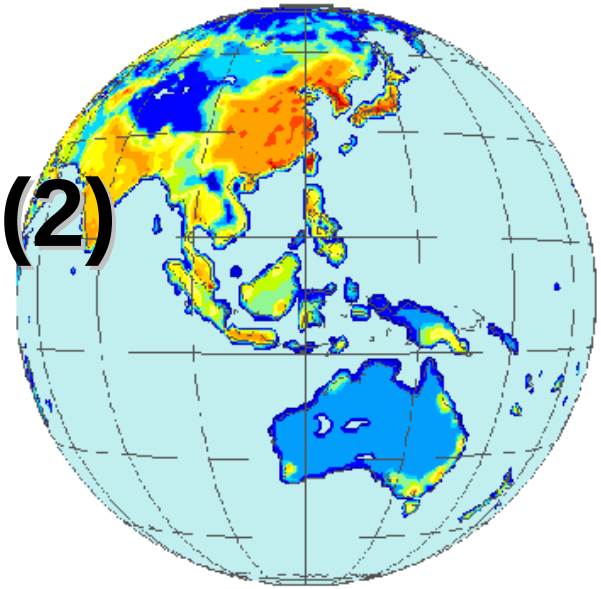
CGE model development (2)

CGE model development based on U&V matrix



Toshihiko MASUI

National Institute for Environmental Studies

AIM Training Workshop 2005

NIES, 7-11 November 2005

From IO to U&V

- IO table shows relationship between commodity and commodity.
  - IO table represents "In order to make a commodity, which commodity and how much commodity is needed."
  - Negative inputs except import and subsidy can be seen.
    - by-products
    - wastes generation
  - For energy analysis, differentiation between sector and commodity is convenient.
    - joint product: oil refinery sector produces gasoline, diesel oil, heavy oil and so on from crude oil.
    - power sector: Hydro power, thermal power and so on produce electricity.
- IO table is converted to U matrix and V matrix.

relationship between IO and U&V in SAM (Social Account Matrix)

		expenditure								
		activity	commodity	factor	enterprise	household	government	capital account	rest of the world	total
receipt	activity		gross output							total sale
	commodity	intermediate demand				household consumption	government consumption	investment	export	aggregate demand
	factor	value added							factor service export	factor income
	enterprise			gross profit			transfer			enterprise income
	household			wage	distributed profits		transfer		foreign remittance	household income
	government	indirect tax	tariff	factor tax	enterprise taxes	direct taxes				government income
	capital account				retained earnings	households savings	government saving		capital transfer	total saving
	rest of the world		import	factor service import		transfers abroad	transfers abroad	capital transfer		foreign payment
	total	total cost	aggregate supply	factor expenditure	enterprise expenditure	household expenditure	government expenditure	total investment	foreign receipt	

relationship between IO and U&V in SAM (Social Account Matrix)

		expenditure								
		activity	commodity	factor	enterprise	household	government	capital account	rest of the world	total
receipt	activity		V matrix							total sale
	commodity	U matrix	IO table			household consumption	government consumption	investment	export	aggregate demand
	factor								factor service export	factor income
	enterprise			gross profit			transfer			enterprise income
	household			wage	distributed profits		transfer		foreign remittance	household income
	government	indirect tax	tariff	factor tax	enterprise taxes	direct taxes				government income
	capital account				retained earnings	households savings	government saving		capital transfer	total saving
	rest of the world		import	factor service import		transfers abroad	transfers abroad	capital transfer		foreign payment
	total	total cost	aggregate supply	factor expenditure	enterprise expenditure	household expenditure	government expenditure	total investment	foreign receipt	

U matrix (Use matrix)

- commodity X sector
- U matrix represents inputs in activity, and supply & demand of commodity.

U matrix	sec A	sec B	con	inv	total
com 1	80	20	60	30	190
com 2	40	100	40	50	230
cap	30	60			
lab	50	40			
total	200	220			

Above table shows following information;

* Sector A demands 80 of commodity 1, 40 of commodity 2, 30 of capital and 50 of labor, and then sales 200 .
* 190 of commodity 1 are supplied, and 80 in sector A, 20 in sector B, 60 in consumption, and 30 in investment are demanded.

V matrix (make matrix)

- sector X commodity
- V matrix represents produced commodities from each sector.

V matrix	com 1	com 2	total
sec A	150	50	200
sec B	40	180	220
total	190	230	

Above table shows following information;
* Sector A produces 150 of commodity 1 and 50 of commodity 2.

Relation between U and V

U matrix	sec A	sec B	con	inv	total
com 1	80	20	60	30	190
com 2	40	100	40	50	230
cap	30	60			
lab	50	40			
total	200	220			

In U, row total is not equal to column total.

