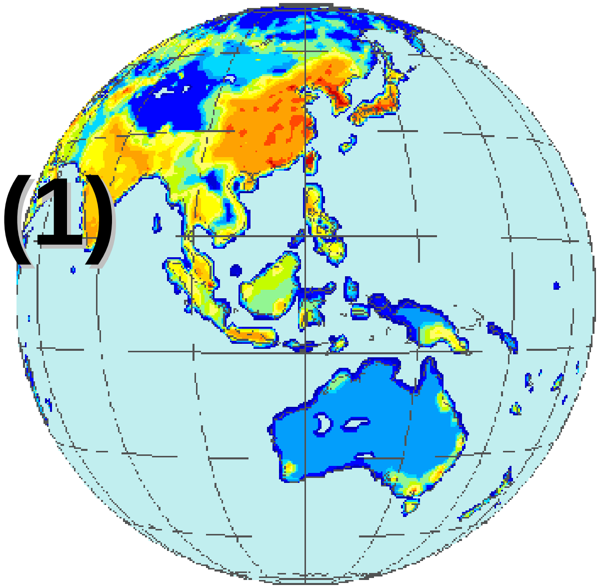


CGE model development (1)

Concept of CGE model and simple CGE model based on IO data

Toshihiko MASUI

National Institute for Environmental Studies

AIM Training Workshop 2005

NIES, 7-11 November 2005

What's "Model"?

- Model represents a specific aspect of real world.
  - When we develop a model, we must understand objectives.
  - We can simulate the future in advance by using model.

- The representation in model is not real world but ideal world.
  - We must take into account difference between actual world and modeled model.

Model for environmental policies

- Not only economic activity but also environment will be taken into account.
- What's the relationship between environment and economy?
  - Provision of services and goods
  - Assimilation of pollutants
  - Degradation of environmental quality
  - Maintenance of environment
- What is key option to protect the environment?
  - Technology: more efficient, renewable energy, ...
  - Institution: tax, regulation, ...
  - Management: operation, skill, ...
- By using model, effectiveness of environmental options can be assessed in advance.

What's CGE?

- "**Computable**": quantitative
- "**General**": treatment of all commodities, sectors and production factors in the treated society
- "**Equilibrium**": demand and supply of each commodity and factor are balanced through the price mechanism

