;

I = LO + CO + SO;

BETA = LO / (CO + LO);

ALPHA = (LO + CO) / I;

SIGMA_S = (ETA - KO / I) / ALPHA;

ETAMIN = KO / I;

ABORT\$(SIGMA_S LT 0) " Error: cannot calibrate SIGMA_S", ETAMIN;

SIGMA_L = (XI*(LSO/LO)-SIGMA_S*BETA*(1-ALPHA)-ALPHA*BETA+ELO/I)/(1-BETA);

XIMIN = -(LO/LSO) * (- SIGMA_S * BETA * (1-ALPHA) - ALPHA*BETA + ELO/I);

ABORT\$(SIGMA_L LT 0) " Error: cannot calibrate SIGMA_L", XIMIN;

DISPLAY "Calibrated elasticities:", SIGMA_S, SIGMA_L;

\$ONTEXT

\$MODEL:CHKCAL

\$COMMODITIES:

PL

PK

PC

PS

```

$SECTORS:
    Y
    S

$CONSUMERS:
    RA

$PROD:Y
    O:PC    Q:(KO+LSO-SO)
    I:PL    Q:(LSO-SO)
    I:PK    Q:KO

$PROD:S
    O:PS    A:RA T:TS
    I:PL

$DEMAND:RA  s:SIGMA_S a:SIGMA_L
    E:PC    Q:MO
    E:PL    Q:ELO
    E:PK    Q:KO
    D:PS    Q:SO
    D:PC    Q:CO a:
    D:PL    Q:L0 a:

$OFFTEXT
$SYSINCLUDE mpsgeset CHKCAL

S.L = SO;
TS = 0;

*      VERIFY THE BENCHMARK:

CHKCAL.ITERLIM = 0;
$INCLUDE CHKCAL.GEN
SOLVE CHKCAL USING MCP;

PL.L = 1.001;

CHKCAL.ITERLIM = 0;
$INCLUDE CHKCAL.GEN
SOLVE CHKCAL USING MCP;

*      Compute induced changes in labor supply using the labor market
*      "marginal", PL.M. This marginal returns the net excess supply of
*      labor at the given prices. We started from a balanced benchmark,
*      with no change in labor demand (the iteration limit was zero).
*      Hence, PL.M returns the magnitude of the change in labor supply.
*      We multiply by the benchmark wage (1) and divide by the benchmark
*      labor supply (LSO) to produce a finite difference approximation
*      of the elasticity:

DISPLAY "CALIBRATION CHECK -- THE FOLLOWING VALUES SHOULD BE IDENTICAL:", XI;
XI = (PL.M / 0.001) * (1 / LSO);
DISPLAY XI;

PL.L = 1.0;

```

```

*          CHECK THE ELASTICITY OF SAVINGS WRT RENTAL RATE OF CAPITAL:

PK.L = 1.001;
PS.L = 1 / 1.001;
TS   = 1 / 1.001 - 1;

CHKCAL.ITERLIM = 0;

*          Compute elasticity of savings with respect to the rental rate of
*          capital. This requires some recursion in order to account for the
*          effect of changes in savings on effective income. When PK increases,
*          PS declines -- there is an effective "subsidy" for saving, paid from
*          consumer income. In order to obtain a difference approximation for
*          the elasticity of savings response, we need to make sure the virtual
*          tax payments are properly handled. In the MPSGE model, this means
*          that the level value for S must be adjusted so that it exactly equals
*          the savings. We do this recursively:

SET ITER /IT1*IT5/;

PS.M = 1;
LOOP(ITER$(ABS(PS.M) GT 1.0E-8),

$INCLUDE CHKCAL.GEN
      SOLVE CHKCAL USING MCP;
      S.L = S.L - PS.M;
);

DISPLAY "CALIBRATION CHECK -- THE FOLLOWING VALUES SHOULD BE IDENTICAL:", ETA;
ETA = ((S.L - SO) / 0.001) * (1 / SO);
DISPLAY ETA;

```

9 Calibrating the Demand for Bequest

Bequests play an important part in the life-cycle income and expenditures of certain classes of households (see Kotlikoff and Summers [1981]). Bequests are important for analyzing the intergenerational impact of economic policy because they provide a mechanism through which impacts on older generations are transmitted to younger generations.

