Introduction to Computable General Equilibrium Model (CGE)

Dhazn Gillig
&
Bruce A. McCarl

Department of Agricultural Economics
Texas A&M University

Course Outline

- Overview of CGE
- An Introduction to the Structure of CGE
- An Introduction to GAMS
- **Casting CGE models into GAMS**
- Data for CGE Models & Calibration
- Incorporating a trade & a basic CGE application
- Evaluating impacts of policy changes and casting nested functions & a trade in GAMS
- Mixed Complementary Problems (MCP)

This Week's Road Map

- **Add-on a simple market clearing problem via GAMS**
- **Casting CGE via GAMS**
  - : Set definitions
  - : Data entry
  - : Variable & Equation specifications
  - : Identifying complementarity relationship
  - : Normalizing prices
  - : Solution reports
  - : Comparative analysis

Formulation of a Simple Market Clearing

- **Demand:** $P \geq Pd = 6 - 0.3*Qd$
- **Supply:** $P \leq Ps = 1 + 0.2*Qs$
- **Equilibrium:** $Qs \geq Qd$ and $P, Qs, Qd \geq 0$

2 commodities: corn and wheat

Corn Demand: $P_c \geq Pd_c = 6 - 0.3*Qd_c - 0.1*Qd_w$

Wheat Demand: $P_w \geq Pd_w = 8 - 0.07*Qd_c - 0.4*Qd_w$

Corn Supply: $P_c \leq Ps_c = 1 + 0.5*Qs_c + 0.1*Qs_w$

Wheat Supply: $P_w \leq Ps_w = 2 + 0.1*Qs_c + 0.3*Qs_w$

Corn Equilibrium: $Qs_c \geq Qd_c$

Wheat Equilibrium: $Qs_w \geq Qd_w$

$P_c, P_w, Qd_c, Qd_w, Qs_c$, and $Qs_w \geq 0$

Formulation of a Simple Market Clearing

Set Definition & Data Entry

```
SET Commodities  commodities used in the model            /Corn,Wheat/    ;
SET Curvetype    supply and demand intercept and slope  /Supply,Demand/ ;

TABLE intercepts(Curvetype,Commodities)  supply and demand intercept terms
                  Corn    Wheat
        demand     6        8
        supply     1        2;

TABLE slopes(Curvetype,Commodities,Commodities)  supply and demand slope te
                   Corn   Wheat
     Demand.Corn   -.3     -.1
     Demand.Wheat  -.07    -.4
     Supply.Corn    .5      .1
     Supply.Wheat   .1      .3      ;
```

Formulation of a Simple Market Clearing

STEPS

1. Set definitions

2. Data entry

3. Variables specification

4. Equations specification

    a. declaration

    b. algebraic structure specification

5. Model statement

6. Solve statement

Set Definitions

In algebraic modeling, we commonly have subscripts.

In GAMS, the corresponding items are sets. A set definition has several potential parts.

SET ItemName optional explanatory text for item

/ element1 optional explanatory text for element ,

element2 optional explanatory text for element / ;

SET or SETS	**to start**
ItemName	**a unique name**
optional explanatory text for item	
/	**opening slash**
Element names	
optional explanatory text for element	
, or line feed	**to separate elements**
/	**closing slash**
;	**a closing ;**

Set Definitions

In our example:

```
SET Commodities  commodities used in the model              /Corn,Wheat/     ;
SET Curvetype    supply and demand intercept and slope  /Supply,Demand/ ;
```

Define set names

Text comments, Optional in command

Assign elements to the sets

Set Definitions

Another example:

```
SET    SECTORS    sectors of the economy
   / Steel        steel mining sector (in millions of tons sold)
     Energy       energy sector (in millions of btus sold)
     Coal         coal sector (in millions of tons sold)
  / ;
```

Element explanatory text

Note: the explanatory text must not exceed 80 characters and must all be contained on the same line as the identifier it describes.

Set Definitions- Alias

ALIAS is used to give another name to previously defined sets.

```
ALIAS (Commodity,Commodities);
```

"**Commodities**" is like a **j** and **j'** in mathematical notation.

Data Entry

Data are entered via three different types of GAMS commands

1) Scalar – for items that are not set dependent

2) Parameters – for items that are vectors (can be multidimensional)

3) Tables – for items with 2 or more dimensions

Data Entry - **SCALAR commands**

Scalar commands:

Basic format:

SCALAR ItemName optional text / value / ;

In the CGE example:

SCALAR Incometax Household tax level / 0.00 / ;

Data Entry - **PARAMETER commands**

Basic format:

```
PARAMETER    ItemName(setdependency)    optional text
   / element1      value1 ,
     element2      value2 /   ;
```

In the CGE example:

```
PARAMETER
   SigmaC(HouseHolds)    Household elas. of substitution
     / NonFarmer            1.5
       Farmer               0.75 /

   Phi(Sector)           Production scale parameter
     / Food                 1.5
       NonFood              2.0  /        ;
```

Data Entry – TABLE commands

Basic format:

```
TABLE   ItemName(set1dep,set2dep)    optional text
                      set2elem1       set2elem2
   set1element1       value11         value12
   set1element2       value12         value22         ;
```