Features of CGE

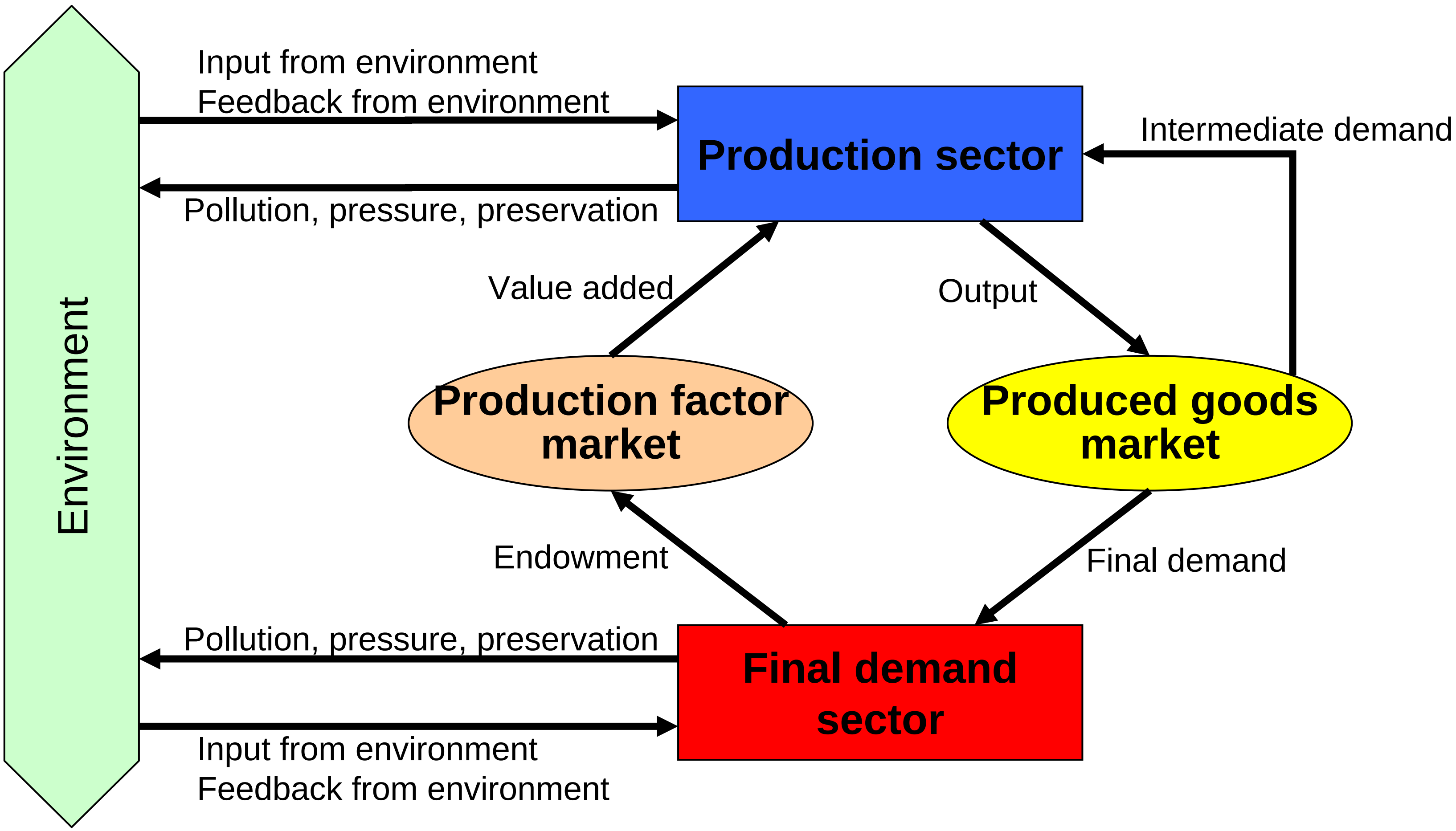
- Multiple interacting agents.
- Individual behavior based on optimization.
- Most agent interactions are mediated by market and prices.
- Typically disaggregate, with many agents and markets.
- Limited data in comparison with the number of behavioral and technological parameters in the model.
- Equilibrium allocations which typically cannot be characterized as the solution to a single (planner's) optimization problem.
- Formulation has as implicit or explicit focus on policy analysis.

By using CGE, detailed impacts of policy on price, activity, income and so on can be simulated in advance.

Procedure of CGE model development

1. Design rough model structure
   - Relationship among production sector, final demand, commodity & environment
2. Define elements
   - Classification of production sector, final demand, commodity, ...
3. Design detailed model structure
   - Commodity flow, function, elasticity of substitution, ...
4. Quantify data
   - Parameters setting
5. Formulate model (programming)
6. Simulate model
   - Replication of benchmark
   - Quantification of policy simulation

1. Rough sketch of model



2. Definition of CGE model

Simple example:

Based on IO table, model with 2 commodities, 2 sectors & 1 final demand is developed.

- Commodity: not only goods & service but also production factor & hypothetical commodity
- Sector: production sector & hypothetical sector
  - input (demand) commodities
  - output (supply) commodities

  Maximizing profit subject to production function
- Final demand:
  - supply endowments and get income
  - demand commodities

  Maximizing utility subject to income.