A simple means of introducing bequests to the OLG exchange model involves the introduction of a separate “bequest market” for each generation. A representative of generation g then demands two *goods*, \hat{u}_g and b_g . The first of these is the composite of lifetime consumption as defined by the utility function:

$$\begin{aligned} \max_{c_{g,t}} u_g(c_{g,t}) &= \sum_{t=g}^{g+N} \left(\frac{1}{1+\rho} \right)^{t-g} \frac{c_{g,t}^{1-\theta}}{1-\theta} \\ \text{s.t.} \quad \sum_{t=g}^{g+N} p_t c_{g,t} &\leq \sum_{t=g}^{g+N} p_t \omega_{g,t}, \end{aligned}$$

The second represents the amount of bequests to future generations. In this approach, we can think about the value of bequests as a reduced form which is calibrated to two parameters: the average bequest rate, β , and the bequest elasticity, ξ . If wealth of generation g is \bar{m}_g , we have:

$$\beta = \frac{\bar{b}_g}{\bar{m}_g} \tag{1}$$

and

$$\xi = \left. \frac{\partial b_g}{\partial m_g} \frac{m_g}{b_g} \right|_{\bar{m}_g, \bar{b}_g}. \tag{2}$$

In order to make the reduced form representation, we model the bequests as a specialized transfer good that is demanded by generation g and carries the price p_g^b . Endowments of this commodity are fixed and sum to \bar{b}_g , hence p_g^b is an index of the aggregate transfer. The ownership of the bequest good then determines which generations receive the bequest and p_g^b , equal to unity in the baseline, determines the value of the bequest.

We write the utility function for a representative generation g as

$$u(b, \hat{u}) = \left[\beta^{\frac{1}{\nu}} b^{\frac{\nu-1}{\nu}} + (1-\beta)^{\frac{1}{\nu}} \hat{u}^{\frac{\nu-1}{\nu}} \right]^{\frac{\nu}{\nu-1}} \tag{3}$$

so the elasticity of substitution, ν , can be used to calibrate the bequest elasticity. The “demand” for bequests arising from budget-constrained utility maximization on the baseline growth path provides the following equation:

$$\bar{b} = \frac{\beta m}{\beta p_b + (1-\beta)p_b^{\nu}}. \tag{4}$$

Evaluating $\partial m / \partial p_b$ permits us to express ξ as a function of ν , and we may invert this relationship to find the elasticity of substitution as a function of the average bequest rate and the bequest elasticity:²

$$\nu = \frac{1 - \xi\beta}{\xi(1-\beta)}. \tag{5}$$

²When the bequest elasticity is unity the ratio of bequest to wealth is constant, and (3) is replaced by a Cobb-Douglas function with β as the value share of bequests.

10 Alternative Functional Forms

A well known dimensionless index of second-order curvature is the compensated price elasticity (CPE), which is defined as:³

$$\sigma_{ij}^C \equiv \frac{\partial \ln C_i}{\partial \ln p_j} = \frac{C_{ij} p_j}{C_i},$$

and a related measure of second-order curvature is the AUES, which has been already discussed. This can also be written as

$$\sigma_{ij}^A = \frac{\sigma_{ij}^C}{\theta_j}.$$

The AUES is a *one-input-one-price* elasticity of substitution (Mundlak, 1968), since, as the definition of σ_{ij}^A makes clear, it measures the responsiveness of the compensated demand for one input to a change in one input price. In contrast, the Morishima elasticity of substitution (MES; Morishima, 1967) constitutes a *two-input- one-price* elasticity measure, being defined as

$$\sigma_{ij}^M \equiv \frac{\partial \ln(C_i/C_j)}{\partial \ln(p_i/p_j)} = \sigma_{ij}^C - \sigma_{jj}^C.$$

Note that, in general, the MES is not symmetric, i.e. $\sigma_{ij}^M \neq \sigma_{ji}^M$.

