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: \( P \geq P_d = 6 - 0.3*Q_d \)
- Supply: \( P \leq P_s = 1 + 0.2*Q_s \)
- Equilibrium: \( Q_s \geq Q_d \) and \( P, Q_s, Q_d \geq 0 \)

2 commodities: corn and wheat

Corn Demand: \( P_c \geq P_{dc} = 6 - 0.3*Q_{dc} - 0.1*Q_{dw} \)
Wheat Demand: \( P_w \geq P_{dw} = 8 - 0.07*Q_{dc} - 0.4*Q_{dw} \)
Corn Supply: \( P_c \leq P_{sc} = 1 + 0.5*Q_{sc} + 0.1*Q_{sw} \)
Wheat Supply: \( P_w \leq P_{sw} = 2 + 0.1*Q_{sc} + 0.3*Q_{sw} \)
Corn Equilibrium: \( Q_{sc} \geq Q_{dc} \)
Wheat Equilibrium: \( Q_{sw} \geq Q_{dw} \)

\( P_c, P_w, Q_{dc}, Q_{dw}, Q_{sc}, \) and \( Q_{sw} \geq 0 \)
Formulation of a Simple Market Clearing

Set Definition & Data Entry

\textbf{SET}  Commodity \text{es} \text{ used in the model} /\text{Corn, Wheat}/ ;
\textbf{SET}  Curvetype \text{ supply and demand intercept and slope} /\text{Supply, Demand}/ ;

\textbf{TABLE} intercepts(Curvetype, Commodity) \text{ supply and demand intercept terms}

\begin{tabular}{ll}
Corn & Wheat \\
\text{demand} & 6 & 8 \\
\text{supply} & 1 & 2 \\
\end{tabular}

\textbf{TABLE} Slopes(Curvetype, Commodity, Commodity) \text{ supply and demand slope terms}

\begin{tabular}{ll}
Corn & Wheat \\
\text{Demand. Corn} & -.3 & -.1 \\
\text{Demand. Wheat} & -.07 & -.4 \\
\text{Supply. Corn} & .5 & .1 \\
\text{Supply. Wheat} & .1 & .3 \\
\end{tabular} ;
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.

\[
\text{SET ItemName optional explanatory text for item} \\
/ element1 optional explanatory text for element, \\
\text{element2 optional explanatory text for element} /; \\
\]

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

<table>
<thead>
<tr>
<th>Corn</th>
<th>Wheat</th>
<th>Elements from Commodities set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Demand</td>
<td>Supply</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>(2nd set)</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>;</td>
</tr>
</tbody>
</table>

Elements from Curvetype set (1st set)
## Data Entry – TABLE commands

More than two dimensional data entry using **TABLE**

<table>
<thead>
<tr>
<th>Define table name</th>
<th>Associated sets</th>
<th>Text comments, Optional in command</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TABLE</strong> Slopes(Curvtetype,Commodities,Commodities)</td>
<td>supply and demand slope</td>
<td>Elements from <strong>Commodities</strong> set (3rd set)</td>
</tr>
<tr>
<td>Demand . Corn</td>
<td>-0.3</td>
<td>-0.1</td>
</tr>
<tr>
<td>Demand . Wheat</td>
<td>-0.07</td>
<td>-0.4</td>
</tr>
<tr>
<td>Supply . Corn</td>
<td>0.5</td>
<td>0.1</td>
</tr>
<tr>
<td>Supply . Wheat</td>
<td>0.1</td>
<td>0.3</td>
</tr>
</tbody>
</table>

Elements from **Curvtetype** set (1st set)  
Elements from **Commodities** set (2nd set)
Formulation – Variable Declarations

Basic format:

```plaintext
VARIALBE VarName1(setdependency) optional text
   VarName2(setdependency) optional text
   ...

... ;
```

to declare variables $< \text{ or } > 0$

Or

```plaintext
POSITIVE VARIABLE
   VarName1(setdependency) optional text
   VarName2(setdependency) optional text
   ...

... ;
```

To declare $\geq 0$ variables
Formulation – Variable Declarations

In our example:

POSITIVE VARIABLES

\begin{align*}
P(\text{Commodities}) & \quad \text{Equilibrium price} \\
Q_d(\text{Commodities}) & \quad \text{Quantity demanded} \\
Q_s(\text{Commodities}) & \quad \text{Quantity supply} \\
\end{align*}