Commodity

In simple example

- Produced commodity
  - commodity 1 (PY("com1"))
  - commodity 2 (PY("com2"))
- Production factor
  - capital (PK)
  - labor (PL)
- Hypothetical commodity
  - aggregated final consumption goods (PC)
  - aggregated investment goods (PI)

Sector

In simple example

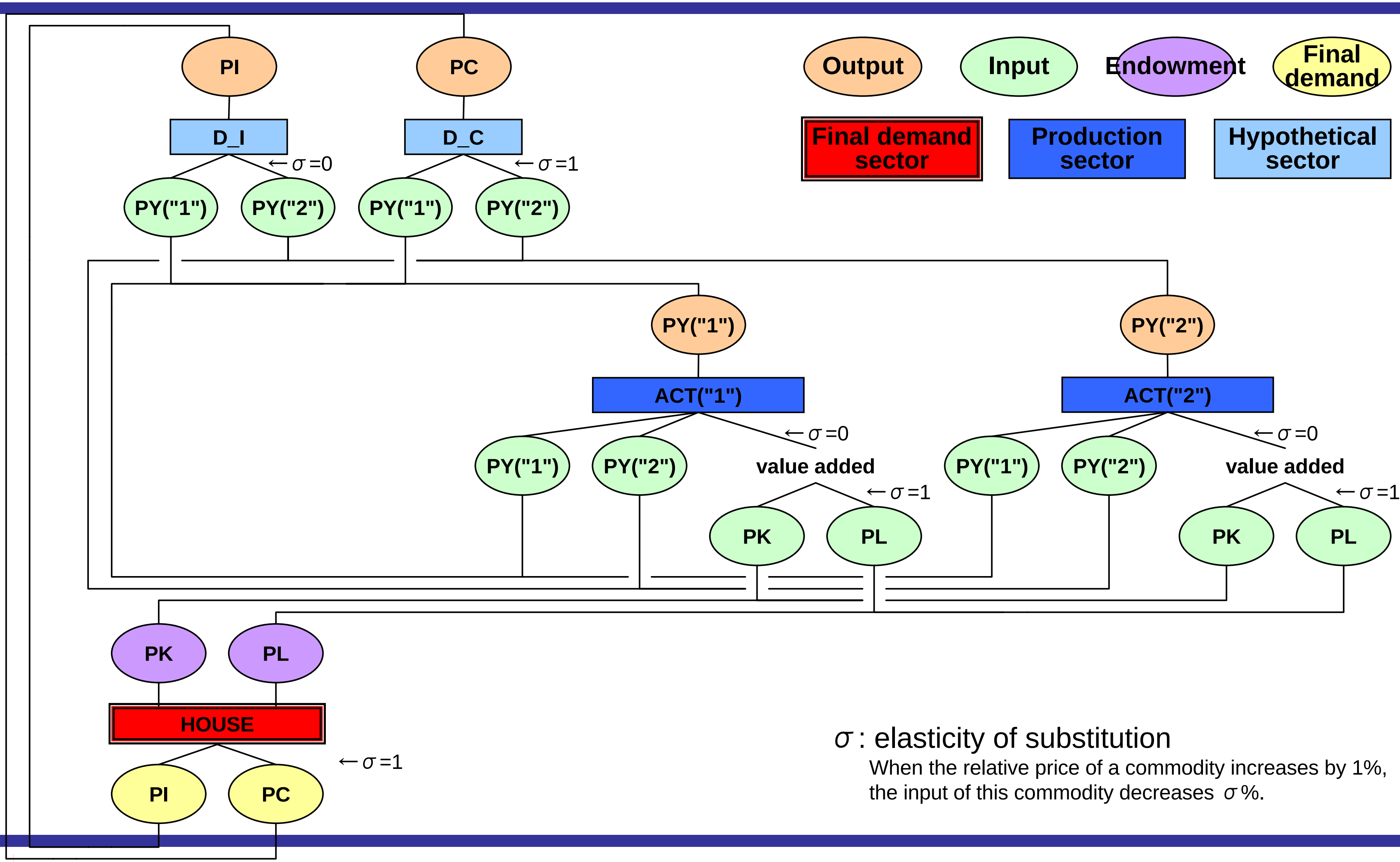
- Production sector
  - sector producing commodity 1
  - sector producing commodity 2
    - Input: com1, com2, CAP & LAB
    - Output: com1 or com2
- Hypothetical sector
  - aggregation of final consumption goods
  - aggregation of investment goods