In our example:

```
TABLE  Intercept(Curvetype,Commodities)    intercept term
                     Corn            Wheat
   Demand            6               8
   Supply            1               2         ;
```

Corn, Wheat → Elements from Commodities set (2nd set)

Demand, Supply ↓ Elements from Curvetype set (1st set)

Data Entry – TABLE commands

More than two dimensional data entry using TABLE

Define table name

Associated sets

Text comments, Optional in command

```
TABLE Slopes(Curvetype,Commodities,Commodities)   supply and demand slope
                       Corn      Wheat
      Demand . Corn   -0.3      -0.1
      Demand . Wheat  -0.07     -0.4
      Supply . Corn    0.5       0.1
      Supply . Wheat   0.1       0.3     ;
```

Elements from Commodities set (3rd set)

Elements from Curvetype set (1st set)

Elements from Commodities set (2nd set)

Formulation – Variable Declarations

Basic format:

VARIALBE VarName1(setdependency) optional text

VarName2(setdependency) optional text

… ;

to declare variables < or ≥ 0

Or

POSITIVE VARIABLE

VarName1(setdependency) optional text

VarName2(setdependency) optional text

… ;

To declare ≥ 0 variables

Formulation – **Variable Declarations**

In our example:

```
POSITIVE VARIABLES
      P(Commodities)                    Equilibrium price
      Qd(Commodities)                   Quantity demanded
      Qs(Commodities)                   Quantity supply     ;
```

Note that this defines a variable for each case in the set commodities and thus encompasses the cases:

$$P_c, P_w, Qd_c, Qd_w, Qs_c, Qs_w \geq 0$$

Formulation – Equation Declarations

Basic format:

Equation EqName1(setdependency) optional text

EqName2(setdependency) optional text

… ;

Formulation – **Equation Declarations**

In our example:

```
EQUATIONS
      PDemand(Commodities)        Demand equation
      PSupply(Commodities)        Supply equation
      Equilibrium(Commodities)    Equilibrium equation ;
```

Note that this defines an equation for each case in the set commodities

Formulation – **Equation Specifications**

General Structure:

DeclaredEquationName(SetDependency)..
LHSalgebra EquationRelationType
RHSalgebra ;

where

DeclaredEquationName was in an equation declaration with this **setdependency**.

LHSalgebra and **RHSalgebra** can contain any mixture of variables, parameters, and data in algebraic relations.

EquationRelationType tells equality or inequality nature

; are mandatory

Formulation – **Equation Specifications**

Algebraic Structure

- **Demand:** $P_c \geq Pd_c = 6 - 0.3*Qd_c - 0.1*Qd_w$

  $P_w \geq Pd_w = 8 - 0.07*Qd_c - 0.4*Qd_w$

```
Pdemand(commodities)..
    P(commodities)
     =G=
    intercepts("demand",commodities)
     + SUM(commodity,
           slopes("demand",commodities,commodity)
             *Qd(commodity));
```

Quotes " " are used to select a specific set elements.
Recall: ALIAS(commodity,commodities);

Summation Digression

$$\sum_{j} X_{j}$$

SUM(index of summation , names(index))

SUM(j, X (j))

$$\sum_{j} \sum_{i} X_{ji}$$

SUM(index1 , SUM(index2 , names(index1,index2)))

SUM(j, SUM(i, X (j,i)))

or SUM((j,i), X (j,i))

Formulation – **Equation Specifications**

Algebraic Structure

- **Supply:** $Ps_c = 1 + 0.5*Qs_c + 0.1*Qs_w \geq P_c$
  $Ps_w = 2 + 0.1*Qs_c + 0.3*Qs_w \geq P_w$

```
Psupply(commodities)..
    intercepts("supply",commodities)
     + SUM(commodity,
        slopes("supply",commodities,commodity)
          *Qs(commodity))
    =G=
   P(commodities);
```

Formulation – **Equation Specifications**

Algebraic Structure

- **Equilibrium:** $Qs_c \geq Qd_c$

  $Qs_w \geq Qd_w$

```
Equilibrium(commodities)..
     Qs(commodities)
      =G=
     Qd(commodities)   ;
```

Formulation – **Model and Solve Statement**

```
MODEL PROBLEM    /Pdemand.Qd
                  Psupply.Qs
                  Equilibrium.P/  ;

SOLVE PROBLEM USING MCP;
```

Recall: MCP Requirements

- consistent dimension (sets) of complementary variables and equations
- no variable is complementary with more than one equation or vice versa
- every variable and equation has a complementary partner

```
POSITIVE VARIABLES
      P(Commodities)
      Qd(Commodities)
      Qs(Commodities)

EQUATIONS
      PDemand(Commodities)
      PSupply(Commodities)
      Equilibrium(Commodities)
```

Solution

```
---- EQU PDemand  Demand equation

           LOWER      LEVEL      UPPER     MARGINAL

Corn        6.000      6.000      +INF       4.373
Wheat       8.000      8.000      +INF       7.510

---- EQU PSupply  Supply equation

           LOWER      LEVEL      UPPER     MARGINAL

Corn       -1.000     -1.000      +INF       4.373
Wheat      -2.000     -2.000      +INF       7.510

---- EQU Equilibrium  Equilibrium equation

           LOWER      LEVEL      UPPER     MARGINAL

Corn         .          .         +INF       3.937
Wheat        .          .         +INF       4.690
```