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

\[ P_c, P_w, Q_{d,c}, Q_{d,w}, Q_{s,c}, Q_{s,w} \geq 0 \]
Formulation – Equation Declarations

Basic format:

Equation   EqName1(setdependency) optional text
           EqName2(setdependency) optional text

... ;
Formulation – Equation Declarations

In our example:

\textbf{EQUATIONS}

\begin{align*}
\text{PDemand}(\text{Commodities}) & \quad \text{Demand equation} \\
\text{PSupply}(\text{Commodities}) & \quad \text{Supply equation} \\
\text{Equilibrium}(\text{Commodities}) & \quad \text{Equilibrium equation} \\
\end{align*}

Note that this defines an equation for each case in the set commodities
Formulation – Equation Specifications

General Structure:

\[
\text{DeclaredEquationName} (\text{SetDependency}).. \text{LHSalgebra} \text{ EquationRelationType} \text{RHSalgebra} ;
\]

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:**
  \[
  P_c \geq P_{dc} = 6 - 0.3*Q_{dc} - 0.1*Q_{dw}
  \]
  \[
  P_w \geq P_{dw} = 8 - 0.07*Q_{dc} - 0.4*Q_{dw}
  \]

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

\[ \sum_j X(j) \]

\[ \sum \sum X(\text{ji}) \]

\[ \text{SUM}(\text{index1}, \text{SUM}(\text{index2}, \text{names(\text{index1},\text{index2})})) \]

\[ \text{SUM}(j, \text{SUM}(i, X(j,i))) \]

or \[ \text{SUM}((j,i), X(j,i)) \]
**Formulation – Equation Specifications**

**Algebraic Structure**

- **Supply:**
  
  \[
  P_{s_c} = 1 + 0.5Q_{s_c} + 0.1Q_{s_w} \geq P_c \\
  P_{s_w} = 2 + 0.1Q_{s_c} + 0.3Q_{s_w} \geq P_w
  \]

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

Algebraic Structure

- Equilibrium: \( Q_{sc} \geq Q_{dc} \)
  \( Q_{sw} \geq Q_{dw} \)

Equilibrium (commodities).
- \( Q_s \) (commodities)
  \( = G = \)
  \( Q_d \) (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

<table>
<thead>
<tr>
<th></th>
<th>EQU PDemand</th>
<th>Demand equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOWER</td>
<td>LEVEL</td>
<td>UPPER</td>
</tr>
<tr>
<td>Corn</td>
<td>6.000</td>
<td>6.000</td>
</tr>
<tr>
<td>Wheat</td>
<td>8.000</td>
<td>8.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>EQU PSupply</th>
<th>Supply equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOWER</td>
<td>LEVEL</td>
<td>UPPER</td>
</tr>
<tr>
<td>Corn</td>
<td>-1.000</td>
<td>-1.000</td>
</tr>
<tr>
<td>Wheat</td>
<td>-2.000</td>
<td>-2.000</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>EQU Equilibrium</th>
<th>Equilibrium equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOWER</td>
<td>LEVEL</td>
<td>UPPER</td>
</tr>
<tr>
<td>Corn</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>Wheat</td>
<td>.</td>
<td>.</td>
</tr>
</tbody>
</table>
Solution

<table>
<thead>
<tr>
<th>Lower</th>
<th>Level</th>
<th>Upper</th>
<th>Marginal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corn</td>
<td>3.937</td>
<td>+INF</td>
<td>.</td>
</tr>
<tr>
<td>Wheat</td>
<td>4.690</td>
<td>+INF</td>
<td>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lower</th>
<th>Level</th>
<th>Upper</th>
<th>Marginal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corn</td>
<td>4.373</td>
<td>+INF</td>
<td>.</td>
</tr>
<tr>
<td>Wheat</td>
<td>7.510</td>
<td>+INF</td>
<td>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lower</th>
<th>Level</th>
<th>Upper</th>
<th>Marginal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corn</td>
<td>4.373</td>
<td>+INF</td>
<td>.</td>
</tr>
<tr>
<td>Wheat</td>
<td>7.510</td>
<td>+INF</td>
<td>.</td>
</tr>
</tbody>
</table>
Casting CGE in GAMS