A third type of curvature measure is represented by the class of *two-input-two-price* elasticities of substitution, which take the form $\partial \ln(C_i/C_j)/\partial \ln(p_j/p_i)$. One such index is the shadow elasticity of substitution (SES; Frenger, 1985), which is defined as

$$\sigma_{ij}^S \equiv \frac{\theta_i \sigma_{ij}^M + \theta_j \sigma_{ji}^M}{(\theta_i + \theta_j)}.$$

When technologies are of the CES type, σ_{ij}^A , σ_{ij}^M and σ_{ij}^S are all identical, but they are generally different otherwise.

10.1 The Translog Cost Function

The Translog unit cost function is defined as

$$\ln C(p) \equiv \ln b_0 + \sum_i b_i \ln p_i + \frac{1}{2} \sum_{ij} a_{ij} \ln p_i \ln p_j \equiv \ln b_0 + L(p).$$

Compensated Demand Functions:

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \frac{\partial \ln C(p)}{\partial p_i} C = \frac{C(p)}{p_i} \left[b_i + \sum_j a_{ij} \ln p_j \right]$$

Restrictions:

$$\begin{aligned} \sum_i b_i &= 1; \\ a_{ij} &= a_{ji}, \quad \forall i, \forall j; \\ \sum_j a_{ij} &= 0, \quad \forall i. \end{aligned}$$

³This exposition is based on Perroni Rutherford, 1998.

Calibration:

$$\begin{aligned} a_{ij} &= \theta_i \theta_j (\sigma_{ij}^A - 1), \quad i \neq j; \\ a_{ii} &= - \sum_{j \neq i} a_{ij}, \quad \forall i; \\ b_i &= \theta_i - \sum_j a_{ij} \ln p_j, \quad \forall i; \\ b_0 &= \bar{C} e^{-L(p)}. \end{aligned}$$

10.2 The Generalized Leontief Cost Function

The Generalized Leontief unit cost function is defined as

$$C(p) \equiv \frac{1}{2} \sum_{ij} a_{ij} \sqrt{p_i p_j}.$$

Compensated Demand Functions:

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \sum_j \frac{a_{ij}}{2} \sqrt{\frac{p_j}{p_i}}.$$

Restrictions:

$$a_{ij} = a_{ji}, \quad \forall i, \forall j.$$

Calibration:

$$\begin{aligned} a_{ij} &= 4\sigma_{ij}^A \theta_i \theta_j \frac{\bar{C}}{\sqrt{p_i p_j}}, \quad i \neq j; \\ a_{ii} &= 2\theta_i \frac{\bar{C}}{p_i} - \sum_{j \neq i} a_{ij} \sqrt{\frac{p_j}{p_i}} \quad \forall i. \end{aligned}$$

10.3 The Normalized Quadratic Cost Function

The Normalized Quadratic unit cost function is defined as

$$C(p) \equiv \frac{1}{2} \frac{\sum_{ij} a_{ij} p_i p_j}{\sum_i b_i p_i}.$$

Compensated Demand Functions:

$$x_i(p) = \frac{\partial C(p)}{\partial p_i} = \frac{\sum_j a_{ij} p_j - C(p) b_i}{\sum_j b_j p_j}$$

Restrictions:

$$\begin{aligned} a_{ij} &= a_{ji}, \quad \forall i, \forall j; \\ b_i &\geq 0, \quad \forall i; \\ \sum_i b_i &= 1. \end{aligned}$$

Calibration:

$$a_{ij} = \frac{\bar{C}\theta_i\theta_j}{p_i p_j} \left(\sigma_{ij}^A \sum_k b_k p_k + \frac{b_i p_i}{\theta_i} + \frac{b_j p_j}{\theta_j} \right), \quad \forall i, \forall j;$$

We examined two alternative specifications, one in which $b_i = \theta_i$, and another in which $b_i = 1/N$. The first specification is reported by Perroni and Rutherford [1998] to produce a more stable function.