Solution

```
---- VAR P  Equilibrium price

           LOWER     LEVEL     UPPER    MARGINAL

Corn         .       3.937     +INF        .
Wheat        .       4.690     +INF        .

---- VAR Qd  Quantity demanded

           LOWER     LEVEL     UPPER    MARGINAL

Corn         .       4.373     +INF        .
Wheat        .       7.510     +INF        .

---- VAR Qs  Quantity supply

           LOWER     LEVEL     UPPER    MARGINAL

Corn         .       4.373     +INF        .
Wheat        .       7.510     +INF        .
```

Casting CGE in GAMS

Lets set up a model depicting a 2x2x2 economy with

Two factors of production (labor and capital)

Two commodities produced (food and nonfood)

Two household classes (farmer and nonfarmer)

STEPS

1. Set definitions
2. Data entry
3. Variables specification
4. Equations specification
    a. declaration
    b. algebraic structure specification

Set Definitions

Sets definition for a 2x2x2 CGE model

SET

Factor	Factors of production	**/ Labor, Capital /**
Sector	Producing industries	**/ Food, NonFood /**
Households	Household types	**/ Farmer,NonFarmer / ;**

Data Entry – TABLE commands

```
TABLE  Alpha(Sector,HouseHolds) Consumption share
                 NonFarmer   Farmer
   Food            0.5          0.3
   NonFood         0.5          0.7 ;

TABLE  Endowment(Factor,HouseHolds) Factor Endow
                NonFarmer Farmer
   Labor            0         60
   Capital         25          0   ;
```

	Non-Farmer	Farmer
Food	0.5	0.3
Non-Food	0.5	0.7
Labor	0.0	60
Capital	25	0.0

Data Entry – Direct Assignment

Basic format:

```
PARAMETER  ItemName(set1dep,set2dep) optional text  ;
           ItemName(set1dep,set2dep) =  0 ;
```

In our example:

```
PARAMETER TaxRate(Factor,Sector) Consumption share ;
          TaxRate(Factor,Sector)  =  0 ;
```

CGE Variable Specification

POSITIVE VARIABLE

FactorPrice(Factor)	**Prices for factors**
FactorQuan(Factor,Sector)	**Factors used by a sector**
ComPrice(Sector)	**Prices of commodities**
DemCommod(Households,Sector)	**Demand by household**
Production(Sector)	**Production quantity level**
HHIncome(Households)	**Household income**
TaxRevenue	**Government tax revenue**

;

CGE Equation Specification

EQUATION

FactorMkt(Factor)	**Factor market balances**
FactorDem(Factor,Sector)	**Factor demand by a sector**
CommodMkt(Sector)	**Commodity market balance**
CommodDem(Households,Sector)	**Commodity demand**
Profit(Sector)	**Zero profit condition**
Income(households)	**Household budget**
GovBal	**Government budget**

;

Equation Specification

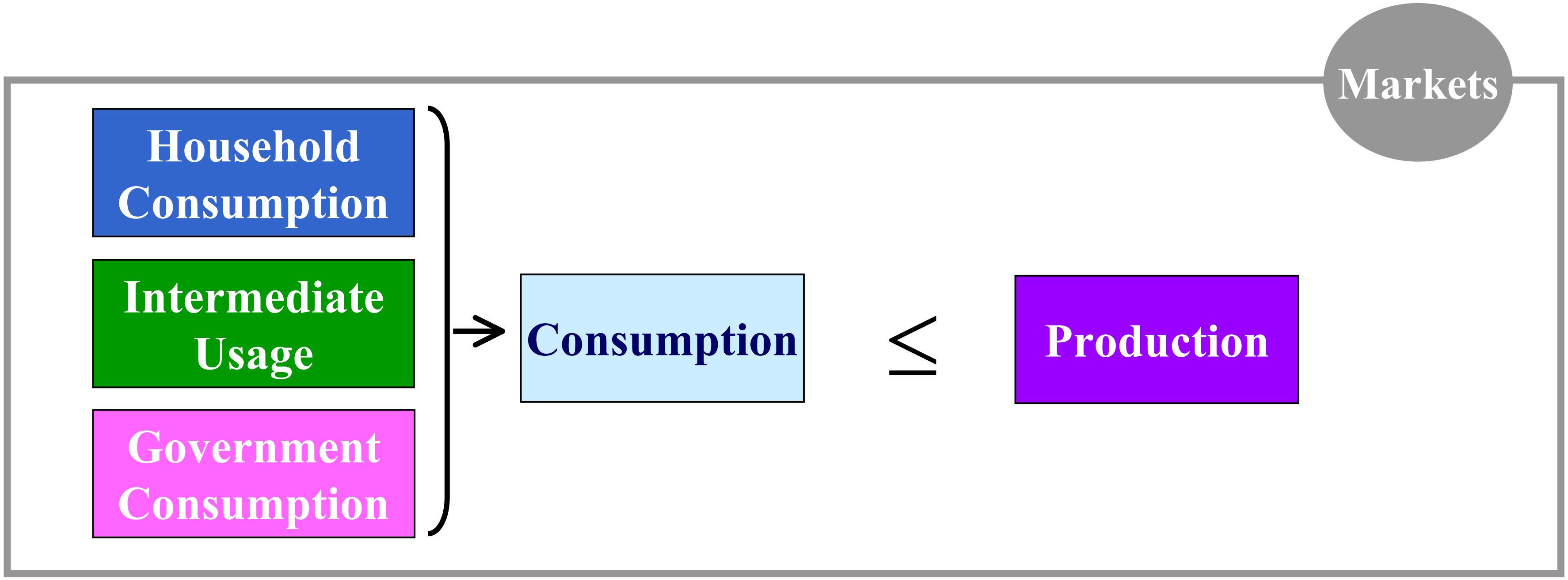
1. **Supply-Demand identities for each factor market**