Let's 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** \( \text{Alpha} \left( \text{Sector, HouseHolds} \right) \) Consumption share

<table>
<thead>
<tr>
<th></th>
<th>NonFarmer</th>
<th>Farmer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Food</strong></td>
<td>0.5</td>
<td>0.3</td>
</tr>
<tr>
<td><strong>NonFood</strong></td>
<td>0.5</td>
<td>0.7</td>
</tr>
</tbody>
</table>

**TABLE** \( \text{Endowment} \left( \text{Factor, HouseHolds} \right) \) Factor Endow

<table>
<thead>
<tr>
<th></th>
<th>NonFarmer</th>
<th>Farmer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Labor</strong></td>
<td>0</td>
<td>60</td>
</tr>
<tr>
<td><strong>Capital</strong></td>
<td>25</td>
<td>0</td>
</tr>
</tbody>
</table>
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(\text{Factor}) \quad \text{Prices for factors}

FactorQuan(\text{Factor, Sector}) \quad \text{Factors used by a sector}

ComPrice(\text{Sector}) \quad \text{Prices of commodities}

DemCommod(\text{Households, Sector}) \quad \text{Demand by household}

Production(\text{Sector}) \quad \text{Production quantity level}

HHIncome(\text{Households}) \quad \text{Household income}

TaxRevenue \quad \text{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

;
1. Supply-Demand identities for each factor market

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

\[ \sum_{j} L_j \leq \sum_{h} L_h \]
\[ \sum_{j} K_j \leq \sum_{h} K_h \]

\[ \Rightarrow \quad \sum_{j} F_{fj} \leq \sum_{h} F_{fh} \]

\text{FactorMkt}(\text{Factor}).. \]
\text{SUM}(\text{Sector,FactorQuan}(\text{Factor,Sector}))
= \text{L}=
\text{SUM}(\text{HouseHolds,Endowment}(\text{Factor,HouseHolds})) ;
2. Supply-Demand identities for each output market

\[
\text{CommodMkt}(\text{Sector}) ..
\]
\[
\sum \text{DemCommod}(\text{Households}, \text{Sector}) + \sum \text{IntermediateUse}(\text{Sector}, \text{OtherSector}) \times \text{Production}(\text{OtherSector}) + \text{GovernmentPurch}(\text{Sector}) \times \text{TaxRevenue}/\text{ComPrice}(\text{Sector}) = \text{Production}(\text{Sector}) ;
\]
\[ \sum_{h} x_{hj} + \sum_{j1} a_{j,j1} q_{j1} + s_{j} R / P_{j} \leq q_{j} \]

\text{CommodMkt(Sector)..}

\text{SUM(Households,DemCommod(Households,Sector))}
+ \text{SUM(OtherSector, IntermediateUse(Sector,OtherSector) * Production(OtherSector))}
+ \text{GovernmentPurch(Sector)*TaxRevenue/ComPrice(Sector)}
= \text{L= Production(Sector)} ;
3. Zero Profit Conditions

\[ \sum_{j_1} P_{j_1} a_{j_1,j} Q_j + \sum_f (1 + t_{fj}) W_f F_{fj} \geq P_j Q_j \]
\[
\sum_{j_1} P_{j_1} a_{j_1,j} Q_j + \sum_f (1 + t_{ff}) W_f F_{ff} \geq P_j Q_j
\]

\text{Profit(Sector)...}

+ \text{SUM(OtherSector, ComPrice(OtherSector) \times IntermediateUse(OtherSector,Sector) \times Production(Sector))}

+ \text{SUM(Factor, (1+TaxRate(Factor,Sector)) \times FactorPrice(Factor) \times FactorQuan(Factor,Sector))}

= G = \text{ComPrice(Sector) \times Production(Sector)} ;
4. Factor demand by producers

\[ F_{fj} = q_j \phi_j^{(\sigma_j-1)} \times (\delta_{fj} \sum_{f'}^{\sigma_j} \left( \delta_{fj}^{\sigma_j} (W_f'(1+t_{fj}))^{(1-\sigma_j)} \right)^{1/(1-\sigma_j)} / \phi_j ) / (W_f(1+t_{fj}))^{\sigma_j} \]