10.4 The Nonseparable Nested CES Cost Function

We restrict our discussion to the case $N = 3$ (for the general N -input case see Perroni and Rutherford, 1995), and focus on a particular nesting structure which we call ‘‘Lower Triangular Leontief’’ (LTL). Let us rearrange indices so that the maximum off-diagonal AUES element is σ_{12}^A . Then the three-input NNCES-LTL cost function can be defined as

$$C(p) \equiv \phi \left[\alpha (a_1 p_1 + a_3 p_3)^{1-\gamma} + (1-\alpha) (b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu})^{\frac{1-\gamma}{1-\mu}} \right]^{\frac{1}{1-\gamma}}.$$

Compensated Demand Functions: We simplify the algebra by defining price indices for the two nests:

$$p_{13} = a_1 p_1 + a_3 p_3$$

and

$$p_{23} = \left[b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu} \right]^{\frac{1}{1-\mu}}$$

we have:

$$\begin{aligned} x_1(p) &= \frac{\partial C(p)}{\partial p_1} = a_1 \alpha \phi \left(\frac{C(p)}{\phi p_{13}} \right)^\gamma \\ x_2(p) &= \frac{\partial C(p)}{\partial p_2} = b_2 (1-\alpha) \phi \left(\frac{C(p)}{\phi p_{23}} \right)^\gamma \left(\frac{\phi p_{23}}{p_2} \right)^\mu \\ x_3(p) &= \frac{\partial C(p)}{\partial p_3} = a_3 \alpha \phi \left(\frac{C(p)}{\phi p_{13}} \right)^\gamma + b_3 (1-\alpha) \phi \left(\frac{C(p)}{\phi p_{23}} \right)^\gamma \left(\frac{\phi p_{23}}{p_3} \right)^\mu \end{aligned}$$

Restrictions:

$$\begin{aligned} \gamma &\geq 0; \\ \mu &\geq 0; \\ \phi &\geq 0; \\ a_i &\geq 0, \quad \forall i; \\ b_i &\geq 0, \quad \forall i. \end{aligned}$$

Calibration:

Let us denote with s_3 the fraction of the total input of commodity 3 which enters the first subnest of the structure:

$$C(p) \equiv \phi \left[\alpha (a_1 p_1 + a_3 p_3)^{1-\gamma} + (1-\alpha) (b_2 p_2^{1-\mu} + b_3 p_3^{1-\mu})^{\frac{1-\gamma}{1-\mu}} \right]^{\frac{1}{1-\gamma}}.$$

(with $(1 - s_3)$ representing the fraction entering the second subnest). If we select

$$\begin{aligned}\gamma &= \sigma_{12}^A; \\ \mu &= \frac{\sigma_{12}^A \sigma_{13}^A - \sigma_{23}^A \sigma_{11}^A}{\sigma_{13}^A - \sigma_{11}^A};\end{aligned}$$

it can be shown that

$$s_3 = \frac{\sigma_{12}^A - \sigma_{13}^A}{\sigma_{12}^A - \sigma_{11}^A}.$$

The remaining parameters can then be recovered as follows:

$$\begin{aligned}\phi &= \bar{C}; \\ \alpha &= \theta_1 + s_3 \theta_3 \\ a_1 &= \frac{\theta_1}{\alpha p_1} \\ a_3 &= \frac{s_3 \theta_3}{\alpha p_3} \\ b_2 &= \frac{\theta_2}{(1 - \alpha) p_2^{1-\mu}} \\ b_3 &= \frac{(1 - s_3) \theta_3}{(1 - \alpha) p_3^{1-\mu}}\end{aligned}$$