   The total demand is less than or equal to the total supply in every factor market.

$$\left.\begin{aligned} \sum_j L_j &\leq \sum_h \bar{L}_h \\ \sum_j K_j &\leq \sum_h \bar{K}_h \end{aligned}\right\} \Rightarrow \sum_j F_{fj} \leq \sum_h \bar{F}_{fh}$$

```
FactorMkt(Factor)..
SUM(Sector,FactorQuan(Factor,Sector))
  =L=
SUM(HouseHolds,Endowment(Factor,HouseHolds))      ;
```

2. Supply-Demand identities for each output market




```
CommodMkt(Sector)..
  SUM(Households,DemCommod(Households,Sector))
+ SUM(OtherSector,
    IntermediateUse(Sector,OtherSector)
     *Production(OtherSector))
+ GovernmentPurch(Sector)*TaxRevenue/ComPrice(Sector)
 =L= Production(Sector) ;
```

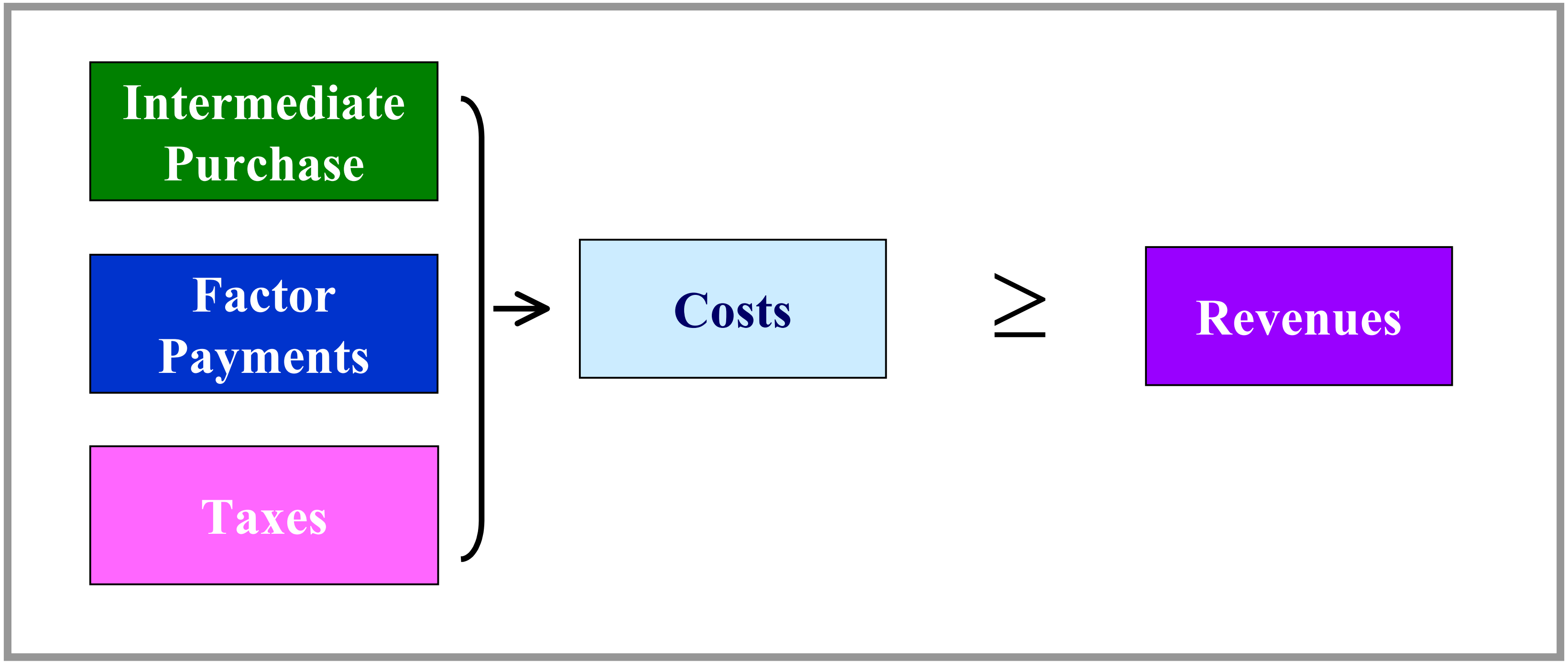
$$\sum_{h} X_{hj} + \sum_{j1} a_{j,j1} Q_{j1} + s_j R / P_j \le Q_j$$

```
CommodMkt(Sector)..
  SUM(Households,DemCommod(Households,Sector))
+ SUM(OtherSector,
    IntermediateUse(Sector,OtherSector)
     *Production(OtherSector))
+ GovernmentPurch(Sector)*TaxRevenue/ComPrice(Sector)
  =L=
 Production(Sector)
   ;
```