Sector total in U must be equal to sector total in V

V matrix	com 1	com 2	total
sec A	150	50	200
sec B	40	180	220
total	190	230	

Commodity total in U must be equal to commodity total in V

Programming

```
set
sec      sector              /sec1, sec2/
com      commodity           /com1, com2, com3/
v_a      value added         /cap,  lab/
;

Table  U(*,*)  use matrix (commodities x
       sectors)
          sec1     sec2      con      inv
com1       80       20       80       20
com2       40      100       40       40
com3       60       50       40        0
cap        30       70
lab        70       50
;

Table  V(sec,com)  make matrix (sectors x
       commodities)
          com1     com2     com3
sec1     160      120        0
sec2      40      100      150
;
```

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

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

Programming

<IO base program>

```
$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:
```

<U&V base program>

```
$ontext
$model:sample
$sectors:
      ACT(sec)  ! 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(sec)     t:0   s:0   va:1
      O:PY(com) Q:V(sec,com)
      I:PY(com)   Q:U(com,sec)
      I:PK        Q:U("cap",sec)       va:
      I:PL        Q:U("lab",sec)       va:
```

Programming

<IO base program>

```
$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:
```

<U&V base program>

```
$ontext
$model:sample
$sectors:
      ACT(sec)  ! 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(sec)     t:0   s:0   va:1
      O:PY(com) Q:V(sec,com)
      I:PY(com)   Q:U(com,sec)
      I:PK        Q:U("cap",sec)       va:
      I:PL        Q:U("lab",sec)       va:
```

Programming

<IO base program>

```
$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;
```

<U&V base program>

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

$prod:D_I               s:0
        O:PI            Q:tot_i
        I:PY(com)       Q:U(com,"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(sec)    I:PK   prod:ACT(sec)
        V:ACTPL(sec)    I:PL   prod:ACT(sec)
$offtext

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

Programming

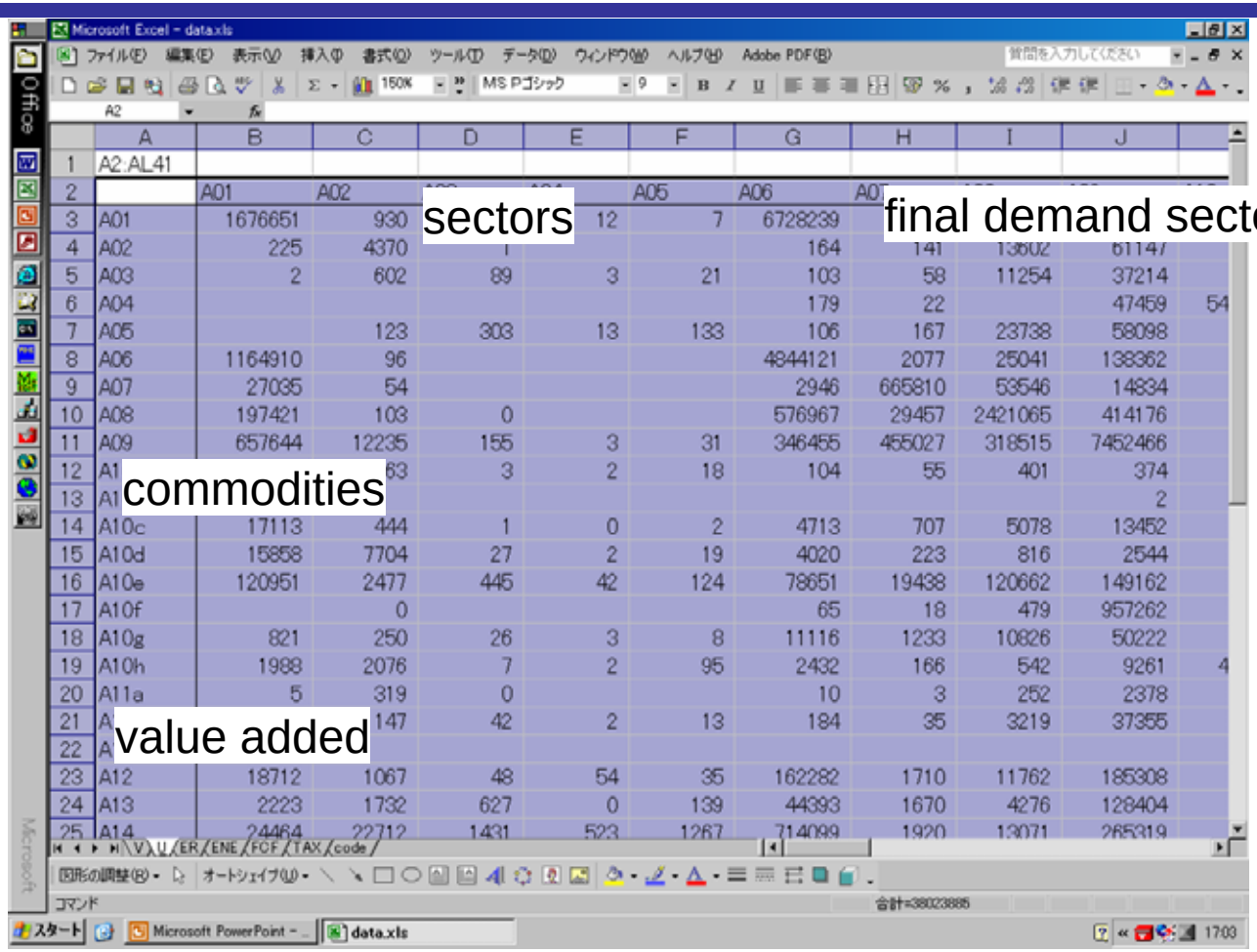
<IO base program>

```
$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:ACTPY(com)  O:PY(com)  prod:ACT(com)
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;
```