References

- Kotlikoff, Laurence J. and Lawrence H. Summers**, “The Role of Intergenerational Transfers in Aggregate Capital Accumulation,” *Journal of Political Economy*, 1981, 89 (4), 706–732.
- Perroni, Carlo and Thomas F. Rutherford**, “Regular flexibility of nested CES functions,” *European Economic Review*, 1995, 39, 335–343.
- and —, “Regular flexibility of nested CES functions,” *Computational Economics*, 1998, 11, 245–263.
- Rutherford, Thomas F.**, “Applied General Equilibrium Modeling with MPSGE as a GAMS Subsystem: An Overview of the Modeling Framework and Syntax,” *Computational Economics*, 1999, 14, 1–46.

Calculating Equivalent and Compensating Variations in CGE Models

Véronique Robichaud, July 2001

Introduction

Most CGE modelers use these models to assess the impacts of given shocks or policies on a specific economy. While it is quite straightforward to measure impacts on aggregate nominal production and consumption levels, relative prices, nominal income and savings, it is less obvious to **quantitatively** evaluate how much better or worse off the households are. As direct and indirect utility functions are purely ordinal in nature, we can only analyse the direction of change. An interesting alternative is provided by using the *money metric utility function*, which measures the nominal income the consumer needs at one set of prices in order to be as well off at an alternative set of prices and nominal income. As such, it can be used to obtain monetary measures of the welfare effects of different policy scenarios. The most common of these measures are equivalent variations (EV) and compensating variations (CV), which we will present in this note.

We might also be tempted to EVs and CVs between households. However, this is misleading as the utility each consumer receives from a given amount of income differs. In the same way, EVs and CVs should not, in principle, be aggregated across households as a welfare measure. Nonetheless, aggregate EVs and CVs could be interpreted as the total lump-sum transfer that is equivalent to the policy change that is examined. Note, finally, that even "disaggregate" EVs and CVs in CGE models are evaluated at the semi-aggregate level of household categories rather than at the individual consumer level. This brings us into the larger debate of aggregation in CGE models and the representative agent assumption that goes beyond this note (see Kirman 1992).

The objectives of the present memo is to provide MIMAP teams with a concise presentation of money metric welfare indicators and their calculation using the most popular utility functions found in CGE models. The general formulation will be presented as well as its derivation for the Cobb-Douglas and the LES utility functions.

Theory

Let us first define the *utility function*, $\mu(C)$, the *indirect utility function*, $v(P, Y)$ and the *money metric indirect utility function*, $m(P, \mu)$. In these expressions, C represents the vector of goods consumed, P the vector of prices and Y household income. Equation (1A) represents the utility function for a Cobb-Douglas specification and equation (1B) for the LES specification:

$$\mu^{CD}(C) = \prod_{i=1}^I C_i^{\alpha_i} \quad (1A)$$

$$\mu^{LES}(C) = \prod_{i=1}^I (C_i - \gamma_i)^{\beta_i} \quad (1B)$$

where $\sum_{i=1}^I \alpha_i = 1$ and $\sum_{i=1}^I \beta_i = 1$ and γ_i represents the minimum consumption level.