3. Zero Profit Conditions

$$\sum_{j1} P_{j1}\, a_{j1,j} Q_j + \sum_f (1+t_{fj}) W_f F_{fj}$$

$$\geq \; P_j Q_j$$



$$\sum_{j1} P_{j1}\, a_{j1,j} Q_j \;+ \sum_f (1+t_{fj}) W_f F_{fj} \;\geq\; P_j Q_j$$

```
Profit(Sector)..

 +  SUM(OtherSector,
          ComPrice(OtherSector)*IntermediateUse(OtherSector,Sector)
             * Production(Sector) )
 +  SUM(Factor, (1+TaxRate(Factor,Sector))*FactorPrice(Factor)
             *FactorQuan(Factor,Sector) )
   =G=
 ComPrice(Sector)*  Production(Sector)      ;
```

4. Factor demand by producers

$$F_{fj} = Q_j \phi_j^{(\sigma_j - 1)} \times (\delta_{fj} (\sum_{f'} (\delta_{f'j}^{\sigma_j} (W_{f'} (1 + t_{f'j}))^{(1-\sigma_j)})^{1/(1-\sigma_j)} / \phi_j) / (W_f (1 + t_{fj})))^{\sigma_j}$$

```
FactorDem(Factor,Sector)..
  FactorQuan(Factor,Sector)
   =E=
  Production(Sector)*Phi(Sector)**(sigma(Sector)-1)
   *(  Delta(Factor,Sector)
        *( SUM(Factor1,Delta(Factor1,Sector)**sigma(Sector)
                        *(FactorPrice(Factor1)*(1 + taxrate(Factor1,Sector)))
                              **(1 - sigma(Sector))
                   )**(1/(1-sigma(Sector))) /Phi(Sector)
        ) / (FactorPrice(Factor) * (1+taxrate(Factor,Sector)))
     ) **sigma(Sector) ;
```

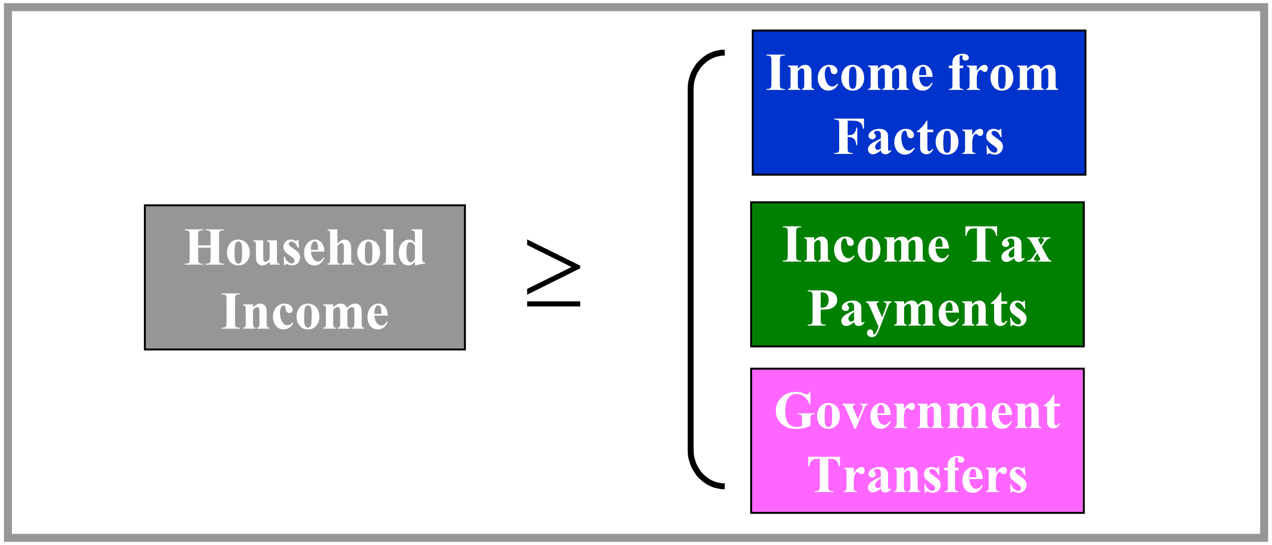
Note: using ALIAS (f,f');

5. Product demand by households

$$X_{hj} = \frac{\alpha_{jh} \; Income_h}{P_j^{\sigma_h} \sum_{j'} \left(\alpha_{j'h} \left(P_{j'} \right)^{1-\sigma_h} \right)}$$

```
CommodDem(Households,Sector)..
  DemCommod(Households,Sector)
   =E=
    (Alpha(Sector,HouseHolds) * HHIncome(HouseHolds))
  / (ComPrice(Sector)**sigmaC(HouseHolds)
       * SUM(Sector1,
                Alpha(Sector1,HouseHolds)
                   * ComPrice(Sector1)**(1-SigmaC(HouseHolds) ) )
    ) ;
```