Final demand

In simple example

- Endowment
  - Capital
  - Labor
- Final demand
  - Final consumption
  - Investment (fixed capital formation) = saving

3. Detailed model structure



σ : elasticity of substitution

When the relative price of a commodity increases by 1%, the input of this commodity decreases σ%.

In order to check consistency, draw the diagram like this slide!

4. Quantify data (IO table)

		Commodity		Final demand		
		com1	com2	con	inv	total
Commodity	com1	80	20	80	20	200
	com2	40	100	40	40	220
Value added	cap	30	60			
	lab	50	40			
	total	200	220			

Total demand = Total supply

Total cost = Total sale

Total in column = Total in row

5. Programming

- Based on GAMS/MPSGE format.
- Solution by MCP (Mixed Complementarity Problem)

$$P_i * f_i(P_i)=0,\ P_i \geq 0,\ f_i(P_i) \geq 0$$

P_i: *price*

$f_i(P_i)$: *excess supply*

When demand equal supply ($f_i(P_i)=0$), price is positive ($P_i > 0$).

When supply exceeds demand ($f_i(P_i)>0$), price is 0 ($P_i=0$).

– Optimization model is converted to simultaneous equations.

- See manual for installation of GAMS.

5. Programming

```
set
com          commodity    /com1, com2/
v_a          value added  /cap,  lab/
;

alias (com,c_m) ;

Table  IO(*,*) input output table
             com1    com2    con    inv
com1           80      20     80     20
com2           40     100     40     40
cap            30      60
lab            50      40
;
```

```
scalar
tot_c        total consumption
tot_i        total investment
tot_k        total capital
tot_l        total labor
;

tot_c        = sum(c_m, IO(c_m,"con")) ;
tot_i        = sum(c_m, IO(c_m,"inv")) ;
tot_k        = sum(com, IO("cap",com)) ;
tot_l        = sum(com, IO("lab",com)) ;

parameter
out(com)  total output
;

out(com)     = sum(c_m, IO(c_m,com))
             + sum(v_a, IO(v_a,com)) ;
```

5. Programming

Definition of commodity type as a set.

```
set
com            commodity    /com1, com2/
v_a            value added  /cap,  lab/
;
```

Copy of set.

```
alias (com,c_m) ;
```

```
Table  IO(*,*) input output table
               com1    com2    con    inv
com1             80      20     80     20
com2             40     100     40     40
cap              30      60
lab              50      40
;
```

Dataset by using table format.
Here, input-output table is indicated.

Definition of scalar.

```
scalar
tot_c          total consumption
tot_i          total investment
tot_k          total capital
tot_l          total labor
;
```

Quantification of defined scalar.

```
tot_c          = sum(c_m, IO(c_m,"con")) ;
tot_i          = sum(c_m, IO(c_m,"inv")) ;
tot_k          = sum(com, IO("cap",com)) ;
tot_l          = sum(com, IO("lab",com)) ;
```

Definition of parameter.

```
parameter
out(com)  total output
;

out(com)       = sum(c_m, IO(c_m,com))
               + sum(v_a, IO(v_a,com)) ;
```

Quantification of defined parameter.