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

\[ X_{hj} = \frac{\alpha_{jh} \ Income_h}{\sum_{j'} P_j^{\sigma_{jh}} \left( \alpha_{j'h} (P_j)^{1-\sigma_{jh}} \right)} \]

CommodDem(Households, Sector) =
DemCommod(Households, Sector) =
\( \frac{(\text{Alpha}(\text{Sector}, \text{HouseHolds}) \times \text{HHIncome}(\text{HouseHolds}))}{(\text{ComPrice}(\text{Sector})^{\sigma_{C}})(\sum \text{Sector1}, \text{Alpha}(\text{Sector1}, \text{HouseHolds}) \times \text{ComPrice}(\text{Sector1})^{1-\sigma_{C}} )} \) ;
6. Household income constraint

\[
\text{HHIncome(Households)} \geq \text{Income from Factors} + \text{Income Tax Payments} + \text{Government Transfers}
\]

\[
\text{Income(Households)} = \sum (\text{Factor}, \text{Endowment(Factor,Households)} \times \text{FactorPrice(Factor)}) - \text{incometax(Households)} \times \sum (\text{Factor}, \text{Endowment(Factor,Households)} \times \text{FactorPrice(Factor)}) + \text{TaxShare(Households)} \times \text{TaxRevenue}
\]
\( \text{Income}_h \geq \sum_{f} F_{fj} W_f - t_h \sum_{f} F_{fj} W_f + s_h R \)

\[
\text{Income(Households)} = \sum \text{Endowment(Factor,Households)} \times \text{FactorPrice(Factor)}
\]

\[
- \text{incometax(Households)} \times \sum \text{Endowment(Factor,Households)} \times \text{FactorPrice(Factor)}
\]

\[
+ \text{TaxShare(Households)} \times \text{TaxRevenue}
\]
7. Government income constraint

\[ \text{GovBal..} \]

\[ \text{TaxRevenue} = \sum \text{Households, Incometax(Households)} \times \sum \text{Factor, Endowment(Factor, HouseHolds) * FactorPrice(Factor))} \]

\[ + \sum ((\text{Factor, Sector), TaxRate(Factor, Sector) * FactorPrice(Factor) * FactorQuan(Factor, Sector)}) ; \]
\[ R \leq \sum_{h} \left( t_h \sum_{f} \overline{F}_{fh} W_f \right) + \sum_{fj} t_{fj} W_f F_{fj} \]

GovBal..

\[ \text{TaxRevenue} = \text{L} = \text{SUM(Households, Incometax(Households) * SUM(Factor,Endowment(Factor,Households)* FactorPrice(Factor)))} \]

\[ + \text{SUM((Factor,Sector),TaxRate(Factor,Sector) * FactorPrice(Factor) *FactorQuan(Factor,Sector) ) ;} \]
Model Complementarity Relationship

MODEL CGEModel

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

<table>
<thead>
<tr>
<th>Equation Name</th>
<th>Variable Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>FactorMkt(Factor)</td>
<td>FactorPrice(Factor)</td>
</tr>
<tr>
<td>FactorDem(Factor,Sector)</td>
<td>FactorQuan(Factor,Sector)</td>
</tr>
<tr>
<td>CommodMkt(Sector)</td>
<td>ComPrice(Sector)</td>
</tr>
<tr>
<td>Profit(Sector)</td>
<td>Production(Sector)</td>
</tr>
<tr>
<td>Income(households)</td>
<td>HHIncome(households)</td>
</tr>
<tr>
<td>GovBal</td>
<td>TaxRevenue</td>
</tr>
<tr>
<td>CommodDem(Households,Sector)</td>
<td>DemCommod(Households,Sector)</td>
</tr>
</tbody>
</table>
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. \texttt{FactorPrice.FX("Labor")= 1;}

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

Solution example: \begin{tabular}{lcc}
\textbf{ } & \textbf{NoTax} & \textbf{Tax} \\
\hline
\textbf{Labor} & 1.00 & 1.00 \\
\textbf{Capital} & 1.37 & 1.13 \\
\textbf{Food} & 1.40 & 1.47 \\
\textbf{NonFood} & 1.09 & 1.01 \\
\end{tabular}
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.

\[
\text{FactorPrice.L(Factor)} = 1; \\
\text{FactorPrice.LO(Factor)} = 0.0001;
\]

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”.