6. Household income constraint

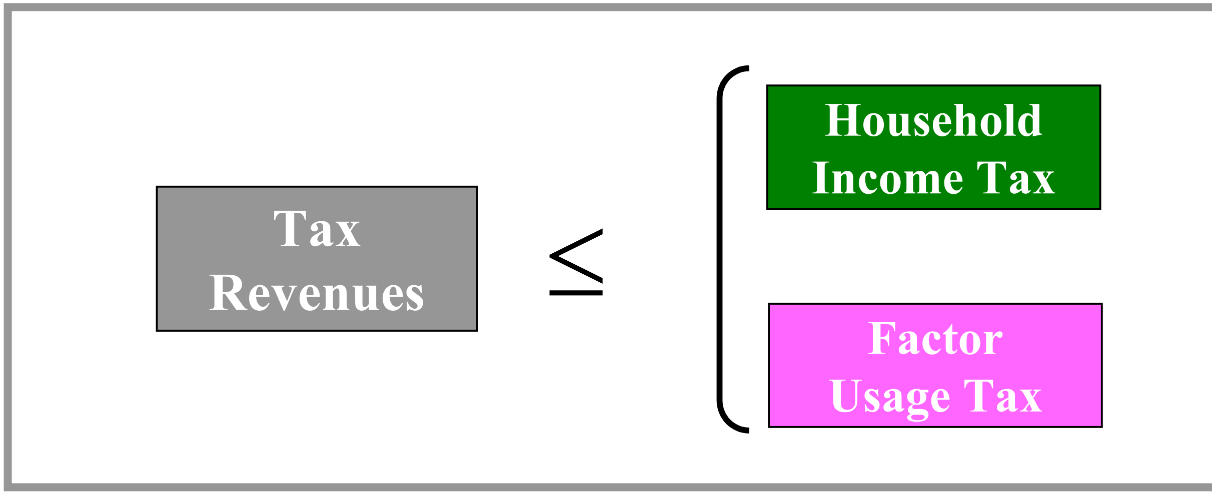



```
Income(HouseHolds)..
  HHIncome(HouseHolds)
   =G=
  SUM(Factor,Endowment(Factor,HouseHolds)
          * FactorPrice(Factor))
– incometax(HouseHolds)*
     SUM(Factor,Endowment(Factor,HouseHolds)
              * FactorPrice(Factor) )
+ TaxShare(HouseHolds) * TaxRevenue             ;
```

$$Income_h \geq \sum_f \bar{F}_{fj} W_f - t_h \sum_f \bar{F}_{fj} W_f + s_h R$$

```
Income(HouseHolds)..
  HHIncome(HouseHolds)
   =G=
  SUM(Factor,Endowment(Factor,HouseHolds)
          * FactorPrice(Factor))
– incometax(HouseHolds)*
     SUM(Factor,Endowment(Factor,HouseHolds)
              * FactorPrice(Factor) )
+ TaxShare(HouseHolds) * TaxRevenue            ;
```

7. Government income constraint




```
GovBal..
   TaxRevenue
   =L=
   SUM(Households, Incometax(Households) *
      SUM(Factor,Endowment(Factor,HouseHolds)* FactorPrice(Factor)))
 + SUM((Factor,Sector),TaxRate(Factor,Sector) * FactorPrice(Factor)
         *FactorQuan(Factor,Sector) )     ;
```

$$R \leq \sum_{h} \left(t_h \sum_{f} \bar{\mathbf{F}}_{fh} W_f \right) + \sum_{fj} t_{fj} W_f F_{fj}$$

```
GovBal..
   TaxRevenue
   =L=
    SUM(Households, Incometax(Households) *
       SUM(Factor,Endowment(Factor,HouseHolds)* FactorPrice(Factor)))
 + SUM((Factor,Sector),TaxRate(Factor,Sector) * FactorPrice(Factor)
          *FactorQuan(Factor,Sector) )     ;
```

Model Complementarity Relationship

```
MODEL  CGEModel

   / FactorMkt.FactorPrice
     FactorDem . FactorQuan
     Commoddem . DemCommod
     CommodMkt . ComPrice
     Profit . Production
     Income . HHincome
     Govbal . TaxRevenue
     CommodDem . DemCommod        / ;
```

Equation Name	Variable Name
FactorMkt(Factor)	FactorPrice(Factor)
FactorDem(Factor,Sector)	FactorQuan(Factor,Sector)
CommodMkt(Sector)	ComPrice(Sector)
Profit(Sector)	Production(Sector)
Income(households)	HHIncome(households)
GovBal	TaxRevenue
CommodDem(Households,Sector)	DemCommod(Households,Sector)

Other Features

Normalizing Prices

Recall: a property of our model is that we are homogenous of degree zero in prices, thus an infinite number of prices will solve above equations. To overcome this problem, we need to normalize on something. We can set

- the income for one household equal to one,
- or the price of a commodity to one.

Recall only **relative prices** affect behavior in CGE, so it does not matter which price is chosen. **FactorPrice.FX("Labor")= 1;**

In the 2x2x2 example, labor price is set as numeraire.