Given these specifications, demand functions are derived by maximizing utility subject to the budget constraint:

$$C_i^{CD}(P, Y) = \frac{\alpha_i Y}{P_i} \quad (2A)$$

$$C_i^{LES}(P, Y) = \gamma_i + \frac{\beta_i}{P_i} \left(Y - \sum_{i=1}^I \gamma_i P_i \right) \quad (2B)$$

The *indirect utility function*, $V(P, Y)$, is obtained by replacing the C_i in the utility functions (1A) and (1B) with the demand functions (2A) and (2B), respectively:

$$V^{CD}(P, Y) = \prod_{i=1}^I \left(\frac{\alpha_i Y}{P_i} \right)^{\alpha_i} \quad (3A)$$

$$\begin{aligned} V^{LES}(P, Y) &= \prod_{i=1}^I \left[\frac{\beta_i}{P_i} \left(Y - \sum_{i=1}^I \gamma_i P_i \right) \right]^{\beta_i} \\ &= \prod_{i=1}^I \left(\frac{\beta_i}{P_i} \right)^{\beta_i} \prod_{i=1}^I \left(Y - \sum_{i=1}^I \gamma_i P_i \right)^{\beta_i} \\ &= \left(Y - \sum_{i=1}^I \gamma_i P_i \right) \prod_{i=1}^I \left(\frac{\beta_i}{P_i} \right)^{\beta_i} \end{aligned} \quad (3B)$$

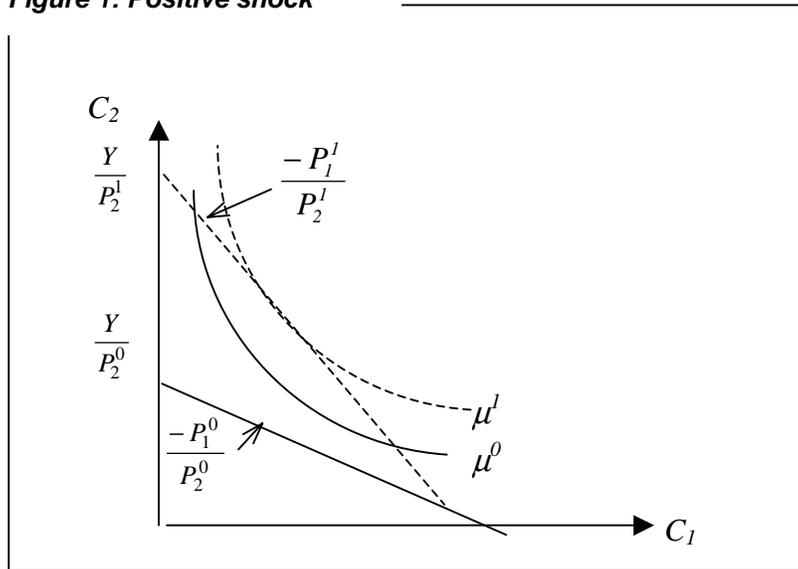
Solving equation (3A) and (3B) for Y gives the *money metric indirect utility function*, $m(P, V)$, which is a measure of the income needed to attain utility level v at the vector of prices P :

$$m^{CD}(P, v) = \prod_{i=1}^I \left(\frac{P_i}{\alpha_i} \right)^{\alpha_i} v \quad (4A)$$

$$m^{LES}(P, v) = \prod_{i=1}^I \left(\frac{P_i}{\beta_i} \right)^{\beta_i} v + \sum_{i=1}^I \gamma_i P_i \quad (4B)$$