SOLVE SUMMARY

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

```plaintext
----  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
2. A display command

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

<table>
<thead>
<tr>
<th></th>
<th>Food</th>
<th>NonFood</th>
</tr>
</thead>
<tbody>
<tr>
<td>NonFarmer</td>
<td>11.515</td>
<td>16.675</td>
</tr>
<tr>
<td>Farmer</td>
<td>13.428</td>
<td>37.704</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Food</th>
<th>NonFood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>24.942</td>
<td>54.378</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Food</th>
<th>NonFood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>24.942</td>
<td>54.378</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Food</th>
<th>NonFood</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>2.000</td>
<td>0.500</td>
</tr>
</tbody>
</table>
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 a parameter to store the report data.

```plaintext
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

```plaintext
DISPLAY Compare;

---- 754 PARAMETER  Compare  consumption

<table>
<thead>
<tr>
<th></th>
<th>NoTax</th>
<th>Tax10%</th>
<th>Tax50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>NonFarmer.Food</td>
<td>11.51</td>
<td>10.83</td>
<td>8.99</td>
</tr>
<tr>
<td>NonFarmer.NonFood</td>
<td>16.67</td>
<td>16.47</td>
<td>15.83</td>
</tr>
<tr>
<td>Farmer.Food</td>
<td>13.43</td>
<td>13.46</td>
<td>13.40</td>
</tr>
<tr>
<td>Farmer.NonFood</td>
<td>37.70</td>
<td>38.72</td>
<td>41.48</td>
</tr>
</tbody>
</table>
```
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)

<table>
<thead>
<tr>
<th></th>
<th>NoTax</th>
<th>Tax10</th>
<th>Tax50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labor.Food</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Labor.NonFood</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Capital.Food</td>
<td>0</td>
<td>0.1</td>
<td>0.5</td>
</tr>
<tr>
<td>Capital.NonFood</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>TaxRate</th>
<th>.Capital</th>
<th>.Food</th>
<th>NoTax</th>
<th>Tax10%</th>
<th>Tax50%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consumption</td>
<td>NonFarmer. Food</td>
<td></td>
<td>11.51</td>
<td>10.83</td>
<td>8.99</td>
</tr>
<tr>
<td>Consumption</td>
<td>NonFarmer. NonFood</td>
<td></td>
<td>16.67</td>
<td>16.47</td>
<td>15.83</td>
</tr>
<tr>
<td>Consumption</td>
<td>Farmer. Food</td>
<td></td>
<td>13.43</td>
<td>13.46</td>
<td>13.40</td>
</tr>
<tr>
<td>Consumption</td>
<td>Farmer. NonFood</td>
<td></td>
<td>37.70</td>
<td>38.72</td>
<td>41.48</td>
</tr>
</tbody>
</table>
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)

<table>
<thead>
<tr>
<th></th>
<th>NoTax</th>
<th>Tax10</th>
<th>Tax50</th>
<th>Tax70</th>
<th>Tax80</th>
<th>Tax100</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labor  .Food</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Labor  .NonFood</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>Capital .Food</td>
<td>0</td>
<td>0.1</td>
<td>0.5</td>
<td>0.7</td>
<td>0.8</td>
<td>1.0</td>
</tr>
<tr>
<td>Capital .NonFood</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
<td>0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
## Comparative Analysis

---

352 PARAMETER Compare Saving comparative report

<table>
<thead>
<tr>
<th></th>
<th>NoTax</th>
<th>Tax10</th>
<th>Tax50</th>
<th>Tax70</th>
<th>Tax80</th>
<th>Tax100</th>
</tr>
</thead>
<tbody>
<tr>
<td>TaxRate.Capital.Food</td>
<td>0</td>
<td>0.10</td>
<td>0.50</td>
<td>0.70</td>
<td>0.80</td>
<td>1.00</td>
</tr>
<tr>
<td>Consumption. Farmer .NonFood</td>
<td>37.70</td>
<td>38.72</td>
<td>41.48</td>
<td>42.37</td>
<td>42.74</td>
<td>43.35</td>
</tr>
</tbody>
</table>
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