5. Programming

```
$ontext
$model:sample
$sectors:
        ACT(com)      ! production
        D_C           ! final consumption
        D_I           ! fixed capital formation

$commodities:
        PY(com)       ! commodity
        PK            ! capital
        PL            ! labor
        PC            ! final consumption
        PI            ! investment

$consumers:
        HOUSE         ! household


$prod:ACT(com)      t:0   s:0   va:1
        O:PY(com) Q:OUT(com)
        I:PY(c_m)   Q:IO(c_m,com)
        I:PK        Q:IO("cap",com)       va:
        I:PL        Q:IO("lab",com)       va:

$prod:D_C                   s:1
        O:PC                Q:tot_c
        I:PY(c_m)           Q:IO(c_m,"con")

$prod:D_I                   s:0
        O:PI                Q:tot_i
        I:PY(c_m)           Q:IO(c_m,"inv")

$demand:HOUSE               s:1
        D:PC                Q:tot_c
        D:PI                Q:tot_i
        E:PL                Q:tot_l
        E:PK                Q:tot_k

$report:
        V:ACTPK(com)        I:PK  prod:ACT(com)
        V:ACTPL(com)        I:PL  prod:ACT(com)
$offtext

$SYSINCLUDE MPSGESET SAMPLE
$INCLUDE SAMPLE.GEN
SOLVE SAMPLE USING MCP;
```

5. Programming

```
$ontext
$model:sample
$sectors:
        ACT(com)      ! production
        D_C           ! final consumption
        D_I           !

$commodities:
        PY(com
        PK            ! capital
        PL            ! labor
        PC            ! final consumption
        PI            ! investment

$consumers:
        HOUSE

$prod:ACT(com)      t:0   s:0   va:1
        O:PY(com)  Q:OUT(com)
        I:PY(c_m)  Q:IO(c_m,com)
        I:PK       Q:IO("cap",com)      va:
        I:PL       Q:IO("lab",com)      va:
```

Sign of start of formulation.

Definition of model name.

Definition of sector.

Words after "!" show comment

Definition of commodity.

Definition of final demand.

Definition of activity in sector.

```
$prod:D_C                   s:1
        O:PC                Q:tot_c
        I:PY(c_m)           Q:IO(c_m,"con")

$prod:D_I                   s:0
        O:PI                Q:tot_i
        I:PY(c_m)           Q:IO(c_m,"inv")

$demand:HOUSE               s:1
        D:PC                Q:tot_c
        D:PI                Q:tot_i
        E:PL                Q:tot_l
        E:PK

$report:
        V:ACTPK(com)        I:PK  prod:ACT(com)
        V:ACTPL(com)        I:PL  prod:ACT(com)
$offtext

$SYSINCLUDE MPSGESET SAMPLE
$INCLUDE SAMPLE.GEN
SOLVE SAMPLE USING MCP;
```

Definition of activity in final demand.

Definition of quantified element.

Sign of end of formulation.

Sign of preparation of simulation and solution of this model.

5. Programming

name of activity
Value represents activity level.
Default is one.

elasticity of transformation

elasticity of substitution

elasticity of substitution among specific inputs

```
$prod:ACT(com)          t:0   s:0   va:1
      O:PY(com)         Q:OUT(com)          P:1
      I:PY(c_m)         Q:IO(c_m,com)       P:1
      I:PK              Q:IO("cap",com)     P:1     va:
      I:PL              Q:IO("lab",com)     P:1     va:
```

O: shows output, and
I: shows input.

name of commodity
Value represents price.
Default is one.

Q: shows reference quantity of input/output when the activity level is one.
Default is one.

P: shows reference price of commodity.
Default is one.

P X Q shows cost of input or sale of output.
Based on these values, share parameters are defined.

6. Simulation

1. Replication of benchmark
2. Sensitivity analysis to check parameters
3. Scenario and policy design
4. Simulation based on scenario
5. Analysis of results
6. Assessment of alternative scenarios and policies

How to apply CGE

1. Translate policy into the model instruments.
2. Guess at the policy results.
3. Run the simulation and compare results.
4. Compare the model results with your earlier guess.
5. Evaluate the outcome and write up your key findings.
6. Develop sensitivity analyses.
7. Write up the model.