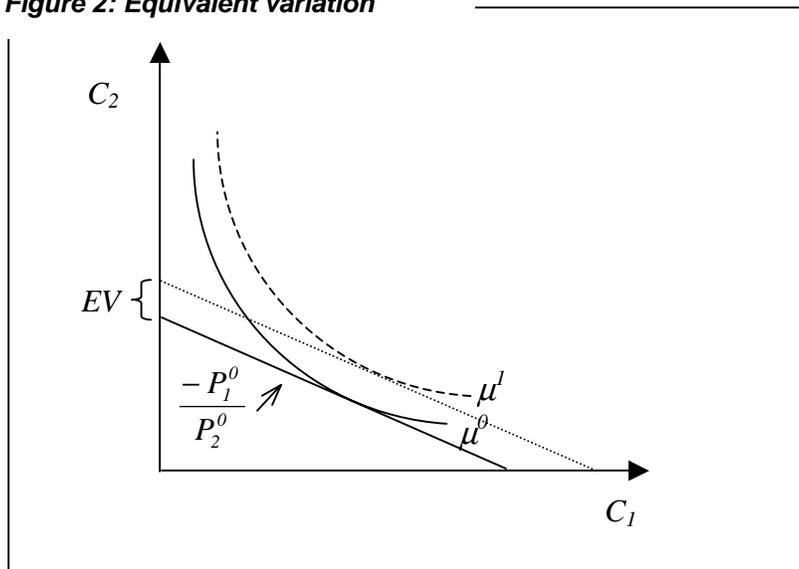
Equivalent and compensating variations are welfare measures based on the money metric indirect utility function. Before deriving the mathematical expressions for these measures, let us quickly recall graphically what they represent using a two-good example. Suppose that initially prices were (P_1^0, P_2^0) . After some policy change, new prices (P_1^1, P_2^1) are observed assuming, for simplicity, that nominal income (Y) is unchanged. Figure 1 shows that, in this example, a higher utility level is attained after the policy change, since μ^1 is further than μ^0 from the origin.

Figure 1: Positive shock



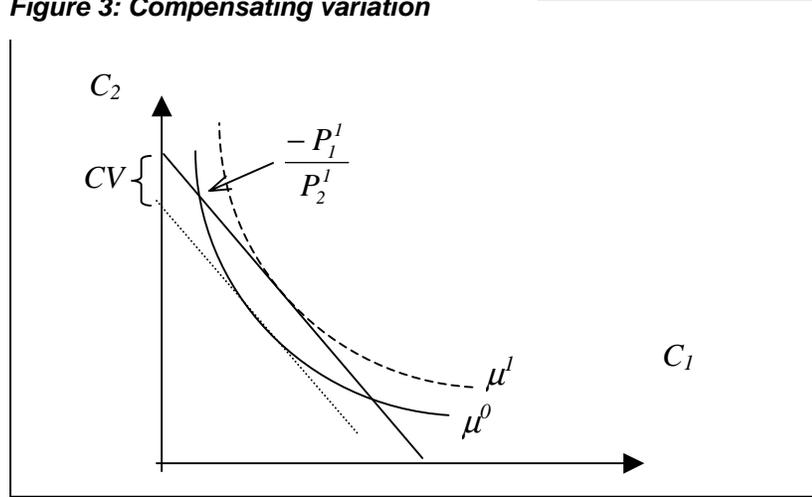
Although we can simply say that the policy had a positive impact on the consumer, we may want to quantitatively estimate the "size" of the impact in order to be able to compare between consumers and/or scenarios. The money metric indirect utility function allows such an evaluation. This is done by measuring the distance between the two indifference curves, at constant prices. Those prices could be the initial ones or the final ones. Figure 2 shows the first measure, which is called equivalent variation. The intuition behind that measure is to calculate what would be the income change, at initial prices, that is welfare-equivalent to the observed change in prices.

Figure 2: Equivalent variation



The second measure is known as the compensating variation (CV) and uses the post-policy prices. The CV thus measures the income change necessary to compensate the consumer for the change in prices. Figure 3 represents that measure.

Figure 3: Compensating variation



Let us now derive these two measures formally in the case where both prices and nominal income change. The equivalent variation is given by:

$$EV = m(P_i^0, v(P_i^1, Y^1)) - m(P_i^0, v(P_i^0, Y^0)) = m(P_i^0, v(P_i^1, Y^1)) - Y^0$$

For the Cobb-Douglas specification, we use equation (3A) and (4A):

$$\begin{aligned} EV &= m^{CD}(P_i^0, v^{CD}(P_i^1, Y^1)) - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{\alpha_i} \right)^{\alpha_i} v^{CD}(P_i^1, Y^1) - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{\alpha_i} \right)^{\alpha_i} \prod_{i=1}^I \left(\frac{\alpha_i Y^1}{P_i^1} \right)^{\alpha_i} - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{P_i^1} \right)^{\alpha_i} Y^1 - Y^0 \end{aligned} \quad (5A)$$