Solution example:

	NoTax	Tax
Labor	**1.00**	**1.00**
Capital	1.37	1.13
Food	1.40	1.47
NonFood	1.09	1.01

Other Features

Starting points, bounds, and SOLVE statements

To avoid numerical problems with lots of zero variables and to speed up convergence, starting points (*.L) and lower bounds (*.LO) are needed.

```
FactorPrice.L(Factor)    = 1;
FactorPrice.LO(Factor)  = 0.0001;
```

The CGE model is best solved with the PATH solver.

(http://www.gams.com/solver.htm#PATH)

OPTION MCP = PATH ; **=> choose PATH as the solver**

SOLVE CGEModel **USING MCP ;**

Solutions

Status Reports

After the solver executes, GAMS prints out a brief “**SOLVE SUMMARY**” indicating “**SOLVER STATUS**” and the “**MODEL STATUS**”.

```
               S O L V E      S U M M A R Y

     MODEL   CGEModel
     TYPE    MCP
     SOLVER  PATH                FROM LINE  306

**** SOLVER STATUS     1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL

 RESOURCE USAGE, LIMIT          0.660      1000.000
 ITERATION COUNT, LIMIT         4         10000
 EVALUATION ERRORS              0             0
```

Solutions

Solution Reports

The report summary gives the total number of non-optimal, infeasible, and unbounded.

```
**** REPORT SUMMARY :          0      NONOPT
                               0  INFEASIBLE
                               0   UNBOUNDED
                               0   REDEFINED
                               0      ERRORS
```

Solutions can be presented in several ways:

1. GAMS solution output format as above
2. Addition of **DISPLAY** commands to write out values associated with identified sets, parameters, variables, and equations
3. Added computed reports using values from solutions

Solutions

1. A standard GAMS solution format

```
---- EQU Profit  Zero profit condition
              LOWER     LEVEL     UPPER    MARGINAL

Food            .         .       +INF      24.942
NonFood         .         .       +INF      54.378

---- VAR Production  Production quantity levels
              LOWER     LEVEL     UPPER    MARGINAL

Food            .       24.942    +INF        .
NonFood         .       54.378    +INF        .
```

The single dot "." represents zeros; INF = infinity

Solutions

2. A display command

DISPLAY DemCommod.**L**, Production.**L,** Profit.**M**, Sigma **;**

```
----     307 VARIABLE  DemCommod.L  Commodity demand by households

                  Food       NonFood
NonFarmer        11.515       16.675
Farmer           13.428       37.704


----     307 VARIABLE  Production.L  Production quantity levels

Food     24.942,     NonFood 54.378


----     307 EQUATION  Profit.M  Zero profit condition

Food     24.942,     NonFood 54.378


----     307 PARAMETER Sigma  elasticity of factor substitution

Food     2.000,     NonFood 0.500
```

for a variable using .L

for an equation using .M

for a parameter, nothing

Solutions

You can also control precision in displays

OPTION DECIMALS = 0 ;

DISPLAY DemCommod **.L,** Production**.L ;**

```
----     309 VARIABLE  DemCommod.L  Commodity demand by households
                      Food      NonFood

NonFarmer               12           17
Farmer                  13           38

----     309 VARIABLE  Production.L  Production quantity levels
Food     25,     NonFood 54
```

OPTION DECIMALS = 2 **;**

```
----     309 VARIABLE  DemCommod.L  Commodity demand by households

                      Food      NonFood

NonFarmer          11.51        16.67
Farmer             13.43        37.70
```

Solutions

You can compute reports involving solution variable values

PARAMETER

ProdRev(Sector)   Producer revenues ;

ProdRev(Sector) = Production.L(Sector) * ComPrice.L(Sector) ;

DISPLAY ProdRev ;

```
----     352 PARAMETER ProdRev  Producer revenues

Food     34.90
NonFood  59.44
```

```
----     356 VARIABLE  ComPrice.L  Commodity price

Food     1.40
NonFood  1.09
```

```
----     354 VARIABLE  Production.L  Production quantity levels

Food     24.94
NonFood  54.38
```

Comparative Analysis

Two ways to conduct comparative analysis

1. **Use multiple GAMS submissions or multiple solves generating report writing output and then manually compare the analysis results**

2. **Use the GAMS LOOP procedure and set up a comparative scenario analysis system that creates cross scenario comparison tables**

Comparative Analysis

1. Use multiple GAMS submissions

```
PARAMETER
TaxRate(Factor,Sector)  Tax rates affect factor prices   ;
TaxRate(Factor,Sector) = 0 ;
OPTION MCP = PATH ;
SOLVE CGEModel USING MCP ;
```

```
TaxRate(“Capital","Food") =  0.5  ;
SOLVE CGEModel USING MCP ;
```

The model is first solved at the original TaxRate 0.

Then the TaxRate is changed to equal 0.5 and model is solved again with the altered TaxRate in effect doing a comparative static analysis of solution sensitivity to TaxRate.