<U&V base program>

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

$prod:D_I              s:0
      O:PI             Q:tot_i
      I:PY(com)        Q:U(com,"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:ACTPY(sec,com) O:PY(com)  prod:ACT(sec)
V:ACTPK(sec)     I:PK       prod:ACT(sec)
V:ACTPL(sec)     I:PL       prod:ACT(sec)
$offtext

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

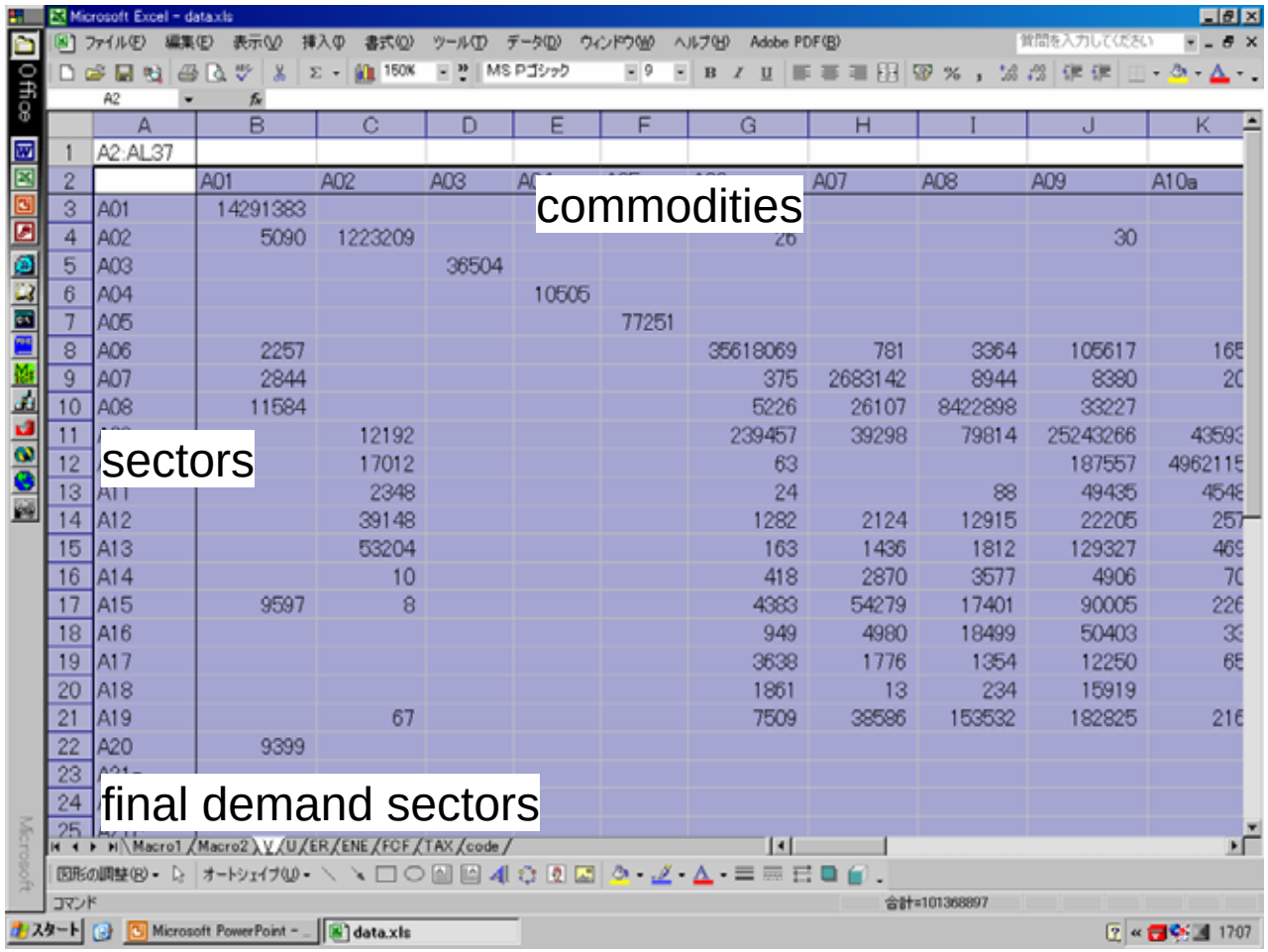
Making U matrix



	A	B	C	D	E	F	G	H	I	J
1	A2:AL41									
2		A01	A02			A05	A06	A07		
3	A01	1676651	930		12	7	6728239			
4	A02	225	4370				164	141	13602	61147
5	A03	2	602	89	3	21	103	58	11254	37214
6	A04						179	22		47459
7	A05		123	303	13	133	106	167	23738	58098
8	A06	1164910	96				4844121	2077	25041	138362
9	A07	27035	54				2946	665810	53546	14834
10	A08	197421	103	0			576967	29457	2421065	414176
11	A09	657644	12235	155	3	31	346455	455027	318515	7452466
12	A1		63	3	2	18	104	55	401	374
13	A1									2
14	A10c	17113	444	1	0	2	4713	707	5078	13452
15	A10d	15858	7704	27	2	19	4020	223	816	2544
16	A10e	120951	2477	445	42	124	78651	19438	120662	149162
17	A10f		0				65	18	479	957262
18	A10g	821	250	26	3	8	11116	1233	10826	50222
19	A10h	1988	2076	7	2	95	2432	166	542	9261
20	A11a	5	319	0			10	3	252	2378
21	A		147	42	2	13	184	35	3219	37355
22	A									
23	A12	18712	1067	48	54	35	162282	1710	11762	185308
24	A13	2223	1732	627	0	139	44393	1670	4276	128404
25	A14	24464	22712	1431	523	1267	714099	1920	13071	265319

On Excel sheet, U matrix (commodity x sector) is created.

Making V matrix



On Excel sheet, V matrix (sector x commodity) is prepared.

How to create U & V matrix

1. If your country distributes both these data, you can use them directly.
2. If you have IO table and V matrix, you can calculate U matrix by yourself.

   From V matrix and input coefficient of IO table, U matrix can be calculated using assumption that "each commodity has the same input structure even though it is produced in any different industries"; $U=AV^{T}$