Similarly, for the LES specification, we use equation (3B) and (4B):

$$\begin{aligned} EV &= m^{LES}(P_i^0, v^{LES}(P_i^1, Y^1)) - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{\beta_i} \right)^{\beta_i} v^{LES}(P_i^1, Y^1) + \sum_{i=1}^I \gamma_i P_i^0 - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{\beta_i} \right)^{\beta_i} \left(Y^1 - \sum_{i=1}^I \gamma_i P_i^1 \right) \prod_{i=1}^I \left(\frac{\beta_i}{P_i^1} \right)^{\beta_i} + \sum_{i=1}^I \gamma_i P_i^0 - Y^0 \\ &= \prod_{i=1}^I \left(\frac{P_i^0}{P_i^1} \right)^{\beta_i} \left(Y^1 - \sum_{i=1}^I \gamma_i P_i^1 \right) - \left(Y^0 - \sum_{i=1}^I \gamma_i P_i^0 \right) \end{aligned} \quad (5B)$$

The compensating variation is given by the following expression:

$$CV = m(P_i^1, v(P_i^1, Y^1)) - m(P_i^1, v(P_i^0, Y^0)) = Y^1 - m(P_i^1, v(P_i^0, Y^0))$$

For the Cobb-Douglas specification and using equations (3A) and (4A):

$$\begin{aligned} CV &= Y^1 - m^{CD}(P_i^1, v^{CD}(P_i^0, Y^0)) \\ &= Y^1 - \prod_{i=1}^I \left(\frac{P_i^1}{\alpha_i} \right)^{\alpha_i} v^{CD}(P_i^0, Y^0) \\ &= Y^1 - \prod_{i=1}^I \left(\frac{P_i^1}{\alpha_i} \right)^{\alpha_i} \prod_{i=1}^I \left(\frac{\alpha_i Y^0}{P_i^0} \right)^{\alpha_i} \\ &= Y^1 - \prod_{i=1}^I \left(\frac{P_i^1}{P_i^0} \right)^{\alpha_i} Y^0 \end{aligned} \quad (6A)$$

And finally, for the LES specification, using equations (3B) and (4B):

$$\begin{aligned} CV &= Y^1 - m^{LES}(P_i^1, v^{LES}(P_i^0, Y^0)) \\ &= Y^1 - \prod_{i=1}^I \left(\frac{P_i^1}{\beta_i} \right)^{\beta_i} v^{LES}(P_i^0, Y^0) - \sum_{i=1}^I \gamma_i P_i^1 \\ &= \left(Y^1 - \sum_{i=1}^I \gamma_i P_i^1 \right) - \prod_{i=1}^I \left(\frac{P_i^1}{\beta_i} \right)^{\beta_i} \left(Y^0 - \sum_{i=1}^I \gamma_i P_i^0 \right) \prod_{i=1}^I \left(\frac{\beta_i}{P_i^0} \right)^{\beta_i} \\ &= \left(Y^1 - \sum_{i=1}^I \gamma_i P_i^1 \right) - \prod_{i=1}^I \left(\frac{P_i^1}{P_i^0} \right)^{\beta_i} \left(Y^0 - \sum_{i=1}^I \gamma_i P_i^0 \right) \end{aligned} \quad (6B)$$

References:

- Kirman, A. (1992), "Whom or What does the Representative Individual Represent?", *Journal of Economic Perspectives*, 6(2): 117-136.
- Varian, Hal R. (1992), Microeconomic Analysis, Third Edition, W.W. Norton & Company, New York.
- Varian, Hal R. (1999), Intermediate Microeconomics: A Modern Approach, Fifth Edition, W.W. Norton & Company, New York.