Comparative Analysis

Report writing commands always use values from the most recent solution, so one must save the data if comparative reports are desired by creating parameter to store the report data.

```
SOLVE CGEModel USING MCP ;
PARAMETER Compare(Households,Sector,* )  ;
Compare(Households,Sector,“NoTax")
  =  DemCommod.L(Households,Sector) ;


TaxRate(“Capital","Food") =  0.1   ;
SOLVE CGEModel USING MCP ;
Compare(Households,Sector,"Tax10%")
  = DemCommod.L(Households,Sector) ;


TaxRate(“Capital","Food") =  0.5   ;
SOLVE CGEModel USING MCP ;
Compare(Households,Sector,"Tax50%")
  = DemCommod.L(Households,Sector) ;
```

Comparative Analysis

DISPLAY Compare;

---- 754 PARAMETER Compare consumption

	NoTax	Tax10%	Tax50%
NonFarmer.Food	11.51	10.83	8.99
NonFarmer.NonFood	16.67	16.47	15.83
Farmer .Food	13.43	13.46	13.40
Farmer .NonFood	37.70	38.72	41.48

Comparative Analysis

2. Use the GAMS LOOP procedure

The code contains a **LOOP** which causes GAMS to repeat execution of statement enclosed in the parentheses defining the **LOOP** .

```
LOOP( Scenario,
    TaxRate(Factor,Sector)  =  sTaxRate(Factor,Sector) ;
    TaxRate(Factor,Sector)  =  scenTax(Factor,Sector,Scenario)   ;
SOLVE CGEModel USING MCP ;
    Compare("TaxRate",Factor,Sector,Scenario)
           = TaxRate(Factor,Sector) ;
    Compare("Consumption",Households,Sector,Scenario)
           = DemCommod.L(Households,Sector) ;
    OPTION  Compare:2:3:1; DISPLAY Compare;
 ) ;
```

Comparative Analysis

2. Use the GAMS LOOP procedure (Con't)

```
SET  Scenario   /  NoTax    NoTax
                   Tax10    "10% Tax on Factor"
                   Tax50    "50% Tax on Factor"        / ;

TABLE ScenTax(Factor,Sector,Scenarios)
                   NoTax   Tax10    Tax50
Labor.Food           0      0.0      0.0
Labor.NonFood        0      0.0      0.0
Capital.Food         0      0.1      0.5
Capital.NonFood      0      0.0      0.0   ;

PARAMETER
   Compare(*,*,*,*)    Saving comparative report
   sTaxRate(Factor,Sector)  save tax rate               ;

   sTaxRate(Factor,Sector)  =  TaxRate(Factor,Sector) ;
```

Comparative Analysis

DISPLAY Compare;

```
----    352 PARAMETER Compare  Saving comparative report

                                      NoTax    Tax10%   Tax50%

TaxRate      .Capital    .Food                   0.10     0.50
Consumption.NonFarmer.Food            11.51    10.83     8.99
Consumption.NonFarmer.NonFood         16.67    16.47    15.83
Consumption.Farmer      .Food         13.43    13.46    13.40
Consumption.Farmer      .NonFood      37.70    38.72    41.48
```

Comparative Analysis

Advantage of using the GAMS LOOP procedure

```
SET  Scenario   /  NoTax    NoTax
                   Tax10    "10% Tax on Factor"
                   Tax50    "50% Tax on Factor"
                   Tax70    "70% Tax on Factor"
                   Tax80    "80% Tax on Factor"
                   Tax100   "100% Tax on Factor"          / ;

TABLE ScenTax(Factor,Sector,Scenario)
                      NoTax    Tax10   Tax50   Tax70    Tax80   Tax100
Labor    .Food          0        0.0     0.0     0        0.0     0.0
Labor    .NonFood       0        0.0     0.0     0        0.0     0.0
Capital  .Food          0        0.1     0.5     0.7      0.8     1.0
Capital  .NonFood       0        0.0     0.0     0        0.0     0.0   ;
```

Comparative Analysis

---- 352 PARAMETER Compare Saving comparative report

	NoTax	Tax10	Tax50	Tax70	Tax80	Tax100
TaxRate.Capital.Food	0	0.10	0.50	0.70	0.80	1.00
Consumption.NonFarmer.Food	11.51	10.83	8.99	8.39	8.13	7.70
Consumption. NonFarmer.NonFood	16.67	16.47	15.83	15.58	15.47	15.27
Consumption. Farmer .Food	13.43	13.46	13.40	13.32	13.28	13.20
Consumption. Farmer .NonFood	37.70	38.72	41.48	42.37	42.74	43.35

Wrap Up

- Casting CGE via GAMS

  1. Sets definition & data entry
  2. Variables & equation specifications
  3. Model complementarity relationship
  4. Solution reports
  5. Comparative analysis

Next:

- Hierarchical (nested) function & functional forms
- Social Accounting Matrices
- Input-output table
- Building benchmark equilibrium data sets
- Parameters calibration

References

Brooke, A., D. Kendrick, and A. Meeraus. “GAMS: A User’s Guide”.

Ferris, M. C. and J. S. Pang. “Engineering and Economic Applications of Complementarity Problems.” *SIAM Review*, 39:669-713, 1997.

McCarl, B. A. and D. Gillig. “Notes on Formulating and Solving Computable General Equilibrium Models within GAMS.”

Shoven, J. B. and J. Whalley. “Applying general equilibrium.” *Surveys of Economic Literature*, Chapters 3 and 4, 1998.

Shoven, J. B. and J. Whalley. “Applied General-Equilibrium Models of Taxation and International Trade: An Introduction and Survey.” *J. Economic Literature*, 22:1007-1051, 1984.