3. If you have only IO table, you can directly use IO table as U matrix, and create table, in which diagonal elements are equal to output of U matrix, as V matrix.

Creation of U matrix from IO and V

IO	com 1	com 2	final demand	total
com 1	80	20	100	200
com 2	40	100	20	160
value added	80	40		
total	200	160		

V	com 1	com 2	total
act A	120	60	180
act B	80	100	180
total	200	160	

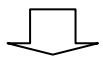
A	com 1	com 2	final demand	total
com 1	0.4	0.125	100	200
com 2	0.2	0.625	20	160
value added	0.4	0.250		
total	1	1		

V^T	act A	act B	total
com 1	120	80	200
com 2	60	100	160
total	180	180	

U	act A	act B	final demand	total
com 1	55.5	44.5	100	200
com 2	61.5	78.5	20	160
value added	63.0	57.0		
total	180	180		

Creation of U & V matrix from IO

IO	com 1	com 2	final demand	total
com 1	80	20	100	200
com 2	40	100	20	160
value added	80	40		
total	200	160		



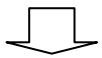
U	act A	act B	final demand	total
com 1	80	20	100	200
com 2	40	100	20	160
value added	80	40		
total	200	160		

V	com 1	com 2	total
act A	200	0	200
act B	0	160	160
total	200	160	

Creation of U & V matrix from IO with negative inputs

IO	com 1	com 2	final demand	total
com 1	80	-10	130	200
com 2	40	100	20	160
value added	80	70		
total	200	160		

Negative demand = output



U	act A	act B	final demand	total
com 1	80	0	130	210
com 2	40	100	20	160
value added	80	70		
total	200	170		

V	com 1	com 2	total
act A	200	0	200
act B	10	160	170
total	210	160	

1. Negative values in IO are transferred in V matrix.
2. In U matrix, negative values are converted in 0.
3. Total values in row and column in U matrix are calculated.
4. Total values in U matrix are transferred in V matrix.
5. Each cell in V matrix is calculated to keep the consistency.

Homework until tomorrow

- Make U matrix and V matrix for your country CGE model on Excel sheet.
- Check consistency between U matrix and V matrix.